A two-phase mechanism for agent’s action selection in soccer simulation

Vahid Salmani[1], Mahmoud Naghibzadeh[2], Farid Seifi[3], Amirhossein Taherinia[4]
[1] salmani@stu-mail.um.ac.ir
[2] naghibzadeh@ferdowsi.um.ac.ir
{[3] fa_se25, [4] am_ta51}@stu-mail.um.ac.ir
Software Simulation and Modelling Lab., Computer Dept., Ferdowsi University of Mashhad, Iran

Abstract*—**Soccer simulation is an effort to motivate researchers and practitioners to do artificial and robotic intelligence research; and at the same time put into practice and test the results. Many researchers and practitioners throughout the world are continuously working to polish their ideas and improve their implemented systems. At the same time, new groups are forming and they bring bright new thoughts to the field. The research includes designing and executing robotic soccer simulation algorithms. In our research, a soccer simulation player is considered to be an intelligent agent that is capable of receiving information from the environment, analyze it and to choose the best action from a set of possible ones, for its next move. We concentrate on developing a two-phase method for the soccer player agent to choose its best next move. The method is then implemented into our software system called *Nexus simulation team of Ferdowsi University*. This system is based on ***TsinghuAeolus[1] team that was the champion of the world RoboCup soccer simulation contest in 2001 and 2002.

***Keywords*—**RoboCup, Soccer simulation, multi-agent environment, intelligent soccer agent, ball controller agent.

I. Introduction

Soccer simulation environment is of client-server type. Two teams compete against each other with the help of the server program. For each player, a corresponding program receives visual, audio, and other sensible information that is sent by the server in every simulation cycle. This program has to analyze this information and perform whatever action it realizes to do [2].

The soccer agent's skills consist of three general types. The first type of skills is called *low-level skills*. Body or neck turning, shooting, moving and speaking are in this category of skills. The player can perform every one of these actions by directly sending appropriate commands to the server. The second type of skills is called *middle-level skills*. It includes actions like turning around, heading towards a designated point, controlling the ball, and shooting the ball along a defined angle. These skills are of a higher level of sophistication compared to low-level skills. The third type of skills is *high-level skills.* These skills are even more sophisticated than middle-level skills. Intercepting the ball, passing, and marking the opponent are samples of high-level skills [3]. The set of possible actions that a ball controller agent can perform is a subset of high-level skills. The ball controller agent can shoot, pass or dribble. It has only one simulation cycle to make a decision to perform one of these actions and inform the server to apply it. Some minor actions such as holding the ball are not dealt with, due to low its usage.

In Sections 2, we examine major parameters that affect soccer agent’s decision-making process. Section 3 presents evaluation methods for possible next moves of the ball-holding agent. Section 4 compares the two methods that are developed in Section 3. Section 5 is a summary.

II. Decision-making parameters

The number of possible actions in each simulation cycle depends on various parameters like the received information resolution, the number of teammates, the number of opponent players in the vicinity, overall team strategy, and coach instructions. As Fig. 1 suggests, the ball controller agent has numerous options to consider, before taking its next action. It has to analyze the situation and determine its best possible action. It then communicates its next action to the server. The server, in turn, updates the setting as though the action has taken place.

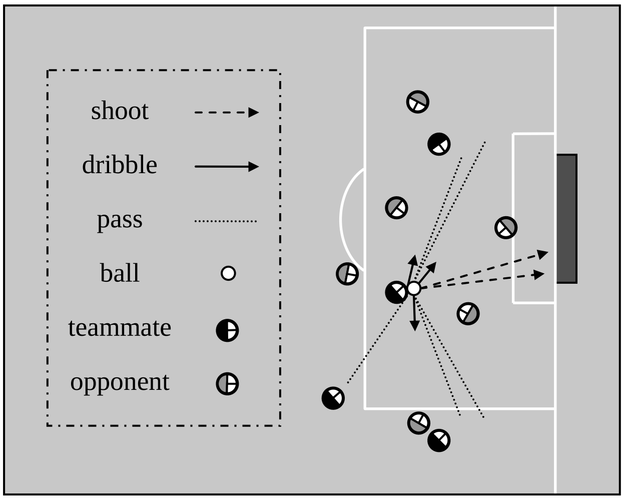

FIGURE 1
POSSIBLE ACTIONS THAT ARE AVAILABLE TO THE BALL CONTROLLER

The best action is the one that helps the most the intelligent agent's utmost success. The attempt chosen has to bring about the most possible positive results in each simulation cycle, given the definition we have about the ideal rational agent [4].

