Journal of Applied Sciences

ISSN 1812-5654

Journal of Applied Sciences 9 (11): 2056-2066, 2009
ISSN 1812-5654

Multiagent Reniforcement Learning in Extensive Form Games with Perfect Information

A. Akramizadeh, A. Afshar and Mohammad-B Menhaj
Computational Intelligence and Large Scale System Research Laboratory,
Department of Electrical Engineering, Tehran Polytechnic, Tehran, Iran

Abstract: In this study, Q-learning has been extended to multiagent systems where a kind of ranking in action selection has been set among several self-interested agents. The process of learning is regarded as a sequence of situations modeled as extensive form games with perfect information. Each agent decides on its actions, in different subgames the higher level agents have decided on, based on its preferences affected by the lower level agents' preferences. These modified Q-values, called associative Q-values, are the estimations of possible utilities gained over a subgame with respect to the lower level agents'game preferences. A kind of social convention can be addressed in extensive form games providing the ability to better deal with multiplicity in equilibrium points as well as decreasing complexity of computations with respect to normal form games. This new process is called extensive Markov game which is proved to be a kind of generalized Markov decision process. It is also provided a comprehensive review on the related concepts and definitions previously developed for normal form games. Some analytical discussions on the convergence and the computation space are also included. A numerical example affords more elaboration on the proposed method.

Key words: Multiagent reinforcement learning, extensive form game, normal form game, Nash equilibrium points, subgame perfect equilibrium points

INTRODUCTION

Multiagent systems are a group of entities interacting with each other and with a common environment, perceiving with their sensors and act upon it through their actuators. Internal interaction is a key point in most of real world applications, especially those aroused during recent years in the field of social problems (Weiss, 1999), robotic teams, distributed control systems, collaborative decision support systems, resource management, data mining, etc. One of the most important issues in this area is learning when there is not considerable information on the environment and the interactions among agents. Although the agents in a multiagent system can be programmed with behaviors designed in advance, but it is often necessary to learn new behaviors such that the performance of the agent or the whole multiagent system gradually improves (Stone and Veloso, 2000). This is usually because of the complexity or insignificant information about the environment and the effects of agents' behaviors on environment and other agents. It causes a priori design of a good agent behavior difficult, or even, impossible. Moreover, in an environment that changes over time, a hardwired behavior may become inappropriate.

Various paradigms in the field of multiagent learning was proposed such as policy gradient method (Sutton *et al.*, 2000), evolutionary learning (Panait and Luke, 2005) and reinforcement learning among which integration of game theory and reinforcement learning seems to be the most promising solution. As a learning method that does not need a model of its environment and can be used online, Reinforcement Learning (RL) has been extensively used in single agent problems. Simplicity and generality of the algorithms make RL attractive even for multiagent systems, where agents know little about other agents and the environment.

In most of initial studies, single-agent RL were applied directly without modifications. Such approaches treat other agents in the system as a part of the environment, ignoring the differences between responsive agents and passive environment. A simplified version of Q-learning to estimate agents' value-functions has been proposed (Claus and Boutilier, 1998). This method fails to converge in some difficult coordination problems, and some improvements aiming to overcome these problems were published (Kapetanakis and Kudenko, 2002). However, cooperative learning through sharing sensation, episodes and learned policies was experimentally shown to outperform the independent learning in multiagent systems (Tan, 1993).

Corresponding Author: Ali Akramizadeh, Computational Intelligence and Large Scale System Research Laboratory, Department of Electrical Engineering, Tehran Polytechnic, Tehran, Iran

The contribution of game theory to MRL algorithms has been reviewed in some papers. A critical survey on some state of the art approaches were presented (Shoham *et al*., 2003) resulting in four well-defined problems in MRL. They, later, tried to start a set of discussions about MRL (Shoham *et al*., 2006). Most of the review papers can be considered as comprehensive reports on MRL methods. The most recent one has been accomplished by Busoniu *et al*. (2008). They, first, classified briefly recent developments on multiagent learning from the basis, including direct policy search, reinforcement learning, and game theory. Then, they categorized the leaning algorithms with respect to the types of tasks they can address ranging from fully cooperative to fully competitive.

Generally, game theory provides a convenient framework to study a number of interactive agents trying to maximize their outcomes which generally focus on stateless (static) situations. They may be set to select their actions simultaneously (flat) or make their decisions sequentially (hierarchical). Simultaneous decision making can be modeled through normal form games. Its applicability is restricted due to a number of cumbersome constraints such as equal resources, authorities, requirements, etc. Nevertheless, normal form games provide easier modeling capabilities which make it interesting for being used in multiagent reinforcement learning. Hu and Wellman (1998) proposed Nash-Q for general-sum Markov games with simultaneous action selection. Unfortunately, their method is guaranteed to converge only under very restrictive conditions. Littman (2001) proposed a new method, which relaxes these limitations by adding some additional (a priori) information about the roles of the agents in the system. Wang and Sandholm (2002) proposed a method that is guaranteed to converge with any team Markov game to the optimal Nash equilibrium 2). Conitzer and Sandholm (2003) presented an algorithm that converges to a Nash equilibrium in self-play and learn to play optimally against stationary opponents.

In some other approaches, a kind of sequential decision making can be addressed while learning. The premier one proposed by Littman (1996), called MinMax-Q. The game was supposed to be between two successive fully competitive agents which latter was partly modified by Asymmetric-Q (Kononen, 2004). In asymmetric-Q, the process of learning was divided into a sequence of two levels zero-sum game states between two agents, so-called leader and follower. The game was still fully competitive but the follower was restricted to select its actions subjective to the leader's preferences in each game state which basically was proposed to deal with multiplicity in equilibrium points. It was proved that optimal action selection, in each game state, can be accomplished via Stackelberg' equilibrium point. The encountered game state is called Stackelberg's duopoly game which is a special kind of extensive form of games.

