THE INSTITUTE FOR SYSTEMS RESEARCH

ISR TECHNICAL REPORT 2013-06

A Distributed Learning Algorithm with Bit-valued Communications for Multi-agent Welfare Optimization

Anup Menon and John S. Baras

A Distributed Learning Algorithm with Bit-valued Communications for Multi-agent Welfare Optimization

Anup Menon and John S. Baras*†

Abstract

A multi-agent system comprising N agents, each picking actions from a finite set and receiving a payoff that depends on the action of the whole, is considered. The exact form of the payoffs are unknown and only their values can be measured by the respective agents. A decentralized algorithm was proposed by Marden et. al. [1] and in the authors' earlier work [2] that, in this setting, leads to the agents picking welfare optimizing actions under some restrictive assumptions on the payoff structure. This algorithm is modified in this paper to incorporate exchange of certain bit-valued information between the agents over a directed communication graph. The notion of an interaction graph is then introduced to encode known interaction in the system. Restrictions on the payoff structure are eliminated and conditions that guarantee convergence to welfare minimizing actions w.p. 1 are derived under the assumption that the union of the interaction graph and communication graph is strongly connected.

1 Introduction

An important direction of research in cooperative control of multi-agent systems is game theoretic control. This refers to the paradigm of: 1. designing individual utility functions for agents such that certain solution concepts (like Nash equilibria (NE)) correspond to desirable system-wide outcomes; and 2. prescribing learning rules that allows agents to learn such equilibria [3]. Also, the utilities and the learning rules must conform to the agents' information constraints. A popular choice is to design utilities such that the resulting game has a special structure so that the corresponding solution concepts are efficient w.r.t. system-wide objectives. NE of potential games, for instance, correspond to the extremal values of the potential function which can then be chosen so that its extrema correspond to desirable system-wide behavior. Examples of such utility design for specific applications range from distributed optimization [4] to coverage problems in sensor networks [5] and power control in wireless networks [6].

The other advantage of designing utilities with special structure is that players can be prescribed already available learning algorithms from evolutionary games that helps them learn to play NE [7], [8], [9]. These algorithms have the desirable feature of being payoff-based, i.e. an agent adjusts its play only on the basis of its past payoffs and actions, and does not require the agents to have any knowledge of the structure of the game. However, the success for most learning procedures is only guaranteed under an assumption on the game such as potential, weakly acyclic or congestion game.

Thus, while an effective paradigm, game theoretic control has the following limitations:

- Since available algorithms are provably correct only for certain classes of games, there is a burden to design utilities that conform to such structure for each application.
- If a compromise is made in the utility design phase in order to obtain required game structure, the resulting utilities may not reflect the desired system-wide outcome.
- If the system requirements prohibit design of utilities with special structure, equilibrating to NE may be inefficient w.r.t. desirable outcome (also, may be unnecessary in non-strategic situations).

*Research partially supported by the US Air Force Office of Scientific Research MURI grant FA9550-09-1-0538 and by the National Science Foundation (NSF) grant CNS-1035655.

†The authors are with the Institute for Systems Research and the Department of Electrical and Computer Engineering at the University of Maryland, College Park, MD 20742, USA amenon@umd.edu, baras@umd.edu

While within the paradigm of game theoretic control, our approach is complementary to that of utility design. Instead, we focus on algorithm design for welfare (i.e. the sum of individual utilities) optimization for arbitrary utilities. To motivate this approach we consider an application where the current paradigm is too restrictive: the problem of maximizing the total production of a wind farm [10]. Aerodynamic interactions between different wind turbines are not well understood and there are no good models to predict the effects of one turbine's actions on another. However, it is clear that the amount of power a turbine extracts from the wind has a direct effect on the power production of turbines downstream (such a region of influence is called a wake, see Figure 1(a)). The information available to each turbine is its own power production and a decentralized algorithm that maximizes the total power production of the farm is sought. Since there are no good models for the interactions, there is little hope to design utilities with special structure that are functions of such individual power measurements. This points towards the need for algorithms that are applicable when there is little structural information about the utilities (for instance, a turbine can be assigned its individual power as its utility which, in turn, can depend on the actions taken by others in complex ways).

A decentralized learning algorithm is presented in [1] with the intent to address this issue of unknown payoff structures. This algorithm allows agents to learn welfare maximizing actions and does not require any knowledge about the exact functional form of the utilities. In our earlier work [2], we provide conditions that guarantee convergence of this algorithm. However, convergence is guaranteed only under an assumption on the utilities called *interdependence* (which, for instance, need not hold for the wind farm problem).

The contribution of this paper is a distributed multi-agent learning algorithm that:

1. eliminates the need for any structural assumptions on the utilities by using inter-agent communication;
2. and, under appropriate conditions, ensures that agent actions converge to global extrema of the welfare function.

Regarding the use of inter-agent communication, in [10], a scheme using proxy utilities computed by inter-agent communication over a undirected connected graph is suggested to satisfy interdependence and enable using the algorithm of [1], [2]. In contrast, we develop a framework for capturing known interaction in the system and prove results that help design "minimal" communication networks that guarantee convergence. The exchanged information in our algorithm is bit-valued which has implementation and robustness advantages. The framework developed also contrasts between implicit interaction between the agents via utilities and explicit interaction via communication. We also wish to point out that the problem formulated here can be thought of as a multi-agent formulation of a discretized extremum seeking problem [11] and the algorithm provides convergence to global optimal states with few restrictions on the functions involved.

The remainder of the paper is organized as follows. In section 2 we formulate the problem, develop the analysis framework, present the algorithm and state the main convergence result. Section 3 introduces *Perturbed Markov Chains* and states relevant results. In section 4, the results of Section 3 are used to prove the main result of section 2. The paper concludes with some numerical illustrations and discussions about future work.

Notation

The paper deals exclusively with discrete-time, finite state space Markov chains. A time-homogeneous Markov chain with Q as its 1-step transition probability matrix means that the i^{th} row and j^{th} column entry $Q_{i,j} = \mathbb{P}(\mathbf{X}_{t+1} = j | \mathbf{X}_t = i)$, where $\mathbf{X}_t$ denotes the state of the chain at time t. If the row vector η_t denotes the probability distribution of the states at time t, then $\eta_{t+1} = \eta_t \; Q$. More generally, if $Q(t)$ denotes the 1-step transition probability matrix of a time-nonhomogeneous Markov chain at time t, then for all $m > n$, $\mathbb{P}(\mathbf{X}_m = j | \mathbf{X}_n = i) = Q_{i,j}^{(n,m)}$, where the matrix $Q^{(n,m)} = Q(n) \cdot Q(n+1) \cdots Q(m-1)$. The time indices of all Markov chains take consecutive values from the set of natural numbers $\mathbb{N}$. A Markov chain should be understood to be homogeneous unless stated otherwise. We denote the N-dimensional vector of all zeros and all ones by the bold font $\mathbf{0}$ and $\mathbf{1}$ respectively. For a multi-dimensional vector x, its i^{th} component is denoted by x_i; and that of x_t by $(x_t)_i$.

2 Problem Statement and Proposed Algorithm

2.1 A Multi-agent Extremum Seeking Formulation

2.1.1 Agent Model

We consider N, possibly heterogeneous, agents indexed by i. The i^{th} agent can pick actions from a set $\mathcal{A}_i$, $2 \leq |\mathcal{A}_i| < \infty$; the joint action of the agents is an element of the set $\mathcal{A} = \prod_{i=1}^{N} \mathcal{A}_i$. The action of the i^{th} agent in the joint action $a \in \mathcal{A}$ is denoted by a_i. Further, given the i^{th} individual's present action is $b \in \mathcal{A}_i$, the choice of its very next action is restricted to be from $\mathcal{A}_i(b) \subset \mathcal{A}_i$.

Assumption 1. *For any $b \in \mathcal{A}_i$, $b \in \mathcal{A}_i(b)$ and there exists an enumeration $\{b_1, ..., b_{|\mathcal{A}_i|}\}$ of $\mathcal{A}_i$ such that $b_{j+1} \in \mathcal{A}_i(b_j)$ for $j = 1, ..., (|\mathcal{A}_i| - 1)$ and $b_1 \in \mathcal{A}_i(b_{|\mathcal{A}_i|})$.*

The former allows for the possibility of picking the same action in consecutive steps and the latter ensures that any element of $\mathcal{A}_i$ is "reachable" from any other. Specific instances of such agent models in literature include the discretized position and viewing-angle sets for mobile sensors in [5], discretized position of a robot in a finite lattice in [12, 13] and the discretization of the axial induction factor of a wind turbine in [10].

