Nemesis Team Description 2010

Mehrab Norouzitallab[1], Amin Javari[1], Alireza Noroozi[1], S.M.A. Salehizadeh[1], Kourosh Meshgi[1]

[1] Amir Kabir University of Technology, Hafez Ave., Tehran, Iran

{m.norouzitallab, a.javari, smasalehizadeh, meshgi}@Gmail.com, ar_noroozi@Yahoo.com

Abstract. In our study, we tried to develop our teams in such a way that machine learning techniques and advanced artificial intelligence tools have the main role in improving skills and increasing team performance. We consider soccer simulation platform as an uncertain and dynamic environment, so we develop learning algorithms according to this important feature and agent's partial observability.

1 Introduction

The Nemesis team was established in 2004 aiming to develop our team in such a way that machine learning techniques and advanced artificial intelligence tools have the main role in improving skills and increasing team performance. Each year, new members are joining our team to further their studies on these fields and use this simulation environment and previously implemented team as their basis of work. Nemesis is founded to serve as a platform for machine learning schemes such as artificial neural networks, evolutionary algorithms and reinforcement learning. Today this team is used as a platform for testing new ideas on as long as implementing latest papers to observe their dynamics. Several course projects and master thsse are implemented on this base too.

Nemesis base is updated by HELIOS 2008 code release under GPL and since then we change our Formation strategy to Fuzzy ARTMAP [8] (which is modified to perform better this year), add a Mark skill based on Maximum Weighted Bipartite Matching [1], improve our Block skill by enhancing a method called neuroHassle [5], and improve Offensive positioning by the means of PSO algorithm [2]. Also we introduce a framework called Mental Simulation [6] for Decision making that we are implementing the platform for it so far, but research on this subject has just begun. There are many more improvements everywhere in the code e.g. major improvement of Passing, minor debug of Dribbling and so on.

As a software cycle policy our team publishes the code at the end of March annually, and lots of complementary material along with source code will be available on: http://mnt.ir/nemesis

2 Marking

Mark skill is one the important defensive skills in soccer simulation. We formulated this task as an assignment problem, in which our players should observe and follow opponent players to confine their collaborative offensive abilities. To address this assignment problem we employ "Maximum Weighted Bipartite Matching" as the framework of this task assignment problem and make use of Hangarian algorithm to solve it [1]. Following subsections describes the MWBM methodology and our approach to utilize this solution in our problem efficiently. Note that this skill is under training and not fully embedded inside the source code so far.

2.1 Maximum Weighted Bipartite Matching

A graph $G = (V,E)$ is bipartite if there exists partition $V = X \cup Y$ with $X \cap Y = \varnothing$ and $E \subseteq X \times Y$. A *matching* is a subset $M \subseteq E$ such that $\forall v \in V$ at most one edge in M is incident upon v and its size, $|M|$, is equal to the number of edges in M. A *maximum matching* M is a matching such that every other matching M' satisfies $|M'| \leq M$. Thus, given bipartite graph G, the task is to find a maximum matching.

In the case of weighted bipartite graphs [1], edges of the graph has a weight, or value, $w(i, j)$. The weight of matching M is the sum of the weights of edges in M, $w(M) = \sum_{e \in M} w(e)$. Hence, the assignment task is reduced to finding a maximum weight matching given bipartite weighted graph G. Without loss of generality G can be assumed as a complete weighted graph by adding edges of weight 0. A *perfect matching*, is one in which every vertex is adjacent to some edge in it. It's clear that a max-weight matching is perfect.

A vertex labeling is a mapping $\ell : V \rightarrow R$. A *feasible labeling* is one such that $\ell(x)+\ell(y) \geq w(x, y)$ and the equality graph with respect to ℓ is noted as $G = (V,E_\ell)$ where

$$E_\ell = \{(x, y) : \ell(x)+\ell(y) = w(x, y)\} \quad \textbf{(1)}$$

Theorem: If ℓ is feasible and M is a perfect matching in E_ℓ then M is a max-weight matching.[1]

2.2 Applying MWBM Structure to Player Marking Problem

There are plenty of considerations regarding the efficient assignment of our defenders to opponent team attackers via Marking. First we establish the bipartite graph comprising teammate defenders on one side and attackers on the other side as the graph's nodes such that, each edge has a weight, $w(i,j)$ which represents the importance of marking player j by player i . In order to calculate these weights a linear function of the effective factors is defined, as follows.

$$w(i,j) = \frac{1}{m} \sum_{k=1}^{m} C_k F_k(i,j) \quad \textbf{(2)}$$

Where m is the number of effective factors, F represents the value extracted for feature, and C is the respective coefficient. Table 1 contains a list of these features which can be classified into two classes: the factors which determine the ability of defender to mark the attacker (F_1,F_2), and those correspondent with the risk of the opponent (F_3,F_4,F_5,F_6). These features are defined based on several expert rules and are based mathematical formulas.

