AN UNSUPERVISED LEARNING METHOD FOR AN ATTACKER AGENT IN ROBOT SOCCER COMPETITIONS BASED ON THE KOHONEN NEURAL NETWORK

M. Mirmomeni* and M. J. Yazdanpanah

Control and Intelligent Processing Center of Excellence, College of Engineering
School of Electrical and Computer Engineering, Tehran University
P.O. Box 14395-515, Tehran, Iran
mirmomeni@ut.ac.ir - yazdanpanah@ut.ac.ir

*Corresponding Author

(Received: December 30, 2006 – Accepted in Revised Form: May 9, 2008)

Abstract RoboCup competition as a great test-bed, has turned to a worldwide popular domains in recent years. The main object of such competitions is to deal with complex behavior of systems which consist of multiple autonomous agents. The rich experience of human soccer player can be used as a valuable reference for a robot soccer player. However, because of the differences between real and simulated soccer world, it needs to fit such precious transcendental knowledge to use in the simulated soccer game. On the other hand, Reinforcement Learning (RL) as a common method in this domain because of its trial-and-error nature does not have great performance in using transcendental knowledge. Thus, this method is limited to complex multi-agent learning problems. Among various frameworks of intelligences, in general, Artificial Neural Networks (ANN) and specially Kohonen neural networks with its feed-forward architecture and its ability in discovering any relationships of interest that may exist in the input data may be considered as a powerful tool in clustering. This paper puts forward an unsupervised learning method based on Kohonen network to create a powerful Tactics layer in decision-making section for an attacker agent. The approach presented in this paper is based on the combination of expert's knowledge and data obtained from the simulated world. This system is applied to the attacker agents of ULA 2006 soccer team. Simulation results revealed that the chosen approach is superior with respect to the other intelligent techniques.

Keywords Robocup, Multi-Agent System, Kohonen Network, Real-Time System, Feed Forward, Unsupervised Learning, Decision-Making

چکیده در سال های اخیر، رقابت های روبوکاپ به عنوان بستر آزمون ایده های نو فراگیر شده است. هدف از برگزاری چنین مسابقاتی تحلیل رفتارهای پیچیده ای است که در سیستم های مشتمل بر چندین عامل مستقل به چشم می‌خورد. به طور حتم، یکی از مهم ترین منابع برای طراحی یک ربات فوتبالیست، یک بازیکن فوتبال واقعی است. هرچند باید اذعان داشت که به علت تفاوت های ذاتی مابین بازی فوتبال و محیط شبیه سازی می‌بایست دانش حاصل از عملکرد تیم های فوتبال واقعی را با شرایط حاکم بر محیط شبیه سازی مطابقت داد. از طرف دیگر، یادگیری تشویقی – تنبیهی با طبیعتی آزمون و خطایی، کیفیت قابل قبولی در استفاده از این دانش اکتسابی ندارد. به عبارت دیگر، این روش یادگیری محدود به مسائل چند عامله ای است که دانش مزبور با استفاده از آزمون و خطا استخراج می‌شود. در میان رویکردهای هوشمند، عموماً شبکه های عصبی مصنوعی، به خصوص شبکه های کوهونن با ساختاری پیش خور و داشتن توانایی بالا در کشف روابط حاکم بر داده ها در مسئله خوشه یابی، از توجه بالاتری برخوردار بوده اند. این مقاله در نظر دارد با استفاده از شبکه های کوهونن با روش یادگیری بدون نقاد، لایه تاکتیکی قدرتمندی در بخش تصمیم گیری یک مهاجم فوتبال در محیط شبیه سازی مسابقات روبوکاپ ارائه دهد. روشی که در این مقاله ارائه می‌شود مبتنی بر دانش حاصل از یک بازیکن فوتبال واقعی توأمان با داده های گردآوری شده از محیط شبیه سازی است. سیستم پیشنهاد شده به مهاجمین تیم ULA 2004 اعمال شده است. نتایج شبیه سازی گواه بر عملکرد بالای روش ارائه شده در مقایسه با روش های هوشمند دیگر است.

1. INTRODUCTION

In recent years, the research on multi-agent systems and distributed Artificial Intelligence (AI) has become a main trend in artificial intelligence domain. Human society as a precious reference can be used by a multi-agent system to focus on collective intelligent behaviour [1]. Such systems have been applied in software agents, intelligent manufacturing systems, electronic commerce [2], [3], predicting stock prices [4], control systems, medicine and etc. [5-7]. In addition, the rapidly expanding complexity of modern systems and our ever-increasing desire for improved performance, durability, economy and intelligence of such systems, inevitably turns our attention to the paradigm of artificial intelligence and soft computing. This is particularly true when we consider distributed and multi-agent systems in noisy and real-time environment, and existence of cooperative and adversarial agents in many benchmark problems such as Robot Soccer Game or in practical applications such as cooperative satellite constellations for more accurate sensing and more area coverage.

RoboCup Simulation Game is a standard simulation game defined by RoboCup League. It is marked by its dynamic environment, the co-existence of cooperation and competition among several agents, limited communication bandwidth, and noisy environment, which makes it a good test-bed for multi-agent system research and application. In other words, RoboCup is an international joint project to promote AI, robotics and related fields. It is an attempt to foster AI and intelligent robotics research by providing standard problems where a wide range of technologies can be integrated and examined.

On the other hand, artificial neural networks especially MLP networks as feed-forward networks because of their ability in modelling nonlinear phenomenon [8-10], and Reinforcement Learning with its online learning attitude [11,12] frequently have been used by teams competing in RoboCup competitions to define complex relations among soccer agents. By such definitions, it could be possible for an agent to make proper decisions in different situation in a soccer game. However, it has to be said that both methods are good in some area, but they all have shortcomings. For instance, for most applications, though simple as the environment may be, it seems to be impossible to get a proper definition for entire system behaviour, because of the dynamic characteristics of such environments. In addition, learning ability and adaptation are the most important characteristics for agents in such environments with their intrinsic uncertainties. There are many classifications of learning in AI literature which are used in this domain. Among these methods, reinforcement learning gets more attention because of its on-line capability and its ability to learn optimal policy in an entirely unknown environment [1]. However, Reinforcement learning like other methods, has its weak points. First of all, trial and error methods do not seem to be a proper way for expressing rules and utilizing transcendental knowledge. Secondly, RL algorithm for multi-agent system is developed from typical RL algorithm, which is for a single agent, and is not competent in dealing with complex multi-agent system learning problem [2].

