Nexus 2D 2008 Team Description

Amin Milani Fard, M. Hossein Ansari, Armin Ildermi, Sina Molavipour, Mahdi Aledaghi, Farid Seifi, Mahmoud Naghibzadeh

Software Simulation and Modeling Laboratory, Department of Computer Engineering
Ferdowsi University of Mashhad, Mashhad, Iran
nexus@um.ac.ir
http://nexus.um.ac.ir

Abstract. This paper presents an overview of our 2D Soccer Simulation Team. The main development features were done on decision making, action selection, and coach agent strategy making module using fuzzy logic mechanism and game theory approach.

1 Introduction

Nexus Soccer Simulation team is developed by a group of M.S and B.S students of Ferdowsi University of Mashhad, Iran and since establishment in 2002, achieved number of honors in international and local competitions. Our aim has been to construct stable and flexible agent architecture for our further development and research. This architecture is organized such that in each release of server we can apply the changes to this architecture, easily [1]. In this paper, we propose a comprehensive review of our research projects done in the RoboCup simulation field from the early stablishement of the team.

Nexus 2D team focused on decision making and action selection module which is considered a high-level skill. The best action is the one that helps towards the agent's utmost success. The selected action has to bring about the most possible positive results in each simulation cycle, consistent with the definition of an ideal rational agent [2]. Every agent has to analyze various conditions as well as to handle newly received information. An intelligent agent should use the recently received information from the server in the best feasible way. It is possible that parts of the received information from the surrounding be of no use or of little importance. Considering parameters of each of the three possible actions (shooting, dribbling, and passing), the information received from the surrounding area and the existing conditions can be divided into two parts: The information that is related to only one *specific* action and the information that is *common* among all three actions [3].

2 One-phase decision making mechanism

In our one-phase evaluation method, we use a specific weight for each parameter that affects an action. Through test runs and analysis of the outcomes, we have experimentally obtained proper weights for these parameters. The analysis was aimed at pinpointing the weaknesses of our team and trying to adjust the weights to improve the efficiency of the system. Each weight can be either a reward or a punishment whose summation for each one of the possible actions can result in a computed priority that recommends the most reasonable action. To obtain the weights, we start with an initial value for each weight. Afterward, the agent is made to contest several times and after each contest, the weights are readjusted. This process is similar to the supervised learning [3], but it is performed offline. The weights will gradually adjust to a stable value. To evaluate the priority for each one of the possible actions, both specific and common measures are used. The highest calculated priority determines the preferred action.

3 Two-phase decision making mechanism

To determine the best action from amongst all possible ones for a given situation, we [3] first recognize the best of each action type, i.e., the best shoot, the best dribble, and the best pass, independently. It is clear that, when the best possible shoot is sought the parameters that affect the shooting action are considered, only. For dribble and pass actions a similar process is followed. In the next phase, we select the best of bests, i.e., the system chooses the best action from amongst the three best actions shoot, dribble, and pass. In this phase, common measures are used in order to evaluate the actions. Table.1 shows the effects of different parameters on the three actions shoot, dribble, and pass. Fig.1 shows the overall work diagram.

Table.1 Parameters' effects on different actions

Code	Parameter	Action
P1	Distance to the penalty point	Pass
P2	Receiver view angle	Pass
P3	Number of opponent around	Pass
P4	Adjacency rate to the goal	Pass
P5	Receiver attackness	Pass
P6	Pass distance	Pass
S1	Shoot speed	Shoot
S2	Attackness	Shoot
S3	Shoot distance	Shoot
S4	Shoot angle view	Shoot
D1	Number of opponent around	Dribble
D2	Distance to offside line	Dribble
D3	Agent stamina	Dribble
C1	Action interception probability	All
C2	Teammate density in target area	All
C3	Target area information novelty	All

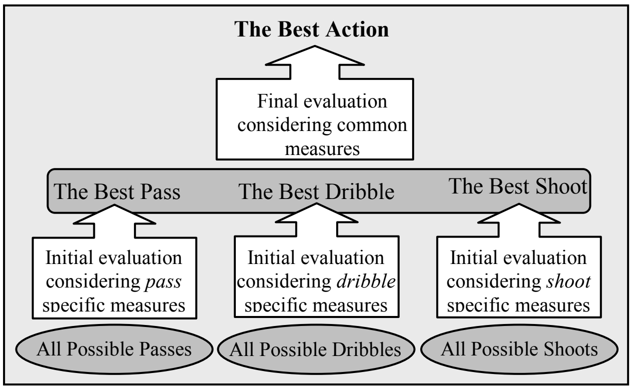

Fig.1 The two-phase selection diagram

4 Fuzzy two-phase decision making mechanism

We expected [4] the fuzzy system to be appropriate for decision-making process in the soccer simulation environment, considering the noise produced by the soccer server and uncertainties which affect all the perceptions and actions of the agents. Fuzzy systems are not sensitive to the completeness of the rule base, and even sometimes by removing half of the rules from a working system the performance does not considerably degrade, as long as the boundary rules are preserved in the fuzzy associative memory [5]. Our fuzzy rule base [4] includes 12 rules. The number of rules is much lower than the number of rules for our crisp system which is 50. For instance, the *high* priority measurement rules for the first phase are as the followings:

> ***IF*** *P1 is Short* ***AND*** *P2 is High* ***AND*** *P3 is Low* ***AND*** *P4 is Long* ***AND*** *P5 is High* ***AND*** *P6 is Medium* ***AND*** *C1 is Low* ***AND*** *C2 is High* ***AND*** *C3 is High* ***THEN*** *Pass priority is High*