Table 1. Effective factors for calculating weights of the MWBM structure for Mark skill

Factor		Description	Value
Defender	F_1	Distance from defender home position to mark position	The less this distance is, the higher value should be set to F_1
	F_2	Distance form defender current position to mark Position	The less this distance is, the higher value should be set to F_2
Attacker	F_3	Confidence of attacker position	The more accurate estimation of attacker point, the more accurate Mark position is calculated. Therefore other factors reach the more accurate values.
	F_4	Distance form attacker position to ball position	The nearer the attacker position to the ball position is, the more likely the attacker can receive the pass
	F_5	How dangerous the marking area is	The areas in which the probability of shoot or one-to-Goalie state is higher, assumed to be dangerous areas. This probability specifies the extent of hazardousness.
	F_6	Distance from mark position to the nearest point on goal line	The less this distance is, the more danger is occurred.

2.3 Parameters Tuning

The weights of MWBM can be optimized in two fashions: First, applying more effective factors (F_i), and second tuning the parameter (C_i).The six factors considered in Table 1 perform good enough to address the first issue. Thus, here we propose an approach to adjust the coefficients in such a way that the resulted weights lead to a promising MWBM structure for the Mark skill.

A training procedure is introduced to fine tune these parameters. In this scenario the attacker players of opponent team and the defender player of our team is in the field and in each episode the ball is granted to an attacker. Our player should mark opponent players based on MWBM. An evolutionary algorithm (e.g. PSO, ES, DE, etc.) could be applied to adjust the parameters based on these trainings. In our implementation, Particle Swarm Optimization method [2] is hired to maximize the fitness function resulted from the outcome of the scenario [3]. This fitness function is indeed the value obtained from Eq. 2 and the optimization procedure adjusts the values of C_i. The desired outcome of this scenario could be conquering the ball by the defenders, clearing the ball to out of field, or having the ball played further than its initial position from own goal. The scenario has undesired outcome when a ball leading opponent player takes the ball where he has the opportunity to shoot the ball toward the goal. The fitness of this outcome is generated automatically using a rule-base. It must be mentioned that this devised approach has been working well so that it outperforms the previously proposed techniques. In addition, our experiences in several competitions and simulation tests show that the above experimental setting can cover most of the cases.

3 Block Skill

Defending against incoming attacks and recapturing the ball is a crucial task for each team. Defending strategy consist of two sub-task: Positioning and Hassling. The former task aims to arrange players in free spaces so that they are capable of intercepting potential opponent passes, covering the direct defending player, marking the attacker player possesses the ball, and avoiding opponent to have clear shoot toward the goal (section 2). The latter task is to improving the aggression skill of defender in the manner that they can interfere the opponent ball leading player, "hassle" him, and bringing ball under their control while simultaneously hindering him from dribbling ahead. Moreover, the assignment of these two tasks is challenging because they can conflict and result in two undesired situations: no one interferes the attacker or two players decide to hassle the ball leader and leave a breach in defensive formation or leaving an opponent player uncovered. Also this assignment should maximize the collaborative defense utility [5]. A common choice for this assignment is to give the task of hassling to the closest player to the ball while others maintain a good defensive coverage formation.

Conquering the ball from an attacking player is risky and difficult to implement, because (i) it's hard to devise a trivial scheme to handle the broad variety of utilized dribbling strategies (ii) risk of over-specializing to some type of dribble strategies and loss of generalization for others that lowers the overall efficiency of the scheme and (iii) the importance of a duel between attacker and defender: if the defending player looses this duel, the attacker overruns him, and will achieve more space and better opportunities with few defenders ahead.

Brainstromers team has employed an effective scheme for the hassling task since RoboCup 2007 competitions called *neuroHassle*[5]. We are working on an enhanced version of this approach to be embedded in our block mechanism. The goal of this problem is to train defensive agents with reinforcement learning to hassle an attacker. In the other words, a given naïve defender finds a policy by trial and error, to conquer the ball from an opponent ball leading player with no a priori knowledge about his dribbling capabilities. The proposed reinforcement learning solution is value function estimation by a multi layer perceptron neural network. The architecture of our proposed solution differs slightly from the one explained in [5] yet use similar basics and training concepts.

Architecture: A MLP neural network with one hidden layer consists of 20 neurons with sigmoidal activation function. The neural network training is run in batch mode and uses back-propagation to minimize the mean square error of the value function approximation.

Inputs: These features are extracted from the environment and fed to the neural network.