It has to be said that decision making in robot soccer games is one of the most important problems that needs many works done to get to a proper performance in any situation, in a simulated game. Many researchers have devoted their efforts to reach this zenith. Among these studies, there are many methods to set a proper decision making system. However, in these methods some improvements can be seen, but there exist some obstacles which needs to be solved. Shi, Jinyi, and Zhen in [13] tried to make a learning method based on a fuzzy reinforcement learning approach, but the main problems in RL methods in this research make its performance poor in some especial situation. Chohra and Scholl designed a decision-making system by a MLP network for robots, so it could be possible for robots to make proper decisions in different situations in the middle-sized RoboSoccer Competition [14]. Unfortunately, this method is not compatible for the real-time environment of the RoboCup Simulation Competitions. Meyer, Adolph, and Stephan in [15] created a system to control agents' movements according to the team strategy such as offside trap based on the potential field approach. This idea is improved in [16]. Although, this method is powerful to control the agents' movements and make decision for agents in the hyper-level, yet it is not a comprehensive method in this real-time domain. Because it needs lots of information even for a simple action such as shooting or dribbling, and obviously this is an obstacle for a real-time environment such as RoboCup simulation game. Yang, Li, and Yue in [27] tried to design a decision-making system based on Decision-Tree approach. It has to be said that because of the complexities of the environment and its dynamic attitudes, it is difficult to make a state-space that could properly define the total environment. In Robot Soccer Game, the rich experience of human soccer players is of great significance to the designation of robot players. In order to make full use of this transcendental knowledge, a method should be considered that could mix rich experience of human soccer player in the real soccer game with simulated game characteristics. In some researches it is tried to use fuzzy logic and similar approaches to use the expert knowledge in designing a skilful agent [18,19]. In addition, according to this fact that the positive and important roles of emotions have been emphasized not only in psychology, but also in AI and robotics in recent years [20-23] Brain Emotional Learning (BEL) is used in some researches to solve the decision making problem of an intelligent soccer player in this simulated domain [24-26].

According to the great performance of the Recurrent Neural Networks [27-29] in clustering which use unsupervised learning methods, these methods introduce themselves as a useful instrument in control and predicting. These networks have been used to control real-time systems and they have shown great performance [30]. In addition, these networks have been used to model dynamical systems and they show great potential in this field [31]. Therefore, these networks seem to be perfect for a decision-making system in a real-time nonlinear environment such as Simulated Soccer Game. In this paper, Kohonen network is proposed as a feed-forward neural network with its unsupervised learning method to mix transcendental knowledge of a real soccer player with some attitudes that some simulated agents must have in this simulated world, to design a decision-making system for attacker agent in this simulated domain. It has to be said that this method could be generalized to the other agents by little changing in the input functions of this system. The proper behaviour of attacker agents in different situations shows the great potential of this method in decision-making.

The remaining sections of this paper are structured as follows: Section 2 provides a brief introduction of RoboCup environments and the agents' architecture of a simulated team in this environment. Section 3 is devoted to give the mathematical model of the attacker agent's performance in the field. In section 4, a portrait of data-mining efforts from real and simulated soccer is given. After that, by using this proper data, a Kohonen network is tuned to give a skilful attacker agent. Simulation results in RoboCup environment are given in section 5. The last section contains concluding remarks.

2. BASIC CONCEPTS OF ROBOCUP ENVIRONMENTS

2.1. Robocup Simulation Environment The simulated league has the most participants among RoboCup's leagues [32]. During a simulated game, a number of rules, which are similar to those of the human soccer games, are enforced by an automated referee within the server, or by a human referee. The simulation game is running on standard computer simulation environment in Client/Server manner. RoboCup Association provides standard Soccer Server System and each participant team design their own client programs. Soccer Server is a system that allows participants to create their competing programs in different kinds of program languages [33]. It provides a virtual soccer field and simulates the movements of all objects in this virtual field. The game is going in discrete time steps. To make the game more real, Soccer Server adds noise into the environment to make the senses and actions of each client noisy like its equivalence of a real soccer player in a real soccer game. In addition, each client can only control one player and no communication channel is permitted except the one that Soccer Server provides. It has to be said that this channel is a narrow-bandwidth multi-access one. The environment is like something shown in Figure 1.

Following is the generalization of RoboCup Simulation Environment:

- Dynamic real-time system.
- Noisy environment.
- Cooperation and coordination.
- Limited communication bandwidth.

The Soccer Server creates a virtual world, which has many unfavorable characteristics for team designers. This world is real-time, noisy, collaborative, and adversarial environment. Therefore, agents have to be adapted in this complex environment.

2.2. Architecture of a Soccer Player in RoboCup In this section, agents' architecture of a soccer team in this simulated world will be given. This architecture seems to be suitable to create a team of agents in RoboCup complex environment.

Figure 2 shows the agents' architecture made by three major parts: analyzer, world-modeling and the decision-making parts.

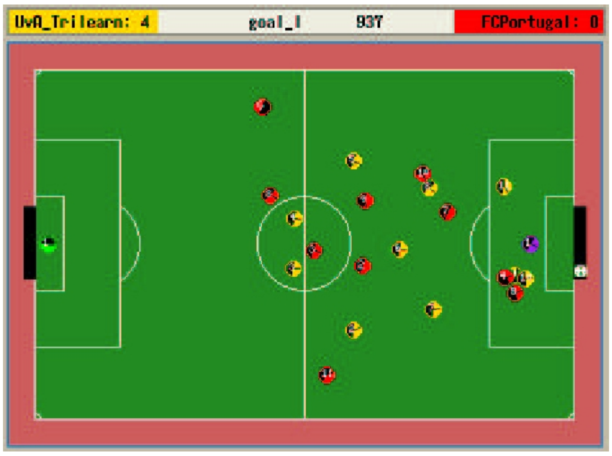


Figure 1. The robocup simulation game environment.

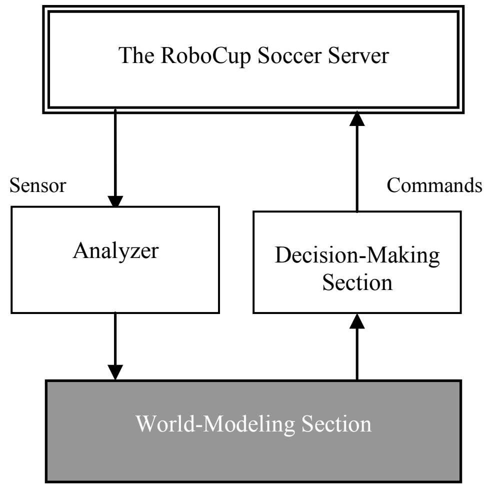


Figure 2. Agents' architecture.

