SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

March 2021, Vol. 64 132206:1–132206:14
https://doi.org/10.1007/s11432-019-2825-y

The greedy crowd and smart leaders: a hierarchical strategy selection game with learning protocol

Linghui GUO[1,2], Zhongxin LIU[1,2*] & Zengqiang CHEN[1,2]

[1] *College of Artificial Intelligence, Nankai University, Tianjin* 300350*, China;*
[2] *Tianjin Key Laboratory of Intelligent Robotics, Nankai University, Tianjin* 300350*, China*

Received 22 October 2019/Revised 24 December 2019/Accepted 5 February 2020/Published online 7 February 2021

Abstract In this paper, a general resource distribution game with a hierarchical structure on the bipartite graph is proposed. In this system, the game is divided into two interacting levels, the agent level and the group level, with negotiations taking place on both levels. Each agent can belong to multiple groups, resulting in a system topology with a bipartite structure. On the agent level, decisions are based on the greedy principle, with the game being a state-based potential game. In contrast, some participants on the group level behave more "smartly" and are more likely to adopt a sophisticated strategy maximizing their personal interest. Strategies on both levels are based on distributed protocols, and the social welfare increases as the system approaches a Nash-equilibrium point. The designed protocols are theoretically analyzed from stability and efficiency. Furthermore, a reinforcement learning algorithm is introduced in the group level, where the smarter players are allowed to refine their strategies in the multi-step decision-making process by learning from historic game outcomes. In theory and according to simulations, agents with the learning behavior improve not only their personal interest but also the efficiency of the systemic resource distribution.

Keywords multi-agent system, reinforcement learning, game theory, complex network, bipartite graph

Citation Guo L H, Liu Z X, Chen Z Q. The greedy crowd and smart leaders: a hierarchical strategy selection game with learning protocol. Sci China Inf Sci, 2021, 64(3): 132206, https://doi.org/10.1007/s11432-019-2825-y

1 Introduction

The theory of nature selection is one of the most fundamental research topics inspired by the evolution of animals. Its application seems ubiquitous, ranging from natural communities to human societies and some artificial systems [1–8]. One key idea of nature selection is that a group evolves as a whole, and the population structure plays a much more important role than just a gathering of individuals. Many studies focus on the cooperation and altruism behaviors in evolutionary systems. It has been suggested that, the strategy of cooperation is more preferred by individuals in certain population structures than others [2, 5, 8], whereas social properties such as reputation, reciprocity and kin selection have certain effects on individual behaviors within a system [6, 9–12]. The majority of evolutionary cooperative models are based on complex networks. In traditional networks, all interactions in a system are modeled as pairwise connections between vertices, each of which denotes an individual. Despite its simplicity, the network approach facilitates the study of multiple individual interactions in complex systems [2, 6, 13, 14].

However, the pairwise connections in traditional networks cannot represent some complicated structures. Hence, networks with more sophisticated architectures such as multi-layer, multiplex and bipartite networks have gained a lot of attention in recent years [15–18]. The bipartite network is proved to be an intuitive and elegant model for systems such as citation and actor networks that feature triple or multiple connections between individuals [15–18].

Any economy system is essentially an ecosystem in many ways. In particular, the development of markets in social systems is the result of mutual selections between multiple producers and consumers,

[*] Corresponding author (email: lzhx@nankai.edu.cn)

employers and employees. Different behaviors of partners are driven by self-interest and the market regulation, also known as the invisible hand, which in theory helps non-cooperative crowds to achieve a good commercial order and an optimal resource distribution. However, this mechanism has not been thoroughly studied yet. In the field of artificial system optimization, many solutions are inspired by the principles of natural systems [19, 20]. While designing a centralized control protocol is essential for some gigantic systems, distributed protocols are often more robust and cheaper to design. Inspired by the market law, models based on game theory are widely adopted in designing efficient and decentralized control protocols. The target in a game-based model is to maximize the global optimization target by encouraging each agent to maximize its private utility function. The strategic set of agents can be complicated, and the relationship between individual agents is more than just cooperation or defection. On the one hand, all agents have a common global optimization target, which is the sum of all local utility functions. If the game is a convex potential game [21], a Nash-equilibrium point satisfies the first-order Karush-Kuhn-Tucker (KKT) condition [22–26]. Even if agents are non-cooperative, and only concern about their personal utility functions, their selfish behavior promotes the overall welfare, pushing the overall system to a global optimization solution. Some non-convex systems can be modeled as state-based potential games, allowing the equilibrium points to be adjusted by tuning model parameters, avoiding the solution being trapped in local optimal solutions [26]. On the other hand, agents would compete for scarce resources, and their selfish behavior may harm the social welfare, which is familiarly known as "the tragedy of the commons". One widely studied challenge is the resource distribution problem, derived from the real-world economic dilemma how selfish agents must negotiate to allocate the scarce resources efficiently. Agents typically use greedy algorithms to maximum the local utility functions, making it difficult to guarantee an efficient global solution. Some promising studies focus on the Lagrange method and bidding protocol. The transparent auction is a distributed greedy algorithm guaranteeing the resource consumption boundary, which deserves a further study [27, 28]. One of the problems with this algorithm is that the result of the auction does not guarantee the global efficiency even if the Nash-equilibrium is reached [27]. When the population is small, a weight modification has to be imposed on the price function of each agent, which makes the algorithm not fully distributed. The idea of the weight control method is also applicable to some distributed solutions of combinatorial optimization problems. For example, the minimum vertex cover problem in graphs is extended to the weighted vertex cover problem [29]. This constraint can be neglected when a system is sufficiently large. However, not much research has been done toward theoretical analysis of the relationship between the system size and its efficiency.

The bipartite graph opens up a new way of studying complex game theoretic systems, while its combination with the bidding and other restricted optimization protocols has not been fully studied yet. In the bipartite graphs, vertices have two hierarchies, and their interactions are more complicated than in traditional networks. In many previous studies, agents are considered to be either greedy (i.e., only concerned about the current information) or smart (i.e., learning from history). However, in a game with hierarchical structure, agents in different layers may have different behaviors. If an agent is smart, it realizes that the strategies adopted in current step may have long-term effects to the future payoff, and while the long-term utility function cannot be calculated directly, it can be estimated from the history. Reinforcement learning algorithms have been proved to be an efficient tool for such multi-step games, especially in complex systems [30–35]. Hence this study focuses on applying reinforcement learning to games that can be represented as bipartite graphs.

In particular, the model proposed in this paper is based on the state-based potential game with a hierarchical structure. On the lower level, agents are greedy and short-sighted. On the upper level, the state parameters are adjusted by "smart leaders" using a reinforcement learning algorithm. Hence, the topology of the system is considered to be a bipartite graph with two levels, the agent level and the group level.

The rest of the paper is organized as follows. The game is proposed and discussed in Section 2 first. The protocols for the game on both levels are defined, and the learning behaviors of the game participants are designed next. Then, the stability and efficiency of the employed distributed algorithms are studied. Finally, the simulation results are presented to verify the improvement of the systematic efficiency brought by the "smart leaders".

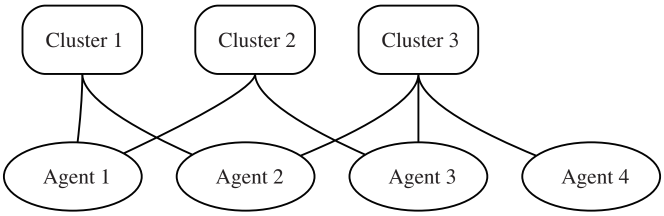

Figure 1 Connection topology of the proposed model, where groups (upper level) and agents (lower level) form a bipartite graph, and the links exist only between these two sets.

