ECAI 2016
G.A. Kaminka et al. (Eds.)

doi:10.3233/978-1-61499-672-9-1344

Accelerating Norm Emergence Through Hierarchical Heuristic Learning

Tianpei Yang [1] and **Zhaopeng Meng** [1,2] and **Jianye Hao** [3] and **Sandip Sen** [4] and **Chao Yu** [5]

Abstract. Social norms serve as an important mechanism to regulate the behaviours of agents and to facilitate coordination among them in multiagent systems. One important research question is how a norm can rapidly emerge through repeated local interaction within agent societies under different environments when their coordination space becomes large. To address this problem, we propose a hierarchically heuristic learning strategy (HHLS) under the hierarchical social learning framework. Subordinate agents report their information to their supervisors, while supervisors can generate instructions (rules and suggestions) based on the information collected from their subordinates. Subordinate agents heuristically update their strategies based on both their own experience and the instructions from their supervisors. Extensive experiment evaluations show that HHLS can support the emergence of desirable social norms more efficiently and can be applicable in a much wider range of multiagent interaction scenarios compared with previous work. The influence of key related factors (e.g., different topologies, population, neighbourhood and action space size, cluster size) are also investigated and new insights are obtained as well.

1 INTRODUCTION

In multiagent systems, social norms play an important role in regulating agents' behaviors to ensure coordination among agents and functioning of agent societies. One commonly adopted characterization of a norm is to model it as a consistent equilibrium that all agents follow during interactions where multiple equivalent equilibria coexist [20]. How social norms can emerge efficiently in agent societies is a key research problem in the area of normative multiagent systems.

There exist two major approaches for addressing norm emergence problem: the top-down approach and the bottom-up approach. The former approach investigates how to efficiently synthesize a norm for all agents beforehand, while the latter one focuses on investigating how a norm can emerge through repeated local interactions by learning among agents. In distributed multiagent interaction environments, it is usually difficult to come up with any norm before agents interactions start since there may not exist such a centralized controller and also the optimal norm may vary frequently as the environment dynamically changes and therefore, the bottom-up approach via local learning promises to be more suitable for such kinds of distributed and dynamic environments.

[1] School of Computer Software, Tianjin University, China, email: {tpyang,mengzp}@tju.edu.cn
[2] Tianjin University of Traditional Chinese Medicine, China
[3] School of Computer Software, Tianjin University, China, email: jianye.hao@tju.edu.cn; Corresponding author
[4] University of Tulsa, USA, email: sandip-sen@utulsa.edu
[5] Dalian University of Technology, China, email: cy496@dlut.edu.cn

Until now, significant efforts have been devoted to investigating norm emergence problem from the bottom-up research direction [2, 3, 5–9, 12–17, 21–25]. Sen and Airiau [13] investigated the norm emergence problem in a population of agents within randomly connected networks where each agent is equipped with certain existing multiagent learning algorithms. The local interaction among each pair of agents is modeled as two-player normal-form games, and a norm corresponds to one consistent Nash equilibrium of the coordination/anti-coordination game. Later a number of papers [2, 9, 12, 17] subsequently extended this work by using more realistic and complex networks (e.g., small-world network and scale-free network) to model the diverse interaction patterns among agents. Additionally, different learning strategies and mechanisms have been proposed to better facilitate norm emergence among agents within different interaction environments [3, 6, 8, 11, 25].

Most of the previous works only focus on games with relatively small size, which do not accurately reflect the practical interaction scenarios where the action space of agents can be quite large. With the increasing of the action space, unfortunately, most of the existing approaches usually result in very slow norm emergence or even fail to converge. Recently Yu et al. [21] proposed a hierarchical learning strategy to improve the norm emergence rate for the huge action space problem. However, this work only considers the case in which a norm corresponds to a Nash equilibrium where all agents select the same action. This usually can be modelled as a two-player n-action *coordination game*. One simple example with n=2 is shown in Table 1. In contrast, in realistic interaction scenarios, a norm may correspond to agents coordinating using different actions. One notable example is considering two drivers arriving at a road intersection from two neighbouring roads. To avoid collision, one possible norm is "yield to the left", i.e., waiting for the car on the left-hand side to go through the intersection first. This kind of scenario can naturally be modelled as an *anti-coordination game* shown in Table 2, which exist two different norms (a, b) and (b, a).

Furthermore, agents may be faced with the challenge of high miscoordination cost and stochasticity of the environment. One representative example is shown in Table 3, which we call it *fully stochastic coordination game with high penalty*. In this game, there exist two optimal Nash equilibriums each of which corresponds to one norm, and one suboptimal Nash equilibrium. Two major challenges coexist in this game: agents are vulnerable to converge to the suboptimal Nash equilibrium due to the high penalty when agents mis-coordinate on the outcomes; agents need to effectively distinguish between the stochasticity of the environment and the explorations of other learners. It is not clear, a priori, how a population of agents can efficiently evolve towards a consistent norm given the large space of possible norms in such challenging environments.

Table 1. An example of coordination game

		Agent 2's actions	
		a	b
Agent 1's actions	a	1	-1
	b	-1	1

Table 2. An example of anti-coordination game

		Agent 2's actions	
		a	b
Agent 1's actions	a	-1	1
	b	1	-1

Table 3. Fully stochastic coordination game with high penalty

1's payoff / 2's payoff		Agent 2's actions		
		a	b	c
Agent 1's actions	a	8/12	-5/5	-20/-40
	b	-5/5	0/14	-5/5
	c	-20/-40	-5/5	8/12

To answer this question, in this paper we propose a novel hierarchical heuristic learning strategy (HHLS) under the hierarchical social learning framework to facilitate the rapid norm emergence in agent societies. In the hierarchical social learning framework, the agent society is separated into a number of clusters of subordinate agents, where each cluster's strategies are monitored and guided by one supervisor agent. For each supervisor agent, in each round, it collects the interaction information of the subordinate agents under its supervision and generates guided instructions in the forms of rules and suggestions for its subordinates. On the other hand, for each subordinate agent, apart from learning from its local interaction, it also adjusts its strategy based on the instructions from its supervisor. The main feature of the proposed framework is that through hierarchically supervised subordinate agents, an effective compromise solution can be generated to effectively balance distributed interactions and centralized control towards efficient and robust norm emergence. We evaluate the performance of HHLS under a wide range of games and experimental results show that HHLS can efficiently facilitate the rapid emergence of norms compared with the state-of-the-art approaches. We also investigate the influence of a number of key factors on norm emergence: the population size, the neighbourhood size, the size of action space, cluster size, different network topologies, etc.

