MT2018: Team Description Paper

Zheng Yang, Ziqiang Liu, Xiaorui Wang, Ning Dong, Xiangben Hu, JingLi
Shengbing Chen, Gang Lv

Department of computer science and technology,
Hefei University,
Anhui Province, P.R.China, 230601
hfuumt@163.com

Abstract. MT2018 is a simulation soccer team for the RoboCup soccer 2D simulation league which is consisted of the students who are coming from Hefei University and all of them are with strong robot enthusiasm. Since 2012, the MT2018 team has participated in RoboCup China open competition every year, the RoboCup competition since 2015, and has achieved many good results. This paper briefly describes the background of MT2018 and the main works of our team since RoboCup 2017 competitions. Through these works we have greatly improved the overall capacity of our team.

Keywords: RoboCup, Move strategy, Dribble strategy, Log mining

1. Introduction

MT was founded in 2012, by the Hefei University Department of Computer Science and Technology Innovation Laboratory of a group of robot-loving soccer students. During the five years, we have taken an active part in annual competitions of RoboCup. And there are some achievements, in 2012 and 2013 we won the second prize, in 2014 we won the grand prize, in Portuguese Open 2016 competition the champion, we sixth in RoboCup WorldCup 2017 in Soccer 2D Simulation competition. By the communication with other teams, we found some deficiencies, and then proposed improvement measures. We hope to verify the effect of improved code in this year's competition, and improve the team's level gradually.
For this competition, We started the strategy improvement in August, the attack and defense ability of the team has been greatly improved on the basis of last year. We hope we can get remarkable achievements, make more friends, and learn more things in this year.

2. The underlying of the MT2018

We use agent2d-3.1.1 as the underlying code. Using librcsc as the underlying database, the team’s underlying is action-chained style. We have been the repaired and improved on the basic of MT2017, and we added a lot of new documents to the project.

3. Keep dribble strategy

Owing to several tests our team ball keeper is easy to slide tackle and other reasons lose the ball.

To avoid such a situation as much as possible and improve the Usage Percentage, we do some changes on the player's dribble.

The basic technology of a single Agent is a very important part of the RoboCup competition. Dribble is also the basic technology of agent. It means that the players take the ball fast forward, and also keep control of the ball at the same time. The

player is properly adjusted after getting the control of the ball, kick the ball in front of its body, then intercepte the ball as soon as possible. In fact, accomplish to dribble is to repeat the process. In a real competition, the strikers in the two wings are able to take the ball at high speed, get rid of the opposing defender, thus break through the whole line of defense from the side. Therefore, It's a very effective technology for striker.

The dribbling process can be divided into five steps:

(1)Determine the goal of the ball; (2)Adjust own position, body orientation and the direction of the neck in order to catch the ball at high-point; (3)Kick the ball in a small force to a intermediate transit point near the body and turn around to the target point at the same time; (4)Kick the ball out of the controllable range with the proper force; (5)Intercepte the ball at the fastest speed and prepare for the next dribble action.

In the past, our team's strikers were often too weak to take the ball, so that in the offensive was intercepted or tackled, lead to losing the ball, offensive failure. Therefore, we have done a lot of attempts to improve the player's dribbling ability. We are based on a lot of dribble data in order to get the appropriate ball action reward value. When meeting defensive players, by predicting the opponent's behavior and the score of the reward value to determine the action parameters of dribble. Through a large number of training so that players try to cross the opposing player's defense, improve the success rate of a single agent dribble.

4. Move strategy

Some of the previous move strategies in a number of offensive and defensive combats have shown some weaknesses gradually, for example, the use of formation, some geometric algorithms, using Voronoi Diagram etc. they can quickly give each agent to find some reasonable target point positioning. However, because the RoboCup competition has the characteristics of uncertainty and dynamics, most of the time, the target point selected by the above method is not the best strategy, they always miss some of the coordinates of the assessment. Each player runs on the team cooperation and reasonable positioning is highly needed , we have been groping and trying many times. In MT2018, we use as many search and evaluation methods as possible.

