Exploration and Exploitation Tradeoff using Fuzzy Reinforcement Learning

Seyed Mohammad Hossein Nabavi[1], Somayeh Hajforoosh[2]

[1,2] Department of Electrical Engineering, Payame Noor University, PO BOX 19395-3697 Tehran,IRAN.

ABSTRACT

Difficulty of making a balance between exploration and exploitation in multiagent environment is a dilemma that does not have a clear answer and there are still different methods for investigation of this problem that all refer to it. In this paper, we provide a method based on fuzzy variables for making exploration and exploitation in multiagent environment. In this method, an effective agent (ε in ε-greedy method) is obtained which is updated using fuzzy variables in each step to manage tradeoff between exploration and exploitation.The proposed algorithm is investigated for determination an optimized path in the Grid World. In this method, agents effort to reach locations with a highest gain in a cooperative environment. Outcomes of the suggested fuzzy based algorithm compared with the results by conventional ε-greedy method. In addition, quality improvement of interaction between exploration and exploitation is discussed.

KEYWORDS:

Reinforcement learning, Multiagent environment, Balance between exploration and exploitation, Q-Learning.

1. 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 [1,2]. Internal interaction is a key point, increasing the ability to solve a wide variety of applications including robotic teams, distributed control systems, collaborative decision support systems, resource management, data mining, etc. This is because of the nature of the most real world applications, especially those aroused during recent years in the field of social problems. Although the agents in a multiagent system can be programmed with behaviors designed in advance, it is often necessary that they learn new behaviors online, such that the performance of the agent or of the whole multiagent system gradually improved in [3]. Reinforcement learning, as a learning method that does not need a model of its environment and can be used online, is well suited for multiagent systems. Simplicity and generality of the RL setting make it attractive even for multiagent learning. However, several new challenges arise for RL in multiagent systems. Foremost among them is the difficulty of defining a good learning goal for the multiple RL agents. Furthermore, each learning agent must keep track of the other learning agent that makes it nonstationary. The environment nonstationarity invalidates the convergence properties of most single-agent RL algorithms. In addition, the scalability of algorithms to realistic problem sizes, already problematic in single-agent RL, is an even greater cause for concern in multiagent reinforcement learning (MRL).

The MRL field is rapidly growing, and a wide variety of approaches to exploit its benefits and address its challenges have been proposed over the last few years. These approaches integrate developments in the field of machine learning (RL), game theory, and direct policy search techniques. Among various paradigms, integration of game theory and machine learning seems to be the most promising solution. Many MRL algorithms in mixed tasks are designed for normal (stateless) repeated games [4], or work in a stage wise fashion that arises in each state of Markov game. Learning in a stage game is an important issue in multiagent learning system. This is due to the theorem proved in [5], stating that the Nash solution in single stage normal game is a part of the solution of the whole Markov game. Convergence proof in Nash-Q [6] is based on this theorem which latter had been extended to Asymmetric-Q [7]. It had also being used in [8].

Among various proposed method in MRL, Nash-Q which is proved to be convergent to the unique equilibrium of the game, provides the most significant concept in MRL techniques [6-8].

Various algorithms were extended from single agent learning to multiagent learning such as Evolutionary learning [9], Coevolutionary learning [10], [11], and the combination of reinforcement learning (RL) and game theoretic solvers based on Nash equilibrium points, which seems to be a good soultion of solving MRL problems[12].

Contribution of game theory to MRL was first implied in [13] by MinMax-Q. It was proposed for two successive fully competitive agents. Due to simplicity, MRL were later developed for agents with simultaneous action selection modeled as normal form games. It was first considered in Nash-Q for general-sum Markov games [14]. Agents decide on their actions to reach the presumed unique equilibrium point of the current game. It was proved that the algorithm gradually converges to optimal policy. Unfortunately, their applicability is restricted due to some drawbacks [15], especially with large number of agents. Littman proposed another method, [16], which some of the presumed limitations in [14] were relaxed by adding some additional information about the roles of the agents in the system. In addition, other modifications are also proposed that can be reviewed in [12].

Aforementioned works were mainly involved with the model of learning and convergence proof. Practically, some important issues including exploration-exploitation tradeoff and computing Nash equilibria are disregarded.