An individual has a private utility that can be an arbitrary time-invariant function of the action taken by the whole but is measured or accessed only by the individual. Agent i's utility is denoted by $u_i : \mathcal{A} \to \mathbb{R}^+$. Examples include artificial potentials used to encode information about desired formation geometry for collaborative control of autonomous robots in [12, 13] and the measured power output of an individual wind turbine in [10]. At any time t, agent i only measures or receives $(u_t^{mes})_i = u_i(a_t)$ since neither the joint action a_t nor any information about $u_i(\cdot)$ is known to the agent.

The objective of the multi-agent system is to collaboratively minimize (or maximize) the welfare function $W^* = \min_{a \in \mathcal{A}} W(a)$, where $W(a) = \sum_{i=1}^{N} u_i(a)$. Achieving this objective results in a desirable behavior of the whole like a desired geometric configuration of robots in [12, 13], desired coverage vs. sensing energy trade-off in [5] and maximizing the power output of a wind farm in [10]. Thus we seek distributed algorithms for the agents to implement so that their collective actions converge in an appropriate sense to the set

$$\mathcal{A}^* = \{\arg\min_{a \in \mathcal{A}} W(a)\}.$$

Before proceeding further we must point out that this is a general combinatorial optimization problem and is NP hard. In fact, even if a brute force search were used, it is not clear how elements of $\mathcal{A}^*$ can be identified if the agents do not exchange any information.

2.1.2 Interaction Model

Interaction in a multi-agent setting can comprise of explicit communication between agents via communication or can be implicit with actions of an agent reflecting on the payoff of another. The latter is an artifact of the given problem at hand and must be modeled appropriately. We present a general modeling framework that allows the designer to encode known inter-agent interactions in the system. The communication network varies from one application to another and we choose a simple signaling network where only a bit-valued variable is exchanged amongst the agents.

1. **Interaction Graph**
   Consider a directed graph $\mathcal{G}_I(a)$ for every $a \in \mathcal{A}$ with a vertex assigned to each agent. Its edge set contains the directed edge (j, i) if and only if $\exists\, b \in \mathcal{A}_j$ such that $u_i(a) \neq u_i(b, a_{-j})$.[1] Thus, for every action profile a, $\mathcal{G}_I(a)$ encodes the set of agents whose actions can (and must) affect the payoffs of other specific agents. We call this graph the *interaction graph*.

   In the case of a wind farm, power production of a turbine downstream of another may be affected by the actions of the latter (see Figure 1). For the collaborative robotics problem, all robots that contribute to the artificial potential of a given robot constitute the latter's in-neighbors in the interaction graph.

[1] We borrow notation from the game theory literature: a_J denotes the actions taken by the agents in subset J from the collective action a and the actions of the rest is denoted by a_{-J}.

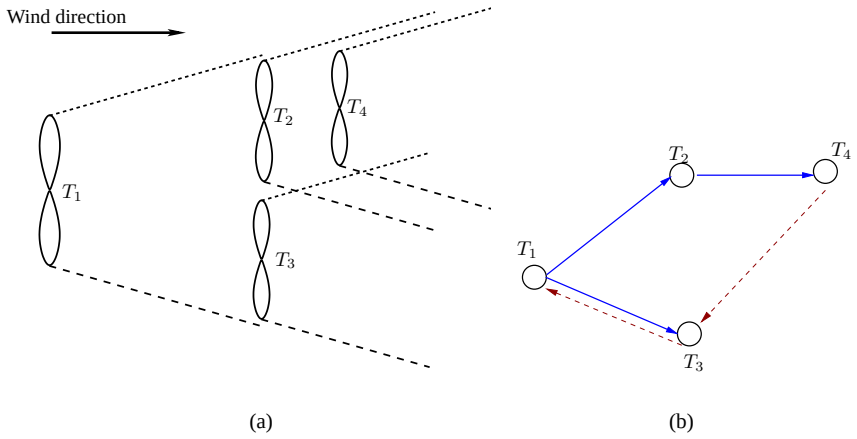


Figure 1: (a) Schematic diagram of a wind farm; a loop represents a wind turbine and the dotted lines its corresponding wake. (b) Solid arrows represent edges in $\mathcal{G}_I$ and the dotted arrows edges in $\mathcal{G}_c$.

Essentially, the interaction graph is a way of encoding certain "coarse" information about the structure of the payoff functions even in the absence of explicit knowledge of their functional forms.

2. **Communication Graph**
   The agents are assumed to have a mechanism to transmit a bit-valued message to other agents within a certain range. The mode of communication is broadcast and an agent need not know which other agents are receiving its message. For each $a \in \mathcal{A}$, we model this explicit information exchange by a directed graph over the set of agents $\mathcal{G}_c(a)$ called the *communication graph*. A directed edge (j, i) in $\mathcal{G}_c(a)$ represents that agent j can send a message to agent i when the joint action being played is a. Let $\mathcal{N}_i(a)$ denote the in-neighbors of agent i in the communication graph. Each transmission is assumed to last the duration of the algorithm iterate.

The dependence of the above graphs on $a \in \mathcal{A}$ is required for accurate modeling; for instance, in the collaborative robotics example where $\mathcal{A}_i$ represents the set of discretized positions of robot i, which robots are within the interaction or communication range of specific others depends on the position of all robots ($\in \mathcal{A}$). On the contrary, in the wind farm example, these graphs can be considered constant for all $a \in \mathcal{A}$ (see Figure 1(b)). We stress that this framework is for modeling and analysis at the level of the system designer; the agents neither know the joint action nor the corresponding neighbors in $\mathcal{G}_I(\cdot)$ or $\mathcal{G}_c(\cdot)$ and simply go about measuring their utilities, broadcasting messages and receiving such broadcast messages from other agents when in range.

2.2 The Decentralized Algorithm

Endow agent i with a state $x_i = [a_i, m_i]$; $a_i \in \mathcal{A}_i$ corresponds to the action picked and m_i is the $\{0, 1\}$-valued 'mood' of agent i. When the mood variable equals 1 we call the agent "content" else "discontent". The collective state of all agents is denoted by $x = (a, m)$. Additionally, each agent maintains a variable $\overline{u}_i$, which records the payoff it received in the previous iterate. At $t = 0$, the agent i initializes $(m_0)_i = 0$, picks an arbitrary $(a_0)_i \in \mathcal{A}_i$ and records the received payoff $\overline{u}_i = (u_0^{mes})_i$. With slight abuse of notation, we let $\mathcal{G}_c(a_t) = \mathcal{G}_c(t)$ and $\mathcal{A}_i((a_t)_i) = \mathcal{A}_i(t)$.

For a certain pre-specified monotone decreasing sequence $\{\epsilon_t\}_{t\in\mathbb{N}}$, with $\epsilon_t \to 0$ as $t \to \infty$, and constants $c > W^*$, $\beta_1, \beta_2 > 0$, agent i performs the following sequentially at every ensuing time instant $t > 0$.

Start

Step 1: Receive $(\mathbf{m}_{t-1})_j$ from all $j \in \mathcal{N}_i(t-1)$, i.e. the in-neighbors of i in $\mathcal{G}_c(t-1)$. Compute temporary variable $\tilde{m}_i$ as follows.

1. If $(\mathbf{m}_{t-1})_i = 0$, set $\tilde{m}_i = 0$;

2. else, if $(\mathbf{m}_{t-1})_i = 1$ and $\prod_{j \in \mathcal{N}_i(t-1)} (\mathbf{m}_{t-1})_j = 1$, set $\tilde{m}_i = 1$;

3. and if $(\mathbf{m}_{t-1})_i = 1$ and $\prod_{j \in \mathcal{N}_i(t-1)} (\mathbf{m}_{t-1})_j = 0$,
set $\tilde{m}_i = \{0, 1\}$ w.p. $\{1 - \epsilon_t^{\beta_1}, \epsilon_t^{\beta_1}\}$.

Step 2: Pick $(\mathbf{a}_t)_i$ as follows.