The remainder of the paper is organized as follows. Section 2 discusses related work. Section 3 introduces the hierarchical social learning framework and the heuristic learning strategy. Section 4 presents experimental evaluation results comparing with two representative state-of-the-art approaches. Finally Section 5 concludes the paper and points out future directions.

2 RELATED WORK

Norm emergence problem has received a wide range of attention in MASs literature. Shoham and Tennenholtz [14] firstly investigated the norm emergence problem in agent society based on a simple and natural strategy - the highest cummulative reward (HCR). In this study, they showed that HCR achieved high efficiency on social conventions in a class of games. Sen and Airiau [13] investigated the norm emergence problem in a population of agents within randomly connected networks where each agent is equipped with certain existing multiagent learning algorithms. They firstly proposed the model of learning *social learning*, where each agent learns from repeated interactions with multiple agents in a given secenario. In this study, the local interaction among each pair of agents is modeled as two-player normal-form games, and a norm corresponds to one consistent Nash equilibrium of the game. Later a number of papers [2,9,12,17] subsequently extended this work by leveraging more realistic and complex networks (e.g., small-world network and scale-free network) to model the interaction patterns among agents and evaluated the influence of heterogeneous agent systems and space-constrained interactions on norm emergence. Savarimuthu [11] recapped the existing mechanisms on the multiagent-based emergence, and investigated the role of three proactive learning methods in accelerating norm emergence. The influence of the presence of liars on norm emergence is also considered and simulation results showed that norm emergence can still be sustained in the presence of liars. Villatoro et al. [17] proposed a reward learning mechanism based on interaction histories. In this study, they investigated the influence of different network topologies and the effects of memory of past activities on convention emergence. Later, they [15, 16] introduced two rules (i.e., re-wiring links with neighbors and observation) to overcome the suboptimal norm problems. They investigated the influence of Self-Reinforcing Substructure (SRS) in the network on impeding full convergence towards society-wide norms, which usually results in reduced convergence rates. Hao et al. [5] investigated the problem of coordinating towards optimal joint actions in cooperative games under the social learning framework by introducing two types of learners (IALs and JALs). Yu et al. [24] proposed a novel collective learning framework to investigate the influence of agent local collective behaviours on norm emergence in different scenarios and defined two strategies (collective learning-l and collective learning-g) to promote the emergence of norms where agents are allowed to make collective decisions within networked societies. Later Hao et al. [6] proposed two learning strategies under the collective learning framework: collective learning EV-l and collective learning EV-g to address the problem of high mis-coordination cost and stochasticity in complex and dynamic interaction scenarios. Recently Yu et al. [22] proposed an adaptive learning framework for efficient norm emergence. However, all the aforementioned works usually focus on relatively small-size games, and do not address the issue of efficient norm emergence in large action space problems.

Hierarchical learning framework, as a promising solution to accelerate coordination among agents, has been studied in different multiagent applications (e.g., package routing [25], traffic control [1], p2p network [4] and smart-grid [19]). For example, Zhang et al. [25, 26] studied the package routing problem and proposed a multi-level organizational structure for automated supervision and a communication protocol for information exchange between higher-level supervising agents and subordinate agents. Simulation shows that the organization-based control framework can significantly increase the overall package routing efficiency than traditional non-hierarchical appraoches. Abdoos et al. [1] proposed a multi-layer organizational controlling framework to model large traffic networks to improve the coordination between different car agents and the overall traffic efficiency. Until recently, Yu et al. [21] firstly proposed a hierarchical learning framework to study the norm emergence problem. In this study, they proposed a two-level hierarchical framework. Agents in the lower level interact with each other and report information to their supervisors in the higher level, while agents in the higher level called supervisors pass down guidance to the lower level. Agents in the lower level follow guidance in policy update. However, their

framework is designed for coordination game only where each agent only needs to coordinate on the same action for norm emergence.

3 HIERARCHICAL SOCIAL LEARNING FRAMEWORK

3.1 Framework Overview

We consider a population N of agents where each agent is connected following the underlying network topology. In each round, each agent interacts with one randomly selected agent from its neighbourhood. An agent's neighbourhood consists of all agents which it is physically connected with. We model the interaction between each pair of agents as a normal-form game. At the beginning of each interaction, one agent is randomly assigned as the row player and the other as the column player. We assume that each agent can only have access to its own action and payoff information during interaction. On the other hand, the population of agents are divided into multiple levels, and the agents in each level supervise the behaviours of agents from its neighbouring lower level. For the sake of exposition, we present the hierarchical social learning framework in two levels. However, it is straightforward to extend the hierarchical social learning framework into $k > 2$ levels.

One illustrating example of two-level hierarchical network is shown in Figure 1. Each supervisor agent i in the higher level is in charge of a group of subordinate agents (denoted as $sub(i)$) in the bottom level surrounded by dashes lines. For each subordinate agent j, its supervisor agent is denoted as $sup(j)$. For subordinates, the topological connections between them are determined by the original network topology; for supervisors, a pair of supervisors are neighbouring agents if the corresponding group of subordinates they supervise are connected. Note that a supervisor agent can be viewed as a special subordinate agent within the original network, which is also allowed to communicate with its neighbouring supervisor agents.