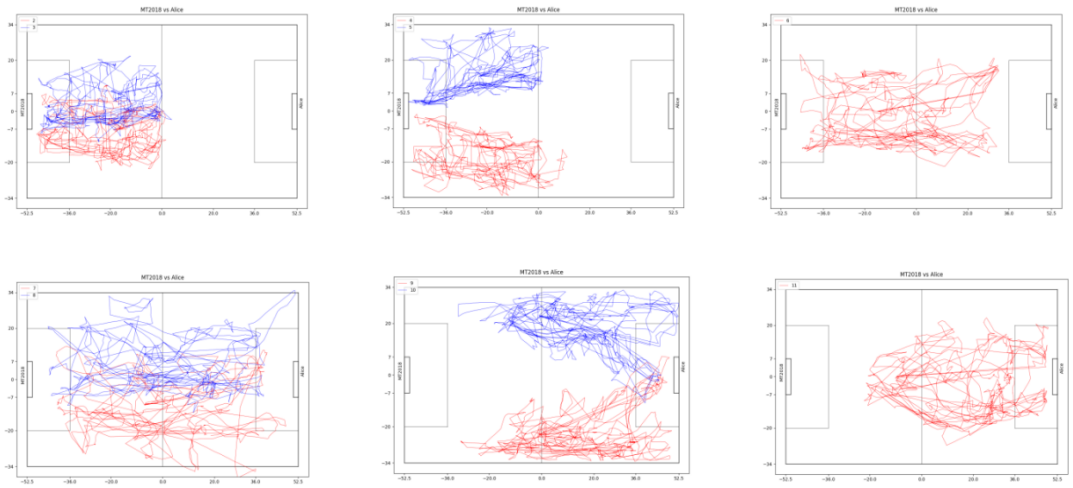

Fig.1. Player move track

We choose some points as the center, such as the formation point of each player, the coordinate point of the ball, each player's coordinates, some points of Voronoi diagram and so on, generating a target point with a certain density and ergodic each

target point for evaluation and sorting, so we can select the optimal target point to run. Furthermore, we use Neural network algorithm to train parameters in evaluation and the optimal target point to the specific circumstances of the intercept, block, mark and movement, improve the team's offensive and defensive ability.

5. Game log mining

In the past, when we developed the simulation 2D soccer project, the strategy model could only be designed through human experience, the problems could be found in the team by watching the test competition or the game playback, and then, the team code could be modified for these problems. This kind of development idea can solve the problem, but the cycle is longer, and the strategy design is short of reference and waste lots of time.

In the process of RoboCup simulation of 2D competition, a log file will be produced to record the detailed data of the game;using simulated 2D log files as data source, using data mining and machine learning algorithms and a large number of data sources. Finally, the model is applied to the development of the simulation 2D team strategy, enhance the team's offensive and defensive abilities.

The rcl file records the commands sent by each cycle player to the Server (competition server) and the execution parameters of the command, as follows:

Cycle, StopTime Recv Team_num (Action + Parameters)

```
0,16399 (referee kick_off_l)
1,0     Recv HELIOS_base_3: (turn 116.858)(turn_neck 43)(change_view normal)(attentionto our 2)
1,0     Recv MT2018_11: (turn -159)(turn_neck -86)(change_view narrow)(attentionto our 2)
1,0     Recv MT2018_3: (turn 169.935)(turn_neck -26)(change_view normal)(attentionto our 2)
1,0     Recv HELIOS_base_5: (turn -137.207)(turn_neck 41)(attentionto our 2)
1,0     Recv HELIOS_base_9: (turn 169.674)(turn_neck 90)(attentionto our 11)
1,0     Recv HELIOS_base_8: (turn -157.647)(turn_neck -90)(attentionto our 2)
1,0     Recv MT2018_1: (turn -108.077)(turn_neck 80)(attentionto our 2)
1,0     Recv HELIOS_base_4: (dash 61.414)(turn_neck -48)(attentionto our 2)
1,0     Recv HELIOS_base_7: (turn -34.749)(turn_neck 90)(attentionto our 2)
1,0     Recv HELIOS_base_11: (turn -68.825)(turn_neck 39)(attentionto our 7)
1,0     Recv MT2018_6: (turn 166.76)(turn_neck -90)(attentionto our 11)
1,0     Recv HELIOS_base_2: (turn 44.774)(turn_neck -70)(attentionto off)(say "R)xb9H+I h")
1,0     Recv MT2018_10: (turn 65.529)(turn_neck -66)(attentionto our 2)
1,0     Recv MT2018_9: (turn -27.512)(turn_neck 29)(attentionto our 2)
1,0     Recv MT2018_7: (turn 57.356)(turn_neck -56)(attentionto our 2)
1,0     Recv HELIOS_base_6: (turn 166.496)(turn_neck -79)(attentionto our 2)
1,0     Recv MT2018_8: (turn 23.485)(turn_neck -21)(attentionto our 2)
1,0     Recv MT2018_2: (turn 126.081)(turn_neck -90)(attentionto off)(say "P4yDs_")
1,0     Recv HELIOS_base_10: (turn -163.589)(turn_neck -90)(attentionto our 11)
1,0     Recv HELIOS_base_1: (dash 45)(turn_neck 90)(attentionto our 2)
1,0     Recv MT2018_5: (turn 83.656)(turn_neck -90)(attentionto our 2)
1,0     Recv MT2018_4: (dash 100)(turn_neck -11)(attentionto our 2)
2,0     Recv HELIOS_base_4: (dash 61.257)(turn_neck 138)(change_view normal)(attentionto our 9)
2,0     Recv MT2018_2: (dash 100)(turn_neck 134)(change_view normal)(attentionto our 3)
2,0     Recv MT2018_8: (turn 28.371)(turn_neck 80)(change_view narrow)(attentionto our 11)
2,0     Recv MT2018_4: (turn -134.774)(turn_neck -70)(change_view normal)(attentionto our 3)
2,0     Recv MT2018_11: (turn 0)(turn_neck 50)(attentionto our 3)
2,0     Recv HELIOS_base_8: (dash 74.538)(turn_neck 71)(change_view narrow)(attentionto our 11)
2,0     Recv MT2018_5: (turn 69.125)(turn_neck 69)(change_view normal)(attentionto our 3)
2,0     Recv HELIOS_base_5: (dash 61.638)(turn_neck -131)(change_view normal)(attentionto our 8)
2,0     Recv HELIOS_base_6: (dash 73.14)(turn_neck 121)(change_view normal)(attentionto our 11)
2,0     Recv HELIOS_base_3: (dash 61.273)(turn_neck -0)(attentionto off)(say "R-SXnaE hY")
2,0     Recv HELIOS_base_2: (dash 71.885)(turn_neck -0)(attentionto our 3)
2,0     Recv HELIOS_base_11: (dash 63.702)(turn_neck -11)
2,0     Recv HELIOS_base_9: (dash 68.9)(turn_neck 0)
2,0     Recv MT2018_10: (turn 4.977)(turn_neck -5)(attentionto our 3)
2,0     Recv MT2018_1: (dash 100)(turn_neck 0)(attentionto our 3)
2,0     Recv MT2018_7: (dash 100)(turn_neck -0)(attentionto our 3)
2,0     Recv MT2018_6: (dash 100)(turn_neck 0)
2,0     Recv HELIOS_base_10: (dash 69.876)(turn_neck 0)
2,0     Recv MT2018_3: (dash 100)(turn_neck -0)(attentionto off)(say "R7RGeEaTEX")
```

