Robocup 2013- Soccer Simulation League
2D Soccer Simulation
<Riton – Iran>

Masoud Alavi[1], Mohammad Falaki Tarazkouhi[2], Arezoo Azaran, Alieh fazeli, Parichehr Rezazadeh Bavil Soflae[1], Mohsen Abedini[3]

[1]Rajaee Teacher Training University, [2]Iran University of Science and Technology
[3]Islamic Azad University of Mahdishahr

{masoud.alavi, falaki.t.mohammad, riton.2d}@gmail.com

Abstract. This paper is a quick overview of algorithms which have been applied by Riton soccer simulation team in skills and decision making layers. We also intend to improve Riton by means of restoring to previous experiences which has been gained by Riton 2D simulation team in Robocup simulation league. In this competition we have tried to use new algorithms for marking and with the ball skill and using learning algorithm for shoot skill. We have also tried to improve our pass system by new formulas.

1 Introduction

Riton has started again its professional activities since 2009 on the basis of what it has done during student's 2d simulation league. This team was ranked among the top 7 teams in AUTCup 2010 and 4th in PNUCup 2010 and PNUCup 2011. It was among the 5 finalist teams in Iran open 2011 and also it has got the 2nd place in Iran Open 2012 and 4th place in Dutch Open 2012. We also qualified for World Competitions in Mexico 2012. We started our project by applying logic scoring. Our primary focus is on new ideas to improve processing and to gain success by means of using new algorithms in the field of artificial intelligence. For this goal we are conducting a research on probability functions to create acknowledgment base from the opponent with no need to have heavy deductions on their base, so we can have a faster, more precise and more successful decision making process. We will explain in this paper our Decision Mechanism, Skills Architecture, Danger Rate Function, and Future Research Programs.

2 Skills Architecture

This architecture consists of actions with enable players to act. In general, these actions are performed by using the saved data in World Model, and after processing the data a comprehensive instruction is made. The action selection is done by prior architecture [2]. We described our main structure of each skill in our last year paper, so we describe our new ideas in this paper.

2.1 Direct Pass

Our pass system is almost the same as the one we applied last year; and we improved it by changing its scoring system as below. First we describe the outline of our pass system; in this system, both sender and receiver should have a correct perception over the field, so the probability of success is high. In passing method, we check several cases and list some of teammates with the least acceptable situations. We choose the best player with exercising our scoring system. Some of the conditions that we consider for scoring are:

1. Least and most distance
2. The non-existence of opponent's player within the 1 distance from the line
3. Last received data from teammate in less than 3 cycles
4. The non-existence of teammate in offside

Scoring System: We use a formula for scoring this year which figure1 illustrates its graph. And the formula is:

$$y = -0.02(x^2) + 0.63x$$

By using this kind of equation we can score more precisely in comparison with considering the distance only. The best distance got the highest score and this score will be multiply the other factor as up than the best point will be chosen.