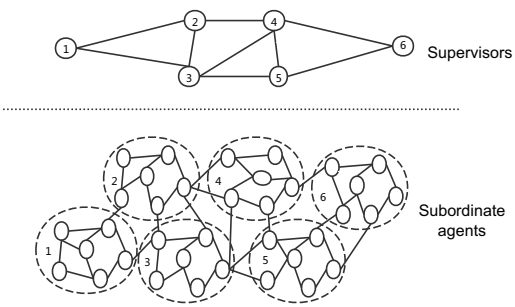

Figure 1. An example of the two-level hierarchical network

The interaction protocol of agents under the hierarchical social learning framework is summarized in Algorithm 1. In each round, each agent is paired with another agent randomly selected from its neighbourhood to interact with (Line 3), and their roles are randomly assigned (Line 4). Each agent then chooses an action following its learning strategy (Line 5), and then updates its strategy based on its current-round feedback (Line 6). After that, each subordinate agent reports its action and reward information to its supervisor (Line 7-9). At the end of each round, each supervisor collects all subordinate agents' information, generates and issues the instructions to its subordinate agents (Line 12-14). Finally each subordinate agent updates its strategy based on the instructions accordingly (Line 15-17).

Algorithm 1 The interaction protocol of hierarchical framework

1: **for** each round of interaction **do**
2: **for** each agent $i \in N$ **do**
3: Randomly choose a neighbouring agent j to interact;
4: Assign distinct roles randomly $i \rightarrow$ state s_i, $j \rightarrow$ state s_j
5: Select actions a_i and a_j and get rewards r_i and r_j;
6: Update its strategy based on $\langle s_i, a_i, r_i \rangle$.
7: **if** agent i is a subordinate agent **then**
8: Reporting its experience $\langle s_i, a_i, r_i \rangle$ to sup(i);
9: **end if**
10: **end for**
11: **for** each supervisor agent j **do**
12: Generate instructions based on the information from $sub(j)$;
13: Provide the instructions to $sub(j)$;
14: **end for**
15: **for** each subordinate agent k **do**
16: Update its strategy based on the instructions from $sup(k)$;
17: **end for**
18: **end for**

3.2 Information Exchange between Supervisors and Subordinates

In the hierarchical social learning framework, subordinate agents send their feedback information to their corresponding supervisors, while supervisors pass down instructions to their corresponding subordinate agents. In details, each subordinate agent i reports its current-round interaction experience $\langle s_i, a_i, r_i \rangle$ to its supervisor $sup(i)$. For supervisors, we distinguish two different forms of instructions that they can provide to their subordinates: $suggestion$ and $rule$ [25]. Intuitively, a $rule$ is a hard constraint that specifies an action that subordinate agents are forbidden to select under certain state next round; in contrast, a $suggestion$ is a soft constraint which indirectly affects the strategies of the subordinate agents next round.

A set F of $rules$ consists of all the forbidden actions for subordinates under different states. Formally we have,

$$F = \{\langle s, a \rangle | a \in A, s \in S\} \tag{1}$$

where each element $\langle s, a \rangle$ denotes that action a is forbidden to take under state s; A and S are the action space and state space of the subordinates.

A set D of $suggestions$ specifies the recommendation degrees for different state-action pairs, which can be formally represented as follows,

$$D = \{\langle s, a, d(s, a) \rangle | a \in A, s \in S\} \tag{2}$$

where $d(s, a)$ is the recommendation degree of action a under state s. Given an action and a state $\langle s, a \rangle$, if $d(s, a) < 0$, it indicates that action a is not recommended to select under state s; if $d(s, a) > 0$, it indicates subordinate agents are encouraged to select action a when they are in state s. The way of determining rules and suggestions will be covered in details in Section 3.3.2.

3.3 Learning strategy

In this section, we first present the learning strategy of supervisors and how the rules and suggestions are generated in Section 3.3.1 and

$$freq(s,a) = \frac{|\{\langle s_k, a_k, r_k\rangle \mid \langle s_k, a_k, r_k\rangle \in RepInf, s_k = s, a_k = a, r_k = r_{max}(s,a)\}|}{|\{\langle s_k, a_k, r_k\rangle \mid \langle s_k, a_k, r_k\rangle \in RepInf, s_k = s, a_k = a\}|} \tag{3}$$

3.3.2 respectively. Following that, we describe the learning strategy of subordinate agents and how they utilize the instructions from supervisors in Section 3.3.3. Without loss of generality, let us assume that there is a set S of supervisors, and each supervisor $i \in S$, it supervises the set $sub(i)$ of subordinate agents. Each subordinate agent j has a set $neigh(j)$ of neighbours, and each supervisor agent i communicates with a set $com(i)$ of other supervisors.

3.3.1 Supervisor's strategy

We propose that each supervisor i holds a Q-value $Q_i(s,a)$ for each action a under each state s (row or column player). Let us denote the set of information from its subordinates as $RepInf_i = \{\langle s_k, a_k, r_k\rangle \mid k \in sub(i)\}$. For each piece of information $\langle s,a,r\rangle \in RepInf_i$, supervisor agent i updates its Q-value following the optimistic assumption shown in Equation (4),

$$Q_i(s,a) = (1-\alpha_i) * Q_i(s,a) + \alpha_i * r \tag{4}$$

where α_i is its learning rate reflecting its updating degree between using the past experience and using the current round information.

After that, supervisor i further updates its Q-values based on optimistic assumption and the frequency information of each action similar to the FMQ heuristic [7]. Formally we have,

$$FMQ_i(s,a) = Q_i(s,a) + freq(s,a) * r_{max}(s,a) * C \tag{5}$$

where $r_{max}(s,a)$ is the max reward of each action a, $freq(s,a)$ is the frequency of receiving the reward of $r_{max}(s,a)$ by choosing action a under state s and C is a weighting factor.

The value of $r_{max}(s,a)$ is obtained from the reported information of its subordinate agents $sub(i)$. Specifically, The value of $r_{max}(s,a)$ is computed as the maximum reward that all of its subordinates receives under state s by choosing action a in the current round experience. Formally we have,

$$\mathcal{R}(s,a) = \{r_k \mid \langle s,a,r_k\rangle \in RefInf\} \tag{6}$$

$$r_{max}(s,a) = max\{\mathcal{R}(s,a)\} \tag{7}$$

The frequency information $freq(s,a)$ is calculated as the empirical probability of receiving the maximum reward $r_{max}(s,a)$ under state s when action a is selected based on the reported information $RepInf$ collected from the subordinates which is shown in Equation (3).