Generally, when the game is played among several agents with alternative action selection, in game theory, it is called extensive form game. Majority of applications in multiagent systems can be considered in extensive form games which are not necessarily competitive. Even though sequential decision making is more complex to be used in MRL but, it has some key benefits due to its hierarchical structure with respect to simultaneous movements:

- Computation space is reduced due to hierarchical structure. It is not necessary to model the decision space of higher level agents during action selection
- Equilibrium points are always in pure strategies, (Osborne, 2000), which provide better convergence properties
- Computing equilibrium points is easier using backward induction algorithm
- Many problem instances are inherently hierarchical. This is true e.g., in semi-centralized multiagent systems (Kononen, 2004)

As the main contribution, in this study, Q-learning has been extended to general-sum extensive form games with perfect information. To better understand the results, it is needed to provide more elaborative definitions on previously used concepts in MRL, as a joint area in reinforcement learning and game theory, which has been enriched by some related concepts on extensive form games. Self-interested agents sequentially decide to maximize their rewards such that each agent knows about other agents' actions and rewards. After each action selection, the game state is trimmed down to one of its subgames. Agents maintain all other agents' Q-functions together with their own Q-functions. A new concept, named associative Q-value, has been introduced, which is the estimation of the possible utilities gained over a subgame with respect to subsequent agents' preferences. Agents will not need to bear in mind the higher level agents' decision space (game preferences) during action selection. They only need to know about the game preferences in different subgames, the higher level agents may decide on. This will not only decrease the computation space, but also provide a kind of social convention or communication which can better deal with multiplicity in equilibrium points.

Greedy action selection based on associative Q-values results in subgame perfect equilibrium points while it is also possible to use other directed exploration strategy such as Boltzmann. Exploring new strategies is another forte which is not clearly addressed in normal form game based MRL algorithms.

This new learning process has been named Extensive Markov game and proved to be a kind of Generalized Markov Decision Process. Finally, a numerical example and a computer simulation have been given to better present the method and the concepts.

PRELIMINARY DEFINITIONS

Learning in multiagent systems is the process in which less than fully rational players inspect for optimality over time (Fundenberg and Levine, 1998). Regarding Markov property, the whole process for dynamic tasks can be divided into a number of static situations in which agents have assigned to select one action only. The encountered situation has been widely explored in game theory. Game theory initially was introduced for reasoning in economic theory, which later has been widely used in social, political, and behavioral phenomena. It provides the necessary tools to model an interactive situation in which self interested agents interact to gain more according to their preferences and a set of game rules.

Definition 1: A game state is a situation among several self-interested agents which interact to gain more according to their preferences and a set of game rules such that each of them selects an action and the utilities are assigned when there is no other agent to select its action.

A game state can be presented in different forms among which normal and extensive form games are mostly used (Hu and Wellman, 1998).

Definition 2: A Normal game (with ordinal preferences) is a tuple $\Gamma = (P, \Sigma, R)$ consists of

- $P = \{p_1, p_2, \ldots, p_N\}$ is the set of players
- $\Sigma = \{\sigma_1, \sigma_2, \ldots \sigma_k\}$ is the set of possible joint actions $\sigma_i \in (A_1 \times A_2 \times \ldots \times A_N)$, where A_i is the set of admissible actions for agent i
- $R = \{R_i \mid R_2: \Sigma \rightarrow \Re\}$ for each player assigns the preferences over the set of joint actions

In normal form games, agents simultaneously decide on their actions.

Definition 3: An extensive game with perfect information is a tuple $\Psi = (P, \Sigma, f, R)$ where,

- $P = \{p_1, p_2, \ldots, p_N\}$ is the set of player
- $\Sigma = \{\sigma_1, \sigma_2, \ldots \sigma_k\}$ is the set of joint actions called terminal histories in extensive form game $\sigma_i \in \langle A_1 \times A_2 \times \ldots \times A_N \rangle$, where A_i is the set of admissible actions for agent i
- f(h) is the agent function that assigns an agent to every subhistory h of a terminal history. (Assign the priority in action selection)
- $R = \{R_i \mid R_2: \Sigma \rightarrow \Re\}$ for each player assign the preferences over the set of terminal histories (joint actions)

Any sequence $h = (a_1, a_2, \ldots, a_m)$ with respect to a terminal history $\sigma = (a_1, a_2, \ldots, a_N)$ where $m<N$ is called a subhistory.

Definition 4: Extended Q-function is the agent preference to select its actions with respect to other agents' preferences.

In Q-learning based MRL algorithms, an extended Q-function assigned to each agent is defined as:

$$\bar{Q} = [Q_1, \ldots, Q_i, \ldots, Q_N]$$

Equilibrium concept: In single agent case with only one decision maker, it is adequate to maximize the expected utility of decision maker. However, in games, there are many players each of which tries to maximize their own expected utilities. Thus, it is necessary to elaborate solution concepts in form of equilibrium points in which all the agents are, to some extant, satisfied and do not volunteer to decide on another movements.

The idea of Nash Equilibrium (NE) solution is that the strategy choice of each player is a best response to her opponents' play and therefore there is no need for deviation from this equilibrium point for any player alone.

Definition 5: The action profile σ^* in a strategic game with ordinal preferences is a Nash equilibrium if, for every player i and every action a_i of player i, σ^* is at least as good according to player i's preferences as the action profile (a_i, σ^*_{-i}) in which player i chooses a_i while every other player j chooses σ^*_{-i} (Osborne, 2000). Equivalently, for every player i,

$$R_i(\sigma^*) \geq R_i(a_i, \sigma^*_{-i}) \tag{1}$$

In accordance to the aforementioned definition which is well suited for normal game, Nash equilibrium in extensive game is defined as follows:

Definition 6: The strategy profile σ^* in an extensive game with perfect information is a Nash equilibrium if, for every player i and every strategy a_i of player i, the payoff of the terminal history R(σ^*) generated by σ^* is at least as good according to player i's preferences as the payoff R(a_i , σ^*_{-i}) generated by the strategy profile (a_i , σ^*_{-i}) in which player i chooses a_i while every other player other than i chooses σ^*_{-i} (Osborne, 2000). Equivalently, for each player i,

$$R_i(\sigma^*) \geq R_i(a_i, \sigma^*_{-i}) \tag{2}$$

In the aforementioned definition, there is no assumption on the structure of the game, to be flat or not. A method to deal with extensive form game is to model it as a normal game, considering only the payoff matrix. The resulting game is called a flattened game.