To achieve better action quality, agents should generate actions such that they explore environment suitably, and yet exploit their experiences to avoid punishment. Because of these conflicting objectives, balancing exploration and exploitation is an important issue in RL [17,18,19]. Depending on the type of the problem and the aim of learning, different authors assessed the quality of balance in the terms of the number of successes [20,21,22], the learning time-period [21], and the number of failures [20]. However, they did not offer any comprehensive balance measure that includes the effective parameters in balance. In continuous fuzzy reinforcement learning (FRL), fuzzy inference system is used to obtain an approximate model for the value function in continuous state space and to generate continuous actions [23,24,19].

In this paper, we implement learning method with linear function approximates using fuzzy systems. This paper investigates multiagent learning problem with available fuzzy control parameters for making interactions between exploration and exploitation. ε-greedy selection is considered for making balance between exploitation and exploitation. In addition, fuzzy controlling parameters were responsible for exploitation value of ε and α learning value. Results suggest that this method is applicable if there is appropriate time for learning of agents. Although interaction between agents in this interaction logarithm is considered as cooperative.

we use ε-greedy method for action selection in each rule, where the ε gradually decreases as a function of episode number.

2. REINFORCEMENT LEARNING

Reinforcement learning can be expressed in different frameworks. A rough but informative categorization of the learning model. Learning, not only takes place based on the immediate rewards, but also delayed rewards have a great effect on the optimal policy. Traditionally, reinforcement learning has been used in single agent sequential decision making. The learning agent is in permanent interaction with its environment in such a way that the agent and the environment interact at discrete time steps.

In time step *t* the agent performs an action *t a* , receives a reward *t*+1 *r* , and observes the new state of the environment *t* +1 *s* .

Every learning algorithm is considered a reinforcement learning algorithm, if it is able to provide an appropriate mapping from states to actions in a way that the expected sum of the reward is maximized in the long run. In RL the goal of the agent is formalized in terms of the reward signal. Reinforcement Learning agents try to maximize the cumulative reward they receive from the environment. The model of the environment is represented with a Markov Decision Process.

3. MARKOV DECISION PROCESSES

A Markov decision process is a tuple, (S,A,T ,R), where S is

a set of states, A is a set of actions, T is a transition function S ×A × S → [0, 1], and R is a reward function S × A →R.

The transition function defines a probability distribution over next states as a function of the current state and the agent's action. The reward function defines the reward received when selecting an action from the given state.

Solving MDPs consists of finding a policy, π : S → A, which determines the agent's actions so as to maximize discounted future reward, with discount factor .MDPs are the focus of much of the reinforcement learning work . The crucial result that forms the basis for this work is the existence of a stationary and deterministic policy that is optimal. It is such a policy that is the target for RL algorithms.

In multiagent learning, several interactive agents are evolved to reach a target such that a kind of improvement can be addressed through repetitions. The evolution can be regarded as a number of situations which can be considered as a game among several agents.

Game theory initially was introduced for reasoning in economic theory which later has been widely used in social, political, and behavioral phenomena. Game theory provides the necessary tools to model a game.

4. FUZZY INTERACTION BETWEEN EXPLORATION AND EXPLOITATION

Complexity of interaction between exploration and exploitation in single agent environment is a known problem which different methods provided for its solution. We can classify available solutions in two general classes based on ε-greedy selection and Soft max policy methods. First method is based on making random movements with ε possibility in each function that decreases as problem solution improves. Second method is based on a special possible density function for each act of problem in any conditions. Most important property of first method is its simplicity and existence of no complexity in providing an index possible density function for each act of problem in any conditions. But second method is a better method for providing an index possible density function for each act of problem because this method is effected by any reward in any motion .So in any case this method is the best choice for providing possible density to each action . In this study fuzzy interaction based on ε-greedy selection is used for facilitate the operation. For adjustment of exploration and exploitation with ε-greedy selection method the interaction between exploration and exploitation is just affected by ε parameter. In this method the ε parameter shows exploration possibility in any movement of agent and it becomes updated in each time step. Updating is based on the position of 3 fuzzy parameters of $\hat{Q}$, ΔV and E explained in the following lines.