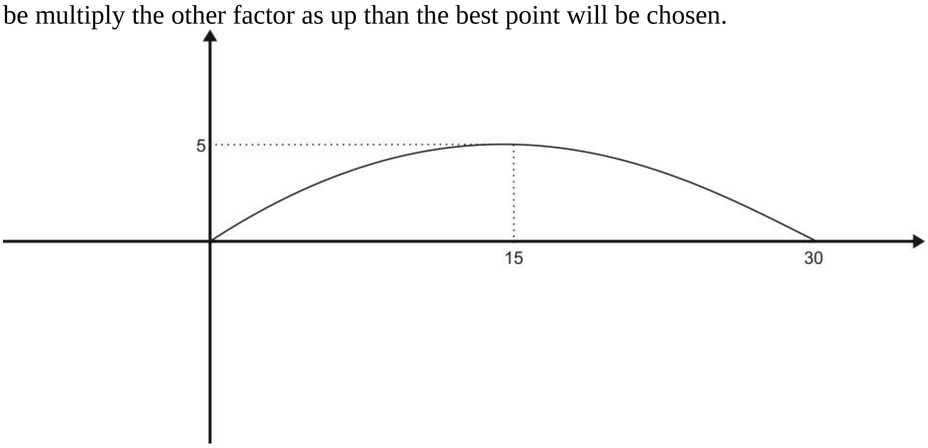


Fig1. Scoring system graph

2.2 With Ball Skill

In our dribble system we choose a fuzzy natural [3][4] algorithm, at first we draw four circles in all our sides, then we choose the best circle by considering these parameters:

1. Circle which leads our movements near to goal has the highest priority, then the circle which takes the ball to the middle after that the circle toward the border lines and the circle which is located behind us has the lowest priority and the least score.
2. The number of opponents in the circles
3. Distance between us and nearest opponent

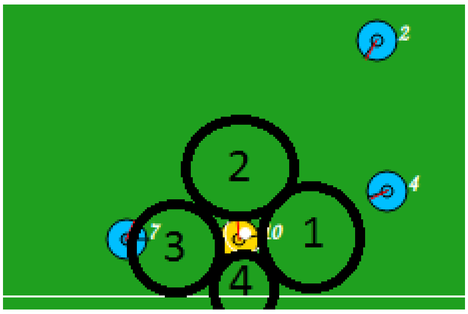


Fig2. The drawn circles and their priorities

After selecting the best area to dribble, we create 7 lines from our position to different degrees (0, 30, 60, 75) then check those parameters on this lines and finally the best way for moving will be chosen.
If in one of the first 3 circles we don't find any opponent agent, we draw 2 more circles frequently and if we can find a suitable situation we will kick the ball more far in compare to our simple dribble and do a fast dribble.

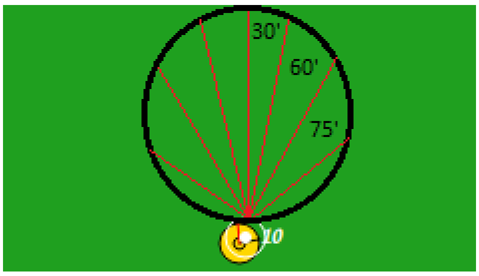
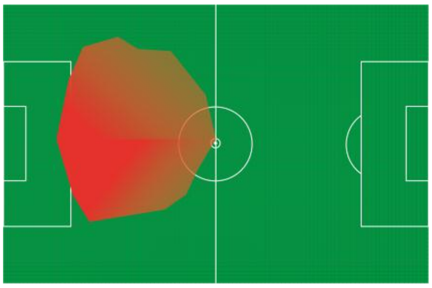


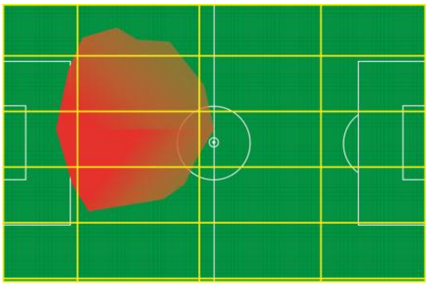
Fig3. The different lines which is drawn

2.3 Opponent selection for marking

We can be more successful in defending if each agent has the responsibility to mark one of the opponent's players. It is essential that the agent choose the nearest opponent to his strategic area and also it should be checked that two agents do not choose the same opponent. Since each team has its own formation and it differs from other teams, when the match kicked off till 1500th cycle, each agent registers his position information and opponents' position information, so according to obtained information the agent can recognize the nearest opponent to his strategic area. We use a two-dimensional 68*105 matrix in order to keep positional information; all elements of this matrix are initialized with 0 value. For each cycle, x and y values of each opponent's position in certain area of agent are obtained and after removing the decimal part and converting to the indexes of the matrix of that opponent, value of that element will have one-unit increase. Certain area of agent has two conditions, the distance between the agent and each opponent should be less than 15 and also the information we have should belong to three past cycles or less, so we will not register the information of those opponents that are not in strategic area of the agent. The information of the agent's position is registered in the same way.
After 1500 cycles movement area of each opponent can be determined more precisely. In order to do this we generate some rectangular matrixes with the dimensions 10*25 on each opponent's matrix and after calculating the sum of the values of the elements of each rectangular matrix, the matrix which belongs to maximum value is the area of the opponent and should be kept. This can be done for opponents that their position information is registered for more than 750 times. Determining the agent's movement area is the same way as determining it for opponents.