After updating the strategy based on the information collected from its subordinates, we also allow each supervisor to learn from its neighboring peers (supervisors). Specifically each supervisor communicates with a neighboring supervisor randomly selected and imitates the neighbor's strategy. The motivation of imitating peers comes from evolutionary game theory [18], which provides a powerful methodology to model how strategies evolve over time based on their relative performance. One of the widely used imitation rules is the proportional imitation [10], which is adopted here as shown in Equation (8),

$$p = \frac{1}{1 + e^{-\beta * (FMQ_j(s,a) - FMQ_i(s,a))}} \tag{8}$$

where parameter β controls the degree of imitating the strategy (the FMQ-value) of the neighbouring supervisor.

Finally each supervisor i updates its strategy (denoted as E-value $E_i(s,a)$) for each action a under state s as the average between the FMQ-values of its own and its neighbour j weighted by parameter p. Formally we have,

$$E_i(s,a) = (1-p) * FMQ_i(s,a) + p * FMQ_j(s,a) \tag{9}$$

3.3.2 Supervisor Instruction Generation

Next we introduce how a supervisor generates instructions for its subordinates at the end of each round. As previously mentioned, there are two forms of instructions from a supervisor: rules and suggestions. First each supervisor i normalizes the E-values, which serves as the basis for generating instructions for its subordinates. Formally we have,

$$E'_i(s,a) = \frac{E_i(s,a) - \overline{E_i(s,a)}}{\sigma} \tag{10}$$

where $\overline{E_i(s,a)}$ is the mean of E-values averaged over all state-action pairs shown in Equation (11),

$$\overline{E_i(s,a)} = \frac{\sum_{a\in A} E_i(s,a)}{|A|} \tag{11}$$

The parameter σ is the standard deviation of FMQ-value following Equation (12),

$$\sigma = \sqrt{\frac{1}{|A|}\sum_{a\in A}(E_i(s,a) - \overline{E_i(s,a)})^2} \tag{12}$$

Given a state-action pair $\langle s,a\rangle$, if the E'-value $E'(s,a)$ is smaller than a given threshold, it indicates that selecting action a is not a wise choice under state s, thus it is encoded as a rule. Formally we have,

$$F = \{\langle s,a\rangle | E'(s,a) < \delta\} \tag{13}$$

where δ is the threshold which is set to the value of -0.5 in this paper.

For each state-action pair (s,a), its recommendation degree $d(s,a)$ is set to the value of $E'_i(s,a)$. Thus the set of suggestions from supervisor i can be represented as follows,

$$D = \{\langle s,a,E'_i(s,a)\rangle \mid a \in A, s \in S\} \tag{14}$$

Given a state-action pair (s,a), if $E'_i(s,a) < 0$, it indicates that selecting action a is not recommended under state s; if $E'_i(s,a) > 0$, it indicates subordinate agents are encouraged to select action a when they are in state s.

3.3.3 Learning Strategy of Subordinates

Similar to the strategies of supervisors, each subordinate agent j also keeps a record of a Q-value $Q_j(s,a)$ for each action $a \in A_j$ under each state s. The Q-value $Q_j(s,a)$ indicates the past performance of choosing action a under state s and serves as the basis for making decisions [3]. For each subordinate agent j, let us first denote its feedback information received by the end of round t as $FeedInf_j^t = \{\langle s_m, a_m, r_m\rangle | m \in [1,t]\}$. At the end of each round

t, subordinate agent j updates its Q-value based on its feedback $\langle s_t, a_t, r_t \rangle$ as follows,

$$Q_j(s_t, a_t) = (1 - \alpha_j) * Q_j(s_t, a_t) + \alpha_j * r_t \quad (15)$$

where α_j is the learning rate modelling its updating degree between using the previous experience and using the most recent information.

Additionally each subordinate agent also updates its Q-values by taking into consideration both the optimistic assumption and the frequency information [7]. Formally we have,

$$FMQ_j(s, a) = Q_j(s, a) + freq(s, a) * r_{max}(s, a) * C \quad (16)$$

where $r_{max}(s, a)$ is the max reward of each action a based on its own experience, $freq(s, a)$ is the frequency of getting the payoff of $r_{max}(s, a)$ until now for action a and C is a weighting factor defining the trade-off between updating using Q-values and maximum payoff information. $freq(s, a)$ is calculated the same as shown in Equation (3).

After receiving supervisor's suggestions, each subordinate further adjusts its estimation of the goodness of each state-action pair based on the FMQ-values as follows,

$$E_j(s, a) = FMQ_j(s, a) * (1 + d(s, a) * \rho) \quad (17)$$

where $d(s, a)$ is the suggestion degree on the state-action pair (s, a), and ρ is a weighting factor controlling the influence of the recommendation degree on the E-values.

Besides, supervisors also influence the subordinate agents' exploration rates. Let us suppose a subordinate agent j selects action a under current state s. If the supervisor i's recommendation degree $d(s, a) < 0$, which indicates subordinate agent j's current choice is not recommended, and agent j should increase the exploration rate to have more chance to select the recommended actions next time. On the other hand, if the recommendation degree $d(s, a) > 0$, it indicates the subordinates' current choice is recommended. Thus subordinate agent j decreases its exploration rate to avoid selecting discouraging actions in the future. For both cases, the adjustment degree varies depending on the absolute value of the state-action pair's recommendation degree. Formally each subordinate agent updates its exploration rate as follows,

$$\epsilon_j = \epsilon_j * (1 - d(s, a) * \gamma) \quad (18)$$

where γ is a weighting factor controlling the influence degree of the supervisor's suggestion on the subordinates' exploration rates.