According to equation

$$\hat{Q}_t = \max_a(Q(s_t,a)) - \min_a(Q(s_t,a)) \qquad (1)$$

where that $\hat{Q}$ is difference weighted between maximum and minimum move value in the current state.This parameter is calculated based on current state data and can be used as a measure of obtained experience in this state. In the beginning of learning process this difference is not significant but learning of this parameter increases as time passes.

Accordingly small measures of $\hat{Q}$ usually represent that the agent is in a new state with little data that requires state exploration. ΔV Shows difference value of current and previous state that is as following.

$$\Delta V_t = V(s_t) - V(s_{t-1}) \qquad (2)$$

where

$$V(s_t) = \max_a(Q(s_t,a)) \qquad (3)$$

According to above equation it is clear that those positive values of ΔV show the agent go to a state with longer reward; so, if exploration causes to state with positive ΔV ,then we will reinforce that policy and accordingly the agent should perform more explorations in these cases. E Shows exploration rate in recent steps based on following equation.

$$E_t = \nu E_{t-1} - (1-\nu)\varepsilon_{t-1} \qquad (4)$$

In above equation, $0 < \nu < 1$ is a coefficient weight which shows the value of giving validity to recent steps. By the help of this parameter we can have a measure for exploration rate in recent steps. Applicable algorithm in this paper controls ε parameter through 3 fuzzy parameters, $\hat{Q}$, ΔV and E . ε controlling and adjusting process is based on following conditions:

- If $\hat{Q}$ and ΔV is negative, then ε will be high.
- If $\hat{Q}$ is low, ΔV is positive and E is low, then ε will be high.
- If $\hat{Q}$ is low, ΔV is positive and E is high, then ε will be low.
- If $\hat{Q}$ is high and ΔV is negative, then ε will be high.
- If $\hat{Q}$ is high and ΔV is positive, then ε will be low.

In five cases that mentioned above, consider different state of controlling parameters. For example, first state is a state in which the agent has little information about environment and therefore requires more exploration.

5. LEARNING RATE ADJUSTMENT

One of the properties of controlling parameters defined in previous part, that includes increasing ability of controlling different learning variables. Learning rate is also among instances that through its adjustment we can have a correct perception of environment. In this paper learning rate parameter is adjusted by a control parameter ($\hat{Q}$). A proper learning strategy should make an uniform exploration in environment and avoid various exploitations associated with punishment. Dependence of this parameter to the number of repeated state experiences, is a proper strategy for learning improvement by learning rate. Thus in the case that learning experience has been repeatedly reduced. This strategy is presented by control parameters as dependent parameter α to $\hat{Q}$.

6. SIMULATION

This part examines the fuzzy model presented for making balance between exploitation and exploitation in sample issue. Our case study was 10 ×10 Grid World with two agents (Figure 1),and our problem in this simulation is finding an optimal path in grid world.

Two agents a / b (Illustrated in figure 1 as two geometrical figures) move in environment and learn their optimized movement. Agent a (Figure△) starts its movement from the location (1, 1) and agent b (Figure○) starts its movement from the location (10, 10). Point locations in this game are static. Locations (2, 9) and (8,3) are point locations illustrated in the figure by hatched area . The two agents a / b start moving to up, down, right and left directions and look for point locations of the issue .There are two locations with positive points in the grid and going to locations other than these two locations will have -1 point for the agent. Agents share the obtained information that achieve from environment with each other. Therefore, this problem is theoretically a sample of cooperative game. In this simulation going in to location(x, y) = (2, 9) will have 15 points and going in to location (x, y) = (8, 3) will have 10 points.

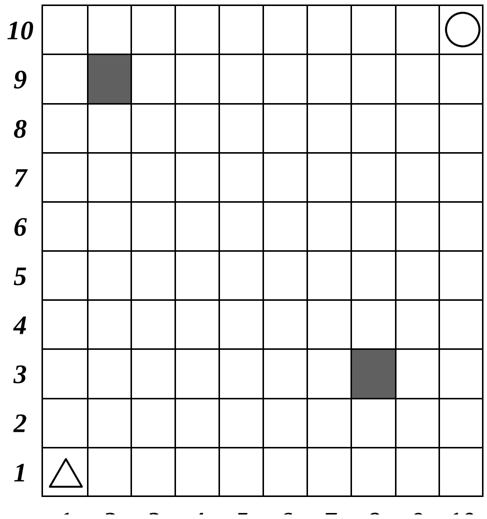

Figure 1: Diagram related to the investigated Grid World

After entering of each agent in to point location the episode will be finished and agents will come back to start location. Agents act cooperatively and Q table for each agent updates based on its information and information of other agents. Each agent that does not consider exploitation behavior in assessment of other agent's behavior .

