GDUT_TiJi 2013 2D Soccer Simulation Team Description

Haohui Huang, Wanlu Yang, Zijie Mei and Wei Chen

School of Automation, Guangdong University of Technology
Guangzhou, Guangdong Province, P.R.China, 510006
yelwong@gmail.com

Abstract

GDUT_TiJi is a 2D soccer simulation team which has been participating in RoboCup since 2006. This paper briefly describes the background of GDUT_TiJi 2013 and recent innovations since RoboCup2012 in Mexico which will mainly focus on optimizing online planning during offensive situation and developing a new decision-making module called anti-calculation which make it possible that agents can forecast the opponents actions whilst defending so that agents can take proper actions in advance to stop opponents offense. Although this module is still at the preliminary stage, we are looking forward to improving our defensive skills.

1 Introduction

GDUT_TiJi, a 2D soccer simulation team from Guangdong University of Technology, was founded in 2006 and has been taking part in annual competitions of RoboCup China Open since 2007. In the preliminary stage, we used UvA_TriLearn as base code but we had not got the ideal achievements. Later on we changed our base code and began working on agent2d. In the new team base, we had made a deep research to the structure of Agent2D-3.1.0 [4] and we had got the first prize in RoboCup China Open 2011, second prize in RoboCup China Open 2012 and the 5th place in RoboCup 2012 in Mexico.

RoboCup simulated soccer has been proved to be an excellent domain for researchers to test their ideas in artificial intelligence [5]. In spite of rigid, our works in the past was mostly focused on the position and movement for both offensive and defensive though some geometric algorithms. In GDUT_TiJi2013, with the purpose of making our agents more flexible and dynamic, more effort have been made on online planning

This paper is organized as follows. Section 2 introduces the Anti-Calculation Module designed for our team. Section 3 briefly presents the changes of the Field-Evaluator since last games. Section4 describes some defensive skills modified in GDUT_TiJi2013. Finally, Section 5 concludes this paper and proposes the future work plan of GDUT_TiJi.

2 Anti-Calculation Module

This module is designed to calculate the opponents' actions during their offense. And that's the reason why we call it Anti-Calculation. First of all, we would like to introduce the structure of this module and how it works.

As shown on the figure below, agents who want to predict the actions that the ball holder may take, will call the function – Anti_Offensive to activate the module. And then Anti_Offensive will

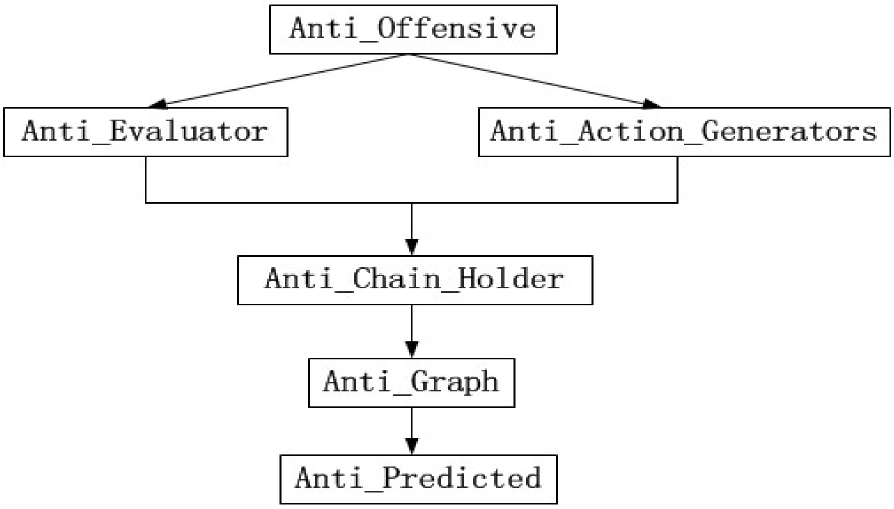

Figure 1: Structure of Anti-Calculation

put the Anti_Evaluator and Anti_Action_Generators together into Anti_Chain_Holder to generate the action chain. Anti_Evaluator is used to evaluate the actions generated by Anti_Action_Geneartors according to state of the world. On the other hand, Anti_Action_Generators contains the actions of the ball owner such as pass, dribble and shoot. Finally Anti_Graph, an online search algorithm, would sort the action chain refer to each action's evaluated value and then return the action with the best evaluation.