Finally, given the current state s, each subordinate agent j chooses its action from those actions whose corresponding state-action pair do not belong to the set F of rules based on the corresponding set of E-values according to the ϵ-greedy mechanism. Specifically each agent chooses its action with the highest E-value with probability $1-\epsilon_j$ to exploit the action with best performance currently (randomly selection in case of a tie), and makes random choices with probability ϵ_j to explore new actions with potentially better performance.

4 EXPERIMENTAL SIMULATION

In this section, we start with evaluating the norm emergence performance of our approach HHLS under different types of games by comparing with the state-of-the-art strategies. Following that we explore the influence of some parameters on norm emergence. Unless otherwise mentioned, all simulation results are obtained under a population of 500 agents within a small-world network. The average connection degrees of small-world and scale-free network are set to 6. All results are averaged over 1000 runs. The parameter settings are shown in Table 4.

Table 4. The initial value of parameters.

Parameters	α	ϵ	β	γ	ρ	θ
Value	0.99	0.93	0.1	0.05	0.01	0.005

4.1 Performance evaluation

We compare our approach HHLS with two previous works: hierarchical learning in [21] and social learning in [2]. All these three learning approaches are within the same social learning environment, i.e., each agent is allowed to interact with only one of its neighbours each round. The work in [2] is the representative state-of-the-art approach tackling norm emergence problem under multiagent social learning framework without considering any hierarchical organization. The work in [21] is the most recent approach introducing hierarchical learning into multiagent social learning framework to improve norm emergence efficiency. Four representative 6-action games are considered shown from Table 5 to Table 8.

Table 5. The payoff matrix of coordination game.

		Agent 2's actions					
		a	b	c	d	e	f
Agent 1's actions	a	1	-1	-1	-1	-1	-1
	b	-1	1	-1	-1	-1	-1
	c	-1	-1	1	-1	-1	-1
	d	-1	-1	-1	1	-1	-1
	e	-1	-1	-1	-1	1	-1
	f	-1	-1	-1	-1	-1	1

4.1.1 Coordination game (CG)

We first consider agents playing a 6-action coordination game (Table 5) in which there exist six norms. Agents are preferred to choose the same action. Figure 2 shows the dynamics of the average payoffs of agents with the number of rounds averaged for the three learning approaches. We can observe that all learning methods enable agents to achieve an average payoff of 1. Our hierarchically heuristic learning strategy converges faster than the hierarchical learning method [21], and the social learning method [2] is the slowest. This is because HHLS enables supervisor to influence subordinate agents in a more efficient manner, thus accelerating norm emergence.

4.1.2 Anti-coordination game (ACG)

Similarly, we consider agents playing a 6-action anti-coordination game (Table 6) in which there also exist six equivalently optimal norms. However, different from coordination game, each norm requires agents to choose different actions. Figure 3 shows the dynamics of the average payoffs using three learning methods. We can observe that both social learning [2] and HHLS enable agents to achieve an average payoff of 1, while the hierarchical learning fails. Besides, our HHLS converge faster than the social learning approach [2], which justifies the efficiency of introducing a hierarchical learning

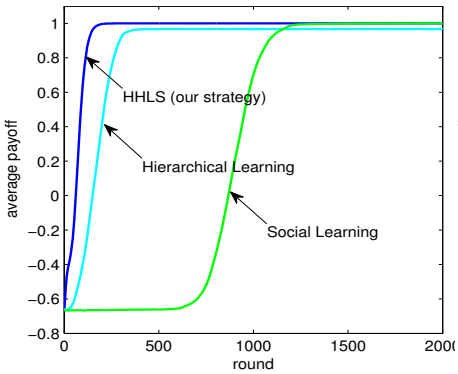

Figure 2. The dynamics of the average payoffs of agents in coordination games under different strategies

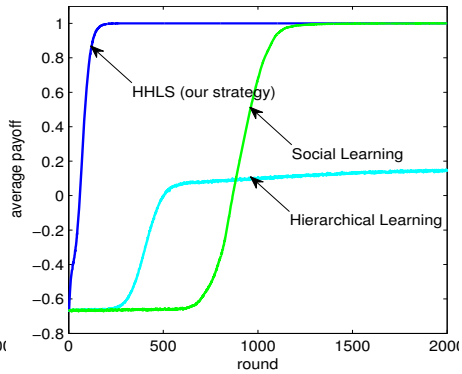

Figure 3. The dynamics of the average payoffs of agents in anti-coordination games under different strategies

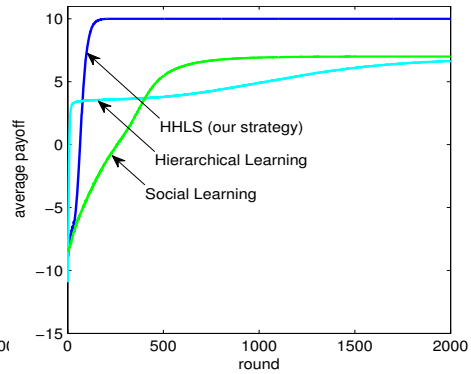

Figure 4. The dynamics of the average payoffs of agents in CGHP under different strategies

structure. For the hierarchical learning [21], it does not distinguish the state information and thus cannot adaptively select different actions for different states.

Table 6. The payoff matrix of anti-coordination game.

		Agent 2's actions					
		a	b	c	d	e	f
Agent 1's actions	a	-1	-1	-1	-1	-1	1
	b	-1	-1	-1	-1	1	-1
	c	-1	-1	-1	1	-1	-1
	d	-1	-1	1	-1	-1	-1
	e	-1	1	-1	-1	-1	-1
	f	1	-1	-1	-1	-1	-1

4.1.3 *Coordination game with high penalty (CGHP)*

Next, we consider 100 agents play a 6-action coordination game with high penalty (Table 7), in which there exist four optimal norms and two suboptimal norms. In this kind of games, agents are vulnerable to converge to suboptimal norms due to the existence of high mis-coordination cost (-30). Figure 4 shows the dynamics of the average payoffs of agents with the number of rounds for the three learning approaches. We can see that only HHLS enables agents to achieve an average payoff of 10 (i..e, converging to one optimal norm). The other two learning methods converge to one of the suboptimal norms, and they also converge slower than HHLS. We hypothesize the superior performance of HHLS is due to the integration of optimistic assumption during strategy update (to overcome mis-coordination cost effect) and efficient hierarchical supervision (to accelerate norm emergence speed).