1. If $\tilde{m}_i = 1$, pick $(\mathbf{a}_t)_i$ from $\mathcal{A}_i(t-1)$ according to the p.m.f.

$$p(b) = \begin{cases} 1 - \epsilon_t^c & \text{if } b = (\mathbf{a}_{t-1})_i \\ \frac{\epsilon_t^c}{|\mathcal{A}_i(t-1)|-1} & \text{otherwise.} \end{cases} \tag{1}$$

2. Else, if $\tilde{m}_i = 0$, pick $(\mathbf{a}_t)_i$ according to the uniform distribution

$$p(b) = \frac{1}{|\mathcal{A}_i(t-1)|} \text{ for all } b \in \mathcal{A}_i(t-1). \tag{2}$$

Step 3: Measure or receive payoff $(\mathbf{u}_t^{mes})_i (= u_i(\mathbf{a}_t))$.
Step 4: Update $(\mathbf{m}_t)_i$ as follows.

1. If $\tilde{m}_i = 1$ and $((\mathbf{a}_t)_i, (\mathbf{u}_t^{mes})_i) = ((\mathbf{a}_{t-1})_i, \overline{u}_i)$, then set $(\mathbf{m}_t)_i = 1$;

2. else, if $\tilde{m}_i = 1$ and $((\mathbf{a}_t)_i, (\mathbf{u}_t^{mes})_i) \neq ((\mathbf{a}_{t-1})_i, \overline{u}_i)$,
set $(\mathbf{m}_t)_i = \{0, 1\}$ w.p. $\{1 - \epsilon_t^{\beta_2}, \epsilon_t^{\beta_2}\}$;

3. and if $\tilde{m}_i = 0$, set

$$(\mathbf{m}_t)_i = \begin{cases} 1 & \text{w.p.} \quad \epsilon_t^{(\mathbf{u}_t^{mes})_i} \\ 0 & \text{w.p.} \quad 1 - \epsilon_t^{(\mathbf{u}_t^{mes})_i}. \end{cases} \tag{3}$$

Update $\overline{u}_i (\leftarrow \mathbf{u}_t^{mes})_i$.
Step 5: Broadcast $(\mathbf{m}_t)_i$ to all out-neighbors in $\mathcal{G}_c(t)$.
Stop
It is easy to see that the algorithm defines a nonhomogeneous Markov chain on the state space $S = \mathcal{A} \times \{0, 1\}^N$.

2.3 Convergence Guarantees

The following is the main convergence result for the decentralized algorithm described above.

Theorem 1. *Let*

1. $\sum_{t=1}^{\infty} \epsilon_t^c = \infty$ *and*

2. *for every* $a \in \mathcal{A}$, $\mathcal{G}_c(a) \cup \mathcal{G}_I(a)$ *be strongly connected.*

Then, if $\mathbf{X}_t = [\mathbf{a}_t, \mathbf{m}_t]$ *denotes the collective state of the agents at time* t,

$$\lim_{t \to \infty} \mathbb{P}[\mathbf{a}_t \in \mathcal{A}^*] = 1.$$

From a practical view-point, the result provides a system-designer with guidelines on how to guarantee convergence of the above algorithm. The first assumption translates to a constraint on the sequence $\{\epsilon_t\}_{t \in \mathbb{N}}$ (or an "annealing schedule") on how fast it may approach zero.

The second provides flexibility to design a 'minimal' communication network by utilizing information about the payoff structure (such a choice is made in Figure 1.b). For instance, if the designer has no information about the structure of the payoff function ($\mathcal{G}_I(\cdot) = \emptyset$), a communication network such that $\mathcal{G}_c(a)$ is strongly connected for all $a \in \mathcal{A}$ can be installed to guarantee convergence. The other extreme case is for all $a \in \mathcal{A}$, $\mathcal{G}_I(a)$ is strongly connected; then the algorithm converges even in the absence of any explicit communication (It can be seen that strong connectivity of $\mathcal{G}_I$ implies interdependence and indeed the result holds true under analogous weaker but more cumbersome-to-state conditions.).

3 Perturbed Markov Chains

The analysis of the algorithm described in the previous section relies on the theory of *perturbed Markov chains* developed by Young [14]. Consider a homogeneous Markov chain with possibly several stationary distributions. Perturb the transition matrix of this chain by adding appropriate functions of a certain noise parameter ϵ to obtain an ergodic chain with a unique stationary distribution. These functions are such that, as $\epsilon \to 0$, the transition matrix of the perturbed chain converges to that of the 'unperturbed chain' with individual elements converging to their respective limits with asymptotic rates $r(\cdot,\cdot)$ (i.e. as $\Theta(\epsilon^{r(x,y)})$). What is interesting is that, under appropriate conditions, as $\epsilon \to 0$, the stationary distribution of the perturbed chain converges to a certain stationary distribution of the unperturbed chain and the support of the latter can be characterized in terms of the rates $r(\cdot,\cdot)$. Thus, one can effectively "choose" amongst the possibly several stationary distributions of the unperturbed chain. This section describes this theory in detail and also proves results on how to reduce ϵ over time while evolving according to the perturbed chain (rendering the chain nonhomogeneous) while retaining ergodicity.

3.1 Perturbed Markov Chains

Let $P(0)$ be the 1-step transition probability matrix of a Markov chain on a finite state space S. We refer to this chain as the *unperturbed chain*.

Definition 3.1. *A regular perturbation of $P(0)$ consists of a stochastic matrix valued function $P(\epsilon)$ on some non-degenerate interval $(0, a]$ that satisfies, for all $x, y \in S$,*

1. *$P(\epsilon)$ is irreducible and aperiodic for each $\epsilon \in (0, a]$,*
2. *$\lim_{\epsilon \to 0} P_{x,y}(\epsilon) = P_{x,y}(0)$ and*
3. *if $P_{x,y}(\epsilon) > 0$ for some ϵ, then $\exists\ r(x, y) \geq 0$ such that $0 < \lim_{\epsilon \to 0} \epsilon^{-r(x,y)} P_{x,y}(\epsilon) < \infty$.*

An immediate consequence of the first requirement is that there exists a unique stationary distribution $\mu(\epsilon)$ satisfying $\mu(\epsilon)P(\epsilon) = \mu(\epsilon)$ for each $\epsilon \in (0, a]$. The other two requirements dictate the way the perturbed chain converges to the unperturbed one as $\epsilon \to 0$.

It follows that for a sufficiently small ϵ^*, $\exists\ 0 < \underline{\alpha}(x, y) < \overline{\alpha}(x, y) < \infty$, such that

$$\underline{\alpha}(x, y) < \epsilon^{-r(x,y)} P_{x,y}(\epsilon) < \overline{\alpha}(x, y),\ \forall\ \epsilon < \epsilon^*.$$

By denoting $\min_{x,y \in S} \underline{\alpha}(x, y) = \underline{\alpha}$ and $\max_{x,y \in S} \overline{\alpha}(x, y) = \overline{\alpha}$, we have

$$\underline{\alpha}\ \epsilon^{r(x,y)} < P_{x,y}(\epsilon) < \overline{\alpha}\ \epsilon^{r(x,y)},\ \forall\ \epsilon < \epsilon^*. \tag{4}$$

Let $\mathfrak{L} = \{f \in \mathfrak{C}^\infty |\ f(\epsilon) \geq 0, f(\epsilon) = \sum_{i=1}^{L} a_i \epsilon^{b_i}$ for some $a_i \in \mathbb{R}, b_i \geq 0\}$ for some large enough but fixed $L \in \mathbb{N}$. The following assumption will be invoked later.

Assumption 2. *For all $x, y \in S$, $P_{x,y}(\epsilon) \in \mathfrak{L}$.*

We develop some notation that will help state the main result regarding perturbed Markov chains. The parameter $r(x, y)$ in the definition of regular perturbation is called the *1-step transition resistance* from state x to y. Notice that $r(x, y) = 0$ only for the one step transitions $x \to y$ allowed under $P(0)$. A *path* $h(a \to b)$ from a state $a \in S$ to $b \in S$ is an ordered set $\{a = x_1, x_2, \ldots, x_n = b\} \subseteq S$ such that every transition $x_k \to x_{k+1}$ in the sequence has positive 1-step probability according to $P(\epsilon)$. The resistance of such a path is given by $r(h) = \sum_{k=1}^{n-1} r(x_k, x_{k+1})$.

Definition 3.2. *The resistance from x to y is given by $\rho(x, y) = \min\{r(h) |\ h(x \to y)$ is a path$\}$.*

Definition 3.3. *Given a subset $A \subset S$, its co-radius is given by $CR(A) = \max_{x \in S \setminus A} \min_{y \in A} \rho(x, y)$.*[2]

[2]These definitions are adopted from relevant literature [15],[16].