2.2.1. Analyzer Analyzer receives the noisy and unprocessed information from Soccer Server by agent's sensors. This part tries to filter the noise from noisy information, so it could be possible for world-modeling part to create a global coordinated image of the field's state. Therefore, its output has to be compatible with two other parts. Because of the asynchronous sensing and acting [34], and the discrete form of the model, Analyzer has to be an independent thread. When visual, aural, or physical information arrives, Analyzer is triggered and after it gets the information, analyzes the sensory to perform a better data process.

2.2.2. World-modeling It is obvious that in a real soccer game, player's visual and aural qualities are limited. Therefore, this simulated world which is based on the rules of real world has the same limitations for an agent in a simulated game. This means that visual information from a field is gathered in a limited angle for an agent. In addition, its precision decreases with distances. For example, player A is in a situation that can see his teammate B. if he sends the Server a turn command that is executed; he will not see the teammate B anymore. Also, if player B runs away from his teammate A, it could be difficult for player A to recognize his teammate.

Because of the single channel, low-bandwidth, and unreliable communication paradigm, communication among teammates is just used as an auxiliary method, and the architecture could not depend on the communication mechanism. Therefore, it seems that this agent's architecture needs a part to estimate the present state without any communication. Such part is embedded in the world-modeling part to solve this problem. In addition, decision-making part needs some future information occasionally. Therefore, a predictor part is needed to be embedded in the agents' architecture to predict the future states. When an agent estimates or predicts some information, the confidence of the information is set. For example, agent A has to check the degree of confidence of existence of teammate B in its former position. In fact, the confidence will be decreased as time goes on (e.g., teammate B sends a dash command to the Server and the Server executes it).

Figure 3 shows the World-modeling part architecture. It can be seen that this system reflects both current information and the future information, and includes two mechanisms: estimation, updating, and prediction.

2.2.3. Decision-making Decision-making part is the kernel of the agents' architecture. The estimator and predictor adapt two characteristics of multi-agent system: real-time and noisy characteristics. The decision-making part adapts the other two: collaborative and adversarial characteristics. The ULA2006 soccer team is supposed to have three teamwork structures: roles, formations, and specific situations. The decision-making part depends on both the world-modeling part and these structures.

Roles: each agent has an independent task (e.g. defense, attack) that can be switched under different situations.

Formations: collections of roles build into team formations.

Specific Situations: arrangement for a specific situation (e.g. kickoff, corner ball).

Figure 4 shows the decision-making part. From receiving world-modeling part to send command, decision-making part is classified to three layers: strategy layer, tactics layer, and execution layer.

2.2.3.1. Strategy Strategy is in the top of the decision-making layer, which decides the roles, formations, or specific situations.

2.2.3.2. Tactics Tactics layer decides the player's individual skills. Individual skills include going to a profitable position, tackling, interception, dribbling, kicking, passing, etc. This paper focuses on this layer and especially designs a decision-making system for the attacker agent in attack mode who has to decide between kick the ball, pass the ball to the teammate in a better place to kick or dribble to a better place to kick.

2.2.3.3. Execution The execution layer is the lowest layer in decision-making part, which executes player skills. In every cycle, it decides the basic command that changes the state of the game and finishes the long-term goal.

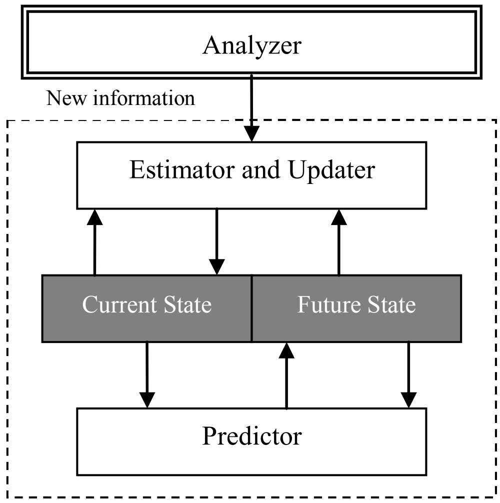


Figure 3. The world-modeling section.

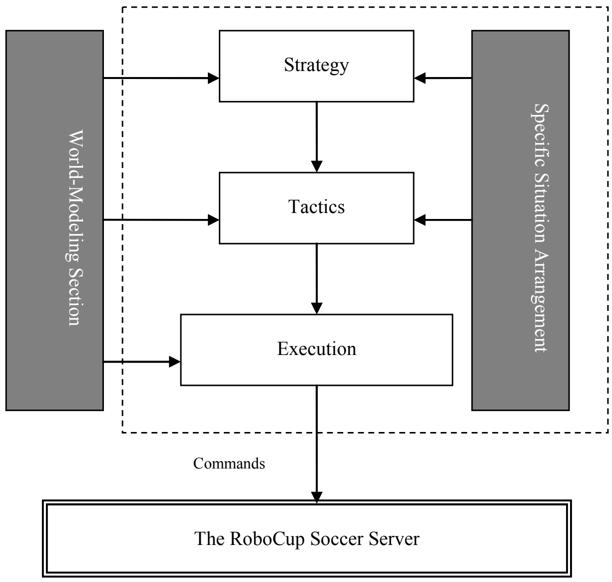


Figure 4. The three layered decision-making architecture.

3. MATHEMATICAL MODEL FOR A SKILLFUL ATTACKER AGENT IN SOCCER

It is obvious that the most important duty for an attacker in a soccer game is scoring for his or her team. This section is devoted to give a mathematical model for the behavior of a skillful attacker during a soccer game close to the rival's penalty region.

It seems that, an attacker close to the rival's penalty region has to choose one of these tasks:

- Find the best point to kick the ball, if it is possible.
- Move to a better place to kick the ball and score, if there is such a place.
- Pass the ball to a teammate with some advantages in positioning in comparison.
- In a difficult situation, pass the ball to a teammate, which is not marked by opponents.

To create a decision-making part based on the mathematical model of an attacker who has to decide one of these tasks near the rival's goal, first of all, it is better to elicit and model the real attacker's logic in a real soccer game. To do so, suppose that there is an attacker agent who has the ball near the rival's goal. If his/her point of view is good, which means that no opponents especially goalkeeper can reach the ball if he kicks it, it is obvious that he/she should kick the ball as soon as possible. In another situation, this skillful attacker may be near the rival's penalty region who has no one upon him (except the rival's goalkeeper, probably). In this situation, it is obvious that the attacker agent should search the field by himself for a better place to go and after that, kick the ball to the goal and make a score. If there is such a place, he has to go there as soon as possible.