4.1.4 *Fully stochastic coordination game with high penalty (FSCGHP)*

Last, we consider agents playing a 6-action fully stochastic coordination game with high penalty (Table 8). In FSCGHP, each outcome is associated with two possible payoffs and the agents receive one of them with probability 0.5, which models the uncertainty of the interaction results. This game is in essence the same with the CGHP in which there also exist four optimal norms and two suboptimal norms. But it is more complex and difficult to emerge norms due to the stochasticity of the environments. Figure 5 shows the dynamics of the average payoffs of agents as the number of rounds for the three learning strategies. We can observe that in this challenging game, only HHLS enables agents to achieve an average payoff of 10 (one optimal norm is converged to). In contrast, the other two learning strategies converge to one of the suboptimal norms with a slower convergence rate. Finally it is worth to mention that if the size of norm space is further increased, the social learning method [2] and hierarchical learning method [21] cannot converge (to a suboptimal norm) within 10000 runs. However HHLS still can support converging to one optimal norm within approximately 200 rounds. The influence of action size will be discussed in Section 4.2.2 in details.

Table 7. The payoff matrix of coordination game with high penalty.

		Agent 2's actions					
		a	b	c	d	e	f
Agent 1's actions	a	10	0	-30	-30	0	-30
	b	0	7	0	0	0	0
	c	-30	0	10	-30	0	-30
	d	-30	0	-30	10	0	-30
	e	0	0	0	0	7	0
	f	-30	0	-30	-30	0	10

4.2 Influence of key parameters

In this section, we turn to investigate the influence of key parameters on the performance of norm emergence. We present the results for hierarchically heuristic learning under the small-world network and the CGHP game. The rest of parameters follow the same settings in Section 4 except the parameter being evaluated is changed.

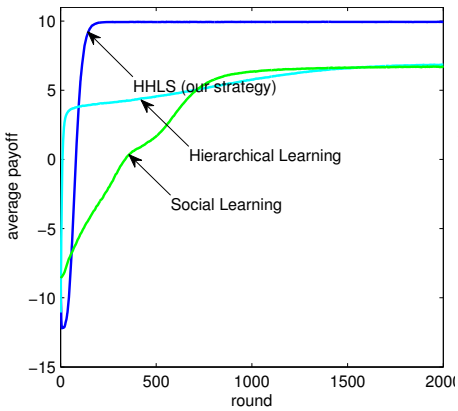

Figure 5. The dynamics of the average payoffs of agents in FSCGHP under different strategies

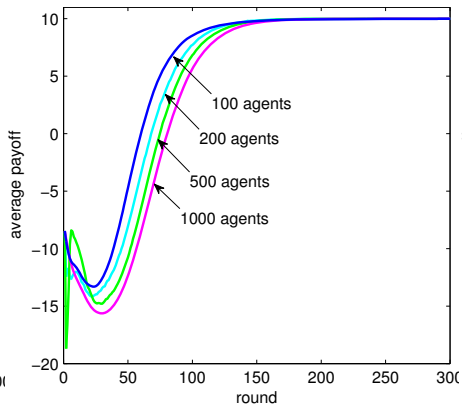

Figure 6. The influence of population size

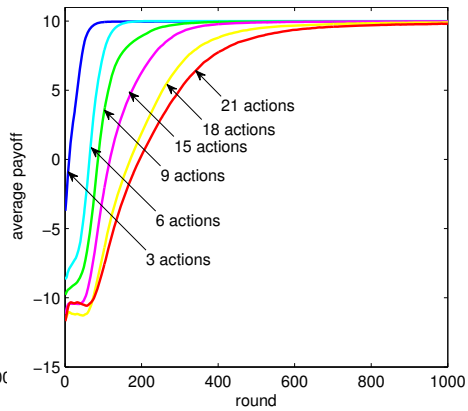

Figure 7. The influence of action size

Table 8. The payoff matrix of fully stochastic coordination game with high penalty.

1's payoff / 2's payoff		Agent 2's actions					
		a	b	c	d	e	f
Agent 1's actions	a	12/8	5/-5	-20/-40	-20/-40	5/-5	-20/-40
	b	5/-5	14/0	5/-5	5/-5	5/-5	5/-5
	c	-20/-40	5/-5	12/8	-20/-40	5/-5	-20/-40
	d	-20/-40	5/-5	-20/-40	12/8	5/-5	-20/-40
	e	5/-5	5/-5	5/-5	5/-5	14/0	5/-5
	f	-20/-40	5/-5	-20/-40	-20/-40	5/-5	12/8

4.2.1 *Influence of population size*

The influence of population size is shown in Figure 6. We can clearly observe the norm emergence efficiency is reduced as the increase of the population size. Given the cluster size unchanged, the number of clusters increase as the population size becomes larger. Thus it takes more time for each supervisor to coordinate between each other, and also more efforts are required for supervisors to guide all of their subordinate agents towards a consistent norm.

4.2.2 *Influence of action size*

Figure 7 shows the dynamics of the average payoffs of agents for different action sizes. We can see that the average convergence rate is decreased as the increase of the action space. This is reasonable because the coordination space becomes larger when the action size increases. Besides, larger action size usually results in more chances of mis-coordination cost and suboptimal norms, which additionally increases the coordination difficulty for agents toward a consistent norm. Finally it is worth noting that with the increase the action space, our framework can still efficiently support norm emergence without significantly degrading the performance (supporting norm emergence within 1000 rounds for all cases). In contrast, in previous socially learning framework without utilizing a hierarchical organization [2], the norm convergence speed is decreased significantly when the action space is increased.

Table 9. The average number of rounds needed before convergence under different network topologies.