2 Model definition and main results

2.1 Game introduction

The model proposed in this study can be presented as a bipartite graph, with its vertices divided into two sets: agents and groups. Let $V^c = \{v_1^c, v_2^c, \ldots, v_{n_g}^c\}$ denote the set of group vertices, and $V^a = \{v_1^a, v_2^a, \ldots, v_{n_a}^a\}$ denote the set of agent vertices. The edge set is denoted by $E \subseteq V^a \times V^c$, with each edge associating an agent with a group. All agents connected to the same group vertex belong to the same group. For each agent $i \in V^a$, $N_i = \{k \in V^c \mid (i,k) \in E\}$ is called its neighbor set, which comprises all groups it belongs to. For each group $k \in V^c$, $N_k = \{i \in V^a \mid (i,k) \in E\}$ is called its neighbor set, which comprises all agents it contains. The proposed model is illustrated in Figure 1.

Agents are free to choose their strategies from a strategy set $A \triangleq \{a_1, a_2, \ldots, a_{n_A}\}$; mixed strategies are allowed. For each agent $i \in V^a$, the preference for strategy $a \in A$ is p_i^a, where $0 \leqslant p_i^a \leqslant 1$ and $\sum_{a\in A} p_i^a = 1$. The mixed strategy of an agent i is denoted by $p_i \triangleq [p_i^{a_1}, p_i^{a_2}, \ldots, p_i^{a_{n_A}}]^{\mathrm{T}}$.

For each group $k \in V^c$, the payoff depends on the strategies of all the agents it contains. Each strategy a has an independent contribution U_k^a to the group's utility function $U_k(\cdot)$, which is defined as

$$U_k(p^k) \triangleq \sum_{a\in A} U_k^a(p_k^a), \tag{1}$$

where $p_k \triangleq \{p_i \mid i \in N_k\}$ and $p_k^a \triangleq \{p_i^a \mid i \in N_k\}$ contain only the local information within N_k.

For each agent $i \in V^a$, its utility function $u_i(\cdot)$ is the linear combination of the payoff of all neighbor groups:

$$u_i(p) \triangleq \sum_{k\in N_i} w_k^a U_k(p_k) = \sum_{k\in N_i} \sum_{a\in A} w_k^a U_k^a(p_k^a), \tag{2}$$

where the weight w_k^a is a non-negative value. As w_k^a increases, U_k^a weights more in the agent's utility function, and the agent's preference is affected by its utility function accordingly. For each strategy $a \in A$, all its relevant weights sum up to the positive value K, namely

$$\sum_{k\in V^c} w_k^a = K. \tag{3}$$

According to Eq. (3), each strategy has equivalent "importance" from the systematic view. However, the weights of strategies in each group may be different, resulting in the agents in different groups having different preferences of strategies. When each w_k^a is fixed, the outcome depends only on the actions of the agents, and the game can be presented as a one-layer network. However, if the weights vary over time, or they are determined by group nodes with some competitive protocols, then the game's outcome is determined by the actions of the nodes in both layers, and the game has a two-level hierarchy.

With this model, a variety of nonlinear problems can be addressed via the choice of utility functions $U_k(\cdot)$. For example, Eq. (4) can present a model similar to the Kelly's route competition game [19], if we properly define the limit parameter $C_i^a > 0$ of the resource i and the weight parameter $h_k^a \geqslant 0$ of the route a:

$$U_k^a(p_k^a) = h_k^a \min_{i\in N_k} \{C_i^a p_i^a\}. \tag{4}$$

Alternatively, we can get the maximum coverage problem represented as

$$U_k^a(p_k^a) = \max_{i\in N_k} C_{ik}^a p_i^a, \tag{5}$$

or the probability consensus problem represented as

$$U_k^a(p_k^a) = \prod_{i\in N_k} p_i^a. \tag{6}$$

In the probability consensus problem Eq. (6), U_k^a can be viewed as the probability of the route k to be "connected" under some channel $a \in A$.

One interesting property of the model Eq. (6) is that for each group k, $U_k(p_k) = 1$ if and only if any local consensus state is reached (i.e., $\exists a \in A$, $\forall i \in N_k$, $p_i^a = 1$). In any other cases, the reward is inferior than in the consensus case because

$$1 = \prod_{i\in N_k}\left(\sum_{a\in A} p_i^a\right) > \sum_{a\in A}\prod_{i\in N_k} p_i^a = U_k(p_k). \tag{7}$$

Hence, the consensus state is the optimal condition for both levels. However, a global consensus may not be easily reached providing that independent decision makers have different neighbors, while the individuals with different preferences compete with each other. To analyze these complex situations, we propose a two-level model comprising the agent level, where members make direct decisions using a greedy behavior, and the group level, where members try to indirectly influence the neighbor agents in the future games and exhibit a learning behavior. These two levels are described below in turn.

2.2 Problem formation on the agent level

On the agent level, given a set of all group weights and all strategies of other agents, if an agent i adopts a pure strategy $a \in A$, the agent's payoff is

$$u_i(a, p_{-i}) = \sum_{k\in N_i} w_k^a \left(\prod_{j\in N_k, j\neq i} p_j^a\right), \tag{8}$$

where p_j denotes the strategy of player j, who belongs to the neighbor set of i, and p_{-i} denotes the current strategies of all members in the agent level except agent i. When the agent i adopts a mixed strategy p_i, the agent's utility function can be written as

$$u_i(p_i, p_{-i}) = \sum_{a\in A} p_i^a u_i(a, p_{-i}). \tag{9}$$

In other words, each utility function $u_i(p)$ is a linear combination of the agents' pure strategy payments.

At each step, the agent i might have the motivation to adjust its strategy p_i to achieve a higher reward. The system's Nash-equilibrium is the state where no agent has such a motivation. One stable Nash-equilibrium point is a solution of p_is satisfying the following condition:

$$p_i = \arg\max_{p_i} u_i(p_i, p_{-i}) \quad \text{s.t.} \sum_{a\in A} p_i^a = 1. \tag{10}$$

For any state deviating from the Nash-equilibrium, the strategy of each agent p_i may be improved using its local information. For a given agent, since its probabilities to adopt each strategy sum up to 1, the set of values $[q_i^a] \subset R$ can be defined to satisfy the following conditions:

$$p_i^a = \frac{q_i^a}{\sum_b q_i^b}, \tag{11}$$

$$\dot{q}_i^a = f(q_i^a, u_i^a), \tag{12}$$

where q_i^as are non-negative and the function f satisfies $f(0, u_i^a) = 0$. One reasonable choice is to set $f(q_i^a, u_i^a) = q_i^a u_i^a$, which would result in the p_i being updated according to the following replicator equation:

$$\dot{p}_i^a = \alpha \sum_{b\in A} p_i^b p_i^a (u_i(a, p_{-i}) - u_i(b, p_{-i})) = \alpha p_i^a (u_i(a, p_{-i}) - u_i(p_i, p_{-i})) = \alpha p_i^a(t) g_i^a(t), \tag{13}$$

where $g_i^a(t) \triangleq u_i(a, p_i(t)) - u_i(p_i(t))$. Eq. (13) has the formation of the replicator dynamic equation; hence the solution would remain on the simplex $\sum_{a\in A} p_i^a(t) = 1$, and if $0 \leqslant p_i(0) \leqslant 1$, the continuous dynamic equation guarantees $0 \leqslant p_i(t) \leqslant 1$ for any $t > 0$.