At this moment, we just focus on the pass action of opponents. With the aid of Anti-Calculation module, we will be able to intercept it more accurately while the opponent passes the ball. So it is an efficient way to block the ball owner undoubtedly and also it can improve the performance of defense a lot.

Furthermore, we will show the experiment of the Anti-Calculation module which is play against with the latest agent2d at the following figures.

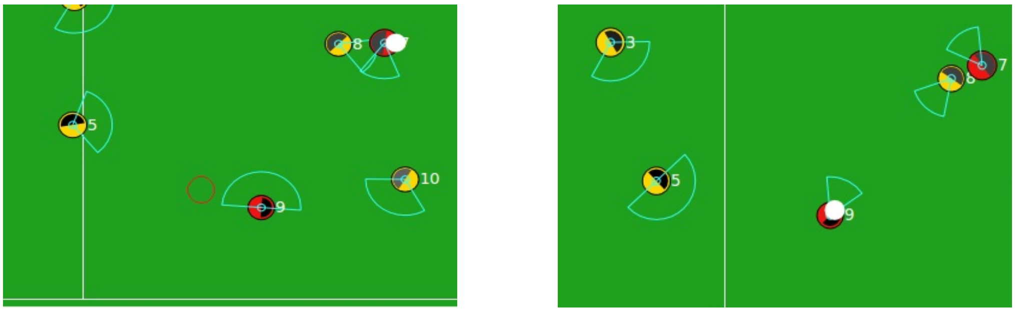

Figure 2: Passing prediction against with Agent2d

We can see from the left image in Figure 2 that the opponent, player NO.7 from agent2d, held the ball at cycle 1524. And player No.5, from our team, called the Anti-Calculation module and the result that the function returned was the ball holder would pass the ball at the next cycle and the target point was [9.01795, 25.9907], the red circle, which meant that the ball owner would pass the ball to his teammate, player NO.9 from agent2d. As expected, player NO.9 from agent2d had received the ball at cycle 1534 and the receive point was [8.02962, 26.5551] as shown in the right image.

Since a good overall evaluation function has not yet been constructed, we have noticed that this algorithm will have a significant decrease in accuracy while the team plays against with other team. Taking this reason into account, we have not put it into practice. But anyway, we have finished the structure of this algorithm and we believe it will make our defensive system more perfect in the near future.

3 State Evaluation

It is well-known that the evaluation function in agent2D, just considering its X-coordinate, is quite simple but stable. Through the efforts of our teammates last year, we have a profound understanding of chain-action in agent2d. In GDUT_TiJi2013, we had re-structured the field evaluator by taking full advantage of the y axis of ball position which somehow reduced the rigid dribble and increase the effective number of passing..

In GDUT_TiJi2013, inspired by this idea from USTC [2], we have rewritten with a specific evaluation function which evaluates the value of the generated Action-State Pair instances by the following equation that can adjust the returned evaluated values and enable the action that the agent takes be more reasonable:

$$\begin{aligned} tmp = & \sqrt{(Ball\_x + Pitch\_halflenth)^2 + Ball\_y^2} \\ & -\sqrt{(Ball\_x - Pitch\_halflenth))^2 + Ball\_y^2} \end{aligned} \tag{1}$$

$$Eval = a \times \frac{tmp}{Pitch\_Lenth + b} - a \tag{2}$$

The simulation for the evaluated values are shown in Figure 3, which depicts two characteristics of the ball's position: its X-coordinate (the larger the better) and the distance from it to the opponent's goal (the smaller the better). In order to prevent the agent just dribble along the x-coordinate on the sideline, some hand-code according to the prior knowledge had been added based on the modified evaluation function model. Howeve, since the evaluation function we modified cannot generate smoothly returned value, the agent always holds the ball and stays in place when the evaluated value mutated

In action chain, evaluated value mainly determines the action's generation which contains five ActionGenerators: shoot, pass, cross, dribble and selfpass. Therefore it can be seen that the evaluated values is the core of our online planning.