On the other hand, for the third option, it is obvious that there must be a teammate positioned in a better place than the ball owner is. Therefore, as a ball owner it is meaningful to pass the ball to that teammate to kick. In the last situation, it is urgent to find the teammate to save the ball.

As it said before, each agent draws a portrait of the field with the sensory information of the world-modeling part. After that, according to this portrait which is obtained by the mathematical model of the present state of the environment, agent needs to make a proper decision. To do so, a mathematical model based on some useful functions is developed to evaluate the present state and make a good choice. Actually, these functions indicate some factors, which are important for the attacker to choose the best decision in decision-making part. These important factors are:

3.1. Distance to the Centre of the Goal this function is designed in such a way that the shorter distance to the center of the rival's goal gives the higher value for this function. The rule for this function is:

$$\text{dist.} = 1 - \sqrt{(x_P - x_{\text{center of goal}})^2 + (y_P - y_{\text{center of goal}})^2} \Big/ \text{dist}_{\max} \tag{1}$$

which (x_p, y_p) is the coordination of the player and $(x_{\text{center of goal}}, y_{\text{center of goal}})$ is the coordination of the center of the goal. $\text{dist}_{\max}$ is a normalization factor.

3.2. View Angle this function gets the angle that the ball owner sees the rival's goal. Rule that is used for this function is:

$$\text{V\_Angle.} = \begin{cases} \alpha & \text{dist.} < (\text{goal's length}/2) \\ \pi + \alpha & \text{dist} > (\text{goal's length}/2) \end{cases} \tag{2}$$

$$\alpha = \arctan \left| \frac{m_2 - m_1}{1 + m_1 m_2} \right| \tag{3}$$

Where m_1, m_2 are slopes of lines that pass from the ball owner and the goal posts.

3.3. Kick Safety (Confidence): this function actually is a criterion to determine the situation of the ball owner. The higher value for this function means the better and safer situation of the ball owner. The rule for this function is:

$$\text{Confidence} = 1 - \frac{\text{Opponent}}{11} \tag{4}$$

where the opponent factor is the factor that models the number of the opponents who can intercept the ball after ball owner kicked it.

3.4. Open Space this function gives the free space in front of the ball owner. The rule for this function is:

$$open\_space = \frac{\pi * r^2}{S_{max}} \quad (5)$$

where *r* is the distance between ball owner and the closest rival and the S_{max} is a number to normalize this value. It is obvious that, the ball owner kicks whenever it has a short distance to the rival's goal, a high view angle and a high confidence.

The attacker agent moves to a better place whenever he/she has many places to go. Therefore, it must be many spaces upon him/her. The attacker agent will pass the ball to another teammate if there is such a teammate who can receive the ball and kick it. At the end, the attacker agent passes the ball to a player, which is not marked by the opponents if there is no way to act.

It has to emphasize that the decision-making process should be done in a limited time because of the real-time characteristics of the environment. Therefore, it is better to work with primary-analyzed data than pure data gained from the modeling section.

4. DESIGNING A SKILLFUL ATTACKER BASED ON KOHONEN NEURAL NETWORK

4.1. Data Mining Artificial neural networks are data driven approaches. With no exception, the Kohonen network which is used for tactic layer in decision-making part needs to be trained by some proper data for each task that will be chosen by attacker agent. In this subsection, a short description of data mining task, which provides some proper data for Kohonen network, is given. As it said before, the rich experience of human soccer player is of great service to the robot players. In addition, RoboCup's ultimate long-term goal is stated in such a way that it could be possible to make a team of fully autonomous humanoid robot soccer players, who can win the soccer game, complying with the official rules of FIFA, against the winner of the most recent world cup [35]. Therefore, it is obvious that the acme for a robot attacker is the best real attacker in soccer world. One way to get to this point is to make a robot who can act like these well-known real human attackers. Therefore, these robot attackers must be taught in such a way that it is possible for them to do things like these famous attackers and rationally, the best data, which can be used for teaching these attacker agents, is the actions of these famous attackers in different situations in some significant soccer competition such as World Cup Competitions. To do so, a software that work as an analyzer in soccer world, can be used to analyze the actions of these human attackers in some real soccer game and get some good data from these situations.

Figure 5 shows a situation, which depicts a state that "Move to the best point to kick" decision, is the best action to do.

It can be seen that this situation is a one to one situation, and obviously, the best decision for such a situation is to get closer to the rival's goal, as close as possible; then kick the ball to make a score.

Figure 6 shows a situation, which specifies that "Pass to a better teammate to make a point" is the best decision to choose.

Because of the differences between real soccer and the simulated soccer game, a rational action in a real game may seem a ridiculous one in a simulated game. Therefore, the differences between real soccer and the simulated soccer game make it a must to fit the transcendental knowledge into the new environment. Therefore, it seems to be important to get some data from some simulated games. Rationally, the best resource to this subject is the RoboCup Tournament itself. However, an important question rises, when this resource is used to elicit data:

Does the agent do the best and optimal action in a particular situation in a simulated soccer game? there seems to be no proper answer for this question. Therefore, only those completely rational actions in the simulated world can be used as a resource for training data. For example, Figure 7 shows a situation that the attacker agent by moving to an empty space behind the defender opponents tricks those three rivals and gets an advantage to kick the ball and make a score.

These kinds of data can be used to train the Kohonen network in the tactics layer in decision-making part.

Figure 5. Move to the best point to kick.

Figure 6. Pass to a better teammate to make a point.

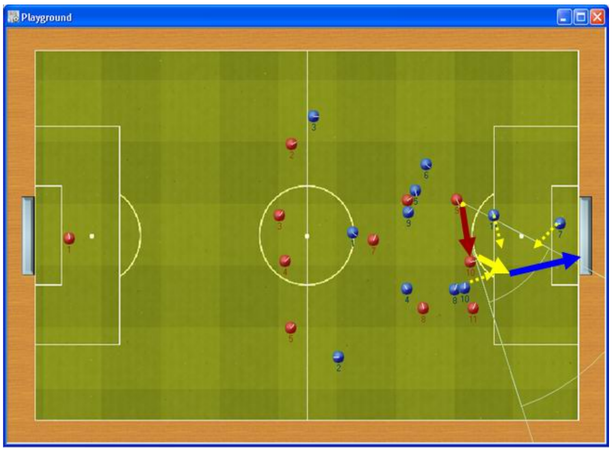


Figure 7. Tricking three rivals by moving to a empty space.