The agent has to recognize various conditions as well as to handle newly received information. The intelligent agent makes its decision based on the information it receives from the server. This that defines the agent’s surrounding thus proper handling of the information is of high importance. It is possible that part of the received information from the surrounding be of no use or of little importance. For example, for the evaluation of the next possible actions the information that defines the area that is close to the ball controller agent is more valuable than the information about far distances. The target area refers to the region in which the ball keeps moving, while the action is in process. Fig. 2 demonstrates the clear area that is required for the three actions of shooting, dribbling and passing and displays each action’s target area.

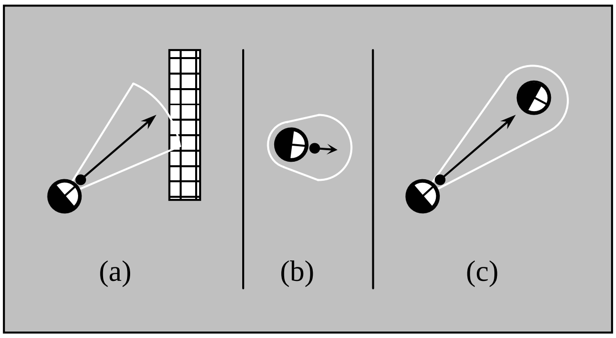

FIGURE 2
TARGET AREA FOR (A) SHOOT, (B) DRIBBLE AND (C) PASS ACTIONS

Given each one of the three actions of shooting, dribbling and passing has its own appointed target area and specific parameters, the information received for the surrounding area and the existing conditions can be divided into two parts:

1-The information that is related to one specific action.
2-The information that is common among all the actions.

Here, we like to elaborate on defining parameters that are specific to each type of action. Ball distance from the center of the opponent's goal, the relative direction of the goalie (goal keeper), and the ball motion are examples of shooting parameters. Dribbling length, the relative angle of player's body, and the dribbling direction are dribbling parameters. The distance between the passer and the ball receiver and the movement direction of the pass receiver are pass parameters.

The particular parameters of each action can be used as measures to evaluate the different cases of that action and find out which one is the best possible case. Similarly, common parameters among all actions, i.e., shoot, dribble, and pass, can be used to evaluate and prioritize the three types of actions. Of the most important common parameters among the three actions are: the sensitivity of the target area, the density of the opponent players in that area, the probability of the ball interception by opponent players, the freshness of the received information about the target area can be named [5]. The density degree of the opponents in a region explains the degree of ability of acting in that region. It is closely related to the probability that the opponent players being able to overtake the ball. Soccer agents suffer from view restriction. Every soccer agent can see only a limited area of its surroundings and it receives only the information related to that area, in each simulation cycle [2]. Therefore, as time goes by the confidence degree of the information pertaining to the remaining areas decreases. However, the confidence degree of the information related to the different parts of the field is not the same. Table 1 shows which parameters affect each of the three actions shoot, dribble, and pass.

TABLE I
EFFECT OF DIFFERENT PARAMETERS ON SHOOT (S), DRIBBLE (D) AND PASS (P) ACTIONS

Parameter	S	D	P
Ball distance from the center of the opponent's goal	y	n	n
Relative direction of the goalie and the ball motion	y	n	n
Dribbling length	n	y	n
Relative angle of player's body and the dribbling direction	n	y	n
Distance between the passer and the ball	n	n	y
Movement direction of the pass receiver	n	n	y
Sensitivity of the target area	y	y	y
The density of the opponents in the target area	y	y	y
Probability of the ball interception by the opponents	y	y	y
Updating degree of the received information as to target area	y	y	y

Every parameter may have different values for different situations. Table 2 shows the weights of parameters with respect to the density of the opponents in the target area and the length of the target area. These weights were experimentally obtained in our investigation based on many test runs.