In order to take into account the hierarchical structure of extensive form games, subgame has been introduced by which more reasonable results on equilibrium points in extensive form games can be derived.

Definition 7: Let Ψ be an extensive game with perfect information, with player function f (Osborne, 2000). For any non-terminal history h of Ψ, the subgame Ψ(h) following the subhistory h is the following extensive game.

- **Players:** The players in Ψ
- **Terminal histories:** The set of all sequences h' of actions such that (h, h') is a terminal history of Ψ
- **Player function:** The player f(h, h') is assigned to each proper subhistory h' of a terminal history
- **Preferences:** Each player prefers h' to h" if and only if (h, h') is preferred to (h, h") in Ψ

Definition 8: A Subgame Perfect Equilibrium (SPE) is a strategy profile σ^* with the property that in no subgame can any player i do better by choosing a strategy different from σ^*_i, given that every other player adheres to σ^*_{-i} (Osborne, 2000),

$$R_i(O_h(\sigma^*)) \geq R_i(O_h(a_i, \sigma^*_{-i})) \tag{3}$$

where, $O_h(\sigma)$ is the terminal history consisting of h followed by the sequence of actions generated by σ after h.

In a subgame perfect equilibrium every player's strategy is optimal.

Proposition 1: Every subgame perfect equilibrium is also a Nash equilibrium.

Proof: If h=∅, then $O_\varnothing$ (σ) = O (σ).

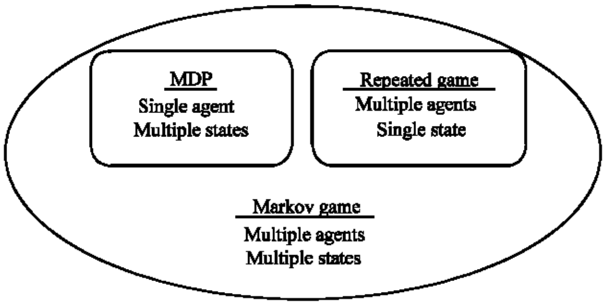

Fig. 1: Different frameworks used in reinforcement learning

The aforementioned proposition means that SPE are a subset of NE, $\{Eq\}_{SPE} \subseteq \{Eq\}_{NE}$. There may emerge some extra equilibrium points in flattened games which are not robust in steady state. This is the main reason of treating extensive form games in their hierarchical form using SPE (Osborne, 2000).

SINGLE AGENT REINFORCEMENT LEARNING

Reinforcement learning can be expressed in different frameworks. A rough but informative categorization of the learning model is depicted in Fig. 1. Finite Markov Decision Process is the basis of most of the reinforcement learning methods.

Definition 9: A Markov Decision Process (MDP) is a tuple (S, A, R, T), where:

- S is the set of all states
- A is the set of all actions
- R = {R|R: S×A → ℜ is the reward function
- T : S × A → Δ(S) is the state transition function

Δ(S) is the set of probability distributions over the set S.

The agent's objective is to learn a Markov policy, a mapping from states to probabilities of taking each available action, π : S × A → [0, 1], that maximizes the expected discounted future reward from each state s:

$$\begin{aligned} V^\pi(s) &= E\{r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \ldots \mid s_t = s, \pi\} \\ &= E\{r_{t+1} + \gamma V^\pi(s_{t+1}) \mid s_t = s, \pi\} \\ &= \sum_{a \in A} \pi(s,a)\left[r_s^a + \gamma \sum_{s'} T_{ss'}^a V^\pi(s')\right] \end{aligned} \tag{4}$$

where, π (s, a) is the probability with which the policy π chooses action a_∈ A in state s, and γ ∈ [0, 1] is a discount-factor. $V^\pi(s)$, is called the value of state s under policy π, and V^π is called the state-value function for π. The optimal state-value function gives the value of each state under an optimal policy:

$$\begin{aligned} V^*(s) &= \max_{\pi} V^{\pi}(s) \\ &= \max_{a\in A} E\left\{r_{t+1} + \gamma V^*(s_{t+1}) \mid s_t = s, a_t = a\right\} \\ &= \max_{a\in A}\left[r_s^a + \gamma\sum_{s'} T_{ss'}^a V^*(s')\right] \end{aligned} \tag{5}$$

Planning in reinforcement learning refers to the use of models of the environment to compute value functions and thereby to optimize or improve policies. Particularly useful in this regard are Bellman equations, such as Eq. 4 and 5, which recursively relate value functions to themselves.

The value of taking action a in state s under policy π, denoted $Q^{\pi}(s, a)$, is the expected discounted future reward starting in s, taking a and henceforth following π:

$$\begin{aligned} Q^{\pi}(s,a) &= E\left\{r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \ldots \mid s_t = s, \pi\right\} \\ &= r_s^a + \gamma\sum_{s'} T_{ss'}^a V^{\pi}(s') \\ &= r_s^a + \gamma\sum_{s'} T_{ss'}^a \sum_{a'} \pi(s',a') Q^{\pi}(s',a') \end{aligned} \tag{6}$$

The optimal action-value function is,

$$\begin{aligned} Q^*(s,a) &= \max_{\pi} Q^{\pi}(s,a) \\ &= r_s^a + \gamma\sum_{s'} T_{ss'}^a \max_{\pi} Q^*(s',a') \end{aligned} \tag{7}$$

It was shown finding the optimal policy is equal to finding the optimal state-action value function through the following recursive equation (Watkins, 1989),

$$Q(s,a) := (1-\alpha)Q(s,a) + \alpha\left(r_s^a + \max_{a'} Q(s',a')\right) \tag{8}$$

MULTIAGENT REINFORCEMENT LEARNING

Markov Games (MG) (Owen, 1995) is a generalized framework that can be use to extend single agent into multiple interactive agents in multiagent systems.

Definition 10: A Markov game with perfect information is a tuple (G, P, Σ, R, T) where,

- G is the set of all game states
- P = $\{p_1, p_2, \ldots, p_N\}$ is the set of player
- Σ = $\{\sigma_1, \sigma_2, \ldots \sigma_k\}$ is the set of possible joint actions $\sigma_i \in (A_1 \times A_2 \times \ldots \times A_N)$, where A_i is the set of admissible actions for agent i
- R = $\{R_i \mid R_i\colon G \times \Sigma \rightarrow \Re\}$is the reward function
- T : G × Σ → Δ(G) is the state transition function

It has been presented (Littman, 1996) that MG is included in a more general framework called Generalized Markov Decision Process (GMDP).