> ***IF*** *S1 is Medium* ***AND*** *S2 is High* ***AND*** *S3 is Short* ***AND*** *S4 is High* ***AND*** *C1 is Low* ***AND*** *C2 is High* ***AND*** *C3 is High* ***THEN*** *Shoot priority is High*

> ***IF*** *D1 is Low* ***AND*** *D2 is Short* ***AND*** *D3 is High* ***AND*** *C1 is Low* ***AND*** *C2 is High* ***AND*** *C3 is High* ***THEN*** *Dribble priority is High*

And the high priority measurement rule for the second phase as bellow:

> ***IF*** *C1 is Low* ***AND*** *C2 is High* ***AND*** *C3 is High* ***THEN*** *selected action priority is High*

4 Game theory-based strategy making of coach agent

The most recent and final work done by Nexus on 2D simulation environment was developing a game theory-based data mining technique for strategy making of the soccer simulation coach agent [7]. A data mining process in the field of RoboCup soccer simulation involves gathering useful information out of the game data and acquires useful knowledge about the game situation known as strategy.

Game theory provides us with the mathematical tools to understand the possible strategies that utility-maximizing agents might use when making a choice. The simplest type of game considered in game theory is the *single-shot simultaneous-move* game. In this game all agents must take one action. All actions are effectively simultaneous. A single-shot game is a good model for the types of situations often faced by agents in a multi-agent system where the encounters mostly require coordination [8]. In a 2 player game, consider player *A* chooses a strategy and plays with it. Player *B* tries to learn *A*'s strategy and design his strategy as the best response to it. We assume *A* restricts itself to strategies realizable by *Deterministic Finite State Automata* (DFA). This is due to DFS strategies have been accepted widely as a model of bounded rationality [9, 10], and also learning the structure of an automaton has been shown to be a very hard problem [11].

In soccer simulation environment the coach agent is a privileged client used to provide assistance to the players. There are two kinds of coaches, the *online coach* and the *trainer*. The trainer can exercise more control over the game and may be used only in the development stage, whereas the online coach connects to server during the game and provides additional advice and information to the players. The coach agent can control the play-mode, broadcast audio messages containing information, and getting noise-free information about the movable objects. The online coach is thus a good tool for opponent modeling, game analysis, and giving strategic tips to its team mates.

In our proposed model, the coach agent constructs a knowledge-base of the game in the main memory containing 11 game matrixes for each 11 soccer player agents and assumes opponent's strategy realizable by a DFA. The number of states in that DFA is a complexity measure. The coach would then apply the polynomial time learning algorithm of $O(n)$ in which n is the number of states of the opponent automaton for all 11 game matrixes with respect to the payoffs assigned by the game knowledge-base as shown in Fig.3. A team strategy is mostly made using a knowledge-base or a set of <*state*, *action*> pairs. Using a special formation is another way in which each player has some predefined duties. These predefined duties are divided into static and dynamic.

	Intercept	Outplay	Pass	Shoot	Dribble	...
Intercept	1,1	0,3	0,4	0,4	0,3	...
Outplay	3,0	0,2	3,0	3,0	2,1	...
Pass	4,0	0,3	0,1	0,2	0,2	...
Shoot	4,0	0,3	0,1	0,1	0,2	...
Dribble	3,0	1,2	0,2	0,2	0,1	...
...	...	...	...	...	...	...

Fig.3 A sample agent game matrix

References

1. *M. Chen, E. Foroughi.* Robocup Soccer Server manual 7.07, August, 2002. RoboCup Federation. available: http://sserver.sourceforge.net.
2. *S. J. Russell, P. Norvig.* Artificial Intelligence A modern Approach, Prentice Hall, 1995.
3. *V. Salmani, M. Naghibzadeh, F. Seifi, and A. Taherinia.* A Two-Phase Mechanism for Agent's Action Selection in Soccer Simulation, The Second World Enformatika Conference, WEC'05, Istanbul, Turkey, pp. 217-220, February 2005.
4. *V. Salmani, A. Milani Fard, M. Naghibzadeh.* A Fuzzy Two-Phase Decision Making Approach for Simulated Soccer Agent, IEEE International Conference on Engineering of Intelligent Systems, pp. 134-139, Islamabad, Pakistan, April 22-23 2006.
5. *B. Kosko*. Neural Networks and Fuzzy Systems, 1992, Englewood Cliffs: Prentice Hall.
6. SoccerDoctor v1.0, by Wright Eagle, 2002. http://wrighteagle.org/sim/tools/SoccerDoctor.exe.
7. *A. Milani Fard, V. Salmani, M. Naghibzadeh, S. K. Nejad, H. Ahmadi.* Game Theory-based Data Mining Technique for Strategy Making of a Soccer Simulation Coach Agent, the 6th International Conference on Information Systems Technology and its Applications, (ISTA2007), May 23-25, 2007, Kharkiv, Ukraine.
8. *J. M. Vidal.* Learning in Multiagent Systems: An Introduction from a Game-Theoretic Perspective In Eduardo Alonso, editor, Adaptive Agents: LNAI 2636. Springer Verlag, 2003.
9. *A. Rubinstein.* Finite automata play the repeated prisoner's dilemma. ST/ICERD Discussion Paper 85/109, London School of Economics, 1985.
10. *A. Neyman.* Bounded complexity justifies cooperation in finitely repeated prisoner's dilemma. Economic Letters, pages 227-229, 1985.
11. *M. J. Kearns and U. V. Vazirani.* An Introduction to Computational Learning Theory. MIT press, Cambridge, Massachusetts, 1994.