Convergence Speed		Game type			
		CG	ACG	CGHP	FSCGHP
Network topology	Grid	142	141	131	153
	Ring	144	146	135	157
	Random	146	136	122	162
	Small-world	141	141	124	162
	Scale-free	149	144	129	163

4.2.3 *Network topology*

We evaluate the influence of five different networks: random network, grid network, ring network, small-world and scale-free network. Table 9 shows the average number of rounds needed before convergence. We find that hierarchical social learning framework is robust to different network topologies. HHLS enables agents to converge to norms in approximately the same number of runs under all the above five network topologies for different types of games.

Table 10. The influence of neighborhood size

Neighbour size	2	6	8	10	20	30	50	99
Convergence Rate	144	141	143	139	141	141	142	139

4.2.4 *Influence of neighborhood size*

We empirically evaluate the influence of neighborhood size varying it from 2 up to 99 (fully connected) with a population of 100 agents.

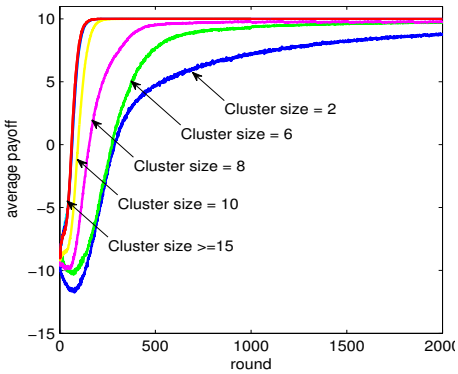

Figure 8. The influence of cluster size

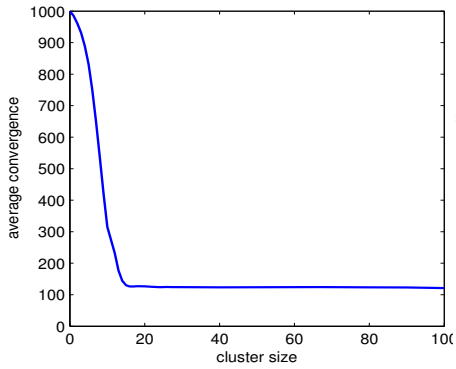

Figure 9. The average number of rounds needed before convergence under different cluster sizes

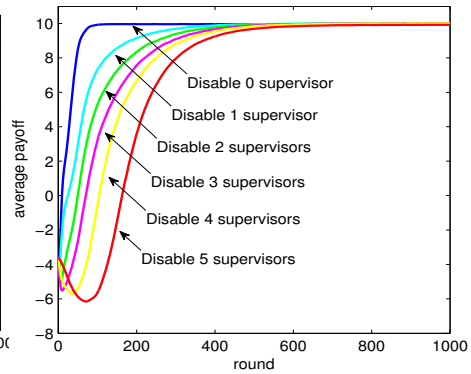

Figure 10. The influence of disabling supervisors

Table 10 shows the the average number of rounds needed before convergence for different neighborhood sizes. We can see that the average number of rounds required is stabilized around 140 rounds. This finding is different from the results usually observed in the traditional socially learning framework without a hierarchical structure [2]. This is because in hierarchical social learning framework, each supervisor supervises and guides a cluster of subordinate agents, which can overcome the low connectivity disadvantage when the neighbourhood size is small.

4.3 Influence of cluster size

One unique feature of the hierarchical social learning framework is the division of clusters of agents. Figure 8 and 9 show the influence of cluster size on norm emergence with a population of 100 agents. From Figure 8, we can see that the norm emergence rate is gradually increased as the increase of the cluster size, and stabilized when the cluster size is larger than 15. This phenomenon can be observed more clearly in Figure 9, which shows the average no. of rounds needed before convergence is reduced with the increase of cluster size and stabilized around 100 rounds.

When the cluster size is increased to 100 in the extreme case, it is essentially reduced to centralized control in which only one supervisor agent supervises all the rest of agents. In this case, all the communication and computation burden would fall on this single supervisor agent. When the cluster size is 1, it is essentially equivalent with the case of the traditional social learning without a hierarchical structure. When the cluster size varies between 1 and 100, with the increase of the cluster size, each supervisor agent can supervise more subordinate agents and thus it is easier for agents to coordinate among each other. However, as the cluster size exceeds certain threshold, the advantage of centralized supervision diminishes. This property is desirable since the same level performance as fully centralized supervision can be achieved under distributed supervision, which not only increases the robustness of the HHLS and the framework itself but reduces the communication and computation burden of supervisor agents.

Next we examine the robustness of HHLS in details by investigating the following questions: whether a consist norm can still rapidly emerge and how is the emergence efficiency changed when certain amount of supervisors are disabled? Figure 10 shows the dynamics of expected payoffs of agents with some supervisors disabled, and the results are averaged over 6-action coordination game with high penalty. We can see that hierarchically heuristic learning still enables agents to converge to a consist norm when certain amount of supervisors are disabled. Though the convergence rate is gradually decreased as the increased of the number of disabled supervisors, better performance can still be achieved than the traditional social learning framework. This is expected since those subordinate agents without supervisors can only learn based on their local information, and the hierarchical social learning framework would be reduced into the traditional social learning framework when all supervisors are disabled.

5 CONCLUSION AND FUTURE WORK

We propose a hierarchically heuristic learning strategy to ensure efficient norm emergence in different distributed multiagent environments. Extensive simulation shows that our strategy can enable agents to reach consistent norms more efficiently and in a wider variety of games compared with previous approaches. The influence of different key parameters (e.g., population size, action space, neighbourhood size and network topology) is also investigated in details. We also evaluate the influence of centralized and decentralized hierarchically design by examining the effects of different cluster sizes (e.g., different number of supervisors). We find that sufficient degree of distributed supervision (large number of supervisors) can achieve the same performance as fully centralized supervision (only one supervisor), and thus making the HHLS robust towards the failure of certain supervisors.

In this paper, we divide subordinate agents randomly into several clusters. As future work, it is worthwhile investigating whether there exists an optimal way of clustering agents in terms of maximizing norm emergence rate and how an optimal clustering structure can be formed automatically among agents.