TABLE II
WEIGHTS OF PARAMETERS WITH RESPECT TO THE DENSITY OF THE OPONENTS IN THE TARGET AREA AND THE LENGTH OF THE TARGET AREA

	5 units	10 units	More than 10 units
No player	0.15	0.19	0.25
1 player	0.10	0.14	0.19
2 players	-0.03	0.00	0.12
More than 2 players	-0.08	-0.03	0.05

In this section the evaluating measures for possible actions were studied. To be able of choosing the best action the soccer agent has to use an efficient real-time algorithm based on the mentioned measures.

III. EVALUATING METHODS OF CHOOSING THE BEST ACTION

To evaluate possible actions various methods have been suggested [5, 6, 7]. We use a specific weight for each parameter that affects an action, in our evaluation method. We obtained these weights, experimentally, through test runs and analysis of the outcomes. This analysis was aimed at pin pointing the weaknesses of our team and trying to adjust the weights to improve the ability of our 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. For example, in evaluation of the two actions A1 and A2, assuming A1 is better than A2, if evaluation module computes a higher priority for A2, the weights are adjusted by increasing the weights of those parameters which have more positive effect on A1 and decreasing those which have more positive effect on A2 (more negative effect on A1). This process is similar to the supervised learning [4], but it is performed offline. The weights will gradually adjust to a stable value.

There are two ways to evaluate each possible action, given the specific as well as the common measures:

A) 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. As fig. 3 shows, in this method, for each feasible action its priority is computed as the summation of all related measures. The action with the highest priority is then recognized.

```
max_priority = 0
selected_action = no_action
for each feasible action (FA) do
   priority = 0
   for each evaluation measure (EM) do
      priority +=  EM.weight
   end for
   if priority > max_priority then
      max_priority = priority
      selected_action = FA
   end if
end for
```

FIGURE 3
THE BEST ACTION SELECTION ALGORITHM IN ONE-PHASE METHOD

Many parameters affect the agent's next move. These parameters may be adjusted so that the decision-making process follows a reasonable sequence of actions for limited number of situations. Since there is unlimited number of difference situations, it is not possible to adjust the weights so that the process works best all the times. On the other hand, affecting parameters varies for different actions. In our experiments, we realized that if the decision-making process is broken into two phases the number of parameters to deal with is reduced and the process is better managed. This lesson is what we learned by monitoring and analysis of numerous test runs. As the next section describes, the set of all affecting parameters are broken into two subsets, those that are common to all actions form the first one and others form the second one. It is worth mentioning that the second subset is different for different actions.

B) To determine the best action from amongst all possible ones for a given situation, we first recognize the best of each action, 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 the same kind of process is followed. In the next phase, we select the best of bests, i.e., the system chooses the best action from amongst three best actions shoot, dribble, and pass. In this phase, common measures are used in order to evaluate the actions. Fig. 4 shows the two-step evaluation method in which in the first phase it finds the best possible shoot, pass and dribble using specific measures. In the second phase, it selects the actual action to take, using common measures. To determine the priority in the second step, the calculated priorities in the first step is not considered. This is the very method that is currently used in our team.

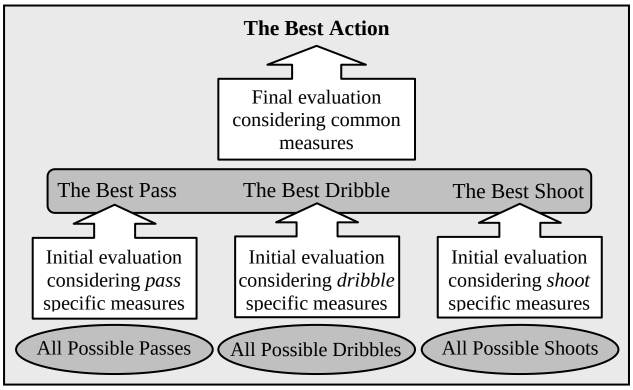
The Best Action
Final evaluation considering common measures
The Best Pass
The Best Dribble
The Best Shoot
Initial evaluation considering *pass* specific measures
Initial evaluation considering *dribble* specific measures
Initial evaluation considering *shoot* specific measures
All Possible Passes
All Possible Dribbles
All Possible Shoots

FIGURE 4
THE BEST ACTION SELECTION DIAGRAM CURRENTLY USED IN NEXUS TEAM (TWO-PHASE METHOD)

Method (A) consumes much more processing time than method (B). Therefore, it does not leave any time for the simulator to augment further precision and increase intelligence. The soccer agent might not be able to complete the evaluation process in one simulation cycle period, if the number of alternatives to be compared becomes large. Note that, the two mentioned methods were explained without considering the overall team's strategy and the coach's guidance. To evaluate the actions, while considering team's strategy and the coach's guidance, other parameters have to be added to the list of parameter affecting the evaluation process.