4 Defensive Skills Improvement

It is obvious that before any modification, the defensive player in agent2d does not intercept when the opponent dribbles breakthrough our defensive line until it reaches the penalty area. Based on the hand code defensive policies which were used in GDUT_TiJi2012, we optimize the strategies and adopt some new approaches to finding the efficient solution for assignment of opponent's attackers to our defenders.

4.1 Block

The most challenging part in this improvement is determined by two aspects. On one side, firstly, we should realize which defender of us should be assigned to block the opponent's ball

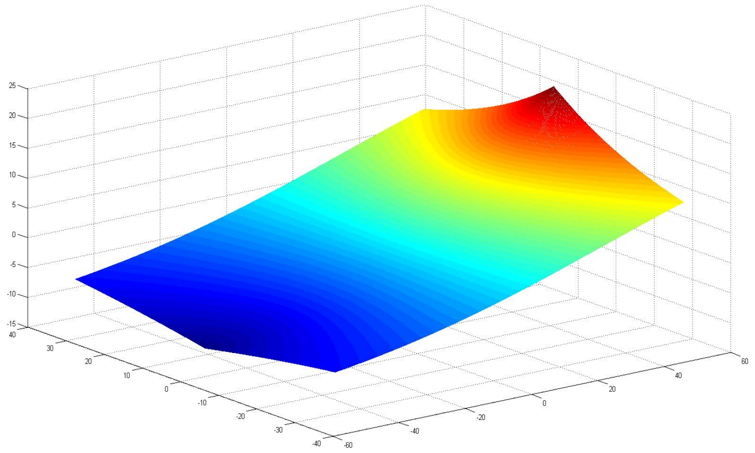

Figure 3: Simulation of the Evaluation Function

owner and calculate the point where the agent should move so as to tackle or intercept the ball more accurately and actively according the distance to the goal and opponents direction.

After last competition, we have re-analyzed the structure of agent2d base and on the basis of the teams' performance which was also used the agent2d base in RoboCup 2012, we found that the agents always tended to dribble towards the bottom line when they were in offensive situation. In GDUT_TiJi2013, therefore, we have strengthened the defensive and intercept ability of the side back players; especially we put more attention to the interception when the opponent has done a through pass. In this aspect, we considered the following parameter to improve the function of the original interception and propose a better interception strategy: the distance between the ball position and the defender position — S_1; the distance between the ball position and the goal — S_2; the velocity of the ball — $ball\_vel$; the running direction of the ball — $ball\_dir$; the weight for the intercept position according to the ball velocity and direction — k. Then the intercept point calculation function is given as follows:

$$Intercept\_pos = ball\_pos + Pos(\frac{S_1}{S_2} \times F(ball\_vel, ball\_dir)) \tag{3}$$

Actually, the side back players, instead of running back to the basic point, can provide more protection in case the opponent passes the ball to the center opponent or breaks through. Although it is simple, it works well indeed.