Considering two locations with different points can be a proper problem for investigation of the exploration environmental quality by agents. In this problem, there is the possibility of finding a path leading to 10-point location by an agent in its exploitation. In these situations, this response may be considered as an optimized response by the agent or may discover with exploitation a path leading to a location with 15 points. It is an idea as a proper measure for quality assessment of suggested algorithm in this dissertation. In other words, optimized algorithm of exploitation is a kind of algorithm that makes a good balance between exploration and exploitation and do that in a way that an agent doesn't miss desired response because of exploration and also doesn't miss the better response because of more exploitation. According an optimal path solution, which is solved using dynamic programming, function values will be available for different modes.

Therefore states of value function for the new condition (V^{π}) calculates based on figure 2.

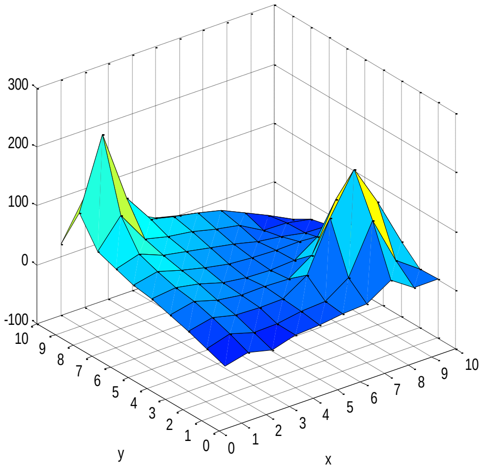

Figure 2: V^{π} for problem of the studied Grid World

According to obtained V^{π} , it is an optimized method for this problem. As you can see in this picture according to obtained optimized strategies for two agents.It is necessary to say that the aim of solving problem with dynamic method is to become aware of the reason for solving the optimized path problem with Nash-Q method (Based on a method with a base algorithm of temporal difference learning) and to know agents should finally converge to which response (path).

7. ε -greedy algorithm

Figure 3 shows position of agent *'a'* reward in ε-greedy algorithm during learning the investigated problem. As you can see in the picture agent a after about 700 episodes converges in to its final response.

Since both agents are aware of obtained information from each other, so it can be expected that player b also converges in to its final response in a similar way. Figure 4 shows position of agent b reward. This agent also has converged to its final response at about 700 episodes.

In solution of investigated problem by ε-greedy algorithm, the final result obtained by ***'a'***, ***'b'*** agents has been shown in figure 5. Proper result of this simulation in obtained optimized path is observable. Agents have chosen paths that provide the most benefits according to their environment. However, this path is not the best possible path.