1. The movement area which is stored in matrix. Darker colored area shows that the opponent's movements take place in that area more.



2. The rectangle areas generated to find more precise movement area.

Fig4. Positional information stored in matrix

Finally, the rectangular area of agent is compared with the areas of the opponents; opponent whose movement area has the most intersection with the agent's movement area is the one that agent must guard.

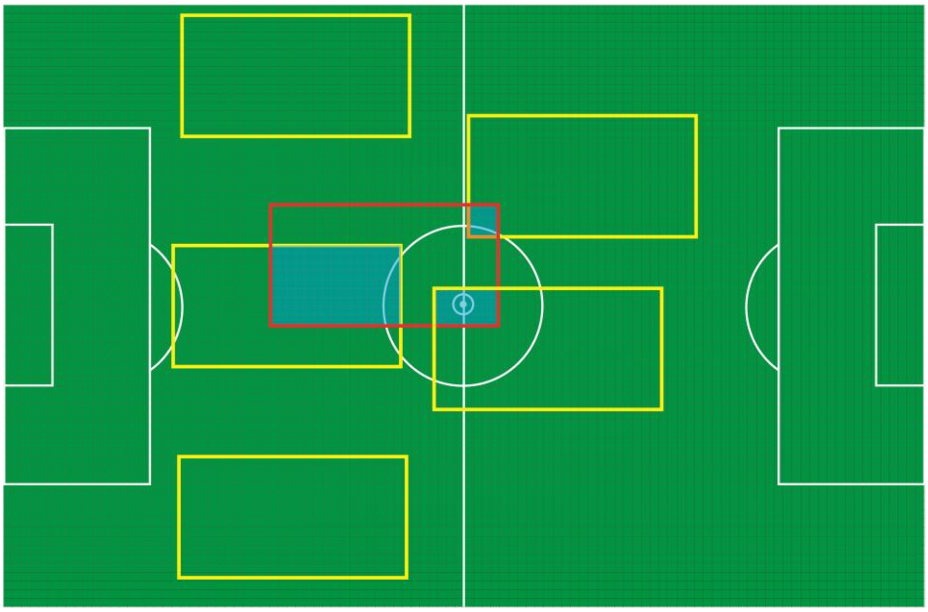


Fig5. The rectangles of opponents in comparison with our agent's rectangle

3 Learning Function

Since all teams are not at the same level, if we use static formulas in skills layer, we must consider opponent team at the highest level. But when we face a team at lower level we can have more behaviors, actions and better performance with higher reliability if we consider its level. Therefore we decide to apply an algorithm which its decision mechanism works according to the approach of the opponent team, so opponent's pervious actions and behaviors should be perused and next decisions should be taken according to obtained information. For example, after a number of passes according to our success rate in pass action, we can estimate opponent's skill in intercepting the ball and then use it as an effective parameter in our next decisions.

In fact, our main idea is to have learning with the use of the probability distributions. Because there are many parameters should be considered in pass skill, we decide to implement this algorithm for shoot skill, in order to find the weak points of the opponent's goalie and choose the best condition to shoot as the first step and then generalize it to other skills.