Definition 11: A generalized Markov decision process is a tuple $\langle G,\Sigma,T,R,N,\gamma,\oplus,\otimes\rangle$ where the fundamental quantities are a set of games G, a finite set of actions σ, a transition function $T: G\times\Sigma \rightarrow \Delta(G)$, a reward function $R: G\times\Sigma \rightarrow \Re$, a next-state function N mapping G×Σ to finite subsets of G, a discount factor γ, a summary operator ⊕ that defines the value of transitions based on the value of the successor game, and a summary operator ⊕ that defines the value of a state based on the values of all state-action pairs (Littman, 1996).

One of the basic algorithms in multiagent Q-learning is proposed by (Hu and Wellman, 1998), called Nash-Q. Nash-Q was proved to be convergent to the unique equilibrium point of the game. It provides the most significant concepts in MRL techniques based on normal form games. Agents decide on their actions to reach the equilibrium point of the current game state. Even though, it is one of the basic methods in MRL, but some restrictive assumptions hinder widespread use of it,

- Each agent computes the Nash point independently which may cause divergence in the presence of multiple equilibrium points
- Each agent must record all the other agents Q-values.
- Nothing has been proposed to explore new joint actions, (exploration vs. exploitation)
- Computing the Nash equilibrium is very complex when the number of agents increases

The proposed reinforcement learning is based on well-known Q-learning.

$$\begin{aligned} Q_{t+1}^i(g_t,a_1,\ldots,a_N) &= (1-\alpha_t)Q_t^i(g_t,a_1,\ldots,a_N) \\ &\quad + \alpha_t\left[r_t^i(g_t,a_1,\ldots,a_N) + \gamma \,\mathrm{Nash}\, Q_t^i(g')\right] \end{aligned} \tag{9}$$

where, $\mathrm{Nash}\, Q_t^i(g')$ is the Nash equilibrium point value in the next game state g' for the learning agent i.

A wide class of the problems cannot be modeled through normal form games. As it was previously mentioned, it is not always proper to flatten an extensive form games. Actually, MRL is a complex problem and can be considered as a large scale system. Conventionally, when the number of the parameters is enormous, it is more practical to tackle with the problem in a hierarchical form.

Sequential decision making, to some extent, has been implemented in MRL. It was first proposed by Littman (1996) where the structure of the game state has been supposed to be in a special kind of two levels zero-sum game with leader and follower. It was called Minimax-Q which the process of learning was named Alternating Markov Game (AMG). Later, in Asymmetric-Q, Stackelberg' equilibrium point was used for joint action selection. In the proposed method, the follower was forced to pursue the leader, somehow deal with multiplicity in equilibrium points. Some drawbacks in the method are,

- Agents are divided only in two levels, leader and follower
- Generalization of the algorithm to a group of leaders and a group of followers is not as easy as it is supposed and needs more investigations
- Nothing has been proposed to explore new joint actions
- The equilibrium concept is only introduced for zero-sum games

The last assumption is a very restrictive one, since there are a few real life applications in which the leader compete the follower. Majority of applications in multiagent systems are in a hierarchical form which are not necessarily competitive. However, the algorithm is proposed based on Q-learning,

$$Q_{t+1}^{i}(g_t, a_1, a_2) = (1-\alpha_t)Q_t^{i}(g_t, a_1, a_2) + \alpha_t\left[r_t^{i}(g_t, a_1, a_2) + \gamma SE^{i}(g')\right], \quad i = 1,2 \tag{10}$$

where, $SE^i(g')$ is the Stackelberg's equilibrium value of the next game g' for the learning agent i.

In this study, Extensive Markov Game (EMG) as another framework in MRL has been introduced as an extension to AMG.

Definition 12: In extensive Markov game agents sequentially decide on their actions which the process can be explained as a tuple $\Psi = \langle G, P, \Sigma, R, T\rangle$ where,

- G is the set of game states
- P is the set of players ordered based on their priority in action selection $\langle p_1, \ldots, p_N\rangle$
- $\Sigma = \{\sigma_1, \sigma_2, \ldots \sigma_k\}$ is the set of terminal histories $\sigma_i \in \langle A_1 \times A_2 \times \ldots \times A_N\rangle$, where A_i is the set of admissible actions for agent i
- T is the state transition function, $T: G \times \Sigma \to \Delta(G)$ where $\Delta(G)$ is the set of probability distributions over the set G,
- R is the reward function $R = \{R_i \mid R_i : G \times \Sigma \to \Re\}$

The objective for the learning agent i is to select its actions such that rewards over the whole game is maximized. The game states in EMG are modeled in extensive form games with perfect information which can be presented as,

$$\overbrace{g_1^0 \xrightarrow{\sigma_1^0} \cdots \xrightarrow{\sigma_N^0} g_N^0}^{g^0} \xrightarrow{\sigma^0} \overbrace{g_1^1 \xrightarrow{\sigma_1^1} \cdots \xrightarrow{\sigma_N^1} g_N^1}^{g^1} \xrightarrow{\sigma^1} \cdots \xrightarrow{\sigma^F} \overbrace{g_1^F \xrightarrow{\sigma_1^F} \cdots \xrightarrow{\sigma_N^F} g_N^F}^{g^F}$$

where, σ_i^m is the action performed by agent i, at subgame g_i^m, i = 1,…,N which is in game state $g^m \in \{g^0, \ldots, g^F\}$.

Agents' rewards are influenced by two factors. One of them is related to the way agents act to reach another game state, π_g and the second one is related to the interaction among agents in a game states to reach one of the possible equilibria, π_{eq}. Learning happens over the game states which is a Markov process. On the other hand, decision making in game state is a semi-Markov process which is related to acting optimally. Actually, the equilibrium policy for learning agent i is a function of possible histories, $\pi_{eq}^i : h_{i-1} \times A_i \to [0,1]$, where, h_{i-1} is all the possible history ending to agent i-1, and A_i is the set of admissible actions for agent i.