4.2 Mark

On the other hand, furthermore, we devote to desire a practical one-by-one marking function after GDUT_TiJi2012 which is done in order to keep the opponents away from ball or disable them get the ball easily. Nevertheless, the defender may miss the opponents that should be marked since we just consider two opponents nearest from the defender. For this reason, we have defined the defensive half player, number 6 in GDUT_TiJi2013, as an important offensive agent to undertake more defensive mission. To be specific, our one-by-one marking function is constructed by the following flow chart:

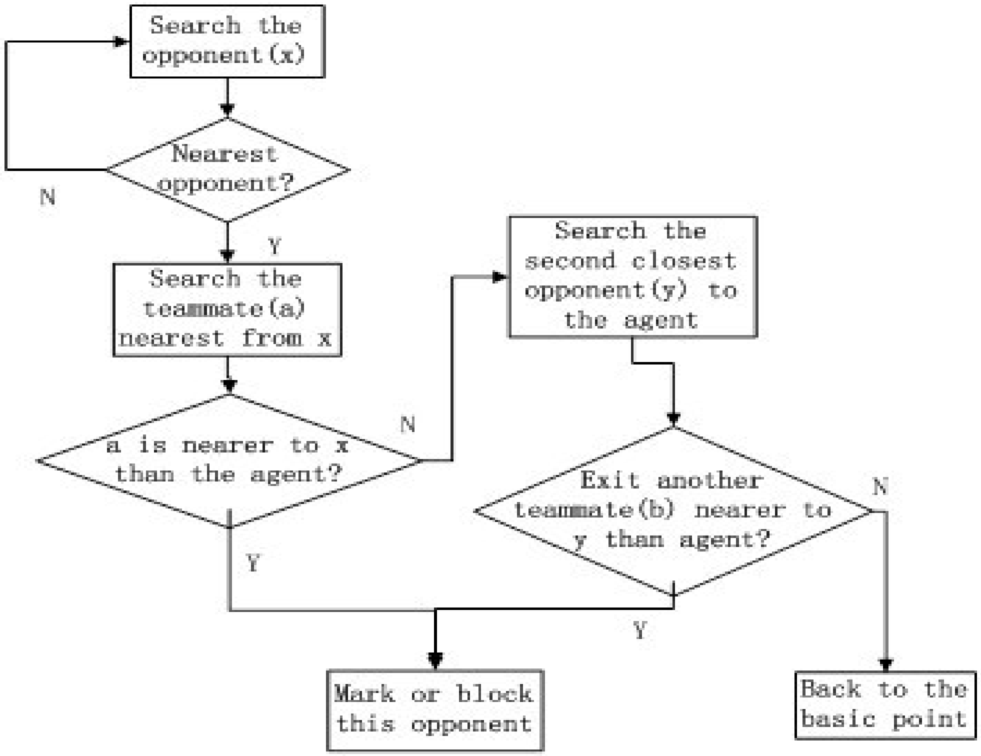
Search the opponent(x)
Nearest opponent?
N
Y
Search the teammate(a) nearest from x
a is nearer to x than the agent?
N
Y
Search the second closest opponent(y) to the agent
Exit another teammate(b) nearer to y than agent?
N
Y
Mark or block this opponent
Back to the basic point

Figure 4: the Flow Chart of the Mark Tactics

5 Experimental Results

In order to evaluate the effectiveness of GDUT_TiJI2013, vast amount of tests were performed against two different teams and each game had run over 50 games to reduce the contingency which definitely exist in 2D simulation. Also we will compare and analyze with GDUT_TiJi2012 both in offensive and defensive.

From Figure 5, it is shown obviously that GDUT_TiJi2013 has maximized the winning rate compared with GDUT_TiJi2012.