Fig.2. rcl file content

The rcg file records the stadium state of each simulation cycle, including the ball and player information, the number of action execution, and so on. The format of the rcg file is described as follows:

(Show Cycle ((BallInfo) (PlayerInfo1)… (PlayerInfo22))

The Cycle record for the current period, BallInfo record for the ball information, BallInfo format in detail:

((b) ball's x coordinates, ball's y coordinates, the x value of the velocity of the ball, the y value of the velocity of the ball)

Fig.3. rcg file content

Here is our idea of parsing the game log file:

First of all, we use regular expressions in Python and string processing function to extract important information in rcg and rcl, such as position information、speed information in rcg and action request instruction in rcl. Then,we stored these information as CSV file format and MySQL database. Based on the extracted data, we can count the important behavioral information in each game by algorithm, and sum up the information of each game, so that we can find out the strengths and weaknesses of the team easily. Through the game log file to guide us to develop the team, improve the efficiency and quality of development.

Now, we can use log to get all the information in a game and determine the player's strategy according to the log.

We use these information for positioning analysis, movement analysis, shoot analysis, etc. The efficiency and accuracy of the development have been greatly improved.

6. 0thers

In addition to the above mentioned, MT2018 also has done a lot of changes in other areas, for example, action evaluation, stamina model, tackles, formation and so on.

6.1 Tackle strategy

In the 2D simulation competition, tackling and intercepting the opponent's attacker is a regular and vital action; to have a better action here, the timing, intensity , and tackle angle are very important factors. In MT2018, we have added more strategies in the tackle.

When the opponent attack with ball, our defense player (usually Blocker) judges the opportunity of tackle, quickly intercept within a certain distance, at the appropriate time in the tackle.

We have optimized the generation and evaluation of tackles and consequently, we have gotten the best result to perform optimal tackles to shovel the ball as far as possible to the opponent's goal, our teammates, or our goal far.

Through the optimization strategy of tackles in MT2018, defensive ability and possession rate has been improved.

6.2 Stamina model

As we all know, midfielders, player number 6,7 and 8, are the most stamina

consuming players. In most cases, they appear with stamina failure at the end of half match; but, the defenders are still sufficient.

In the past, we reduced the midfielder offensive or defensive on move, through the choice of strategies to regulate the stamina. We are trying to use online coaches to achieve dynamic role interchangeability. For example, in a teammate frail condition, we can exchange his role function with a less stamina consuming role so as to achieve the purpose of the use of physical balance.

6.3 Formation

In the past, we always used the 433 formation, MT2018 is starting to use more formations, such as 4231 and 442 formations to cope with the style of different opponents.

7. Summary and Outlook

In the past six years, MT has been improving, but we still have a lot of inadequacies that need to be improved, and there are still a lot of ideas that have not been realized. We will do our best to study RoboCup. We are trying to use deep learning and intensive learning to train our team and we are going to use more smarter algorithms in RoboCup. The RoboCup is a good opportunity to learn from other teams, we will learn the strengths of other teams, improve our weaknesses, and with teams from all over the world, we will make the RoboCup simulation 2D soccer better and better.

8. Reference

[1] Akiyama, H.: Agent2D Base Code. http://www.rctools.sourceforge.jp(2010)

[2] Chen Zhiwen. Development Idea of Team Based on Simulation 2D Log File Data Mining[PPT]. 2016

[3] Defu Long. Mining and Application of RoboCup Simulation 2D Game Log[D], May 2016:32-42

[4] ZHANG Jiawang, HAN Guangsheng, ZHANG Wei. Application of Q-Learning Algorithm in Dribbling Ball Training of RoboCup[C]. College of Electronic Information & Control Engineering, Beijing University of Technology.2005,7:85-87

[5] Deng Yong: Alice Code. https://github.com/windywinter0101/Alice

[6] 方宝富. 机器人足球仿真[M]. 合肥工业大学出版社, 2011.

[7] RoboCup Official Site, http://www.robocup.org/