Thus, $\rho(x, y)$ is the minimum resistance over all possible paths starting at state x and ending at state y and the co-radius of a set specifies the maximum resistance that must be overcome to enter it from outside. We will overload the definition of resistance to include resistance between two subsets $S_1, S_2 \subset S$:

$$\rho(S_1, S_2) = \min_{x \in S_1, y \in S_2} \rho(x, y).$$

Since $P(\epsilon)$ is irreducible for $\epsilon > 0$, $\rho(S_1, S_2) < \infty$ for all $S_1, S_2 \subset S$.

Definition 3.4. *A recurrence or communication class of a Markov chain is a non-empty subset of states $E \subseteq S$ such that for any $x, y \in E$, $\exists\ h(x \to y)$ and for any $x \in E$ and $y \in S \setminus E$, $\nexists\ h(x \to y)$.*

Let us denote the recurrence classes of the unperturbed chain $P(0)$ as $E_1, ..., E_M$. Consider a directed graph $\mathcal{G}_{RC}$ on the vertex set $\{1, ..., M\}$ with each vertex corresponding to a recurrence class. Let a *j-tree* be a spanning subtree in $\mathcal{G}_{RC}$ that contains a unique directed path from each vertex in $\{1, ..., M\} \setminus \{j\}$ to j and denote the set of all j-trees in $\mathcal{G}_{RC}$ by $\mathcal{T}^j_{RC}$.

Definition 3.5. *The stochastic potential of a recurrence class E_i is*

$$\gamma(E_i) = \min_{T \in \mathcal{T}^i_{RC}} \sum_{(j,k) \in T} \rho(E_j, E_k).$$

Let

$$\gamma^* = \min_{E_i} \gamma(E_i).$$

We are now ready to state the main result regarding perturbed Markov chains.

Theorem 2 ([14], Theorem 4)**.** *Let $E_1, ..., E_M$ denote the recurrence classes of the Markov chain $P(0)$ on a finite state space S. Let $P(\epsilon)$ be a regular perturbation of $P(0)$ and let $\mu(\epsilon)$ denote its unique stationary distribution. Then,*

1. *as $\epsilon \to 0$, $\mu(\epsilon) \to \mu(0)$, where $\mu(0)$ is a stationary distribution of $P(0)$ and*
2. *a state is stochastically stable i.e. $\mu_x(0) > 0 \Leftrightarrow x \in E_i$ such that $\gamma(E_i) = \gamma^*$.*

3.2 Ergodicity of nonhomogeneous Markov chains

We now recall some results on ergodicity of a nonhomogeneous Markov chain on a finite state space S, with $Q(t)$ being the 1-step transition probability matrix at time t.

Definition 3.6 (Ergodicity)**.** *The chain is*

- *weakly ergodic (WE) if for all $t' \in \mathbb{N}$ and all $x, y, z \in S$,*

$$\lim_{t \to \infty} |Q^{(t',t)}_{x,z} - Q^{(t',t)}_{y,z}| = 0.$$

- *strongly ergodic (SE) if there exists a probability distribution π on S such that for any initial distribution $\eta(0)$ on S and any $t' \in \mathbb{N}$,*

$$\lim_{t \to \infty} \eta(0) Q^{(t',t)} = \pi.$$

  We call π the limiting distribution of the chain.

Both definitions of ergodicity capture a certain notion of forgetfulness in that the chain forgets where it started after sufficiently large time steps. It is also clear that *SE* implies *WE*. Before proceeding to results that give conditions for ergodicity, we make the following definition.

Definition 3.7 (Ergodic Coefficient)**.** *Given a row stochastic matrix $Q \in \mathbb{R}^{|S| \times |S|}$, its ergodic coefficient is given by*

$$\delta(Q) = 1 - \min_{x,y \in S} \sum_{z \in S} \min\{Q_{x,z}, Q_{y,z}\}.$$

The following result due to Doeblin provides a characterization for *WE* based on the ergodic coefficient.

Theorem 3 (Weak Ergodicity, see [17], Theorem 8.2)**.** *The chain is weakly ergodic if and only if there exists a strictly increasing sequence of positive integers* $\{t_n\}_{n\in\mathbb{N}}$ *such that*

$$\sum_{n\in\mathbb{N}}(1-\delta(Q^{(t_n,t_{n+1})}))=\infty. \tag{5}$$

The next Theorem provides a sufficiency condition for *SE*.

Theorem 4 (Strong Ergodicity, see [17], Theorem 8.3)**.** *Suppose the chain is weakly ergodic and at all* t*, there exists* π_t *such that* $\pi_t Q(t)=\pi_t$ *and*

$$\sum_{t\in\mathbb{N}}\|\pi_{t+1}-\pi_t\|_1<\infty, \tag{6}$$

then the chain is strongly ergodic. Furthermore, the limiting distribution π *as in the definition of SE is the same as the limit of the sequence* $\{\pi_t\}_{t\in\mathbb{N}}$*.*

Proof. This standard result can be found in [17], Theorem 8.3, or [18], Theorem V.4.3. To see why π is the limiting distribution of the nonhomogeneous chain, note that (6) implies (8.12) in [17], pp. 242-243, which is equivalent to the definition of *SE*. □

3.3 Ergodicity of Nonhomogeneous Perturbed Markov Chains

Consider the nonhomogeneous Markov chain resulting from picking the ϵ along the evolution of $P(\epsilon)$ at time instant t as the corresponding element ϵ_t of the sequence $\{\epsilon_t\}_{t\in\mathbb{N}}$. We henceforth refer to this sequence as the annealing schedule and the resulting Markov chain as the nonhomogeneous perturbed chain. Theorem 5 provides conditions on the annealing schedule that guarantee ergodicity of the nonhomogeneous perturbed chain with $\mu(0)$ (as in Theorem 2) being the limiting distribution. We denote the time-varying transition matrix of the nonhomogeneous perturbed chain by the bold-font $\mathbf{P}$, i.e. $\mathbf{P}(t)=P(\epsilon_t)$.

We will need the following technical Lemma.

Lemma 3.1. *Let* $\sum_{n\in\mathbb{N}} a(n)=\infty$ *and* $a(n)\geq a(n+1)\ \forall\ n$*. Then for any* $n',l\in\mathbb{N}$*,* $\sum_{n\in\mathbb{N}} a(n'+l+n)=\infty$*.*

Proof. The case for $l=1$ is trivially true. If $l>1$, $\forall\ n$,

$$\begin{aligned}
a(n'+ln) &\geq a(n'+ln+m),\ \forall m=1,...,l-1\\
\Rightarrow l\cdot a(n'+ln) &\geq \sum_{m=0}^{l-1} a(n'+ln+m).\\
\text{Thus } l\sum_{n\in\mathbb{N}} a(n'+ln) &\geq \sum_{n\in\mathbb{N}}\sum_{m=0}^{l-1} a(n'+ln+m)\\
&= \sum_{n\in\mathbb{N}} a(n'+n)=\infty.
\end{aligned}$$

□

Define

$$\kappa=\min_{E\in\{E_i\}} CR(E). \tag{7}$$

Theorem 5 ([2], Theorem 3)**.** *Let the recurrence classes of the unperturbed chain* $P(0)$ *be aperiodic and the parameter* ϵ *in the perturbed chain be scheduled according to the monotone decreasing sequence* $\{\epsilon_t\}_{t\in\mathbb{N}}$*, with* $\epsilon_t\to 0$ *as* $t\to\infty$*, as described above. Then, a sufficient condition for weak ergodicity of the resulting nonhomogeneous Markov chain* $\mathbf{P}(t)$ *is*

$$\sum_{t\in\mathbb{N}}\epsilon_t^{\kappa}=\infty.$$

Furthermore, if the chain is weakly ergodic and Assumption 2 holds, then it is strongly ergodic with the limiting distribution being $\mu(0)$ *as described in Theorem 2.*

Proof. Weak Ergodicity: Let E^* be a recurrent class such that $CR(E^*) = \gamma$. Since E^* is aperiodic according to $P(0)$, there exists an $l_1 \in \mathbb{N}$ such that for all $m \geq l_1$ and $x, y \in E^*$, $P^m_{x,y}(0) > 0$ (see [17], Theorem 4.3, pp. 75). Since any path under $P(0)$ has zero resistance, once the chain enters a state in E^*, it can remain there with zero resistance via a path of length greater than l_1.