IV. Comparison of two methods

A team's success is directly influenced by each agent's actions. To calculate an agent's competence, we should consider a measure that commensurate with the agent’s pursuing goal [4]. Each soccer agent attempts to change the game towards its own advantage so that it not only minimizes missing but also maximizes scoring through imposing maximum pressure on the opponent's goal. In other words, the soccer agent's entire endeavor is winning the game. Therefore, to determine a team's efficiency, which in fact demonstrates the degree of the soccer agent's effectiveness, the game result or the scored goals difference can be the preferred approach. To compare the two mentioned methods, two teams were set up accordingly. To diminish the effect of accidental results, these two teams were made to contest five times. As it can be seen in table 3, the results remarkably confirm the second method’s superiority.

TABLE III
The result of competition between two Nexus teams

Games	First	Second	Third	Fourth	Fifth	Average
Nexus-1 and Nexus-2 game result	1-1	0-2	0-1	1-2	0-1	0.4-1.4

A soccer simulation environment is a multi-agent system in which the agents in each team collaborate closely against the opponent agents. Therefore, the acceptability and dependability of each action taken by each team can display the extent of the relative competence of the corresponding method that is used. The results in table 4 are obtained by applying *"SoccerDoctor"* software [8], which is one of the best soccer simulation contest analyzers, for the five matches played between the two Nexus teams using the mentioned methods.

TABLE IV
Average action acceptability of the two teams in five mathes

	pass	dribble	shoot	Ball possession
Nexus with first method	54%	77%	17%	38%
Nexus	86%	93%	33%	62%

V. Conclusion

Common measures are essential but not sufficient to evaluate the various actions. To evaluate them more precisely, specific measures are needed. These measures filter out the same type of actions by choosing one best action of each type. In this research, we concentrated on three action types shoot, dribble, and pass. We devised a two-phase action selection method and compared its efficiency with the one-phase evaluation method. Two Nexus teams were set up. The first one used Method (A) to select agent’s next move and the second one used Method (B) for the same reason. The two Nexus teams competed many times and the results were summarized. The outcome clearly shows the superiority of the two-phase selection method.

References

[1] Yao Jinyi, Chen Jiang, (2001). Architecture of TsinghuAeolus, TsinghuAeulos Team Description Paper. available: http://portal.acm.org.

[2] Ehsan Foroughi and Fredrik Heintz, (2001, June 11). Robocup Soccer Server manual 7.07, RoboCup Federation. available: http://sserver.sourceforge.net.

[3] Remco de Boer and Jelle Kok, (2002). The Incremental Development of a Synthetic Multi-Agent System: The UvA Trilearn 2001 Robotic Soccer Simulation Team, MS Thesis. Faculty of Science, University of Amsterdam. available: http://carol.wins.uva.nl/~jellekok/publications/2002/mastersthesis.pdf

[4] Stuart J. Russell and Peter nerving, *Artificial Intelligence A modern Approach*, Prentice Hall, 1995.

[5] Luís Paulo Reis and Nuno Lau, (2001). FC Portugal Team Description: RoboCup 2000 Simulation League Champion, FC Portugal 2000 Team Description Paper. available: http://www.ieeta.pt/robocup/documents/FCPortugalChampion.ps.zip

[6] Peter Stone and David McAllester, “An Architecture for Action Selection in Robotic Soccer,” pp. 1-8. AT&T Labs - Research. Florham Park, NJ, 2001.

[7] Peter Stone, (1998). Layered Learning in Multiagent Systems: A Winning Approach to Robotic Soccer, PhD Thesis, Computer Science Department, Carnegie Mellon University, Pittsburgh, Available as technical report. Available: http://www.cs.utexas.edu/~pstone/book/

[8] SoccerDoctor v1.0, (2002). Available: http://wrighteagle.org/sim/tools/SoccerDoctor.exe