This algorithm works according to the past behaviors and calculates the probability of success for what we want to do by considering the effective parameters. Effective parameters in shoot skill are:

1. The angle between the goalie's body and the considered point for shooting
2. The distance between the ball and the gate
3. The distance between goalie and the hypothetical line between the ball and considered point for shooting

Then we combine these three parameters gain a particular value. Next, the best case and the worst case are calculated in order to deter1mine the domain of obtained values on the vector. After that, discretization should be done by dividing the vector into 10 equal parts.

To acquire some primary information, the player shoots the gate with the goalie under different random conditions and according to effective parameters, it is determined that each condition relates to which part of the ten parts of the vector.

During the match whenever the player has the opportunity to shoot, some points should be specified on the opponent's gate which none of the opponents can intercept the ball till it reaches to these points. According to effective parameters it should be determined that current case refers to which part on the vector and then with the use of the information saved on that part and the binomial distribution [5][6] which is described below in detail, the probability of scoring a goal is calculated. Eventually, the case with the highest probability is chosen.

The binomial distribution

A situation in which an experiment is repeated a fixed number of times can be modeled, under certain assumptions, by the binomial distribution. Within each trial we focus on a particular outcome. If the outcome occurs we label this as a success. The binominal distribution allows us to calculate the probability of observing a certain number of successes in a given number of trials.

Formula: this formula gives the probability of exactly k success in n trial while p is the probability of success (goal) for n trials and q is the probability of failure (not goal) for n trial.

$$p + q = 1 \qquad\qquad p\{y = k\} = \binom{n}{k} p^k q^{n-k}$$

4 Future works

Our future works focus on positioning algorithm. We are trying to increase the performance of our current algorithm. We got the idea from an article [7]. Our algorithm works in 2 separate layers.

In first layer, ball position, teammate who controls the ball and some other information is obtained and also prediction which is essential for positioning is done in this layer, so the movements of the players and the ball that could be done in some fixed time are extrapolated.

In second layer, according to prediction information feasible area which is the area that the player can be positioned in is generated and then the best point for the player that positioning is applying on him will be found.

Here we shown a simple situation for this algorithm , The responsible are is the area where the teammate controlling the ball can pass and the reference position is the best point to pass. The reachable area is a circle around the player that determines the area that he can reach in some fixed specified time. The feasible area is the intersection of responsible area and reachable area where the player can be positioned in.

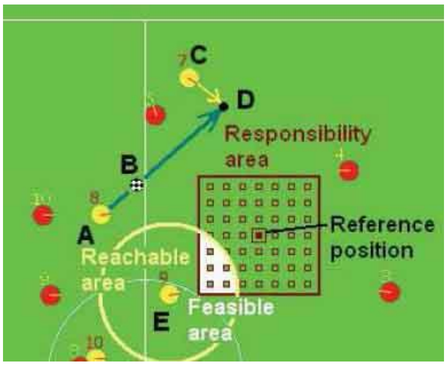


Fig6. Positioning system

5 References

1. Riedmiller, M.Merke, A.Meier, D.Hoffmann, A.Sinner, A.Thate, O.Kill, C.Ehrmann, R.Karlsruhe :brainstormers – a reinforcement learning way to robotic soccer. In Jennings, A.Stone, P.eds : RoboCup-200: Robot Soccer World Cup IV,LNCS. Springer (200)
2. Dr. Mir Hosein Dezfoulian, Nima Kaviani, Mostafa Refaee. Robosina . (2005)
3. Fuzzy sets and fuzzy logic , Dr.koursh Kiani , Amir Kabir University of Technology
4. Fuzzy Logic , Prof.Lotfi Zade
5. Probability, random variables and stochastic process , A.Papoulis and S.U.Pilay, 3rd edition
6. Introduction to Probability Models, Sheldon M. Ross, Tenth Edition
7. V adim Kyrylov and Serguei Razykov (2010). Optimal Offensive Player Positioning in the Simulated Robotic Soccer, Robot Soccer, Vladan Papi (Ed.), ISBN: 978-953-307-036-0