Finally, each agent is concerned with a composition policy $\pi_i = \pi_{g_i} \circ \pi_{eq_i}$ planning to maximize its expected discounted reward over the set of games while interacting with other agents to reach the equilibrium points.

$$E^{\pi_1, \ldots, \pi_i, \ldots, \pi_N}\left[\sum_{k=0}^{\infty} \gamma^t r_i(g^k) \mid g^0\right] \tag{11}$$

where, $r_i(g^k)$ is the accumulated reward for learning agent i at game state g^k.

Rewards are calculated at the end of each game state. This will not affect the learning algorithm, since by definition, the learning agent will not decide on any new actions until the next game state.

$$r_i(g^k) = \{r_i^1(g^k) + \ldots + r_i^N(g^k)\} \tag{12}$$

where, $r_i^j(g_k)$ is the reward of agent i after agent j, j=1,…,N, selects its action in game g^k.

In this study, it is assumed that agents will only gain when the last agent decides on its action. In other words, reward function is introduced such that $r_i^j(g^k) = 0 \quad j = 1, \ldots, N-1$. Thus, $r_i(g^k) = r_i^j(g^k)$.

State transition of the current subgame $\hat{g}_i \in G \times A_1 \times \ldots \times A_{i-1}$ for agent i is,

$$T_{\hat{g}_i g'} = \sum_{\hat{\sigma}} T(g' | \hat{g}_i, \hat{\sigma}) \quad \text{for all} \quad g' \in G, \tag{13}$$

where, $T(g' | \hat{g}_i, \hat{\sigma})$ is the probability that the game terminates to g' starting from $\hat{g}_i$ according to the set of all possible sequences of actions $\hat{\sigma} \in A_i \times \ldots \times A_N$. This kind of model is called here, an extensive model in light of the multi step model (Sutton *et al*., 1999).

The Max operator is proved to be non-expansion (Littman, 1996) and conventionally has been used to maximize reward over game states. It is now necessary to introduce another operator to maximize rewards in a game state.

Definition 13: Associative Q-value is the expected payoff gained by the leader of a subgame $\hat{g}_i$ by selecting an action, over the possible set of followers' actions with respect to their preferences.

$$Q_i^{Asc}(\hat{g}_i, a_i) = \sum_{a_{i+1} \in A_{i+1}} \cdots \sum_{a_N \in A_N} P_i\left(a_{i+1} \ldots a_N \middle| \hat{g}_i, a_i\right) \quad (14)$$

where, $a_i \in A_i$, $\hat{g}_i \in G \times A_1 \times \ldots \times A_{N-1}$ is the current subgame where agent i is the leader and $P_i\left(a_{i+1} \ldots a_N \middle| \hat{g}_i, a_i\right)$ is the probability of selecting $a_{i+1} \ldots a_N$ by the subsequent agents of the corresponding subgame after agent i selects a_i in the current subgame $\hat{g}_i$ and

$$\sum_{a_{i+1} \in \Sigma(p_{i+1})} \cdots \sum_{a_N \in \Sigma(p_N)} P_i\left(a_{i+1} \ldots a_N \middle| \hat{g}_i, a_i\right) = 1 \quad (15)$$

$$\left|P_i(.|.)\right| \leq 1 \quad (16)$$

Based on the proposed concept, utilities (Q-values) are propagated up through the hierarchies. Finally, each agent is concerned with a set of associative Q-values related to its admissible actions.

Lemma 1: Greedy action selection based on Associative Q-values in generic games gradually converges to SPE.

Proof: It is trivial to presents the similarity of backward induction in finite horizon MDP in (Puterman, 1994) and proposed associative Q-values. Based on backward projection, (Kohlberg and Mertens, 1986), the solution in subgame, $\hat{g}_i$, is a part of the solution in the game, g.

Similarly, it was proved in other MRL based algorithms (Laslier and Walliser, 2005).

Associative Q-values are more advantageous than SPE values in MRL, since it provides the possibility of using exploration strategies such as Boltzmann. Exploration strategies are not addressed in most of the proposed MRL.

Learning agent i, at the end of each game state, updates its extended Q-table. Recall that the game state is the one with perfect information and the higher priority agent can view lower level agents' actions and rewards. The proposed update rule for learning agent i is:

$$\begin{aligned} Q_i^{k+1}(g, a_1, \ldots, a_N) &= (1-\alpha) Q_i^k(g, a_1, \ldots, a_N) \\ &\quad + \alpha\left[r_i(g) + \gamma \mathrm{SPE}_i^V\left(\bar{Q}^k(g', a_1, \ldots, a_N)\right)\right] \end{aligned} \quad (17)$$

where, $\mathrm{SPE}_i^V\left(\bar{Q}^k(g', a_1, \ldots, a_N)\right)$ is the SPE value of the ith player for the next game state g'.

The following algorithm can be used for the learning with Boltzmann exploration,

```
(1) Initialize:
    T is big
    All Q-tables are initialized to zero
    The game is initiated, $g_0$
(2) Loop from i=1 to N
    Compute the associative Q-values for agent i,
    Select an action based on associative Q-values
    Play action
(3) Calculate the return values for the resulting game state
(4) Update Q-table for each agent based on Eq. 17
(5) Decrease T
(6) If the goal is not met goto 2
(7) End
```

ANALYTICAL DISCUSSION

Playing the equilibrium solution in game states is an important issue in multiagent learning system. This is due to the theorem proved (Filar and Vrieze, 1997), stating that the Nash solution in a game state is a part of the solution of the whole game. The aforementioned theorem is the basic assumption in most of the proposed MRL algorithms.

Convergence issue: In Nash-Q it is proved that agents converge to the unique equilibrium policy which is mixed. In Asymmetric-Q, the solution is also proved to be convergent. Generally, convergence in extensive form game is faster than the normal form game, since the equilibrium points are pure, while it is not always true in normal form games.

Theorem 1: Every finite extensive game with perfect information has a pure strategy Nash equilibrium point (Nash, 1951).

Convergence in EMG can be verified through Lemma 2 which here is proved form a generic game (Laslier and Walliser, 2005). The convergence in GMDP is presented (Littman, 1996).