Let $e^* \in E^*$ be such that $\exists\ x' \in S \setminus E^*$ such that $\mathbf{r}(x', e^*) = \gamma$ i.e. the transition $x' \to e^*$ has the most resistance among all $x \to e^*$, $x \in S$. For all $x \in S$, consider the shortest paths $h(x \to e^*)$ such that $r(h(x \to e^*)) = \mathbf{r}(x, e^*)$ and denote the length of such paths by $l(x, e^*)$. Let $l_2 = \max_{x \in S} l(x, e^*)$. So by waiting for l_2 transitions, there is a path to E^* from all states $x \in S$ with resistance $\mathbf{r}(x, e^*)$. Thus by allowing more than $l = l_1 + l_2$ transitions, we have for any $x \in S$ and a sufficiently small ϵ^*,

$$P^m_{x,e^*}(\epsilon) > \underline{\alpha}^m \epsilon^\gamma, \quad \forall\ \epsilon < \epsilon^*,\ m \geq l.$$

From (4), since $\epsilon_t \to 0$, for sufficiently large t,

$$\underline{\alpha}\ \epsilon_t^{r(x,y)} < \mathbf{P}_{x,y}(t) < \overline{\alpha}\ \epsilon_t^{r(x,y)}.$$

Consequently, by choosing a subsequence such that $t_{n+1} - t_n = l$, for sufficiently large n,

$$\mathbf{P}^{(t_n, t_{n+1})}_{x,e^*} > \underline{\alpha}^l \epsilon^\gamma_{t_{n+1}},\ \forall x \in S.$$

Then, for sufficiently large n, we can bound

$$\begin{aligned}
&\sum_{z \in S} \min\{\mathbf{P}^{(t_n,t_{n+1})}_{x,z}, \mathbf{P}^{(t_n,t_{n+1})}_{y,z}\} \\
&\quad \geq \min\{\mathbf{P}^{(t_n,t_{n+1})}_{x,e^*}, \mathbf{P}^{(t_n,t_{n+1})}_{y,e^*}\} > \underline{\alpha}^l \epsilon^\gamma_{t_{n+1}},\ \forall x, y \in S.
\end{aligned}$$

Taking minimum over x, y, for sufficiently large n,

$$\min_{x,y \in S} \sum_{z \in S} \min\{\mathbf{P}^{(t_n,t_{n+1})}_{x,z}, \mathbf{P}^{(t_n,t_{n+1})}_{y,z}\} > \underline{\alpha}^l \epsilon^\gamma_{t_{n+1}}. \tag{8}$$

Since $\{t_n\}_{n \in \mathbb{N}}$ is an equally spaced subsequence, from Lemma 3.1 and the hypothesis of the theorem, $\sum_{n \in \mathbb{N}} \epsilon^\gamma_{t_{n+1}} = \infty$. In view of this and (8), *WE* follows by noting that (5) is verified with $Q = \mathbf{P}$.

Strong Ergodicity: Recall the homogeneous perturbed Markov chain $P(\epsilon)$. Consider a graph $\mathcal{G} = (S, \mathcal{E})$ with the state space S as the vertex set and a directed edge $(x, y) \in \mathcal{E}$ if and only if $P_{x,y}(\epsilon) > 0$ for some ϵ. For any vertex $z \in S$, a *z-tree* is a subset of $\mathcal{E}$ that forms a spanning tree in $\mathcal{G}$ such that for every vertex $x \neq z$, there exists a unique directed path from x to z. Let $\mathcal{T}_z$ be the set of all z-trees in $\mathcal{G}$. Then it is known (see [14]) that the stationary distribution $\mu(\epsilon)$ is given by

$$\begin{aligned}
\mu_z(\epsilon) &= \frac{q_z(\epsilon)}{\sum_{x \in S} q_x(\epsilon)} \\
\text{where } q_z(\epsilon) &= \sum_{T \in \mathcal{T}_z} \prod_{(x,y) \in T} P_{x,y}(\epsilon).
\end{aligned} \tag{9}$$

Under assumption 2, both the numerator and denominator of the R.H.S of (9) belong to $\mathfrak{L}$ for a sufficiently large L. Denoting the derivative w.r.t. ϵ by primes and suppressing the argument, $\mu'_z = (1/(\sum_{x \in S} q_x)^2) \cdot (q'_z \sum_{x \in S} q_x - q_z \sum_{x \in S} q'_x)$. Thus, after multiplying and dividing with an appropriate power of ϵ, the numerator of μ'_z also belongs to $\mathfrak{L}$ for a sufficiently large L. For a sufficiently small $\epsilon_z > 0$, μ'_z will be dominated by the term with the least exponent of ϵ for all $\epsilon < \epsilon_z$. Thus, the sign of μ'_z will be either non-positive or positive for all $\epsilon < \epsilon_z$. Let $\epsilon^* = \min_{z \in S} \epsilon_z$, $S^- \subset S$ be such that $z \in S^- \Leftrightarrow \mu'_z \leq 0\ \forall\ \epsilon < \epsilon^*$ and $S^+ = S \setminus S^-$. Let t^*

be such that $\epsilon_t < \epsilon^*,\ \forall t > t^*$. Then,

$$\begin{aligned}
\sum_{t=1}^{\infty}\|\mu(\epsilon_t) - \mu(\epsilon_{t+1})\|_1 &= \sum_{t=1}^{t^*}\|\mu(\epsilon_t) - \mu(\epsilon_{t+1})\|_1 + \\
&\quad \sum_{t=t^*+1}^{\infty}\left[\sum_{z\in S^-}(\mu_z(\epsilon_t) - \mu_z(\epsilon_{t+1})) + \sum_{z\in S^+}(\mu_z(\epsilon_{t+1}) - \mu_z(\epsilon_t))\right] \\
&= M + \sum_{z\in S^-}\sum_{t=t^*+1}^{\infty}(\mu_z(\epsilon_t) - \mu_z(\epsilon_{t+1})) \\
&\quad + \sum_{z\in S^+}\sum_{t=t^*+1}^{\infty}(\mu_z(\epsilon_{t+1}) - \mu_z(\epsilon_t)) \\
&< \infty
\end{aligned}$$

since $M = \sum_{t=1}^{t^*}\|\mu(\epsilon_t) - \mu(\epsilon_{t+1})\|_1$ is a finite sum of finite terms and successive terms cancel within both infinite sums. Since (6) is satisfied with $\pi(t) = \mu(\epsilon_t)$, as shown above, and the chain is *WE*, *SE* follows from Theorem 4. The limiting distribution, in view of Theorem 2, is $\mu(0)$. □

4 Analysis of the Algorithm

The objective of this section is to prove Theorem 1. We will first consider the algorithm of section 2.2 with the parameter ϵ_t held constant at $\epsilon > 0$. The algorithm then describes a Markov chain on the finite state space $S = \mathcal{A} \times \{0,1\}^N$ and we denote its 1-step transition matrix as $P(\epsilon)$. The reason for choosing the same notation here as for the general perturbed Markov chain discussed in section 3 is that we wish to view the Markov chain induced by the algorithm as a perturbed chain and analyze it using results from section 3. Similarly, $\mathbf{P}(t)$ denotes the 1-step transition probability matrix for the duration $(t, t+1)$ of the nonhomogeneous Markov chain induced by the algorithm as described in section 3, i.e. with time varying ϵ_t. Henceforth, the components of any $x \in S$ will be identified with a superscript i.e. $x = [a^x, m^x]$.

Lemma 4.1. *The Markov chain $P(\epsilon)$ is irreducible and aperiodic.*

Proof. Let us consider the transition probability from state $y \in S$ to $z \in S$. For $\epsilon > 0$, irrespective of the values of respective $\tilde{m}_i$, the transition probabilities (1) and (2) let the agents pick a joint action $a' \in \mathcal{A}$ such that $a'_i \neq a^y_i$ for any i with positive probability. Then, again irrespective of the values of $\tilde{m}_i$, by step 4.2 or 4.3, the state can transition from y to $[a', \mathbf{0}]$ with positive probability. Next, starting from state $[a', \mathbf{0}]$, by Assumption 1 and transition probability (2), agent i can pick the action a^z_i with positive probability in a finite number of steps and can keep playing a^z_i with positive probability for any arbitrary finite duration thereafter while maintaining $m_i = 0$ all the while. Thus there is a positive probability for all agents to pick actions that correspond to a', i.e. transition from state $[a', \mathbf{0}]$ to $[a^z, \mathbf{0}]$. Finally, in the very next time instant, agent i can repeat its action with positive probability and update its mood variable to m^z_i with positive probability (3). Hence the transition y to z occurs with positive probability.