By these two resources, about one thousand data is gathered. Among these data, there is two hundred data, which the attacker agent is in doubt to what to do. These data is provided to show the performance of the Kohonen network in some ambiguous situations.

4.2. Decision-Making Layer Based on Kohonen Neural Network As it said before, the simulation soccer game is a real-time environment. Therefore, the most important factor in design a decision-making section for designers is that the calculation must be done in a limited time, called a cycle. All designers try to find a way to reduce the calculations' complexity and time cost. In tactics layer in decision-making part, if one looks at a task that must be chosen by an agent as a cluster, then decision-making problem in this layer can be seen as the clustering problem.

In clustering problem, the recurrent neural networks prove themselves as powerful methods to solve such problems [12,13]. Because of the real-time characteristics of the simulated world, the Kohonen network as a feed-forward clustering network trained in off-line seems to be a good choice for such an environment. This network maps the input almost linearly separable data to their proper clusters with its trained weighting matrix. The architecture of this network is shown in Figure 8.

This network uses the winner takes all training algorithm in its learning phase.

The rule for this training algorithm is:

$$\begin{cases} \hat{w}_m^{k+1} = \hat{w}_m^{k} + \alpha^{k}(x - \hat{w}_m^{k}) & m = \text{winner neuron} \\ \hat{w}_i^{k+1} = \hat{w}_i^{k} & \text{for } i \neq m \end{cases}$$
$$w_i^0 = \frac{1}{\sqrt{n}}[1 \;\; 1 \;\; \dots \;\; 1] \qquad \text{for } i = 1,2,\dots,p \tag{6}$$

In recalling phase, this network uses a max-block to indicate the winner neuron and consequently the proper cluster and the proper task to do [12].

Figure 9 shows the structure of the Tactic layer in decision-making section for an attacker agent.

It can be seen from the figure that the pure data from the world-modeling part that process data sent by Soccer Server initially, goes to the transducer layer.

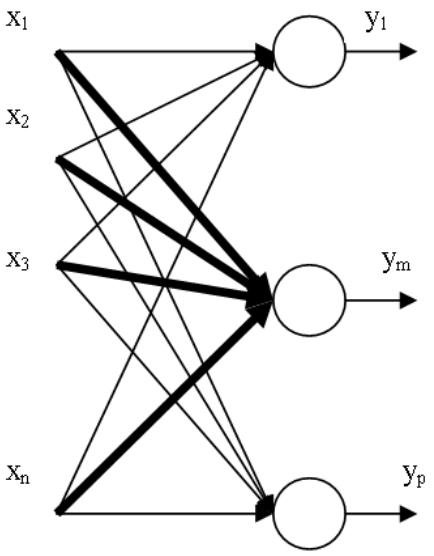


Figure 8. The structure of the kohonen network.

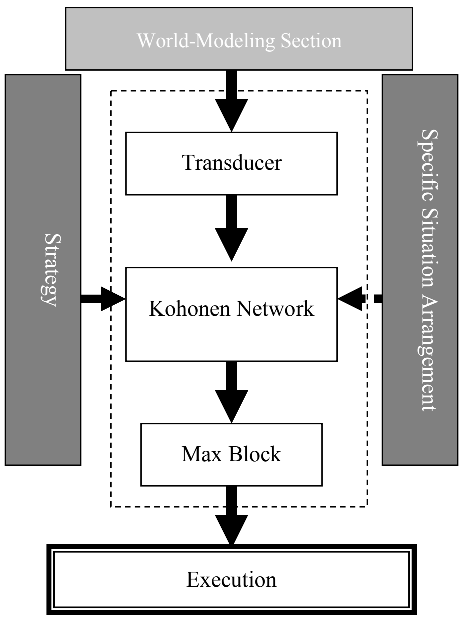


Figure 9. The structure of the tactic layer.

This part of tactics layer includes those functions that model the behavior of a real soccer. In addition, these functions map the pure data to an almost linearly separable space, so it is possible for Kohonen network to work properly. Actually, this section has the most calculation and is a time consuming section and it needs to be designed properly; therefore it is possible for the decision-making part to act properly in this real-time environment. The cause of this problem is that the transducer part includes many functions that have to act simultaneously to produce some linearly separable data. Functions such as, distance to the centre of the rival's goal, agent's density in the penalty region, view angle for ball owner and all other teammates in the penalty region. The worst function among these functions is the confidence function, because it needs to calculate many characteristics in the field. When transducer block produces a state vector, it goes to the Trained Kohonen block, which has been trained by 500 data (the others have been kept to test the performance of the system) to converge to the proper cluster for this state vector by its max block.

After training mode, the weighting matrix of the Kohonen network for the attacker agent is:

$$W_{forward} = \begin{bmatrix} 0.3064 & 0.3728 & 0.2891 & 0.5655 & 0.1468 & 0.0651 & 0.1186 & 0.1324 & 0.2067 & 0.3422 & 0.3829 \\ 0.0961 & 0.1597 & 0.1063 & 0.2283 & 0.3810 & 0.382 & 0.3019 & 0.5039 & 0.4668 & 0.2182 & 0.0082 \\ 0.3903 & 0.2936 & 0.6023 & 0.1134 & 0.2639 & 0.2286 & 0.1278 & 0.0098 & 0.1963 & 0.1303 & 0.4382 \\ 0.2658 & 0.1607 & 0.2267 & 0.1371 & 00973 & 0.0606 & 0.0618 & 0.218 & 0.0967 & 0.7522 & 0.4400 \end{bmatrix} \quad (7)$$

Following is a description for each line corresponded to an action which indicates; The first line is related to the "kicking" option. The second line is related to the "pass to a better teammate" option and the third line is related to the "move to a better situation to kick" option and finally the forth line is related to the "pass to a teammate in a safe situation" option.

The first four columns are related to the ball owner parameters as they were introduced before, such as distance to the centre of the rival's goal, view angle, open space upon the agent, confidence coefficient respectively. The next five columns are related to parameters for the player in the best situation among all the teammates near the rival's penalty region, which are almost like the ball owner parameters except one parameter that is not in the ball owner parameters. This parameter indicates the confidence coefficient that the teammate can intercept the ball after ball owner pass the ball to him. Two last columns are related to the teammate in the safe position and indicate the confidence parameter of intercepting the ball after passing the ball, and the other is the distance from the ball owner.