In large systems, agents typically update their states and strategies asynchronously. At the beginning of each interval $t_n = nT$, the agent i has the opportunity $0 < \delta \leqslant 1$ to update itself during the interval $[t_n, t_{n+1})$. When the system is large and δ is sufficiently small, the update can be viewed as asynchronous. The asynchronous update sequence $[i_{t_1}, i_{t_2}, \dots]$ of all time intervals is called the agents' updating sequence.

For the updating agent $i = i_{i_n}$ at time $t \in [t_n, t_{n+1})$, its personal profit is achieved greedily by modifying the strategy according to the replicator dynamic in Eq. (13):

$$\dot{p}_i^a(t) = \alpha \sum_{b\in A} p_i^a(t) p_i^b(t) g_i^{ab}, \tag{14}$$

where the parameter

$$g_i^{ab} \triangleq \sum_{k\in N_i} \left(w_k^a U_k^a(a, p_{-i}(t_n)) - w_k^b U_k^b(b, p_{-i}(t_n)) \right)$$

is viewed as a constant during the time interval $[t_n, t_{n+1})$.

Under this dynamic setup, each agent makes decisions only based on local information and in a distributed manner. As the greedy motivation of each agent does not necessarily promote the global welfare, there is no guarantee that the system would get close to any equilibrium conditions. This leads to the first question of this study, whether the asynchronous update protocol under Eq. (14) guarantees the system to reach any Nash equilibrium point, given that the parameters of each group level are fixed.

2.3 Problem formation on the group level

According to the behaviors in the agent level, a higher/lower group weight $w_k^a \in W^a$ encourages/discourages the agents in a group k to select a strategy a. For the private interest of each group, the allocation of group weights does not directly affect its payoffs. However, since agents take these parameters into consideration according to Eq. (2), groups with higher weights are more likely to reach a local consensus. While the sum of weights of one strategy in all groups is fixed, its allocation is crucial for the competitiveness of that strategy. The game on the group level is similar to the board of Go, where each strategy has its own domain. Focusing on a strategy may expand its domain across the groups. Hence, more weights should be allocated to one group node if this node brings about larger marginal revenue. If the utility function of each member is known beforehand, a centralized control protocol can be more appropriate, with each parameter being allocated using a tailored optimization algorithm. However, the individual utility functions in most real systems are private information; hence, a distributed protocol is more practical in such situations. The task of the weight allocation problem is to find the efficient assignment of group weights $\{w_k^a \mid k \in V^c\}$ for each strategy a in a distributed manner.

First, we assume that each private payoff function $f_k^a(w_k^a)$ is known by each member, which is its estimation of the utility function in future game. The members can negotiate for the weight allocation by the means of "transparent auction" mechanism: considering the following general distribution problem with global constraints.

$$\max_{w^a \geqslant 0} F^a(w^a) \triangleq \sum_{k\in V^c} f_k^a(w_k^a) \quad \text{s.t. } \mathbf{1}^{\mathrm{T}} w^a = K^a, \tag{15}$$

where $w^a \triangleq [w^{a_1}, w^{a_2}, \dots, w^{a_{n_A}}]^{\mathrm{T}}$, and $K^a > 0$. At each time step, the auction requires each group k to offer a bid $s_k^a \geqslant 0$ for a portion of K^a. Group k finally gets a portion w_k^a of K^a, which is

$$w_k^a = \frac{s_k^a}{\sum_{h\in V^c} s_h^a} K^a. \tag{16}$$

The allocation w_k^a is proportional to its bid s_k^a. The bid also involves a cost $C_k^a(s_k^a)$. We assume that for each k, the profit function is smooth, increasing, upper-bounded and concave, while the cost function is smooth, increasing, and convex. The discounted private payoff is the benefit minus the cost:

$$J_k^a(s_k^a \mid s_{-k}^a) \triangleq f_k^a(w_k^a) - C_k^a(s_k^a). \tag{17}$$

The Nash-equilibrium of the auction is the condition in which each J_k^a reaches a local maximum when no other individuals change their choices, namely $s_k^a = \arg\max_{s\geqslant 0} J_k^a(s \mid s_{-k}^a)$, $\forall k \in V^c$.

Since bids are based on personal interest, the Nash-equilibrium is not necessarily a global optimal solution of $F^a(w^a)$. The global measurement $F_k^a(w^a)$ reveals the overall competitiveness of a strategy a. According to Eq. (15) its necessary optimal condition is: $w_k = 0$ or $f_k^{a\prime}(w_k) = \lambda$ where λ is a constant. This condition can be summarized as follows: for any $k, h \in V^c$,

$$w_k w_h = 0 \ \text{ or } \ |f_k^{a\prime}(w_k) - f_k^{a\prime}(w_h)| = 0. \tag{18}$$

Based on Eq. (18), an approximated solution is ϵ-close to the accurate one if there exists some $\epsilon > 0$ satisfying that for any pair of groups $h, k \in V^c$,

$$\hat{w}_k \hat{w}_h = 0 \ \text{ or } \ |f_k^{a\prime}(\hat{w}_k) - f_k^{a\prime}(\hat{w}_h)| \leqslant \epsilon. \tag{19}$$

In this way the scale of ϵ is the measure of the efficiency of the auction outcome. The smaller ϵ is, the better the allocation is, while the accurate global solution is just a special case of the ϵ-close solutions where the lower bound of ϵ is 0.

This entails the main concern on the group level, which is, whether the efficiency of the auction can be guaranteed in the sense of ϵ-close, given each utility function of the members, and if so, what condition should be satisfied to get an enough small ϵ value.

The questions formulated for the proposed two-level model are analyzed in Section 3 separately. For the agent level, the stability is proved using the conclusion of the state-based potential game. For the group level, the efficiency is proved to have a relation with the system size, while the upper bound of ϵ can be calculated.

3 Theoretical results

First of all, it can be proved that from the view of any agent in the updating sequence, its utility function is non-decreasing during its own updating. According to Eq. (14), when a single agent i in the system updates its strategy p_i, the utility function u_i changes as follows:

$$\dot{u}_i(t) = \sum_{a\in A} \dot{p}^a u_i(a, p_{-i}) = \alpha \sum_{a\in A}\sum_{b\in A} p_i^a(t) p_i^b(t) \left(u_i^a(a, p_{-i}) - u_i^b(b, p_{-i})\right)^2 \geqslant 0. \tag{20}$$

This proves that $u_i(\cdot)$ is non-decreasing, and the protocol is greedy from the perspective of agent i. Furthermore, if $\dot{u}_i = 0$, for any $a, b \in A$, either $p_i^a p_i^b = 0$ or $u_i^a(a, p_{-i}) = u_i^b(b, p_{-i})$ must be true, which means that u_i reaches the first order KKT point.

From the perspective of the whole system, the function defined as below can be proved to be a potential function:

$$V(p) = \sum_{k\in V^c}\sum_{a\in A} w_k^a U_k^a(p_k^a). \tag{21}$$

Since $0 \leqslant U_k^a \leqslant 1$ is true, $V(p)$ is a bounded function.

Theorem 1. Eq. (21) is the potential function for the game on the agent level. Furthermore, the temporal trace $V(t) \triangleq V(p(t))$ is a non-decreasing function and $\lim_{t\to 0} \dot{V}(t) = 0$.

Proof. To prove that $V(t)$ is a potential function, it is just necessary to prove that when one single agent alters its strategy, its utility function and the potential $V(p)$ are changed for the same value. It is obvious that $V(p)$ is a smooth function of p, and for any $i \in V^a$ and $a \in A$,

$$\frac{\partial V(p)}{\partial p_i^a} = \sum_{k\in N_i} \frac{\partial}{\partial p_i^a} w_k^a U_k^a(P_K^A) = \frac{\partial}{\partial p_i^a} u_i^a(p) = \frac{\partial}{\partial p_i^a} u_i(p).$$

Hence, any change about p_i^a would cause $V(p)$ and $u_i(p)$ to change for the same value. Thus it completes the proof that $V(p)$ is a potential function.