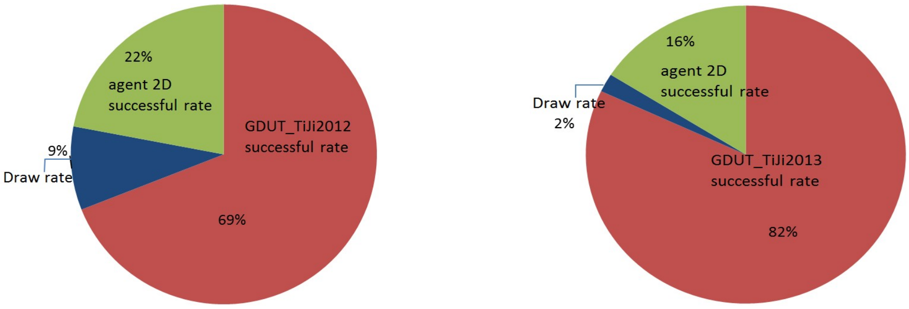

Figure 5: Statistic of GDUT_TiJi && Agent2d

Against with a stronger team as Marlik2012 as shown in Figure 6, however, the percentage of winning had not been improved apparently compared with GDUT_TiJi2012 but seem to be more stable. In addition, table 1 depicts a statistical analysis, gathering from the games of GDUT_TiJi2013 && Marlik2012, about the defensive ability of side back players to test the

Games Numbers	Defendive Numbers	S(Defend Success)	F(Defend Fail)
20	192	126(65.4%)	66(34.6%)

Table 1: Statistic of the Side Back Defensive

utility of interception function as Equation (3). Here, we defined S if the agent could stop the opponent who tried to break through along the bottom line or pass the ball to the center attacker, otherwise it would be F.

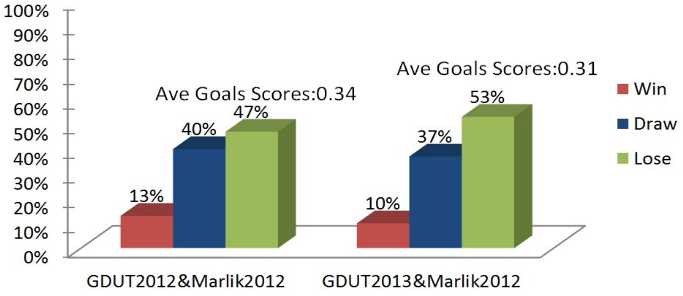

Figure 6: Statistic of GDUT_TiJi && Marlik2013

6 Conclusions and Future Works

In this paper, we have briefly described our latest works since RoboCup2012 mainly focusing on the defensive situation and online-planning in the field of MAS. Although it had enhanced the stability of each agent, there will be more problems and bugs such as the vision information, stamina etc. that need to be solved or fixed in the future. In addition, for the sake of making up for the ignorance of the Y-coordinate and the discontinuous of the return evaluation values, we will perfect the evaluation function which is used both in chain-action and anti-calculation in GDUT_TiJi.

Despite these efforts, in order to apply the on-line planning in the defensive part, we will complete our Anti-Calculation module for our team which is used to predict the opponents action in the next cycle. Even if there are many problems in it, we hope not only can it work on our team in the near future, but also contribute to the defensive strategy. Moreover, we will also pay more attention to improve our cooperation to let the agents act more like human.

References

[1] Ke Shi: Research on Agent Decision Problems Based on the Theory of MDP, Masters thesis, University of Science and Technology of China, 2010.

[2] Zhenyu Li: Research on Multi-Agent System Decision Problems and Application in RoboCup, Masters thesis, University of Science and Technology of China, 2005.

[3] Akiyama, H., Shimora, H., Nakashima, T., Narimoto, Y., Yamashita, K.: HELIOS 2D simulation team description 2012. 2012 RoboCup World, Mexico. 2012.06.

[4] Jing Guo., Haohui Huang., Wei Chen.: GDUT_TiJI 2D soccer simulator team description 2012. 2012 RoboCup World, Mexico. 2012.06.

[5] Kalyanakrishnan, S., Liu, Y., Stone, P.: Half field offense in RoboCup Soccer: A multiagent reinforcement learning case study. RoboCup 2006: Robot Soccer World Cup X pp. 72-85(2007).

[6] Luis Paulo Reis, Nuno Lau and Eugnio C. Oliveira.: Situation Based Strategic Positioning for Coordinating a Team of Homogeneous Agents in Balancing Reactivity and Social Deliberation in Multi-Agent system From RoboCup to Real-World Applications, Springer LNAI, Vol. 2103, pp. 175-197, Berlin, 2001.