It can be seen that in kicking row, the confidence parameter as same as the view angle, is big enough and the attacker agent decides to kick the ball according to these parameters. In the second row, which is related to the pass the ball to another teammate in the best point, it can be seen that this teammate has better confidence of kicking and better view angle. Therefore, this action is rational enough. In the third row which is related to the move to a better place to kick, ball owner has the most open-space value, so rationally he has lots of space upon himself (like a one to one situation). In the last row, there is not any good coefficient to act. However, there exists a teammate in the safe place who can intercept the ball as soon as ball owner pass the ball to him. Therefore, it is rational that ball owner passes the ball to that teammate in the safe place. If there is not such a teammate, ball owner choose one of these action (mostly kick action) according to the performance of the Kohonen network.

5. SIMULATION RESULTS

The Kohonen network which is trained in the former section is used in tactics layer in decision-making part. The success of the trained agents in choosing the best action in attack mode are evaluated for 400 trials in the Simulated Soccer environment is evaluated in this section. As it said before, half of the data elicited from some ambiguous situations, which the attackers are in doubt to what to do. They doubt in kicking, passing, or moving to a better point to kick.

Table 1 shows the performance of the Kohonen network according to the number of the training data. As can be seen, the maximum percentage of success in choosing the proper task occurred with 500 training data. In case of using less than 300 training data, performance are not good enough and this is obvious because, there are still many situations that network and agent has not learnt yet. An interesting thing occurs when 600 or more training data is used for learning. The network's performance is reduced when network is learnt with these amounts of data. Therefore, one can say that "over-training" phenomena occurs when network is trained by more than 500 data.

The next results are reported as a comparison of the quality of decision-making part based on Kohonen network with some other learning methods such as Multi Layer Perceptron neural network (MLP) and Adaptive network based Fuzzy Inference System (ANFIS). All methods are used in their optimal performance. These results are presented in Table 2. It can be seen that Kohonen network's has the best performance in decision-making among these methods. In addition, it can be seen that performance of the ANFIS is better than the performance of the MLP network. An explanation for this phenomenon may be the quality of ANFIS in using transcendental knowledge by its fuzzy nature.

It has to be said that this Kohonen network, which has been used for decision-making part, answers properly in all the clear situations. In these situations, agents have no doubt to do an action. In addition, in "kicking" action and "pass the ball to a teammate in better situation" action, even in ambiguous situations, network converges to the proper cluster. Only in some ambiguous situations of the other two tasks, network does not work properly. When agents are in doubt to move to a better place to kick or in another kind of ambiguous situations that agents are helpless and marked by opponents and there is no such a way to make a score, it is possible that decision-making fails. It is interesting that often network converges to the kick task! However, this phenomenon is not weird and this failing in some way is expectable. Because when a real soccer attacker is considered in these kinds of ambiguous situations, he probably chooses the kick action. Therefore, this Kohonen network seems to be a good model for a skillful attacker in decision-making phase, even in those kinds of ambiguous situations.

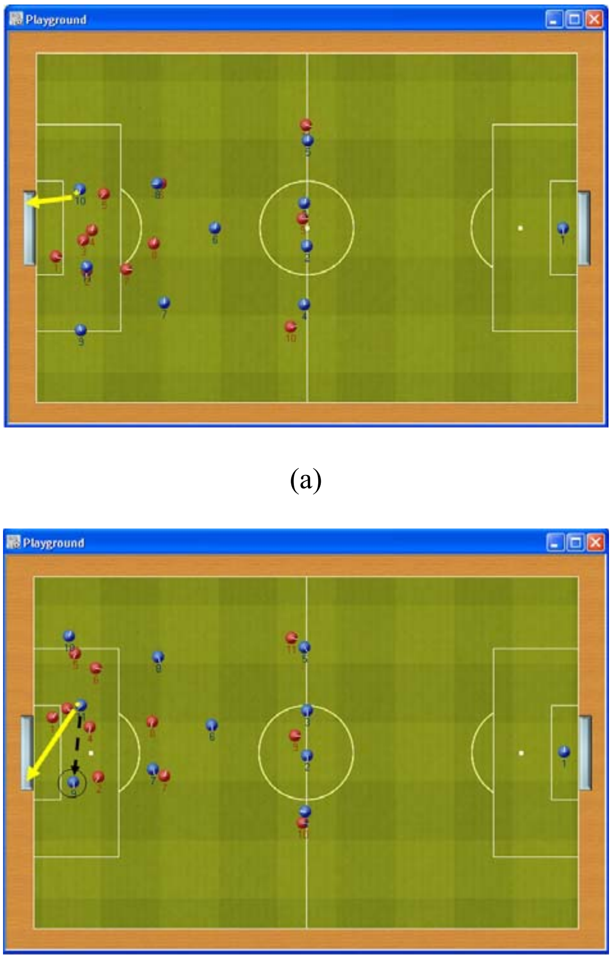
Figure 10 and 11 shows performance of the Kohonen network in the soccer simulation environment. Each figure, contains two situations,

TABLE 1. The Performance of the Kohonen Network in Choosing the Proper Task.

Number of the Input Data	Percentage of Success (%)
200	72.5
300	82
400	84.2
500	88.2
600	85.7

TABLE 2. Comparison of Performances by Different Methods used in Decision-Making Section.

	Specifications	Percentage of Success (%)
Kohonen Network	11 Input Nodes and 4 Output nodes	88.2
ANFIS	8 Rules and 165 Epochs	82.3
MLP	6 Neuron in Hidden Layer	79.4
A Naïve Attacker	--------	35.3



(a)

(b)

Figure 10. (a) Kick with no doubt, (b) Kick with doubt.

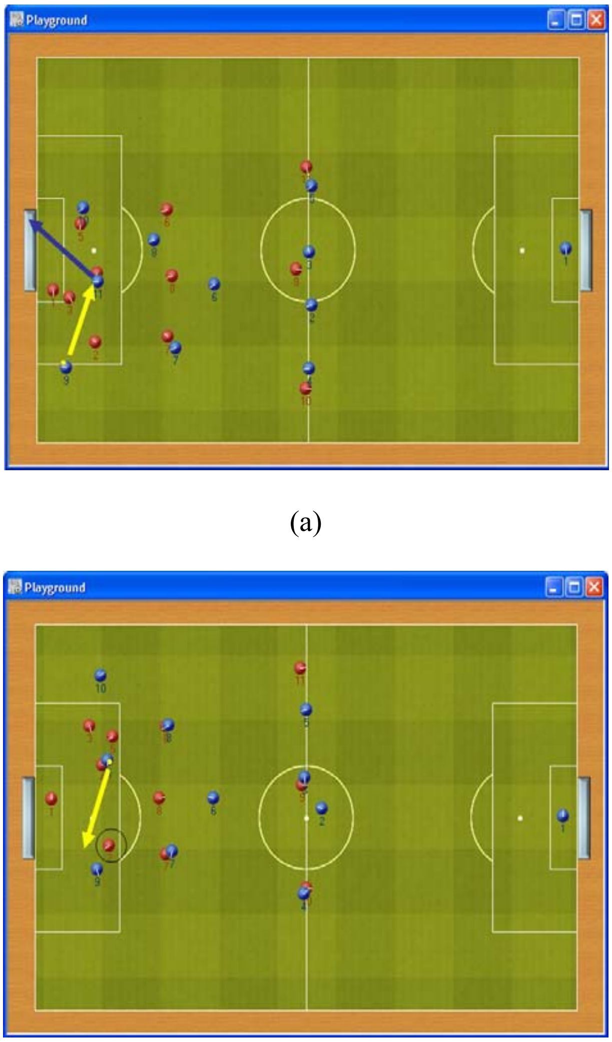
one is a situation that agent is not in doubt and he acts with great confidence, and the other is a situation that the agent is in doubt to what to do.