6 ACKNOWLEDGEMENTS

This work is partially supported by the subproject of the National Key Technology R&D Program of China (No.: 2015BAH52F01-1), National Natural Science Foundation of China (No.: 61304262) and Tianjin Research Program of Application Foundation and Advanced Technology (No.: 16JCQNJC00100).

REFERENCES

[1] Monireh Abdoos, Nasser Mozayani, and Ana LC Bazzan, 'Holonic multi-agent system for traffic signals control', *Engineering Applications of Artificial Intelligence*, **26**(5), 1575–1587, (2013).

[2] Stéphane Airiau, Sandip Sen, and Daniel Villatoro, 'Emergence of conventions through social learning', *Autonomous Agents and Multi-Agent Systems*, **28**(5), 779–804, (2014).

[3] Reinaldo AC Bianchi, Carlos HC Ribeiro, and Anna Helena Reali Costa, 'Heuristic selection of actions in multiagent reinforcement learning.', in *International Joint Conference on Artificial Intelligence*, pp. 690–695, (2007).

[4] Jordi Campos, Marc Esteva, Maite López-Sánchez, Javier Morales, and Maria Salamó, 'Organisational adaptation of multi-agent systems in a peer-to-peer scenario', *Computing*, **91**(2), 169–215, (2011).

[5] Jianye Hao and Ho-fung Leung, 'The dynamics of reinforcement social learning in cooperative multiagent systems.', in *Proceedings of 23rd International Joint Conference on Artificial Intelligence*, volume 13, pp. 184–190, (2013).

[6] Jianye Hao, Jun Sun, Dongping Huang, Yi Cai, and Chao Yu, 'Heuristic collective learning for efficient and robust emergence of social norms', in *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, pp. 1647–1648, (2015).

[7] Spiros Kapetanakis and Daniel Kudenko, 'Reinforcement learning of coordination in heterogeneous cooperative multi-agent systems', in *Adaptive Agents and Multi-Agent Systems II*, 119–131, Springer, (2005).

[8] Mihail Mihaylov, Karl Tuyls, and Ann Nowé, 'A decentralized approach for convention emergence in multi-agent systems', *Autonomous Agents and Multi-Agent Systems*, **28**(5), 749–778, (2014).

[9] Partha Mukherjee, Sandip Sen, and Stéphane Airiau, 'Norm emergence under constrained interactions in diverse societies', in *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 2*, pp. 779–786. International Foundation for Autonomous Agents and Multiagent Systems, (2008).

[10] Jorge M Pacheco, Arne Traulsen, and Martin A Nowak, 'Coevolution of strategy and structure in complex networks with dynamical linking', *Physical review letters*, **97**(25), 258103, (2006).

[11] Bastin Tony Roy Savarimuthu, Rexy Arulanandam, and Maryam Purvis, 'Aspects of active norm learning and the effect of lying on norm emergence in agent societies', in *Agents in Principle, Agents in Practice*, 36–50, Springer, (2011).

[12] Onkur Sen and Sandip Sen, 'Effects of social network topology and options on norm emergence', in *Coordination, Organizations, Institutions and Norms in Agent Systems V*, 211–222, Springer, (2010).

[13] Sandip Sen and Stéphane Airiau, 'Emergence of norms through social learning.', in *International Joint Conference on Artificial Intelligence*, volume 1507, p. 1512, (2007).

[14] Yoav Shoham and Moshe Tennenholtz, 'On the emergence of social conventions: modeling, analysis, and simulations', *Artificial Intelligence*, **94**(1), 139–166, (1997).

[15] Daniel Villatoro, Jordi Sabater-Mir, and Sandip Sen, 'Social instruments for robust convention emergence', in *International Joint Conference on Artificial Intelligence*, volume 11, pp. 420–425, (2011).

[16] Daniel Villatoro, Jordi Sabater-Mir, and Sandip Sen, 'Robust convention emergence in social networks through self-reinforcing structures dissolution', *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, **8**(1), 2, (2013).

[17] Daniel Villatoro, Sandip Sen, and Jordi Sabater-Mir, 'Topology and memory effect on convention emergence', in *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology-Volume 02*, pp. 233–240. IEEE Computer Society, (2009).

[18] Jörgen W Weibull, *Evolutionary game theory*, MIT press, 1997.

[19] Dayong Ye, Minjie Zhang, and Danny Sutanto, 'A hybrid multiagent framework with q-learning for power grid systems restoration', *Power Systems, IEEE Transactions on*, **26**(4), 2434–2441, (2011).

[20] H Peyton Young, 'The economics of convention', *The Journal of Economic Perspectives*, **10**(2), 105–122, (1996).

[21] Chao Yu, Hongtao Lv, Fenghui Ren, Honglin Bao, and Jianye Hao, 'Hierarchical learning for emergence of social norms in networked multiagent systems', in *AI 2015: Advances in Artificial Intelligence*, 630–643, Springer, (2015).

[22] Chao Yu, Hongtao Lv, Sandip Sen, Jianye Hao, Fenghui Ren, and Rui Liu, 'An adaptive learning framework for efficient emergence of social norms', in *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, pp. 1307–1308. International Foundation for Autonomous Agents and Multiagent Systems, (2016).

[23] Chao Yu, Guozhen Tan, Hongtao Lv, Zhen Wang, Jun Meng, Jianye Hao, and Fenghui Ren, 'Modelling adaptive learning behaviours for consensus formation in human societies', *Scientific reports*, **6**, (2016).

[24] Chao Yu, Minjie Zhang, Fenghui Ren, and Xudong Luo, 'Emergence of social norms through collective learning in networked agent societies', in *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, pp. 475–482, (2013).

[25] Chongjie Zhang, Sherief Abdallah, and Victor Lesser, 'Integrating organizational control into multi-agent learning', in *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pp. 757–764, (2009).

[26] Chongjie Zhang, Victor Lesser, and Sherief Abdallah, 'Self-organization for coordinating decentralized reinforcement learning', in *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1*, pp. 739–746, (2010).