Lemma 2: Extensive Markov game with SPE action selection is a generalized Markov decision process.

Proof: The composition policy for each learning agent can be considered as:

$$\begin{aligned}&\pi_1:[G]\times A_1\rightarrow[0,1]\\&\pi_2:[G\times A_1]\times A_2\rightarrow[0,1]\\&\dots\\&\pi_N:[G\times A_1\ldots\times A_{N-1}]\times A_N\rightarrow[0,1]\end{aligned}\qquad(18)$$

where, $\hat{G}_i=G\times A_1\ldots\times A_i$ are the subgames that agent i may lead. Consider the game states of the GMDP to be $G=\bigcup_{i=1}^{N}\hat{G}_i$ and the action space $U=\prod_{i=1}^{N}A_i$, reward function $R=\{r_1(g),\ldots,r_N(g)\}$ and transition function $T_{gg'}=\prod_{i=1}^{N}T_{g'\hat{g}_i}$. For any game policy, π_i, the state-value function can be written as:

$$\begin{aligned}V_i^{\pi}(g)&=E^{\pi}\left[r^k+\gamma r^{k+1}+\gamma^2 r^{k+2}+\ldots\mid g^k=g,\pi\right]\\&=E^{\pi}\left[r^k+\gamma V_i^{\pi}\left(g^{k+1}\right)\mid g^k=g,\pi\right]\\&=\sum_{\sigma\in\Sigma}\pi\left(g^k,\sigma\right)\left[r_i\left(g^k\right)+\gamma\sum_{g_{k+1}}T\left(g^{k+1}\mid g^k,\sigma\right)V_i^{\pi}\left(g^{k+1}\right)\right]\end{aligned}\qquad(19)$$

where, $\pi(g^k,\sigma)=\pi_1(g^k,a_1)\times\pi_2(\hat{g}_2^k,a_2)\times\ldots\times\pi_N(\hat{g}_N^k,a_N)$, such that $\sigma=(a_{1,}a_2,\ldots,a_N)$. (Recall that, $r_i^j(g_k)=0 \quad j=1,\ldots,N-1$)

Equivalently:

$$\begin{aligned}Q_i^{\pi}(g,\sigma)&=r_i\left(g^k\right)+\gamma\sum_{g_{k+1}}T\left(g^{k+1}\mid g^k,\sigma\right)\sum_{\sigma\in\Sigma}\pi\left(g^{k+1},\sigma\right)Q_i^{\pi}\left(g^{k+1},\sigma\right)\\&=r_i\left(g^k\right)+\gamma\sum_{g_{k+1}}T\left(g^{k+1}\mid g^k,\sigma\right)V_i^{\pi}\left(g^{k+1}\right)\end{aligned}\qquad(20)$$

The optimal subgame value function for each agent is the optimal value if the SPE is selected,

$$\begin{aligned}V_i^{*}(g)&=V_i^{\pi_{SPE}}(g)\\&=E^{\pi_{SPE}}\left[r_i\left(g^k\right)+\gamma V_i^{\pi_{SPE}}\left(g^{k+1}\right)\mid g^k=g,\pi_{SPE}\right]\\&=SPE_i^V\left(\bar{Q}^{*}\right)\end{aligned}\qquad(21)$$

According to Lemma 1, greedy action selection based on associative Q-values will result in subgame perfect equilibrium which is the optimal solution in extensive form game (Definition 8). Thus,

$$Q_i^{*}\left(g^k,\sigma\right)=r_i\left(g^k\right)+\gamma\sum_{g_{k+1}}T\left(g^{k+1}\mid g^k,\sigma\right)V_i^{*}\left(g^{k+1}\right)\qquad(22)$$

Now, the well-known Q-learning can be used at the end of each game state,

$$\begin{aligned}Q_i^{k+1}(g,a_1,\ldots,a_N)&=(1-\alpha)Q_i^k(g,a_1,\ldots,a_N)\\&\quad+\alpha\left[r_i(g)+\gamma SPE_i^V\left(\bar{Q}^k(g',a_1,\ldots,a_N)\right)\right]\end{aligned}\qquad(23)$$

It is now necessary to prove that the relevant operator $V^{*}(g)=\otimes Q^{*}(g,a)$ is a non-expansion operator.

Theorem 2: SPE operator is non-expansion, where,

$$\begin{aligned}V_i^{\pi^*}(g)&=SPE_i^V\left(\bar{Q}\right)\\&=\otimes Q(g,a_1,\ldots,a_N)\end{aligned}$$

Proof: For an operator to be non-expansion, it is sufficient to satisfy the following conditions. Given functions f and f' over a finite set X,

$$\min_{x\in X}f(x)\le\underset{x\in X}{\otimes}f(x)\le\max_{x\in X}f(x)\qquad(24)$$

$$\left|\underset{x\in X}{\otimes}f(x)-\underset{x\in X}{\otimes}f'(x)\right|\le\max_{x\in X}\left|f(x)-f'(x)\right|\qquad(25)$$

where, for the learning agent i,

$$f(x)=\bar{Q}(g,\sigma)\text{ and}$$

$$\underset{\sigma\in\Sigma}{\otimes}^{(g,\sigma)}=SPE_i^V(.)$$

The distance norm for the value function is defined as:

$$\left\|V^1-V^2\right\|=\sup_x\left|V^1(x)-V^2(x)\right|$$

That can be extended to Q-function, as well (Littman, 1996),

$$\left\|\bar{Q}^1-\bar{Q}^2\right\|=\sup_g\max_i\left|Q_i^1(g,\sigma)-Q_i^2(g,\sigma)\right|$$

The first constraint is a trivial. As it was proved in (Littman, 1996), the simple max operator is a non-expansion, thus, for any learning agent i, i=1,…,N,

$$\begin{aligned}\left\|\otimes\bar{Q}^1-\otimes\bar{Q}^2\right\|&=\sup_g\left|\otimes\bar{Q}^1(g,\sigma)-\otimes\bar{Q}^2(g,\sigma)\right|\\&=\sup_g\left|SPE_i^V\left(\bar{Q}^1(g,\sigma)\right)-SPE_i^V\left(\bar{Q}^2(g,\sigma)\right)\right|\\&\le\sup_g\left|Nash_i^V\left(\bar{Q}^1(g,\sigma)\right)-Nash_i^V\left(\bar{Q}^2(g,\sigma)\right)\right|\end{aligned}$$

Due to proposition 1 which implies $\{Eq\}_{SPE}\subseteq\{Eq\}_{NE}$. Now, it suffices to prove that Nash operator is non-expansion, which has been previously proved in lemma 16 (Hu and Wellman, 2003).