The output of the Kohonen network in kicking action with confidence and in doubt can be seen in Table 3.

It can be seen from the Figure 10 that in kicking action with no doubt, the attacker is in a place that is near the opponent's goal and his view angle is very good, and there is no one who can intercept the ball after kicking. Therefore, the agent kicks the ball. In the other situation, ball owner still has a good position, but there is a teammate near him who can kick the ball too. In this situation, ball owner himself kicks the ball and makes a score.

The output of the Kohonen network in "pass to a teammate in a better position" action with confidence and in doubt can be seen in Table 4.

It can be seen from Figure 11 that in "pass the ball to a teammate in better situation" with no doubt, ball owner is not in a proper place to make a score, but there is a teammate in the penalty region that has a good view angle, and if he kicks the ball, no one could reach it. Therefore, ball owner pass the ball to that teammate. Then, he scores. In other situation which ball owner is in doubt to pass the ball to the teammate, he still is not in a good position to make a score. Like the former situation, there is a teammate in a good position. However, this time there is an opponent around him who can reach the ball if the ball owner passes the ball to



(a)

(b)

Figure 11. (a) pass the ball to a teammate in better place with no doubt, (b) pass the ball to a teammate with doubt.

his teammate. In addition, that teammate is not facing the ball, and first he has to turn, and then move to intercept the passed ball. Therefore, when ball owner passes the ball to his teammate, this is the opponent who intercepts the ball first. It has to be said that, although this action is failed, but in this situation, this action seems to be the best action to do because ball owner cannot kick the ball. In addition, he cannot move to a better place, and there is no teammate in safe place who can get the ball from the ball owner.

TABLE 3. Output of the Kohonen Network for Kicking Action.

	Kick with no Doubt	Kick with Doubt
Kicking	0.96224	0.91086
Pass to a Teammate in Better Place	0.66941	0.83559
Move to the Better Place	0.74077	0.78213
Pass to the Safe Teammate	0.68168	0.79031

TABLE 4. Output of the Kohonen Network for Passing Action.

	Kick with no Doubt	Kick with Doubt
Kicking	0.64361	0.75633
Pass to a Teammate in Better Place	0.96802	0.87144
Move to the Better Place	0.64979	0.72793
Pass to the Safe Teammate	0.53695	0.4565

6. DISCUSSION AND CONCLUSIONS

In this paper, a structure for tactics layer in decision-making part of an attacker agent in simulated soccer game is proposed. This structure was designed in such a way that it could be possible for such agents to act properly in this real-time dynamic environment. Because of the real-time nature of the simulated world, some simplifications have to be done. Therefore, it could be possible to finish the calculation in a limited time. As a result, this simplification will reduce the performance of the decision-making part. Therefore, it needs to apply an architecture, which can act properly with this undeniable simplification. Kohonen network as a feed-forward neural network with its ability in discovering any relationships of interest that may exist in the input data seems to be a good choice for decision-making section in this real-time environment. Because On one hand, this network with its simple architecture does not need any complex calculations and this is a good attitude in this real-time environment. On the other hand, this network with its great ability in clustering can be used as a method, which can mix transcendental knowledge with simulated world characteristics, by using some elicited data from real and simulated soccer game for its training. The great performance of this network among different kinds of intelligent methods completely depicts its potential in this subject.

7. ACKNOWLEDGEMENTS

The cooperation of RoboCup Simulation Team of National Organization for Developing Exceptional Talents (Farzanegan Qazvin), Ms. Hammidirad, Ms. Namazypour, Ms. PourRajabi, Ms. Roohiha, is gratefully acknowledged.

In addition, the authors wish to thank the reviewers of "International Journal of Engineering" for their helpful suggestions and bringing further references to the attention of the authors.