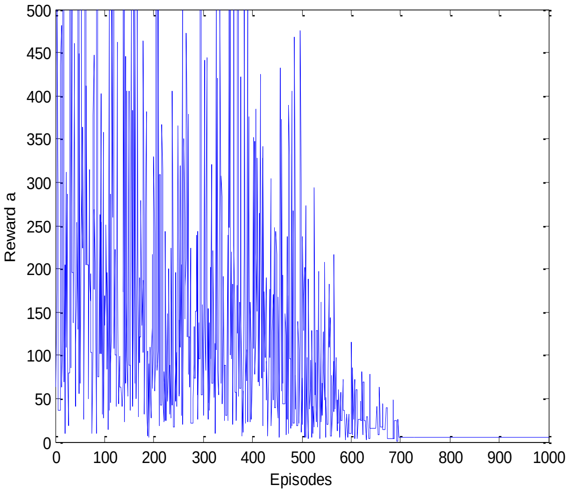

Figure 3:Agent '*a*' reward in ε -greedy algorithm

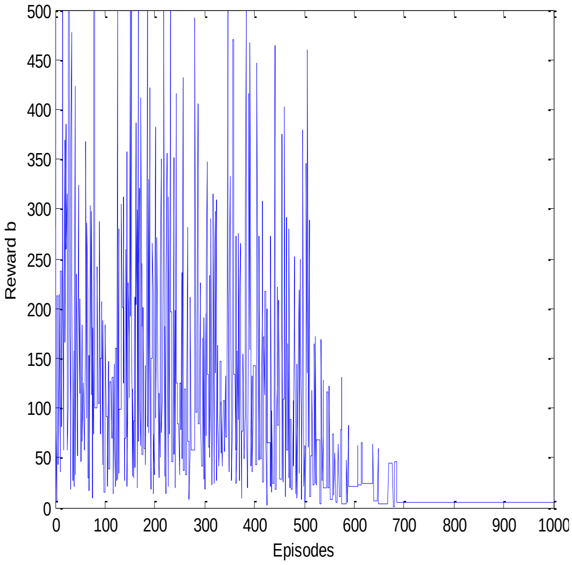

Figure 4: Agent *'b'* reward in ε -greedy algorithm.

It means that if agents have proper exploration in the environment then they attract responses with more rewards. Since agents share obtained information, therefore they both converge to the same response.

According to the results of ε-greedy algorithm, the final convergence of agents to the responses is dependent on first search. By implementation the proposed algorithm and reviewing the convergence frequency to point locations, it can be seen that agents are converged to the points, which are converged more in early episodes.

(that are more exploration than exploitation) have reached that location in more repeated times .Table 1 is comparison between repeated convergences to responses and their dependences to performed explorations in initial episodes. It happens while agents converge in to location (x, y) =(8, 3) and finally they reach to the same location in optimized path .The same situation is observable for states in which agents finally reach to the location (x, y)= (2, 9) .These results are observable in table 2.

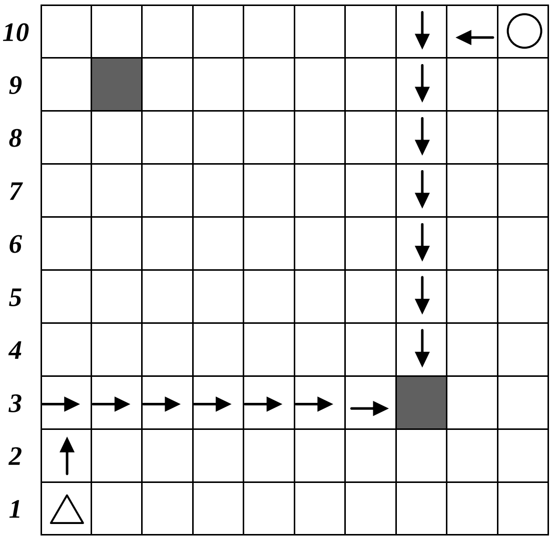

Figure 5: Obtained optimized path for agents in ε -greedy algorithm

Table 1: Percent of agents' convergence to point locations in different episodes while agents are converged to (x, y) = (8, 3).

	Agent a	Agent b
Convergence to location (x, y) = (8, 3) in early episodes <200 (%)	27	23
Convergence to location (x, y) = (2, 9) in early episodes <200 (%)	19	8
Convergence to location (x, y) = (8, 3) in the 1000 episodes (%)	41	29
Convergence to location (x, y) = (2, 9) in the 1000 episodes (%)	16	12

Table 2: Percent of agents' convergence to point locations in different episodes while agents are converged to (x, y) = (2, 9).

	Agent a	Agent b
Convergence to location (x, y) = (8, 3) in early episodes <200 (%)	8	10
Convergence to location (x, y) = (2, 9) in early episodes <200 (%)	24	17
Convergence to location (x, y) = (8, 3) in the 1000 episodes (%)	22	19
Convergence to location (x, y) = (2, 9) in the 1000 episodes (%)	47	52

8. SUGGESTED FUZZY ALGORITHM

We investigate obtained results of problem from solving gridworld and with using Q-Learning algorithm. This stage investigated the results of the problem solution by suggested fuzzy algorithm. Comparison of obtained results from Q-Learning and suggested Fuzzy algorithm can lead to determination of learning state improvement with new method.