From Eq. (20) the temporal trace $V(t)$ can be derived as

$$\dot{V}(t) = \sum_{i\in V^a}\sum_{a\in A} \frac{\partial V}{\partial p_i^a} \dot{p}_i^a = \sum_{i\in V^a} \dot{u}_i(t) \geqslant 0. \tag{22}$$

Since the value of each parameter $0 \leqslant p_i^a \leqslant 1$ is bounded, it is easy to check that both the functions $\dot{p}(t)$, $V(p)$ are uniformly bounded. Furthermore, in each time interval $t \in [t_n, t_{n+1})$ where the updating agent is i, we have

$$\dot{V} = \dot{u}_i(t) = \sum_{a\in A} \dot{p}^a u_i(a, p_{-i}) = \alpha \sum_{a\in A}\sum_{b\in A} p_i^a(t)p_i^b(t)\left(u_i^a(a,p_{-i}) - u_i^b(b,p_{-i})\right)^2 \geqslant 0, \tag{23}$$

$$\begin{aligned}\ddot{V} = \ddot{u}_i(t) &= \sum_{a\in A} \dot{p}^a u_i(a, p_{-i}) = \alpha \sum_{a\in A}\sum_{b\in A} p_i^a(t)p_i^b(t)\left(u_i^a(a,p_{-i}) - u_i^b(b,p_{-i})\right)^2 \\ &= \alpha \sum_{a\in A}\sum_{b\in A}\left(\dot{p}_i^a(t)p_i^b(t) + p_i^a(t)\dot{p}_i^b(t)\right)\left(u_i^a(a,p_{-i}) - u_i^b(b,p_{-i})\right)^2 . \end{aligned}\tag{24}$$

All variables in Eq. (24) are uniformly bounded, so $\ddot{V}(t)$ is uniformly bounded in each time interval, which implies that $\dot{V}(t)$ is uniformly continuous during all the temporal intervals. Similar to the conclusion of the Barbalat's Lemma, if $V(t)$ is bounded, $\dot{V} \geqslant 0$ and is uniformly continuous at each temporal interval, then $\dot{V}(t)$ must converge to 0.

Remark 1. Barbalat's Lemma cannot be directly applied to Theorem 1 because $\dot{V}(t)$ is not continuous and there is an abrupt switch at the beginning of each time interval. However, the conclusion of Theorem 1 can still be achieved by resorting to the technique similar to the proof of Barbalat's Lemma. Since $\dot{V}(t)$ is uniformly continuous in each interval with the same parameters, if $\dot{V} \to 0$ does not hold, namely, there exist an $\epsilon_0 > 0$ and infinitely many updating intervals each of which contains at least a time point $t_h \in [t_{n_h}, t_{n_h+1})$ where $h = 1, 2, \ldots$ such that $\dot{V}(t_h) \geqslant \epsilon_0$, then a contradiction would be produced. Actually, since $V(t)$ is uniformly continuous in all intervals, a unique value $\delta > 0$ can be found, such that at each point $t_h \in [t_{n_h}, t_{n_h+1})$, and for any $t \in [t_{n_h}, t_{n_h+1})$ satisfying $|t - t_h| \leqslant \delta$, $\dot{V}(t) \geqslant \epsilon_0/2$ holds. The value can be set to satisfy $\delta < T$. For each time point t_h, in the inner of its time interval of length T, a neighbor domain $[\tau_h, \tau_h + \delta]$ can be found, where $\tau_h \leqslant t_h \leqslant \tau_h + \delta$, and $\dot{V}(t) \geqslant \epsilon_0/2$ always holds in this neighbor domain. This implies $V(\tau_h + \delta) - V(\tau_h) > \epsilon_0\delta/2$ for every h, and hence no upper boundary can be expected in regard to $V(t)$ when t gets infinitely large, which is a contradiction.

According to the properties of potential games, any local maximum point of the potential function $V(p)$ is a Nash-equilibrium. Furthermore, $V(t)$ is non-decreasing according to Eq. (22). Since any upper-bounded non-decreasing function must converge, $V(t)$ must be converge to some value, $V(t) \to V_0$. Since the updating of agents is asynchronous, the final state is not only associated with the initial state but also with the updating sequence. If the updating sequence is random, the system would approach an equilibrium state.

Theorem 2. If the updating sequence is totally random, and the agents update strategies asynchronously using Eq. (14), the system almost surely converges to the set of Nash-equilibrium points.

Proof. In the Nash-equilibrium state p^*, each agent must be in the stable or boundary condition; i.e., $\dot{u}_i(p^*) = 0$ must be true for any $i \in V^a$. The agent with the largest margin utility can be defined as $\dot{u}_{\max}(t) \triangleq \max_i \dot{u}_i(t)$. Since the updating sequence is random, the probability for agent i to be selected is $1/n_a$. Since $\dot{u}_i \geqslant 0$, $E(\dot{V}(t)) = \sum_i E(\dot{u}_i(t))/n_a \geqslant E(\dot{u}_{\max}(t))/n_a$. According to Theorem 1, $\lim_{t\to\infty} E(\dot{V}(t)) = 0$, so $\lim_{t\to\infty} E(\dot{u}_{\max}(t)) = 0$, and for any selected i, $\lim_{t\to\infty} \dot{u}_i(t) = 0$ is almost surely true. Furthermore, according to Eq. (20), $\dot{u}_{\max}(p)$ is the function of p, and is uniformly continuous in the domain, so it is almost surely true that p converges to the set of points where $\dot{u}_i = 0$, which is the set of Nash-equilibrium points.

Remark 2. Theorem 2 does not exclude some special conditions in which some agent $j \in V^a$ only appears finite times in the updating sequence, and there would be no guarantee that $\partial U_j/\partial p_j \to 0$ as $t \to 0$, which is required by the Nash-equilibrium condition. However, since the system is finite and the updating sequence is random, the probability for such event is only 0, and Theorem 2 is almost surely true.

Theorems 1 and 2 guarantee that since the replicator dynamic is applied, the state of the agent level would converge to and remain in the Nash-equilibrium, given the weight of each group is constant. However, when the allocation of weights is updated, the utility functions are modified, driving the system to a different equilibrium point. This is a kind of state-based potential game defined in the work of Marden [26]. In previous studies, the state of the system is switched based on the random Markov process. In this study, a reinforcement learning framework is applied by each group that leads to a higher overall welfare. Furthermore, the weights would be optimized at early stages of the game before any equilibrium state is approached, if the group nodes can repetitively learn from experience and the protocol is efficient. Although the agents are greedy, their actions are predicted by the sophisticated group members, and hence the final state is selected with caution prospectively.

In the group level, the following conditions are assumed to be true: (a) for any group $k \in V^c$, its payoff $f_k^a(w)$ is upper bounded by N_F, namely $0 \leqslant f_k^a(w) \leqslant N_F$ for any $w \geqslant 0$, and N_F is irrelevant to k; (b) each payoff function is smooth, satisfying $f_k^a(0) = 0$, and $\mathrm{d}f_k^a(w)/\mathrm{d}w$ is a positive, decreasing function when $w \geqslant 0$; (c) each group k shares the same linear cost function, namely $C_k^a(s) = Cs$, where C is a positive constant; (d) the sum of allocated weights K^a is large enough in respect to the system scale n_c. Moreover, there exists some constant $M_a > 0$, such that $K^a \geqslant M_a n_c$.