Thus, the operator is non-expansion and

$$\begin{aligned}\left\|\otimes\bar{Q}^1-\otimes\bar{Q}^2\right\|&\le\sup_g\left|Nash_i^V\left(\bar{Q}^1(g,\sigma)\right)-Nash_i^V\left(\bar{Q}^2(g,\sigma)\right)\right|\\&\le\left\|\bar{Q}^1(g,\sigma)-\bar{Q}^2(g,\sigma)\right\|\end{aligned}$$

Computation space: Consider a game G of N agents with action set, A. All agents need to know about other agents especially to update its Q-function. Thus, each agent has to know all about the other agents. This results in extended Q-table, $\bar{Q}(g,a_1,...,a_N)=[Q_1 \;\; ... \;\; Q_N]$, with the size of N×|G|×|A|^N. The computation space in both extensive and normal form game is equal. But, Nash equilibrium computation in normal form game is not easy and it is still an open problem which suffers complex computations. On the other hand, backward induction in extensive form game, especially for the so-called game state, can be easily used.

The hierarchical structure reduces the action space. Agent i in each game state knows what the higher level agents have done. Thus, it only needs to decide based on N-i+1 Q-table $[Q_i \;\; ... \;\; Q_N]$. For example, the first agent should decide based on all other agents Q-values. The second agent action space is reduced to (N-1)× |G|×|A|^N. The same reduction in action space can be deduced for the later agents as well. Actually, as the agent is placed lower in the hierarchy, its action space is reduced.

Simple example: Consider a game in which three agents are interacting. Each agent has only two possible actions. The environment is divided into 10 states. Thus, there may be 10^3 possible game states. Consider a game state presented in Fig. 2. There are 7 subgames for each history including h=∅ in extensive form game. $\hat{g}_2$ is one of the two possible subgames which agent B may lead. $Q_B^{Asc}(\hat{g}_2,a_1)$ is the associative Q-value if agent B plays a$_1$. It can be calculated simply in each game state,

$$Q_C^{Asc}(\hat{g}_3,a_1)=P_{111}^C=\frac{q_{31}}{q_{31}+q_{32}}$$

where, P_{111}^C is the probability of agent C plays its first action if both agent B and agent A plays their first actions.

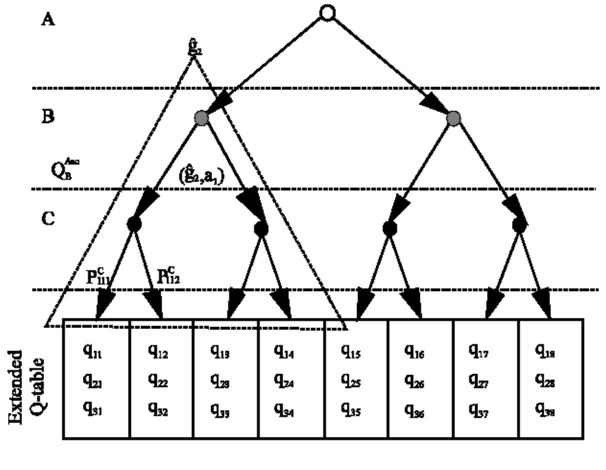
A
B
C
$\hat{g}_2$
Q_B^{Asc}
$(\hat{g}_2,a_1)$
P_{111}^C
P_{112}^C
Extended Q-table
q_{11} q_{12} q_{13} q_{14} q_{15} q_{16} q_{17} q_{18}
q_{21} q_{22} q_{23} q_{24} q_{25} q_{26} q_{27} q_{28}
q_{31} q_{32} q_{33} q_{34} q_{35} q_{36} q_{37} q_{38}

Fig. 2: A three player extensive game. Each player has two actions. Presented numerically as the outcome of the terminal history

where, $\hat{g}_3$ is the right hand subgame and P_{111}^C is the probability of agent C plays its first action if both agent B and agent A plays their first actions.

Equivalently, for agent C's second action:

$$Q_C^{Asc}(\hat{g}_3,a_2)=P_{112}^C=\frac{q_{32}}{q_{31}+q_{32}}$$

Note that, these are the associative Q-values for only one of the subgames which agent C may lead. The other associative Q-values can be calculated similarly for other subgames.

Thus, associative Q-values for agent B corresponding to its first action is,

$$\begin{aligned}Q_B^{Asc}(\hat{g}_2,a_1)&=P_{121}^B+P_{122}^B\\&=q_{21}P_{111}^C\times\left[\frac{P_{121}^C}{q_{21}+q_{23}}+\frac{P_{122}^C}{q_{21}+q_{24}}\right]+q_{22}P_{112}^C\times\left[\frac{P_{121}^C}{q_{22}+q_{23}}+\frac{P_{122}^C}{q_{22}+q_{24}}\right]\end{aligned}$$

The other associative Q-values can be calculated in a same manner.

The same can be done for agent A's associative Q-values.

$$\begin{aligned}Q_A^{Asc}(g,a_1)=&Q_{B_1}^{Asc}(\hat{g}_2,a_1)\left[P_{111}^C\Pi(q_{11})+P_{112}^C\Pi(q_{12})\right]\\&+Q_{B_1}^{Asc}(\hat{g}_2,a_2)\left[P_{121}^C\Pi(q_{13})+P_{122}^C\Pi(q_{14})\right]\end{aligned}$$

Where:

$$\begin{aligned}\Pi(q)=&Q_{B_2}^{Asc}(\hat{g}_2,a_1)\times q\times\left[\frac{P_{121}^C}{q+q_{15}}+\frac{P_{122}^C}{q+q_{16}}\right]\\&+Q_{B_2}^{Asc}(\hat{g}_2,a_2)\times q\times\left[\frac{P_{221}^C}{q+q_{17}}+\frac{P_{222}^C}{q+q_{18}}\right]\end{aligned}$$

Even though, the computations seems a little cumbersome in extensive form game, but they are easy to drive with respect to the fact that estimating Nash equilibrium point in normal game is still a complex issue in game theory (Daskalakis *et al*., 2005).