Control fuzzy parameters with triangle forms adjusted as following:

$\hat{Q}$: Low (zero), High (0.02)

ΔV :Positive (0.03) , Negative (-0.03)

E : Low (-5), High (-2)

As shown in figure 6, by using suggested fuzzy algorithm, agent ***'a'*** has more opportunities for exploration as it explores in some episodes in order to achieve a better response even after convergence to final response. According to figures 3 and 6 we can see approximate increasing of the amount of agent's exploration .Here the question is that whether increasing of exploration will lead to a better response. In order to answer this question figure 7 illustrates final optimized path for a complete plan performance in suggested fuzzy algorithm. Based on this figure the final preferred path is a path with locations which have more points. (X, y)= (2, 9).

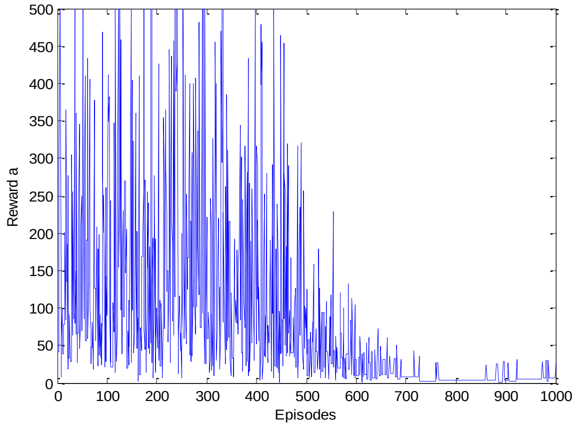

Figure 6: Agent *'a'* reward in suggested fuzzy algorithm

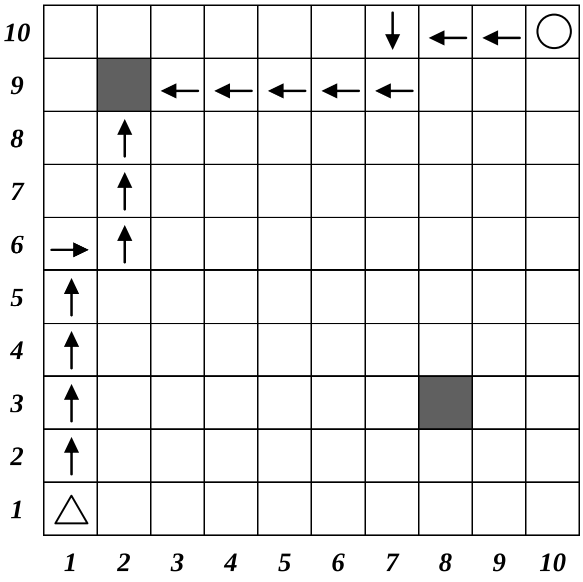

Figure 7: Obtained optimized path for agents in Fuzzy algorithm

9. SUMMARY AND SUGGESTION

This paper investigates multiagent learning problem with available fuzzy control parameters for making interactions between exploration and exploitation. ε-greedy selection is considered for making balance between exploration and exploitation. In addition, fuzzy controlling parameters were responsible for exploration value of ε and α learning value. Results suggest that this method is applicable if there is appropriate time for learning of agents. Although interaction between agents in this interaction algorithm is considered as cooperative, this method can be extended to non-cooperative environments and games with zero summation. We can consider following points in summary of obtained results of Grid World. According to the results, suggested fuzzy algorithm has more opportunities for exploration of states in agents than initial methods of exploration in multiagent systems.

On one hand exploration opportunity available for agents leads to convergence to optimized response and on the other hand, during convergence the agents have no significant effect. In other words, they have not delayed convergence in considerable amount of time.

10. REFERENCES

[1]. Weiss, Gerhard. Multiagent Systems: A Modern Approach toDistributed Modern Approach to Artificial Intelligence. London :MIT Press, 1999.

[2]. Russell, Stuart J. and Norvig, Peter., Artificial Intelligence: A modern approach (2nd Edition). Englewood Cliffs, New Jersey : Prentice Hall, 2003.

[3]. Stone, P. and Veloso, M., “Multiagent systems: A survey from the machine learning perspective.” Auton. Robots, vol. 8, no. 3, pp. 345–383, 2000.