If the assumptions above are true, the efficiency of the group level can be guaranteed using the definition of ϵ-close. Theorem 3 reveals that the larger the system gets, the more efficiency it could guarantee.

Theorem 3. If the assumptions (a)–(d) are true, the allocation in group level is ϵ-close to the optimal condition of Eq. (15). Furthermore, ϵ can be infinitely small if the system scale n_c expands infinitely large. In precise, small ϵ can be selected such that $\epsilon \leqslant 2N_F/(M_a n_c)$.

Proof. To prove the theory, the essential point is that for any $k, h \in V^c$, the requirement of Eq. (19) should be satisfied. In the condition of $w_k = 0$ or $w_h = 0$, Eq. (19) is already true for any $\epsilon > 0$. On other conditions, i.e., $w_k w_h > 0$, without losing of generality, we assume that $s_k > s_h > 0$ for the sake of simplicity. Since the function $f_k^a(w)$ is concave and $f_k^a(0) = 0$, for any $w > 0$, there exists some $0 < \psi < w$ such that $f_k^{a\prime}(\psi) = f_k^a(w)/w$. Since $f_k^{a\prime}(w)$ is decreasing, $f_k^{a\prime}(\psi) \geqslant f_k^{a\prime}(w)$ must be true. This leads to

$$f_k^{a\prime}(w) \leqslant f_k^{a\prime}(\psi) = \frac{f_k^a(w)}{w} \leqslant \frac{N_F}{w}. \tag{25}$$

Moreover, since w is the Nash-equilibrium of the auction and $w_k > 0$, we have

$$0 = \frac{\partial J_k^a(w_k)}{\partial s_k} = f_k^{a\prime}(w_k)\frac{K^a(1 - v_k)}{\sum_{m\in V^c} s_m} - C, \tag{26}$$

where $v_k = s_k/\sum_{m\in V^c} s_m$. This leads to

$$f_k^{a\prime}(w_k) = \frac{C\sum_{m\in V^c} s_m}{K^a(1 - v_k)}. \tag{27}$$

Combining Eqs. (25) and (27) derives the conclusion that for any $k \in V^c$,

$$\frac{1}{1 - v_k} \leqslant \frac{K^a N_F}{C w_k \sum_{k\in V^c} s_m}. \tag{28}$$

Applying the condition of Eq. (19) gets the result:

$$|f_k^{a\prime}(w_k) - f_k^{a\prime}(w_h)| = \frac{C\sum_{m\in V^c} s_m}{K^a}\left|\frac{1}{1 - v_k} - \frac{1}{1 - v_h}\right| = \frac{C\sum_{m\in V^c} s_m}{K^a}\frac{|v_k - v_h|}{(1 - v_k)(1 - v_h)}. \tag{29}$$

Since $v_k \geqslant v_h$ and $0 < v_k + v_h \leqslant 1$, we have

$$\frac{1}{1 - v_h} \leqslant \frac{v_k + v_h}{v_k} \leqslant 2. \tag{30}$$

Furthermore, we have $|v_k - v_h| < v_k$. Eq. (29) can be transformed further:

$$\begin{aligned} |f_k^{a\prime}(w_k) - f_k^{a\prime}(w_h)| &\leqslant C\frac{v_k\sum_{m\in V^c} s_m}{K^a}\frac{2}{1 - v_k} \leqslant 2C\frac{v_k\sum_{m\in V^c} s_m}{K^a}\frac{K^a N_F}{C w_k \sum_{m\in V^c} s_m} \\ &= \frac{2N_F}{M_a n_c}\frac{K^a v_k}{w_k} = \frac{2N_F}{M_a n_c}. \end{aligned} \tag{31}$$

This means that some $\epsilon \leqslant 2N_F/(M_a n_c)$ can always be found satisfying Eq. (18). This upper boundary is irrelevant to the choice of k, h. So, as n_c gets infinitely large, $\epsilon \to 0$ must be uniformly true. Thus the proof is completed.

Remark 3. One precondition of the efficiency of the auction is that each partner k must have full knowledge of its payoff function f_k^a beforehand. However, in multi-step games the payoff function is hard to estimate for three reasons. First, predicting the future game for multiple steps requires a great deal of computation. Second, since each partner might only have incomplete information of the whole system, the prediction is not accurate. Last, the strategies of the members are interdependent, so no one can make valid prediction of the future personal payoff without fully knowing the behavior of all the other participants. To resolve this issue, the reinforcement learning algorithm will be discussed in Section 4.

According to Theorem 3, a large margin increase in the payoff curve f_k^a implies that if a larger weight w_k^a were allocated to group k, a higher likelihood for the strategy a would be observed from the agents within the group. In this case, group k should bid more than the rival groups whose margin payoffs are smaller. If everyone is reasonable and the system is large enough, the result allocation is globally efficient. This efficiency is affected by the select of the cost parameter C, which is the conclusion of Theorem 4.

Theorem 4. For any local discounted payoff function $J_k^a(s_k^a) = f_k^a(w_k^a) - Cs_k^a$, the choice of any parameter $C > 0$ leads to the same outcome of the auction.

Proof. See Appendix A.

Another property of the auction protocol is that no monopoly would come about even if the system is large. In the equilibrium state, the bid s_k^a has the property as follows:

$$\frac{C\sum_{m\in V^c} s_m^a}{K^a(1-v_k^a)} = f_k^{a\prime}(w_k^a) \leqslant \frac{N_F}{w_k^a}.$$

Applying $w_k = K_a s_k^a / \sum_m s_m^a$,

$$s_k^a \leqslant \frac{N_F}{C}(1-v_k^a) \leqslant \frac{N_F}{C}. \tag{32}$$

So despite the system scale, any personal bid s_k is only upper bounded by N_F/C, which is irrelevant of n_c. In real-world economy systems, the total bid $\sum_{m\in V^c} s_m$ is always large, and the effect of s_k would be negligible to the whole. In this case, for each k where $s_k > 0$, the marginal income is

$$f_k^{a\prime}(w_k^a) = \frac{C\sum_{m\in V^c} s_m}{K^a}\frac{1}{1-v_k} \approx \frac{C\sum_{m\in V^c} s_m}{K^a}.$$

So, the values of $f_k^{a\prime}(w_k^a)$s approximately get consensus to the same value. This explains why the competitions in large markets help build and maintain a fair and efficient economy system.

4 Reinforcement learning for group level strategy

According to the previous discussions, the efficiency of the distributed weight allocation in the group level depends on the full knowledge about the personal payoff functions beforehand. However, the exact payoff function is hard to calculate since each group only has incomplete information of the whole system. Furthermore, calculating the expectation of the long-run payoff in multi-step games is difficult. So the agents must learn to estimate. If the game is repeated for many times, and each agent keeps the data of the previous games, a reinforcement learning algorithm can be designed and performed along with the auction process.

At the time t, the target of the learning task for the group k is to optimize the bid $s_k^a(t)$ for a proper amount of group weight w_k^a. Since the payoff function is unknown, a Q-function could be defined to calculate the future payoff, as follows:

$$Q_k^a(X_k(t_n), s_k^a(t_n)) = \sum_{\alpha=0}^{\infty} \gamma^\alpha E\left[r_k^a(t_{n+\alpha}) \middle| X_k(t_n), s_k^a(t_n)\right], \tag{33}$$

where $r_k^a(t)$ denotes the one-step payoff at time t, $X_k(t)$ is the local information available to group k at time t, and the discount parameter $0 < \gamma < 1$ should be less than one so as to guarantee the convergence of the learning algorithm. The most intuitive definition of $r_k^a(\cdot)$ is

$$r_k^a(t_{n+\alpha}) = U_k^a(X_k(t_{n+\alpha})) - s_k^a(t_{n+\alpha}). \tag{34}$$

This definition balances the payoff and cost.