8. REFERENCES

1. Weiss, G., "Multi-Agent Systems: A Modern Approach to Distributed Artificial Intelligence", MIT Press, Cambridge, (1999).
2. Oliviera, E., Fischer, K. and Pankova, O., "Multi-agent Systems: Which Research for Which Applications", ***Robotics and Autonomous Systems***, Vol. 27, (1999), 91-106.
3. Kurdthongmee, W., "A Novel Kohonen SOM-Based Image Compression Architecture Suitable for Moderate Density FPGAs", ***Image and Vision Computing***, Doi: 10.1016/j.imavis.(2007), 11.010.
4. Afolabi, M. O. and Olude, O., "Predicting Stock Prices using a Hybrid Kohonen Self Organizing Map (SOM)", ***Proceedings of the 40th Hawaii International Conference on System Sciences***, (2007), 1-8.
5. Lie, Y. and Yao, X., "Evolving Artificial Neural Networks for Medical Applications", ***Proc. of First Korea-Australia Joint Workshop on Evolutionary Computation***, Korea, (September, 27-29, 1995), 1-16.
6. Davis, L., "Mapping Neural Networks into Classifier Systems", ***Proc. of the 3rd Int. Conf. on genetic Algorithms (ICGA, 89)***, George Mason University, (1989), 375-378.
7. Ersoya, O., Aydara, E., Gourgaudb, A., Artunerc, H. and Bayhan, H., "Clustering of Volcanic Ash Arising from Different Fragmentation Mechanisms Using Kohonen Self-Organizing Maps", ***Computers and Geosciences***, Vol. 33, (2007), 821-828.
8. Stone, P., "Layered Learning in Multi-Agent System", PhD Dissertation, School of Computer Science, Carnegie Mellon University, U.S.A., (1998).
9. Tung, W. L. and Quek, C., "Falcon: Neural Fuzzy Control and Decision Systems Using FKP and PFKP Clustering Algorithms", ***IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics***, Vol. 34, No. 1, (2004), 686-695.
10. Saadatmand-Tarzjan, M., Khademi, M., Akbarzadeh, M.-R. and Abrishami Moghaddam, H., "A Novel Constructive-Optimizer Neural Network for the Traveling Salesman Problem", ***IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics***, Vol. 37, No. 4, (2007), 754-770.
11. Kostiadis, K. and Hu, H., "Reinforcement Learning and Cooperation in a Simulated Multi-Agent System", ***Proc. of the 1999 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems***, (1999), 990-995.
12. Stone, P. and Sutton, R. S., "Scaling Reinforcement Learning Toward Robocup Soccer", ***18th International Conference on Machine Learning (ICML 2001)***, Williamstown, MA, U.S.A., (June 2001), 537-544.
13. Weiss, G., "Distributed Artificial Intelligence Meets Machine Learning: Learning in Multi-agent Environment," Springer, Berlin, Germany, (1997).
14. Shi, L., Jinyi, Y., Zhen, Y. and Zengqi, S., "Multi Rewards Fuzzy Reinforcement Learning Algorithm in RoboCup Environment", ***Proc. of the 2001 IEEE Int. Conf. on Control Applications***, Mexico, (September 5-7, 2001), 317-322.
15. Chohra, A., Scholl, P., Kobialka, H. U., Hermes, J. and Bredenfeld, A., "Behavior Learning to Predict using Neural Networks (NN): Towards a Fast, Cooperative and Adversarial Robot Team (RoboCup)", ***2nd Workshop on Robot Motion and Control***, Germany, (October 18-20, 2001), 79-84.
16. Sharbafi, M., Daneshvar, R., Aminaee, H. and Lucas, C., "New Implicit Approach in Coordination Between Soccer Simulation Footballers", ***International Conference on Computer Science and Information Technologies, CSIT'2005***, Institute for Informatics and Automation Problems, National Academy of Sciences, Yerevan, Armenia, (September 19-23, 2005), 292- 296.
17. Meyer, J., Adolph, R., Stephen, D., Daniel, A., Seekamp, M., Weinert, V. and Vsser, U., "Decision-Making and Tactical Behavior with Potential Fields", ***RoboCup 2002, Lecture Notes in Artificial Intelligence Springer***, (2002), 304-311.

18. Zargari, A. A. and Lucas, C., "A Strategy for Passing by Robot Soccer Players Using Fuzzy Logic", ***15th Iranian Conference on Electrical Engineering, ICEE 2007***, Iran Telecom Research Center, Tehran, Iran, (May 15-17, 2007).
19. Atashpaz Gargari, E. and Lucas, C., "A Strategy for Passing by Robot Soccer Players Using Fuzzy Logic", ***15th Iranian Conference on Electrical Engineering, ICEE 2007***, Iran Telecom Research Center, Tehran, Iran, (May 15-17, 2007).
20. Balkenius, C. and Moren, J., "A Computational Model of Context Processing", ***6th International Conference on Simulation of Adaptive Behavior***, Cambridge, (2000), 113-122.
21. El-Nasr, M., Loerger, T. and Yen, J., "Peteei: A pet with Evolving Emotional Intelligence", ***Autonomous Agents 99***, (1999), 9-15.
22. J. Velasquez, "A Computational Framework for Emotion-Based Control", The Grounding Emotions in Adaptive Systems Workshop SAB'98, Zurich, Switzerland, (1998).
23. Gholipour, A., Lucas, C. and Shahmirzadi, D., "Purposeful Prediction of Space Weather Phenomena by Simulated Emotional Learning", ***Int. Journal of Modeling and Simulation***, Vol. 24, (2004), 65-72.
24. Sharbafi, M. A., Lucas, C., Toroghi Haghighat, A., Ghiasvand, O. A. and Aghazade, O., "Using Emotional Learning in Rescue Simulation Environment", ***Transactions on Engineering, Computing and Technology***, Vol. 13, (May 2006), 333-337.
25. Sharbafi, M. A. and Lucas, C., "Designing a Football Team of Robots from Beginning to End", ***International Journal of Intelligent Technology* (*IJIT*)**, Vol. 3, No. 2, (2006), 101-108.
26. Sharbafi, M. A., Lucas, C., Toroghi Haghighat, A., Ghiasvand, O. A. and Aghazade, O., "Using Emotional Learning in Rescue Simulation Environment", ***13th World Enformatika Conference***, Budapest Hungary, (May 26-28, 2006), 86-92.
27. Yang, Z.-G., Li, L.-S., Yue, Y. and Tang, B., "Study on Shooting Skill in RoboCup Simulator League", ***Proc. of the 2nd Int. Conf. on Machine Learning and Cybernetics***, Xi'an, (November 2-5, 2003), 2089-2092.
28. Zurada, J.-M., "Introduction to Artificial Neural Systems", West Publishing Company, New York, U.S.A., (1996).
29. Bianchia, D., Calogerob, R. and Tirozzia, B., "Kohonen Neural Networks and Genetic Classification", ***Mathematical and Computer Modeling***, Vol. 45, (2007), 34-60,.
30. Hagan, M.-T., Demuth, H.-B. and Beale, M., "Neural Network Design", PWS Publishing Company, New York, U.S.A., (1996).
31. Bailer-Jones, C. and MacKay, D., "A recurrent Neural Network for Modeling Dynamical Systems", ***Network: Computation in Neural Systems***, England, U.K., (November 1998), 531-547.
32. Kitano, H., Asada, M., Kuniyoshi, Y., Noda, I., Osawa, E. and Matsubara, H., "RoboCup: A Challenge Problem for AI and Robotics", RoboCup-97: Robot Soccer World Cup I, Springer, (1998), 1-19.
33. Noda, I., Chen, M., Foroughi, E., Heintz, F., Huang, Z-X., Kapetanakis, S., Kostiadis, K., Kummeneje, J., Obst, O., Riley, P., Steffens, T., Wang, Y., and Yin, X., "Soccer Server Manual, RoboCup Federation", http://www. Robocup.org, (2002).
34. Foroughi, E., Heintz, F. and Kapetanakis, S., "RoboCup Soccer Server User Manual: for Soccer Server Version 7.07 and Later", ***GNU Free Documentation***, (2001), 15-21.
35. RoboCup Federation, "The Goals of RoboCup", On http://www.robocup.org/overview/22.html, 2000, Verified on 12th, (February 2001).