Aperiodicity follows by noting that the $P_{x,x}(\epsilon) > 0$ for any $x \in S$: the same action can be picked by the agents in consecutive time steps with positive probability and (3) permits picking the same mood variable again with positive probability. □

Lemma 4.1 implies that $P(\epsilon)$ has a unique stationary distribution which we denote, as in the previous section, by $\mu(\epsilon)$. It is also clear that $P(\epsilon)$ is a regular perturbation of $P(0)$ (the latter obtained by setting $\epsilon_t \equiv 0$ in the algorithm). Thus, by Theorem 2, $\mu(\epsilon) \to \mu(0)$ as $\epsilon \to 0$ where $\mu(0)$ is a stationary distribution of $P(0)$.

4.1 Stochastically Stable States: Support of $\mu(0)$

Definition 4.1. *Let*

$$C^0 = \{x \in S | m^x = \mathbf{1}\} \text{ and}$$
$$D^0 = \{x \in S | m^x = \mathbf{0}\}.$$

Lemma 4.2. *If for every $a \in \mathcal{A}$, $\mathcal{G}_c(a) \cup \mathcal{G}_I(a)$ is strongly connected, the recurrence classes of the unperturbed chain $P(0)$ are D^0 and the singletons $z \in C^0$.*

Proof. Consider transitions defined by the algorithm with $\epsilon = 0$. Consequently, in Step 1, $\tilde{m}_i = (m_{t-1})_i \cdot \prod_{j \in \mathcal{N}_i(a_{t-1})} (m_{t-1})_j$. Every $z \in C^0$ satisfies $\mathbb{P}(\mathbf{X}_{t+1} = z | \mathbf{X}_t = z) = 1$ since by (1) the same joint action a^z is picked w.p. 1 resulting in the same payoff which in turn results in execution of step 4.1. A state $y \in D^0$ is also constrained to evolve only in D^0 since (3) with $\epsilon = 0$ does not permit a transition to $m_i = 1$ for any i. Also, by Assumption 1 and transition rule (2), there is a positive probability of transitioning from any joint action profile in $\mathcal{A}$ to any other. Thus D^0 and each $z \in C^0$ are recurrence classes of $P(0)$.

Now consider a state $x \in S \setminus \{C^0 \cup D^0\}$. Let $J^x = \{i | m_i^x = 0\} \subset \{1, ..., N\}$ be the non-empty subset of discontent agents. Since $\mathcal{G}_c(a^x) \cup \mathcal{G}_I(a^x)$ is strongly connected, there must exist an outward edge in this graph from at least one vertex $i' \in J^x$ to a vertex $i \notin J^x$. Two cases arise.

1. If (i', i) belongs to $\mathcal{G}_I(a^x)$, then $\exists\ b_{i'} \in \mathcal{A}_{i'}$ that agent i' can pick with positive probability according to (2) and due to Assumption 1, such that $u_i(a^x) \neq u_i(a^x_{-i'}, b_{i'})$. This changes the mood variable of agent i from 1 to 0 in step 4.2.

2. If (i', i) belongs to $\mathcal{G}_c(a^x)$, then agent i receives a 0 from its in-neighbor i' in step 4.1. Agent i sets $\tilde{m}_i = 0$ and consequently m_i is set to 0 in Step 4.3.

Thus x transitions to x' such that $|J^x| < |J^{x'}|$ i.e. at least one more agent becomes discontent with positive probability. Since there are finite number of agents and because of the strong connectivity assumption, repeating this argument for x' yields that there is a positive probability of transitioning from x to D^0; all agents eventually become discontent. Hence no state in $S \setminus \{C^0 \cup D^0\}$ is in a recurrence class. Since all these transitions are according to $P(0)$, we also have for any $y \in D^0$,

$$\rho(x, y) = 0,\ \forall\ x \in S \setminus C^0. \tag{10}$$

□

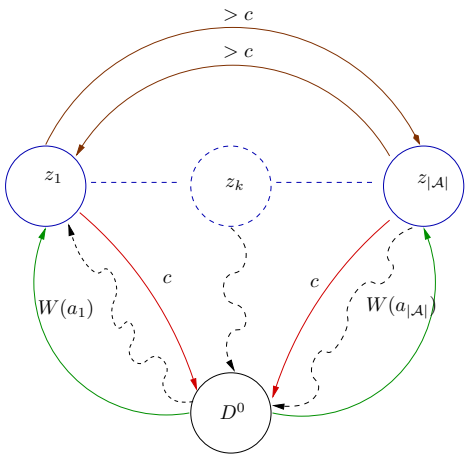


Figure 2: The circles represent recurrence classes of $P(0)$ and weights on the arrows the corresponding $\rho(\cdot, \cdot)$s. If $W(a^{z_1}) = W^*$, the zig-zag lines represent edges in the minimum resistance tree rooted at z_1.

Guided by Theorem 2, we now proceed to calculate the stochastic potential of the recurrence classes of $P(0)$. But first we organize some calculations in the following lemma.

Lemma 4.3. *Under the same assumption as Lemma 4.2, for any $y \in D^0$ and $z \in C^0$,*

$$\rho(x, y) = c, \ \forall \ x \in C^0, \tag{11}$$

$$\rho(y, z) = W(a^z), \tag{12}$$

$$\rho(x, z) \leq W(a^z), \ \forall \ x \in S \setminus C^0, \tag{13}$$

$$\text{and } \rho(z', z) > c, \ \forall \ z' \in C^0, \ z' \neq z. \tag{14}$$

Proof. Consider $x \in C^0$. For any i, a change in m_i^x from 1 to 0 must involve some agent picking a different action. From (1), such a change by an agent has resistance c. Therefore $\rho(x, y) \geq c$. Once such an action is picked by a content agent, its mood can change to 0 with a zero resistance transition in step 4.2. From (10), this intermediate state can now move to $y \in D^0$ with zero resistance. Thus (11) is proved.

For any $y \in D^0$, any $h(y \to z)$ must undergo N discontent to content transitions according to (3) (because $m_i^y = 0 \Rightarrow \tilde{m}_i = 0$ in the ensuing iterate). Thus $\rho(y, z) \geq \sum_{i=1}^{N} u_i(a^z) = W(a^z)$. Since $\tilde{m}_i = 0$ for all i, from (2), all agents can collectively pick a^z via a zero resistance transition and become content with resistance $W(a^z)$. Thus there exists an $h(x \to z)$ such that $r(h) = W(a^z)$. Hence $\rho(x, z) = W(a^z)$ establishing (12). Then (13) follows in view of (10): consider $h(x \to y)$ followed by $h(y \to z)$.

For $z' \in C^0$, $z' \neq z$, there exists at least one agent playing different actions in the two states. Thus any $h(z' \to z)$, must involve this content agent picking a different action with resistance c (from (1)) and becoming content with resistance β_2 or $u_j(a^z)$ by step 4.2 or 4.3 respectively. Hence (14) is established. □

From Lemma 4.2, there are exactly $|\mathcal{A}| + 1$ recurrence classes of $P(0)$; $|\mathcal{A}|$ corresponding to each $a \in \mathcal{A}$ (i.e. each element of C^0) and one for the set D^0. Let $\{z_1, ..., z_{|\mathcal{A}|}\}$ be an enumeration for C^0.

Lemma 4.4. *Under the same assumption as Lemma 4.2, the stochastically stable set is $\{z_i \in C^0 | W(a^{z_i}) = W^*\}$.*

Proof. We will show that the minimum potential z-trees in $\mathcal{G}_{RC}$ are rooted at $\{z_i \in C^0 | W(a^{z_i}) = W^*\}$. The claim then follows as a consequence of Theorem 2. Consider Figure 2 which depicts edges of the $\mathcal{G}_{RC}$ corresponding to the algorithm. The resistances between the recurrence classes are as calculated in Lemma 4.3. For $z_i \in C^0$, consider any z_i-tree in this graph. Any such tree must have one outward edge from each of the $(|\mathcal{A}| - 1)$ states in C^0 and an outward edge from D^0. The former contribute a resistance of at least $(|\mathcal{A}| - 1)c$ and the latter $W(a)$ for some $a \in \mathcal{A}$, hence the least possible stochastic potential for a tree rooted at a state in C^0 is $(|\mathcal{A}| - 1)c + W^*$. It is possible to construct such a tree for any state $z_i \in C^0$ with $W(a^{z_i}) = W^*$ as denoted by the zig-zag lines in Figure 2. By a similar argument, the stochastic potential of D^0 is $|\mathcal{A}| \cdot c$. Since $c > W^*$, any state $z_i \in C^0$ with $W(a^{z_i}) = W^*$ corresponds to the least stochastic potential state.