CONCLUSION

Usually, in MRL algorithms, games are supposed to be in normal forms for general-sum games. On the other hand, most of real life applications are inherently hierarchical, thus, extensive form games have been investigated so much in game theory.

In this study, Q-learning had been extended to be used in extensive form games with perfect information, using subgame perfect equilibrium points. This results in a new version of Markov games, called extensive Markov games. A new concept, called associative Q-values has been introduced which can be used in action selection which provides an estimation on SPE action. Associative Q-values are the probability of reaching a joint action with respect to subsequent agents' preferences. Using the Boltzmann operator during associative Q-values computations, a trade off between exploration and exploitation can be established which cannot easily being implemented in normal form games. Finally, it was proved that the proposed extensive Markov game is a generalized Markov decision process. It was also discussed that the action space is reduced in the proposed extensive Q-learning with respect to normal form game based algorithms.

REFERENCES

Busoniu, L., R. Babuska and B. De Schutter, 2008. A comprehensive survey of multiagent reinforcement learning. IEEE Trans. Syst. Man Cybernet. Part C: Appl. Rev., 38: 156-172.

Claus, C. and C. Boutilier, 1998. The dynamics of reinforcement learning in cooperative multiagent systems. Proceedings of the 15th National Conference of Artificial Intelligence, (AAAI'98), AAAI Press, Madison, USA., pp: 746-752.

Conitzer, V. and T. Sandholm, 2003. AWESOME: A general multiagent learning algorithm that converges in self-play and learns a best response against stationary opponents. Proceedings of the 20th International Conference on Machine Learning, (ICML'2003), Morgan Kaufmann Publishers, Washington DC, USA., pp: 83-90.

Daskalakis, C., P.W. Goldberg and C.H. Papadimitriou, 2005. The complexity of computing a nash equilibrium. Electronic Colloquium on Computational Complexity, Report No. TR05-115. http://eccc.hpi-web.de/eccc-reports/2005/TR05-115/Paper.pdf.

Filar, J. and K. Vrieze, 1997. Competitive Markov Decision Process. 12th Edn., Springer-Verlag, New York, pp: 412.

Fundenberg, D. and D.K. Levine, 1998. The Theory of Learning in Games. MIT Press, Cambridge, Massachusetts, ISBN-10: 0-262-06194-5, pp: 292.

Hu, J. and P. Wellman, 1998. Multiagent reinforcement learning: Theoretical framework and an algorithm. Proceedings of the 15th International Conference on Machine Learning, (ICML'98), Morgan Kaufmann, pp: 242-250.

Hu, J. and M.P. Wellman, 2003. Nash q-learning for general-sum stochastic games. J. Mach. Learn. Res., 4: 1039-1069.

Kapetanakis, S. and D. Kudenko, 2002. Reinforcement learning of coordination in cooperative multi-agent systems. Proceedings of the 18th National Conference on Artificial Intelligence, Edmonton, Alberta (AAAI'02), AAAI Press, Canada, pp: 326-331.

Kohlberg, E. and J. Mertens, 1986. On the Strategic Stability of Equilibria. Econometrica, 54: 1003-1038.

Kononen, V., 2004. Asymmetric multiagent reinforcement learning. Web Intell. Agent Syst. Int. J., 2: 105-121.

Laslier, J.F. and B. Walliser, 2005. A reinforcement learning process in extensive form games. Int. J. Game Theory, 33: 219-227.

Littman, M.L., 1996. Algorithms for sequential decision making. Ph.D. Thesis, Department of Computer Science, Brown University.

Littman, M.L., 2001. Friend-or-foe Q-learning in general-sum games. Proceedings of the 18th International Conference on Machine Learning, (ICML'2001), Morgan Kaufmann, pp: 322-328.

Nash, J., 1951. Noncooperative Games. Annals of Math., 54: 286-295.

Osborne, M.J., 2000. An Introduction to Game Theory. Oxford University Press, USA.

Owen, G., 1995. Game Theory. 3rd Edn., Academic Press, Orlando, Florida.

Panait, L. and S. Luke, 2005. Cooperative multi-agent learning: The state of the art. Autonomous Agents Multi-Agent Sys., 11: 387-434.

Puterman, M.L., 1994. Markov Decision Processes: Discrete Stochastic Dynamic Programming. 1st Edn., John Wiley and Sons, New York.

Shoham, Y., R. Powers and T. Grenager, 2003. Multi-agent reinforcement learning: A critical survey. Computer Science Dept., Stanford University, California.

Shoham, Y., R. Powers and T. Grenager, 2006. If multi-agent learning is the answer, what is the question? Artificial Intell., 171: 365-377.

Stone, P. and M. Veloso, 2000. Multiagent systems: A survey from the machine learning perspective. Auton. Robots, 8: 345-383.

Sutton, R., D. Precup and S. Singh, 1999. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. Artificial Intell., 112: 181-211.

Sutton, R., D. McAllester, S. Singh and Y. Mansour, 2000. Policy gradient methods for reinforcement learning with function approximation. Adv. Neural Inform. Process. Syst., 12: 1057-1063.

Tan, M., 1993. Multi-agent reinforcement learning: Independent vs. cooperative agents. Proceedings of the 10th International Conference on Machine Learning, June 27-29, University of Massachusetts, Amherst, MA, USA., pp: 330-337.

Wang, X. and T. Sandholm, 2002. Reinforcement learning to play an optimal Nash equilibrium in team Markov games. Proceedings of Advances in Neural Information Processing Systems, (NIPS'02), Vancouver, Canada, pp: 1571-1578.

Watkins, C., 1989. Learning from delayed rewards. Ph.D Thesis, Kings College, Cambridge, England.

Weiss, G., 1999. Multiagent Systems: A Modern Approach to Distributed Modern Approach to Artificial Intelligence. MIT Press, London, pp: 643.