The current bid s_k^a affects not only the one-step payoff, but also the payoffs in future steps. If all $r_k^a(\cdot)$s are uniformly bounded, the discounted sum $Q_k^a(\cdot)$ should be a finite value. So the Bellman-equation can be applied to recursively solve the Q-function in each step:

$$\max_{s_k^a(t_n)} Q_k^a(X_k(t_n), s_k^a(t_n)) = r_k^a(t_n) + \gamma \max_{s_k^s(t_{n+1})} Q_k^a(X_k(t_{n+1}), s_k^n(t_{n+1})). \tag{35}$$

The transition from $X_k(t_n)$ to $X_k(t_{n+1})$ is a Markov jump. At every time step, each Markov jump along with other information can be recorded as a tuple: $g_k^a(t_n) = (X_k(t_n), X_k(t_{n+1}), s_k^a(t_n), r_k^a(t_n))$. With the help of these records, the value of $Q_k^a(\cdot)$ can be approximated by training a neural network $\hat{Q}_k^a(\cdot)$. As the record is accumulated along with time, $\hat{Q}_k^a(\cdot)$ can be trained online. Selecting the learning scale as $0 < \beta_1 < 1$, the value of $\hat{Q}_k^a$ is updated as follows:

$$\hat{Q}_k^a(X_k(t_n), s_k^a(t_n)) \leftarrow (1-\beta_1)\hat{Q}_k^a(X_k, s_k^a) + \beta_1 \left[r_k^a(t_n \mid t_n) + \gamma \hat{Q}_k^a(X_k^a(t_{n+1}), s_k^{a*}(t_{n+1})) \right], \tag{36}$$

where the $s_k^{a*}(\cdot)$ denotes the optimal value under the given information, which can be estimated using $\hat{Q}_k^a(\cdot)$:

$$s_k^{a*}(t_{n+1}) \triangleq \arg\max\nolimits_{s \geqslant 0} \hat{Q}_k^a(X_k(t_{n+1}), s). \tag{37}$$

When a better value is updated for the neural network $\hat{Q}_k^a$, its parameters are modified to make its output closer to the better value. To estimate the maximized output of the network $\hat{Q}_k^a(\cdot)$, another neural network $\hat{s}_k^a(X)$ is adopted, whose training target is $\hat{s}_k^a = \arg\max_s \hat{Q}_k^a(X, s)$. Selecting the training scale as $\beta_2 > 0$, the value of $\hat{s}_k^a$ is updated as follows:

$$\hat{s}_k^a(X_k) \leftarrow \hat{s}_k^a(X_k) + \beta_2 \left. \frac{\partial \hat{Q}_k^a(X_k, s)}{\partial s} \right|_{s = \hat{s}_k^a(X_k)}. \tag{38}$$

The learning process in Algorithm 1 can be performed in a variety of forms, for example, all the groups sharing a neural network, or each group having its own network. The better the neural network is trained, the smarter the members in the group level would be, and the more global efficiency the system can achieve.

Algorithm 1 The overall list of the stepwise auction algorithm for the allocation of groups' weights

```
From the initial time $n = 0$, do:
1. For each group $k \in V^c$, record the local information as $X_k(t_n)$.
2. Each individual offers a bid $s_k^a$ for the auction, using the neural network $s_k^a(t_n) = \hat{s}_k^a(X_k(t_n))$.
3. Allocate the weight values.
4. Wait until the next time step $t_{n+1}$.
5. Store the current local information $X_k(t_{n+1})$.
6. Add the tuple $(X_k(t_n), X_k(t_{n+1}), s_k^a(t_n), r_k^a(t_n \mid t_n))$ into the training set.
7. Set $n \leftarrow n + 1$, go back to step 2.
```

5 Simulations and analysis on the motivational model

In the simulation, all agents are placed on the 2D periodic lattices. Each group is composed of each K successive agents along any row, column or diagonal. The lattices and groups are illustrated as Figure 2.

It is obvious that the groups and agents in the lattices have the following properties:

(1) Each group contains the same number of agents, and each agent belongs to the same number of groups.

(2) There is no boundary on the torus surface, so all groups are equivalent.

In the simulation, the strategy set $A = \{a_1, a_2\}$ contains two strategies, and the size of each group is $K = 5$. The value $p_{ij} \in [0, 1]$ denotes the probability of the agent in the i-th row and j-th column for the strategy a_1, so $1 - p_{ij}$ denotes its probability for a_2. The size of the lattices is 70×70. All weights of groups are set to constant 1. The dynamic of each agent follows the replicator dynamic in Eq. (13). The strategy distribution of the agents is illustrated in Figure 3, and the temporal curve of

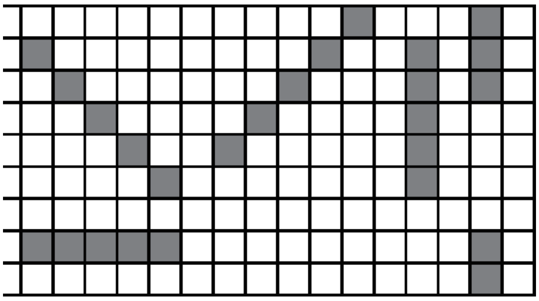

Figure 2 Five groups on the lattices when $K = 5$. Any K successive agents along any direction belong to a common group, while only 5 groups are illustrated here. One group spans across the bound recursively.

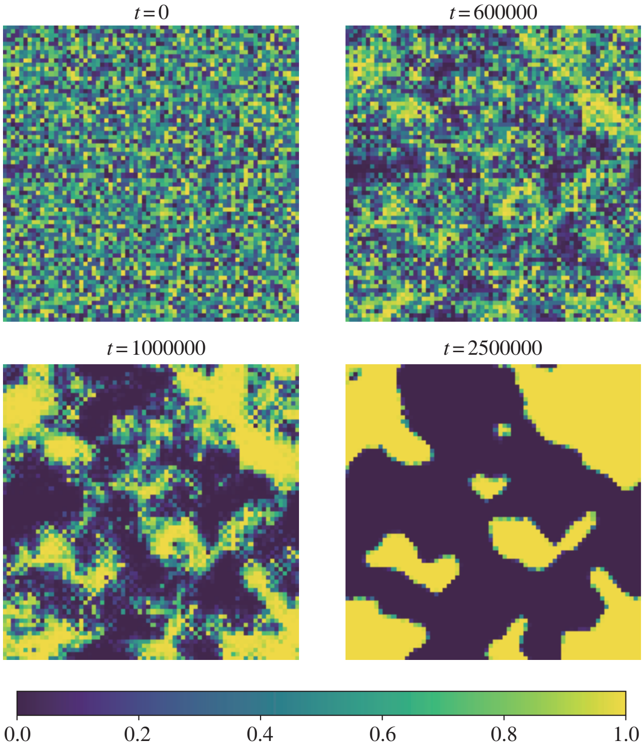

Figure 3 (Color online) The probability of choosing strategy a_1 by the agents in the 70×70 lattices. The agents' initial strategies are selected randomly from the range $[0, 1]$. The agents in lattices adapt to each other's strategy by following the replicator dynamic, until the system reaches the stable state. During the process, agents keep adjusting their strategies trying to get accordance with their neighbour agents. In the final state, the lattices are divided into several areas based on the strategy preferred by agents in each area.