[4]. Busoniu, Lucian, Babuska, Robert and Schutter, Bart De., “A Comprehensive Survey of Multiagent Reinforcement Learning.”, IEEE Transaction on Systems, Man, and Cybernetics,Part C:Applications and Reviews, Vol. 38, No. 2,, pp. 156-172, 2008.

[5]. Filar, Jerzy and Vrieze, Koos., Competitive Markov Decision Process.s.l. : Springer-Verlag, 1997.

[6]. Hu, J. and Wellman, P., “Multiagent reinforcement learning:Theoretical framework and an algorithm.” In Proceedings of the Fifteenth International Conference on Machine Learning. pp. 242– 250, 1998.

[7]. Kononen, Ville., “Asymmetric multiagent reinforcement learning.” Web Intelligence and Agent Systems: An international journal, pp. 105–121, 2004.

[8]. Wang, X. and Sandholm, T., “Reinforcement learning to play an optimal Nash equilibrium in team Markov games.” Vancouver, Canada : Adv. Neural Inf. Process. Syst. (NIPS-02). pp. 1571–1578, 2002.

[9]. l. Panait and S. Luke, “Cooperative multiagent learning: The state of the art,” Autonomous Agents Multiagent Systems, vol.11, no. 3, p-387-434,2005.

[10].M, A, Potter and K. A. D. Jong, “ A cooperative coevolutionary approach to function optimization,” ,Jerusalem, Israel,1994.

[11].S. G. Ficici and J. B. Pollack, “A game-theoretic approach to the simple coevolutionary algorithm,” , Paris,France,2000.

[12]. Lucian Busoniu, Robert Babuska, and Bart De Schutter, "A Comprehensive Survey of Multiagent Reinforcement Learning," IEEE Transaction on Systems, Man, and Cybernetics,Part C: Applications and Reviews, Vol. 38, No. 2, pp. 156-172, 2008.

[13] Michael Lederman Littman, "Algorithms for Sequential Decision Making," Providence, Rhode Island, 1996.

[14].J. Hu and P. Wellman, "Multiagent reinforcement learning: Theoretical framework and an algorithm," in In Proceedings of the Fifteenth International Conference on Machine Learning, 1998, p. 242–250.

[15].Ali Akramizadeh, Ahmad Afshar, and Mohammad –B. Menhaj, "Different Forms of the Games in Multiagent Reinforcement learning: Alternating vs. simultanous movements," in 17th Mediterranean Conference on Control and Automation, Thessaloniki, Greece, 2009.

[16].M. L. Littman, "Friend-or-foe Q-learning in general-sum games," , 2001.

[17]. M. Guo, Y. Liu, J. Malec, A new Q-learning algorithm based on the metropolis criterion, IEEE Trans. Systems Man Cybernet. B 34 (5) (2004) 2140–2143.

[18].R.S. Sutton, A.G. Barto, Reinforcement Learning: An Introduction, MIT Press, Cambridge, MA, 1998.

[19].C. Zhou, Q. Meng, Dynamic balance of a biped robot using fuzzy reinforcement learning agents, Fuzzy Sets and Systems 134 (1) (2003) 169–187.

[20].F. Saadatjou, V. Derhami, V. Majd, Balance of exploration and Exploitation in deterministic and stochastic environment in reinforcement learning, in: 11th Annu. Computer Society of Iran Computer Conf., Tehran, Iran, 2006, pp. 492–498.

[21]. G. Yan, T. Hickey, Reinforcement learning algorithms for robotic navigation in dynamic environment, in: IEEE Internat. Conf. on Neural Network, 2002, pp. 1444–1449.

[22].G. Yen, F. Yang, T. Hickey, M. Goldstein, Coordination of exploration and exploitation in a dynamic environment, in: IEEE Internat. Conf. on Neural Networks, 2001, pp. 1014–1018.

[23].H.R. Berenji, D. Vengerov, A convergent actor-critic-based FRL algorithm with application to power management of wireless transmitters, IEEE Trans. Fuzzy Systems 11 (4) (2003) 478–485.

[24].C.-K. Lin, A reinforcement learning adaptive fuzzy controller for robot, Fuzzy Sets and Systems 137 (3) (2003) 339–352.