All that is left to prove is that any state $z_i \in C^0$ with $W(a^{z_i}) > W^*$ has stochastic potential greater than $(|\mathcal{A}| - 1)c + W^*$. Again, consider any z_i-tree. If the outgoing edge from D^0 is incident on a state $z_k \in C^0$, $z_k \neq z_i$ with $W(a^{z_k}) = W^*$, there must exist at least one edge from a state in C^0 to z_i and $(|\mathcal{A}| - 2)$ outward edges from the rest of elements of C^0 to complete the tree. Such a tree has resistance strictly greater than $(|\mathcal{A}| - 1)c + W^*$ because of the link between two states in C^0. Else, if the outgoing edge from D^0 is to a state $z_k \in C^0$ with $W(a^{z_k}) > W^*$, the outward edges from the $(|\mathcal{A}| - 1)$ states in C^0 result in a resistance at least greater than $(|\mathcal{A}| - 1)c + W(a^{z_k})$. □

4.2 Proof of Theorem 1

We return to the analysis of the nonhomogeneous Markov chain, $\mathbf{P}$, induced by the algorithm with the annealing schedule $\{\epsilon_t\}_{t \in \mathbb{N}}$. The proof relies on noting that if the annealing schedule satisfies $\sum_{t=1}^{\infty} {\epsilon_t}^c = \infty$, $\mathbf{P}$ is strongly ergodic with the limiting distribution having support over states with efficient actions as described by Lemma 4.4.

Lemma 4.5. *Under the same assumption as Lemma 4.2, for the nonhomogeneous Markov chain defined on S by the algorithm, κ as defined in* (7) *equals c.*

Proof. From Lemma 4.2 and (7),
$\kappa = \min\{\{CR(z)\}_{z\in C^0}, CR(D^0)\}$. For any $z \in C^0$, from (13) and (14), $CR(z) > c$. From (10) and (11), $CR(D^0) = c$. Hence $\kappa = c$. □

Proof of Theorem 1. The Assumption of Lemma 4.2 is included in the statement of the Theorem. All transition probabilities in the algorithm of section 2.2 belong to $\mathfrak{L}$; thus Assumption 2 holds. For any $y \in D^0$ and $z \in C^0$, $P_{y,y}(0) > 0$ and $P_{z,z}(0) > 0$. Hence the recurrence classes of the unperturbed Markov chain are aperiodic and, from Theorem 5 and Lemma 4.5, the chain is strongly ergodic if

$$\sum_{t=1}^{\infty} {\epsilon_t}^c \quad = \quad \infty. \tag{15}$$

Next, for any initial distribution η_0 on S and any subset $\tilde{S} \subset S$, $\mathbb{P}(\mathbf{X}_t \in \tilde{S}) = \sum_{j\in\tilde{S}} (\eta_0 \mathbf{P}^{(1,t)})_j$. Since (15) implies *SE* with limiting distribution $\mu(0)$ as in Theorem 2 and from the definition of *SE*, $\lim_{t\to\infty} \mathbb{P}(\mathbf{X}_t \in \tilde{S}) = \sum_{j\in\tilde{S}} \mu_j(0)$. Let $\tilde{S} = \{x \in S | W(a^x) = W^*, m^x = \mathbf{1}\}$, then in view of Lemma 4.4,

$$\lim_{t\to\infty} \mathbb{P}[\mathbf{a}_t \in \mathcal{A}^*] = 1.$$

□

5 Numerical Simulations and Conclusions

To illustrate the setup, we consider the payoff structure in Table 1. As explained earlier, the agents do not know this structure, they can only pick actions simultaneously from $\mathcal{A}_i = \{l, h\}$ for $i = 1, 2, 3$ and measure the resulting payoffs. Let the agents implement the algorithm of section 2.2 to learn the welfare minimizing state. In this example, it is clear (at the level of the system designer) that agent 3's payoff depends only on its own actions and are unaffected by actions of agents 1 and 2. Thus, the interaction graph $\mathcal{G}_c(a)$ is not strongly connected for any a since there is no incident edge on 3. The plot in Figure 3 shows MATLAB simulation runs for the algorithm. The plot on the top is for when $\mathcal{G}_c$ is empty (thereby violating the hypothesis of Theorem 1) and the one below for when $\mathcal{G}_c(a)$ consists of the directed edge $(1, 3)$ for all $a \in \mathcal{A}$ (thereby satisfying the hypothesis of Theorem 1). Observe that the instances where the product of the mood variables equals 1 can be interpreted as "when the agents have learned".

In the first case, since agent 3 cannot be influenced, it seems to learn to play h which offers it an individually rational lower payoff of $\frac{1}{10}$ as opposed to playing l with a payoff $\frac{1}{4}$. Quite intuitively, agents 1 and 2 seem to learn to play (h, h): the welfare minimizing action of the 'sub-game' where 3 chooses h with welfare $= \frac{2}{5}$ which is suboptimal to the global welfare minimal of $\frac{9}{20}$ achieved with (l, l, l). In the second case, it is observed that when the link $(1, 3)$ is added to $\mathcal{G}_c$, the agents lean to play the welfare minimal (l, l, l).

Table 1: Payoff structure of a three agent system

Agent 3 →	l	l	h	h
Agent 2 →	l	h	l	h
Agent 1				
l	$(\frac{1}{10}, \frac{1}{10}, \frac{1}{4})$	$(\frac{1}{2}, 1, \frac{1}{4})$	$(\frac{3}{4}, \frac{3}{4}, \frac{1}{10})$	$(1, \frac{1}{2}, \frac{1}{10})$
h	$(1, \frac{1}{2}, \frac{1}{4})$	$(\frac{3}{4}, \frac{3}{4}, \frac{1}{4})$	$(\frac{1}{2}, 1, \frac{1}{10})$	$(\frac{1}{4}, \frac{1}{4}, \frac{1}{10})$

An interesting question is how does the performance of the algorithm depend on $\mathcal{G}_I$ and $\mathcal{G}_c$. We present results of some numerical experiments to motivate such questions. First of all, we quantify performance as the percentage of times the welfare minimal actions are played in a fixed duration. To analyze the effect of $\mathcal{G}_I$, consider N identical agents with $\mathcal{A}_i = \{0.1, 1\}$. Let us endow agent i with utility function $u_i(a) =$

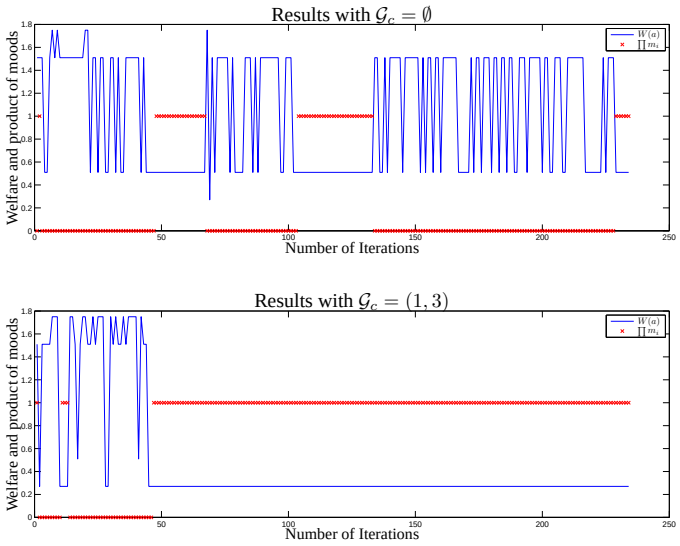


Figure 3: Simulation resuts for the three agent experiment; welfare plotted in blue solid lines and product of moods plotted with red crosses.