1. Distance between defender and ball possessing attacker (Scalar)
2. Distance between ball and our goal (Scalar)
3. Velocity of defender (Vectored and Relative)
4. Velocity of attacker (Scalar: The absolute value of velocity)
5. Position of the ball (Vectored and Relative)
6. Defender body angle (Relative)

7. Attacker body angle (Relative to his direction toward our center of goal)
8. Strategic angle (∠GOM: G is the center of goal, O is the position of the opponent, and M is the position of our player
9. Stamina of the defender

The coordinated system is centered on the center of our player and the abscissa is aligned through our and the opponent player. The degree of partial observability is kept low.

Training: A large training data set should be provided for this task. This data set should cover various velocities and body angles of players and initial position of ball between them (to handle different start up situation for dribbling and defending), various regions of field (because dribbling players are very likely to behave differently depending on where they are positioned on the field), different adversary agent (to avoid over-specialization and maintain generalization), and different stamina size of defender (to consider realistic situation of the game).

Reinforce Signal: The outcome of a training scenario can be categorized in several groups. Regarding this outcome, a different reinforcement should be given to the agent:

- Erroneous Episode: Failure due to losing the ball by attacker because of a mistake, go out of the field, wrong self localization of the agent etc. is known as erroneous episodes and is omitted from training data.
- Success: Conquering the ball by the defender whether he has the ball inside of his kickable area or has a probably successful opportunity of tackling. This outcome will be rewarded by a great value.
- Opponent Panic: A non-dribbling behavior of attacking ball leading opponent player. This behavior takes place (i) when a defender approaches the attacker, (ii) when the defender hassles him too much , or (iii) when he simply do not consider the situation as a suitable one for dribbling. In these cases the attacker kicks the ball as a pass, toward goal or somewhere else (usually forward). This outcome is considered as a draw and with respect to the type of shoot to be toward the goal or not, we penalize or reward the situation by a small value.
- Failure: If none of the other cases has happen. This means that attacker has the ball in his kick range and overrun defender by some distance, or has approached the goal such that a goal shot is hardly stoppable. This outcome is punished by a large value.
- Time Out: If the struggle over the ball doesn't come into one of above mentioned states within a reasonable time. This situation will be punished or rewarded based on the offset of the ball from its initial position.

The learning task of this problem is episodic and the scenario is reset after each episode so there's no need for discounting and the learning rate used should be 1.0. Also to enable exploration to find better and more effective solution for defense we use criteria of energy saving mixed with Boltzman exploration to modify online greedy policy during training. The idea behind this choice is that although large sets and random episodes with start situation brings about a good level of state space exploration as assumed in [5], but the found policy may be not efficient in the terms of stamina, and yet may not cover various dribbling tricks enough and not generalized properly.

Actions: An agent is allowed to choose the low level actions of turn(x) and dash(y) where the domains of bots commands' parameters (x from [−100, 100], y from [−180◦, 180◦]) are discretized such that in total 76 actions are available to the agent at each time step.

Although the effectiveness of policy will be influenced by the presence of other players in the field and the attacker may behave differently, but by a good formation of other defenders, so that passing between opponent players become more risky, this policy gains more importance.

In future works we plan to:

- Enable a defender to shout for help if his stamina level decreases to a critical level;
- When the score of the team is in good winning margin, the defenders tries to reach a state of Time Out and save more energy by preventing a player to dribble ahead;
- When the attacking team has ball in their defensive area and a gap in the midfield, our players start to hassle them from opponent defensive area to conquer the ball and gain good chance of scoring;
- Train a defender to hassle when one more player from each team of attacker and defender are present in the field to enable hassling player to block the passes from the source.

4 Positioning

4.1 Formation Strategy

We propose a new framework for formation strategy in which every agent is capable of extracting features by means of expert knowledge from observing agent behaviors. Last year, we used Fuzzy ARTMAP as knowledge based neural network for extraction of expert knowledge [8]. As a result a model of behavior could be formed combining low-level behavior and expert knowledge. Experimental tests performed last year showed that the proposed model exhibits a higher performance than the conventional BPN. Our current framework is to some extent different. We added the current position of the agent to the current inputs of the neural network and generate the home position. This would result to a completely dynamic positioning framework. It must be mentioned that we implement all of these algorithms in Matlab, and results demonstrate that the current approach outperforms the previous ones.

4.2 Offensive Positioning

Due to the defensive positioning and mark skill improvement in most of the teams, it is important for us to propose a methodology which can improve the offensive positioning task in our team. To do so, we introduce two circular regions, one for each player of our team and the other around the original position obtained by the positioning method, both with the radius of 10. Then, we quantize the intersection area between these 2 circles into 40 points. Finally, we weight these 40 points plus the original positioning point according to the same procedure that proposed for tuning the parameters in Mark skill development but using completely different features as illustrated in Table 2.