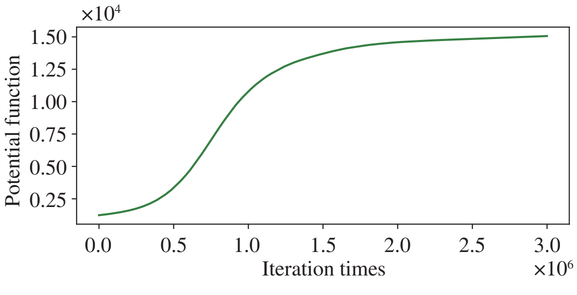

Figure 4 (Color online) The temporal curve of the potential function $V(t)$, which is a monotonic increasing function converging to its upper bound.

the system's potential function is illustrated in Figure 4. It can be seen that in the final state each agent converges to the pure strategy a_1 or a_2. In the next experiment, the game is repeated for several

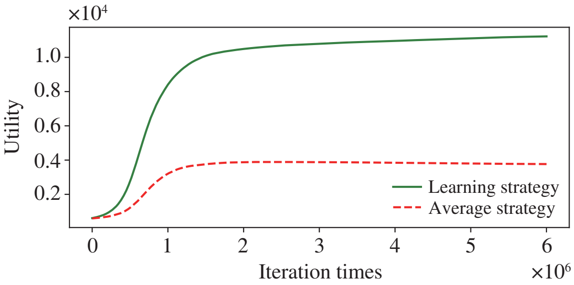

Figure 5 (Color online) The total reward of strategies a_1 and a_2 in the game. The strategy a_1 (learning strategy) uses strategically allocated group weights, while strategy a_2 (average strategy) allocates the weights averagely. In this experiment, strategy a_1 performs better than strategy a_2.

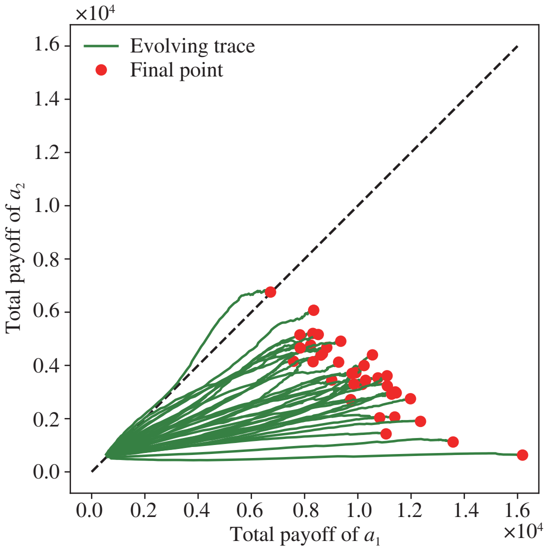

Figure 6 (Color online) The temporal curves of the total rewards of strategies a_1 and a_2 in the system, which are scaled on both axes. The strategy a_1 performs the reinforcement learning method, while the strategy a_2 allocates the group weights averagely. The dashed line denotes the equal points where the rewards on both strategies are equivalent. Most traces end at the right part of the dashed line, which means that the learning strategy a_1 acquires higher rewards than the strategy a_2 by large.

times from randomized initial states. Only strategy a_1 performs reinforcement learning method, while strategy a_2 allocates all group weights averagely. The utility of both strategies is recorded along time, and illustrated as the traces of the points $(U^{a_1}(t), U^{a_2}(t))$. The learning algorithm is repeated several times for learning parameters to converge, and then the multi-step game is performed repeatedly from random initial states, during which the agent payoffs are recorded. The payoff curve of the strategies in a single experiment is shown in Figure 5, while the final outcomes from the repeated experiments are illustrated in Figure 6. In Figure 6, the points with a higher value in the x-label denote a state where strategy a_1 is overall preferred, while those with a higher value in the y-label denote a state where a_2 is overall preferred. From both Figures 5 and 6, it is clear that the payoff of strategy a_1 is significantly higher in total. For experiments with or without reinforcement learning method, a comparison of their final strategy distributions is illustrated in Figure 7.

The better performance of the strategy a_1 implies a higher efficiency in weight allocation for a_1 than for a_2. Since the system is large and the utility functions are bounded, the condition of Theorem 3 is satisfied, and the auction protocol based on greedy strategies leads to the result close to the centralized optimization. Even though the group weights of each strategy sum up to the same value, the allocation of the group weights about strategy a_1 is better refined by the competition in the auction and the reinforcement learning process. So in the agent level, those who adopt a_1 enjoy the higher payoff by large, and in the final state, there is more likelihood that a group reaches the consensus of a_1 than of a_2.

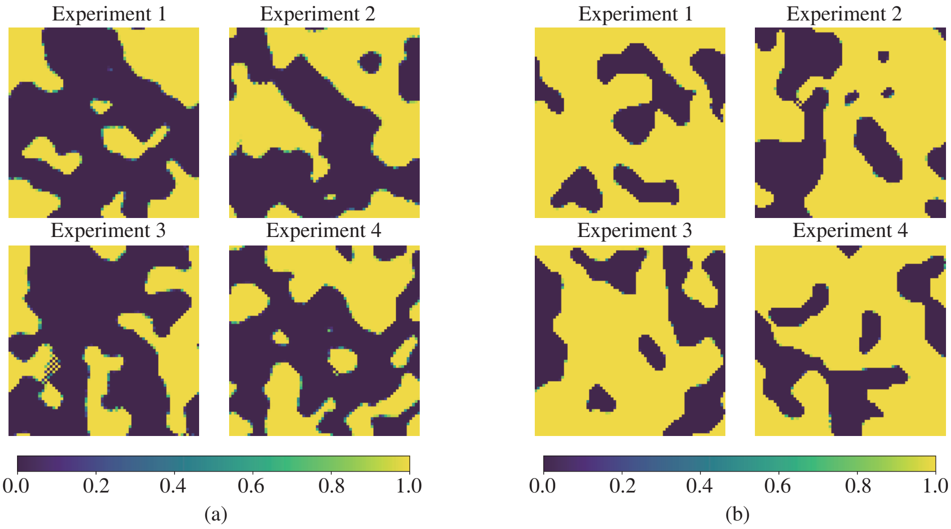

Figure 7 (Color online) The final strategy allocation of agents in eight independent experiments is illustrated, where the values denote the probability of choosing a_1 in agents' mixed strategies. In (a), weights of both strategies are allocated averagely. In (b), the group weights of a_1 are allocated using the reinforcement learning algorithm, while the weights of a_2 are allocated averagely. According to the final states of the agents, each agent trends to choose the same strategy as the neighbor agents in the same groups do, while groups at distance may not reach the consensus about the strategies, dividing the system into several areas. Furthermore, in the experiments with reinforcement learning (b), the agents who prefer a_1 outnumbers the agents preferring a_2, which suggests the advantage of a_1 because of the more efficient allocation of group weights.

6 Conclusion

This paper presented a two-level game theoretic model of a structured population, adding new features to the exiting evolutionary game models. In the proposed model, agents are grouped into overlapped groups, their rewards are connected to the group payoff, and decisions are made on both the agent and group levels. On the agent level, the decisions are made based on the "greedy" strategy. In particular, agents assume that the environment remains stable and adjust their individual strategies towards the maximum of their personal interest. On the group level, long-term strategy programming is performed, and reinforcement learning is used to help each group adapt to the environment. Furthermore, the stability and efficiency of the proposed algorithm was analyzed. On the agent level, the game was proved to be a state-based potential game. On the group level, the auction method was designed and proved to be efficient when utility functions are bounded and the scale of the population is efficiently large. Several simulations of the proposed framework are also presented.

Acknowledgements This work was supported by Tianjin Natural Science Foundation (Grant Nos. 20JCYBJC01060, 20JC-QNJC01450) and National Natural Science Foundation of China (Grant No. 61973175).