$\frac{1}{1+2q}\sum_{j=i-q}^{i+q} a_j$, where the operations in the limits of the summation are mod N. The welfare function $W(a) = \sum_{i=1}^{N} a_i,\ \forall\ q = 1, ..., N/2$, with a unique minimum at $(0.1, ..., 0.1)$. Notice that, for each q, $\mathcal{G}_I^q(a)$ is the same for all $a \in \mathcal{A}$ and can be varied by varying $q = 1, ..., N/2$. In Table 2 we report the performance for the case $N = 10$, $c = 1.1$, $\beta_1 = \beta_2 = 0.5$, $\epsilon_t = \frac{1}{\sqrt[6]{t}}$ and the algorithm is allowed to run while $\epsilon_t > 10^{-4}$. The algorithm is implemented on MATLAB and the reported numbers are averaged over 100 runs for each value of q and the standard deviation is reported as well. Since a greater value of q can be interpreted as more complex interaction, the result seems to agree with the intuitive notion that the speed of convergence reduces with increased interaction complexity. To study the effect of $\mathcal{G}_c$, we use the same set up

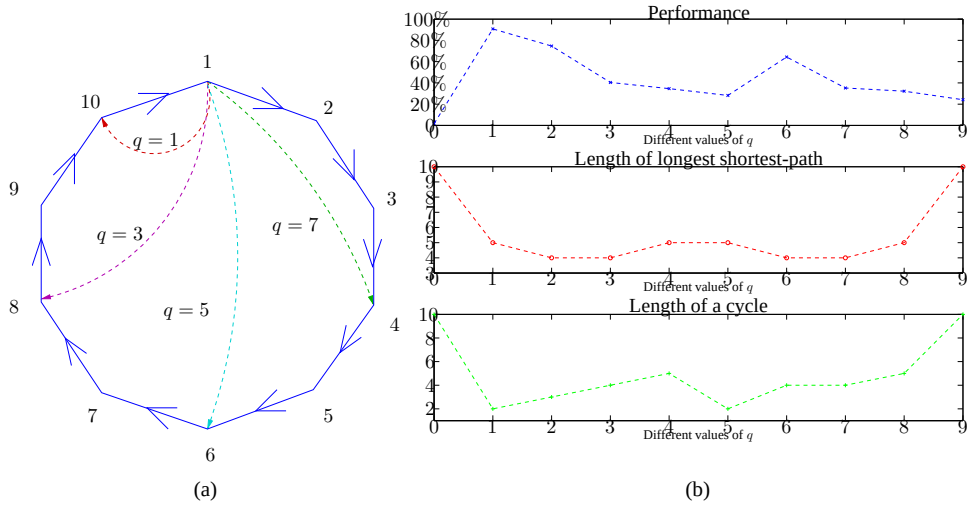
Table 2: Effects of varying $\mathcal{G}_I^q$

q	Performance	Std. Deviation
1	93.78%	2.92%
2	62.21%	7.84%
3	48.15%	9.71%
4	45.35%	11.11%
5	44.31%	11.79%

with $u_i(a) = a_{i-1}$ for $i = 2, ..., N$ and $u_1(a) = a_N$. Thus $\mathcal{G}_I(a)$ is a directed ring for all $a \in \mathcal{A}$ (see Figure 4 (a)). Let directed edges $(i, i-q)$ (where subtraction is mod N) for all i constitute $\mathcal{G}_c^q(a)$ for all $a \in \mathcal{A}$. Let $G(q) = \mathcal{G}_c^q \cup \mathcal{G}_I$. The same experiment as before is carried out with different values of q; the performance measure, the length of the longest shortest-path (SP) in $G(q)$ and length of a cycle in $G(q)$ are plotted for values of q in $\{0, .., (N-1)\}$ in Figure 4 (b). The results suggest a heuristic: To improve performance, pick $\mathcal{G}_c(a)$ to comprise of edges exactly opposite of $\mathcal{G}_I(a)$ and thereby reducing the cycle lengths.

Before concluding we wish to point out that the free parameters $\beta_1, \beta_2 > 0$ in the algorithm can be tuned according to the application to get improved performance; for instance setting $\beta_2 = \max\{0, (u_t^{mes})_i - \overline{u}_i\}$ can allow the agent to remain content when the change in payoff is in the desired direction in step 4.2. These parameters can also be interpreted, in some sense, as weights on the communication and interaction graphs as larger values of β_1 and β_2 correspond to agents being more sensitive to the information from the communication and interaction graphs respectively (see steps 1.3 and 4.2 of the algorithm).

An important open question is determining the rate of convergence of the algorithm. One way to answer this question is to calculate the rate of convergence of $||\eta_t - \mu(0)||$ as $t \to \infty$, where η_t is the density of



(a) (b)

Figure 4: Effects of varying $\mathcal{G}_c$ for $N = 10$. (a) The blue solid arrows represent $\mathcal{G}_I$ and the dotted arrows denote the edge $(1, 1-q)$ in the corresponding $\mathcal{G}_c^q$. (b) Plot of performance, longest SP and cycle length w.r.t. different values of q.

$\mathbf{X}_t = [\mathbf{a}_t, \mathbf{m}_t]$. This is difficult since the Markov chain is nonhomogeneous and the best results we know in such situations are for the simulated annealing algorithm [19]. We will address this issue along such lines in future work. We expect that such an investigation will also shed light on the issue of how the communication and interaction graphs play a role in speed of convergence.

References

[1] J. R. Marden, H. P. Young, and L. Y. Pao, "Achieving Pareto optimality through distributed learning." submitted for journal publication, 2011.

[2] A. Menon and J. S. Baras, "Convergence guarantees for a decentralized algorithm achieving Pareto optimality." To appear in Proceedings of the 2013 American Control Conference.

[3] 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.

[4] N. Li and J. R. Marden, "Designing games for distributed optimization," in *Proc. of 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC), 2011*, pp. 2434–2440, IEEE, 2011.

[5] M. Zhu and S. Martinez, "Distributed coverage games for mobile visual sensor networks," in *Proceedings of the 48th IEEE Conference on Decision and Control*, pp. 175–180, Shanghai, China, Dec. 15-18 2009. Also to appear in *SIAM Journal on Control and Optimization*.

[6] E. Altman and Z. Altman, "S-modular games and power control in wireless networks," *IEEE Transactions on Automatic Control*, vol. 48, no. 5, pp. 839–842, 2003.

[7] J. R. Marden and J. S. Shamma, "Revisiting log-linear learning: Asynchrony, completeness and payoff-based implementation," *Games and Economic Behavior*, 2012.

[8] 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, Feb 2009.

[9] H. P. Young, "Learning by trial and error," *Games and Economic Behavior*, vol. 65, no. 2, pp. 626–643, 2009.

[10] J. R. Marden, S. D. Ruben, and L. Y. Pao, "A model-free approach to wind farm control using game theoretic methods," 2012. submitted for journal publication.

[11] N. Ghods, P. Frihauf, and M. Krstic, "Multi-agent deployment in the plane using stochastic extremum seeking," in *Proc. of 49th IEEE Conference on Decision and Control (CDC), 2010*, pp. 5505–5510, IEEE, 2010.

[12] X. Tan, W. Xi, and J. S. Baras, "Decentralized coordination of autonomous swarms using parallel Gibbs sampling," *Automatica*, vol. 46, no. 12, pp. 2068–2076, 2010.

[13] W. Xi, X. Tan, and J. S. Baras, "Gibbs sampler-based coordination of autonomous swarms," *Automatica*, vol. 42, no. 7, pp. 1107–1119, 2006.

[14] H. P. Young, "The evolution of conventions," *Econometrica: Journal of the Econometric Society*, pp. 57–84, 1993.

[15] J. Robles, "Evolution with changing mutation rates," *Journal of Economic Theory*, vol. 79, no. 2, pp. 207–223, 1998.

[16] M. Pak, "Stochastic stability and time-dependent mutations," *Games and Economic Behavior*, vol. 64, no. 2, pp. 650–665, 2008.

[17] P. Brémaud, *Markov Chains: Gibbs Fields, Monte Carlo Simulation and Queues*, vol. 31. Springer-Verlag, 1999.

[18] D. L. Isaacson and R. W. Madsen, *Markov Chains, Theory and Applications*. RE Krieger Publishing Company, 1976.

[19] D. Mitra, F. Romeo, and A. Sangiovanni-Vincentelli, "Convergence and finite-time behavior of simulated annealing," *Advances in Applied Probability*, pp. 747–771, 1986.