Table 2. Effective factors for calculating weight of each candidate offensive positioning point

Factor	Description	Value
F_1	Distance from the point to the nearest point on the opponent goal line	Lower values are better.
F_2	Difference of the angle between the upper frame of the goal and the point, and the angle between the lower frame of the goal and the point (goal view angle)	The higher this value is, the more likely to shoot toward the goal.
F_3	Distance from the point to the nearest face2face opponent	Higher values are better.
F_4	Probability of receiving the pass in that point	Higher values are better. Calculated by a simple geometric algorithm.
F_5	Number of cycles takes to reach the point	Lower values are better.
F_6	Distance from home position to the point	Lower values are better.
F_7	Distance from ball position to the point	Lower values are better.

5 Decision Making

Klein made a decision model called Recognition-Primed Decision Model [6]. The model solves the problem by considering the situation, recognizing it, recalling the situation's experience and implementing it. Usually the problem's solution has four by-products: goal, cues, expectations and course of action. The decision maker knowing the situation, also know the goal should be followed, cues should be used to gather information, expectations should be monitored and course of action should be implemented. Although, the model has various aspects, we focus here on its mental simulation part. When a decision maker engaged in a complicated situation, after recognizing the situation and recalling the solution, he or she tends to be assured of its success. This is done usually when decision maker senses their insufficient cognition of the environment. Therefore, he or she scans the course of action for tuning, modifying or even omitting it and making a new solution. Klein calls this process, mental simulation. All of the mental simulation is done mentally before implementing the course of action or even while implementing it. The mental simulation makes the advantages and disadvantages of the solution more clear. The mental simulation process defined there based completely on the human cognitive abilities and is similar to the findings of Hastie in [7]. They showed that when jury is going to reach a verdict of guilty or not guilty, they makes their own story or accepts the lawyer's believable story to make final decision. The story should cover the evidences and be a comprehensive one.

In the 2D soccer, the teams plan to win opponent is very similar to this kind of decision making. Especially when an unpredictable change occurs in the middle of the game it is very important to guess the opponent's plan and proposing a proper strategy to conquer. This framework is a new viewpoint to decision making and we plan to further our study in this area and use it as our future work.

6 Conclusion

In this study, we proposed some of our novel strategies and methods to improve the important skills and important tasks in an appropriate way. A new framework introduced to address the assignment problem, player marking, which has been based on the maximum weight bipartite matching structure. A new dynamic formation strategy is suggested and discussed that the current method outperforms our previous one well. Furthermore, due to some improvement in defensive positioning of Robocup teams, we proposed an efficient approach for offensive positioning which can overcome the opponent team defensive and marking strategy. For block skill after implementing various techniques and method we come to the conclusion that reinforcement learning would results in a better performance among others. Thus, the block skill is improved via reinforcement learning approach. Finally we introduced a new framework for the decision making problem and we will further our study on it as our future work.

7 References

[1] M. Paul. Algorithmen für das Maximum Weight Matching Problem in bipartiten Graphen. Master's thesis, Fachbereich Informatik, Universität des Saarlandes, Saarbrücken, 1989.

[2] R.C. Eberhart, J. Kennedy, "*A new optimizer using particle swarm theory,*" Proceedings of the Sixth International Symposium on Micro Machine and Human Science (MHS'95), Nagoya, Japan, 1995, pp. 39–43.

[3] L. Messerschmidt and A. P. Engelbrecht, "*Learning to play games using a PSO-based competitive learning approach,*" IEEE Trans. Evol. Comput., vol. 8, no. 3, pp. 280–288, Jun. 2004.

[4]M. M. Korjani, A. Afshar, M. Norouzitallab, M. B.Menhaj, "*Dynamic Autonomous Agent Positioning Based on Computational Intelligence,*" Proceedings of IAENG Conferences, IMECS 2009.

[5] T. Gabel, M. Riedmiller, F. Trost, "*A Case Study on Improving Defense Behavior in Soccer Simulation 2D: The NeuroHassle Approach,*" RoboCup 2008: Robot Soccer World Cup XII, LNCS, Berlin, Springer Verlag.

[6] G. Klein, **Sources of Power: How People Make Decisions**, Cambridge, MIT Press, MA, United States, (1998).

[7] R. Hastie, and N. Pennington, Explanation-Based Decision Making, In T. Connolly, H.R. Arkes, and K.R.C. Hammond (Eds.), Judgment and Decision Making, An interdisciplinary reader, pp. 212-228, Cambridge University Press, United States, (2000).

[8] M. Norouzitallab, M.M Korjani, S.M.A. Salehizadeh, A. Pourshoghi,D. Khashabi, A. Razaghpanah, P. Kaviani, and M. Vosoughpour, "*Nemesis Team Description 2009,*" RoboCup 2009, Graz, Austria, 2009.