References

1 Quijano N, Ocampo-Martinez C, Barreiro-Gomez J, et al. The role of population games and evolutionary dynamics in distributed control systems: the advantages of evolutionary game theory. IEEE Control Syst, 2017, 37: 70–97
2 Nowak M A, Tarnita C E, Antal T. Evolutionary dynamics in structured populations. Phil Trans R Soc B, 2010, 365: 19–30
3 Fu F, Wang L, Nowak M A, et al. Evolutionary dynamics on graphs: efficient method for weak selection. Phys Rev E, 2009, 79: 046707
4 Taylor C, Fudenberg D, Sasaki A, et al. Evolutionary game dynamics in finite populations. Bull Math Biol, 2004, 66: 1621–1644
5 Ohtsuki H, Nowak M A. Evolutionary games on cycles. Proc R Soc B, 2006, 273: 2249–2256
6 Nowak M A. Five rules for the evolution of cooperation. Science, 2006, 314: 1560–1563
7 Ohtsuki H, Nowak M A, Pacheco J M. Breaking the symmetry between interaction and replacement in evolutionary dynamics on graphs. Phys Rev Lett, 2007, 98: 108106
8 Tarnita C E, Ohtsuki H, Antal T, et al. Strategy selection in structured populations. J Theory Biol, 2009, 259: 570–581
9 Xia C Y, Li X P, Wang Z, et al. Doubly effects of information sharing on interdependent network reciprocity. New J Phys, 2018, 20: 075005
10 Tang C B, Li X, Wang Z, et al. Cooperation and distributed optimization for the unreliable wireless game with indirect reciprocity. Sci China Inf Sci, 2017, 60: 110205
11 Xia C Y, Ding S, Wang C J, et al. Risk analysis and enhancement of cooperation yielded by the individual reputation in the spatial public goods game. IEEE Syst J, 2017, 11: 1516–1525
12 Chen M H, Wang L, Sun S W, et al. Evolution of cooperation in the spatial public goods game with adaptive reputation assortment. Phys Lett A, 2016, 380: 40–47
13 Fudenberg D, Levine D K. The Theory of Learning in Games. Boston: MIT Press, 1998

14 Li J Q, Zhang C Y, Sun Q L, et al. Changing intensity of interaction can resolve prisoner's dilemmas. Europhys Lett, 2016, 113: 58002

15 Perc M, Gómez-Gardeñes J, Szolnoki A, et al. Evolutionary dynamics of group interactions on structured populations: a review. J R Soc Interface, 2013, 10: 20120997

16 Gracia-Lázaro C, Gómez-Gardeñes J, Floría L M, et al. Intergroup information exchange drives cooperation in the public goods game. Phys Rev E, 2014, 90: 042808

17 Gómez-Gardeñes J, Vilone D, Sánchez A. Disentangling social and group heterogeneities: public goods games on complex networks. EPL, 2011, 95: 68003

18 Gómez-Gardeñes J, Romance M, Criado R, et al. Evolutionary games defined at the network mesoscale: the public goods game. Chaos, 2011, 21: 016113

19 Kelly F P, Maulloo A K, Tan D K H. Rate control for communication networks: shadow prices, proportional fairness and stability. J Oper Res Soc, 1998, 49: 237–252

20 Li J, Ma G Q, Li T, et al. A Stackelberg game approach for demandresponse management of multi-microgrids with overlapping sales areas. Sci China Inf Sci, 2019, 62: 212203

21 Monderer D, Shapley L S. Potential games. Games Econom Behav, 1996, 16: 124–143

22 Barreiro-Gomez J, Obando G, Quijano N. Distributed population dynamics: optimization and control applications. IEEE Trans Syst Man Cybern Syst, 2017, 47: 304–314

23 Barreiro-Gomez J, Quijano N, Ocampo-Martinez C. Constrained distributed optimization: a population dynamics approach. Automatica, 2016, 69: 101–116

24 Li N, Marden J R. Designing games for distributed optimization. IEEE J Sel Top Signal Process, 2013, 7: 230–242

25 Li N, Marden J R. Decoupling coupled constraints through utility design. IEEE Trans Autom Control, 2014, 59: 2289–2294

26 Marden J R. State based potential games. Automatica, 2012, 48: 3075–3088

27 Maheswaran R, Basar T. Efficient signal proportional allocation (ESPA) mechanisms: decentralized social welfare maximization for divisible resources. IEEE J Sel Areas Commun, 2006, 24: 1000–1009

28 Yan L, Qu B Y, Zhu Y S, et al. Dynamic economic emission dispatch based on multi-objective pigeon-inspired optimization with double disturbance. Sci China Inf Sci, 2019, 62: 070210

29 Tang C B, Li A, Li X. Asymmetric game: a silver bullet to weighted vertex cover of networks. IEEE Trans Cybern, 2018, 48: 2994–3005

30 Li X X, Peng Z H, Liang L, et al. Policy iteration based Q-learning for linear nonzero-sum quadratic differential games. Sci China Inf Sci, 2019, 62: 052204

31 Watkins C J, Dayan P. Technical note: Q-learning. Mach Learn, 1992, 8: 279–292

32 Lanctot M, Zambaldi V F, Gruslys A, et al. A unified game-theoretic approach to multiagent reinforcement learning. In: Proceedings of the 31st International Conference on Neural Information Processing, 2017. 4190–4203

33 Tuyls K, Pérolat J, Lanctot M, et al. Symmetric decomposition of asymmetric games. Sci Rep, 2018, 8: 1015

34 Zhang K Q, Yang Z R, Liu H, et al. Fully decentralized multi-agent reinforcement learning with networked agents. In: Proceedings of International Conference on Machine Learning, 2018. 5867–5876

35 Busoniu L, Babuska R, de Schutter B. A comprehensive survey of multiagent reinforcement learning. IEEE Trans Syst Man Cybern C, 2008, 38: 156–172

Appendix A Proof of Theorem 4

Proof. We set $J_{1k}(s_k) = f_k^a(w_k) - C_1 s$, $J_{2k}(s_k) = f_k^a(w_k) - C_2 s_k$, and $s_1 = [s_{11}, s_{12}, \ldots, s_{1n_c}]$ is the solutions of J_{1k}. So according to the Nash-equilibrium condition we have

$$0 = \frac{\partial J_{1k}}{\partial s_{1k}} = \frac{{f_k^a}'(w_{1k})(1 - v_{1k})K^a}{\sum_{m \in V^c} s_{1m}} - C_1. \tag{A1}$$

Now we set $s_2 = [s_{21}, s_{22}, \ldots, s_{2n_c}]$ where $s_{2k} = C_1 s_{1k}/C_2$. This makes $v_{1k} = v_{2k}$ and $w_{1k} = w_{2k}$ to be the same allocation of weights. s_2 also satisfies

$$\frac{\partial J_{2k}}{\partial s_{2k}} = \frac{{f_k^a}'(w_{2k})(1 - v_{2k})K^a}{\sum_{m \in V^c} s_{2m}} - C_2 = \frac{C_2}{C_1} \frac{{f_k^a}'(w_{1k})(1 - v_{1k})K^a}{\sum_{m \in V^c} s_{1m}} - C_2 = C_2 - C_2 = 0. \tag{A2}$$

Namely s_2 is also the Nash-equilibrium of each function $J_{2k}(\cdot)$. Since s_1 and s_2 denote the same allocation, we know that all the Nash-equilibrium of J_1 is the Nash-equilibrium of J_2, and the inverse proposition holds the same. So the choice between C_1 and C_2 only affects the scale of the bid s_k but not the final allocation of the auction.