TJNU2010 2D Soccer Simulation

Team Description Paper

Gaoxin Lian, Huaihong Xu, Qi Wang, Chao Li, Meijuan Yu, Jike Sun, Rong Wang
Department of Computer and Information Engineering,
Tianjin Normal University,
Tianjin, China
wangrong0806@163.com

Abstract: In TJNU 2010, we continue to research soccer simulator team based on TJNU 2009. In this paper, we mainly introduce the structure of TJNU 2010 team, the strategy of improved dribble, pass, and special game mode strategies. According to training and experiment, the team works well.

1 Introduction

The TJNU team, attaches to the Tianjin Normal University, was set up in 2008. It is established by the lab center of the College of Computer and Information Engineering in Tianjin Normal University, which is based on the years of research of "Affective Computing and Intelligent Interaction Lab" on Robotics-related fields, which is formed up for robot soccer competition.

The TJNU team is mainly made up by the simulation group and medium-sized group and participated in China RoboCup in 2008 and 2009, then scored excellent achievements. Our simulation group won the third prize and the second prize.

In the past year, we have made our efforts to optimize the strategy of multi-agent collaboration and perfect the part of basic actions. When we give consideration to the offensive and defensive, we also pay attention to the teamwork. The team structure is divided into three decision levels, the overall decision, local decision and individual decision, which strive to get the best results in all horizontal aspect and vertical aspect. The main improvement includes:

(1) Using the predicted information to increase the efficiency of pass and shoot.

(2) The role decision based on the refinement of field.

(3) The strategy of counter attack based on the special game model.

In addition, we optimize the team from a whole and train the formation by optimization algorithm, at the same time, enhance the coach's decision efficiency.

In the second part of this article, it depicts the team's global construct, the third part discusses the team's improvement, and the fourth part gives a summary and brings forward the improving direction in the future.

2 Team structure

Our team is build in the base of agent2d

agent2d-2.1.0，http://sourceforge.jp/projects/rctools/releases/43082

librcsc-3.1.3，http://sourceforge.jp/projects/rctools/releases/47055

The whole structure of our team is as figure 1:

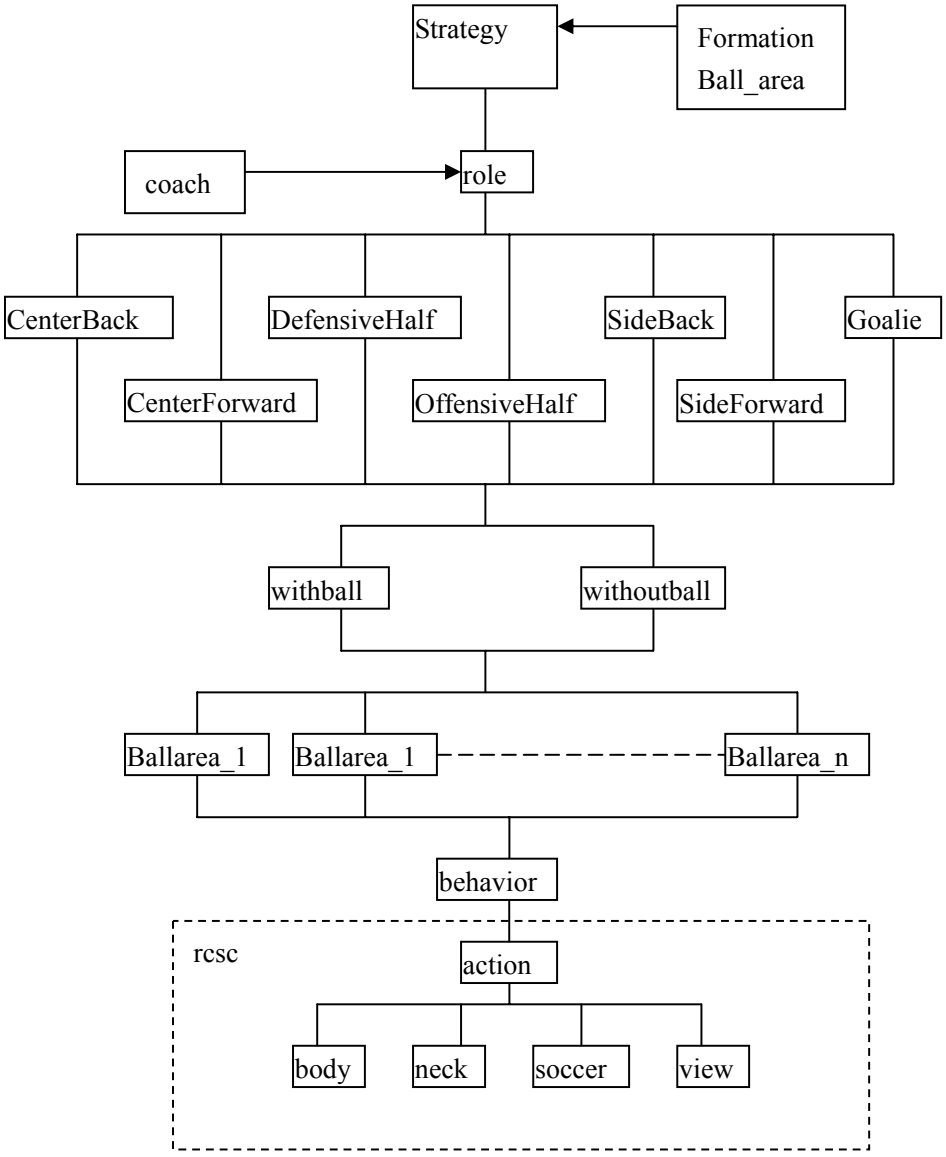


Fig.1 Team Structure

Our team is vertically refined .At the different regions of the field, different role uses different strategy, in this way, it is possible to make formation steady and make team work well. Team's main skills: Action with ball and Action without ball, it works well and strength team in the region offense or defense. Actions without ball—it's vital for player to go to a reasonable position and have a clear look on the fields and affect other players' action. Actions with ball—dribble, pass, those action is the key to the team. In all, players make decisions according to ball's position regardless of with ball or not. We refined the fields to archive team's high level strategy.

Strategy are added along with the different game mode in the behavior layer, this make it easy to have a thunder attack. In the base of actions, we develop the action of dribble, shoot, pass, etc. These keep the fluency of the basic actions.

In order to achieve the team's steady, we develop some formations. After lots of test, we use 343 as our normal formation, it works well. At the same time, team will change its formation according the formation of the field, focusing on the control of midfield. Online coach can change formation in the play_on game mode. When the game begins the communication between the coach and player is limited, player need to change the formation dependently using the

information of world model. Team uses a more balanced defensive and offensive formation 433. Once our team in the advantage, team uses the offensive formation, cut down the amount of defensive player, use more players to attack. When team loses some goals, it will use defensive formation avoiding losing too many goals. In all, our team uses different formations according to the information of the field.

Fig.2 The formation before kick off

3. Kickable strategy

The definitions of the actions related to attack, such as dribble, pass, shoot, etc, are placed in the bottom librcsc in agent2d, while it uses high level strategies function to make decision in the top. In the bottom, such actions as dribbling, passing, shooting are optimized by adding some prediction decision-making method. And at the same time, we optimize the division of the stadium and make a distinction and refinement about the function of different role players.

3.1 Dribbling

It is mainly about the dribble strategy decision for different players in different regions, with adding some judgment to control whether the player can dribble and which way he should choose. And at the same time, we tune some parameters related to the dribble action in the bottom of agent2d by using experiments many times.

3.2 passing

It is mainly about the optimization of the decision-making of the passing route's choice. To reduce the route choice scope and improve efficiency, we optimize the discrimination of passing direction through correcting the step length. And some manmade judging rules and regulations (e.g. it can't pass the ball to their goal, no players' area or outside of the stadium, etc) are added to eliminate some invalid direction. In decision-making, we predict the ball's and players' movement through the feedback information from perceptual system, and use the high reliability information as a basis for decision-making to choose a pass line with a high success rate.

Passing will involve players' cooperation and movement. To the local cooperation, according to

the ball holder in different situations, there are three passing styles: directly pass, lead pass and through pass. As shown in figure3.

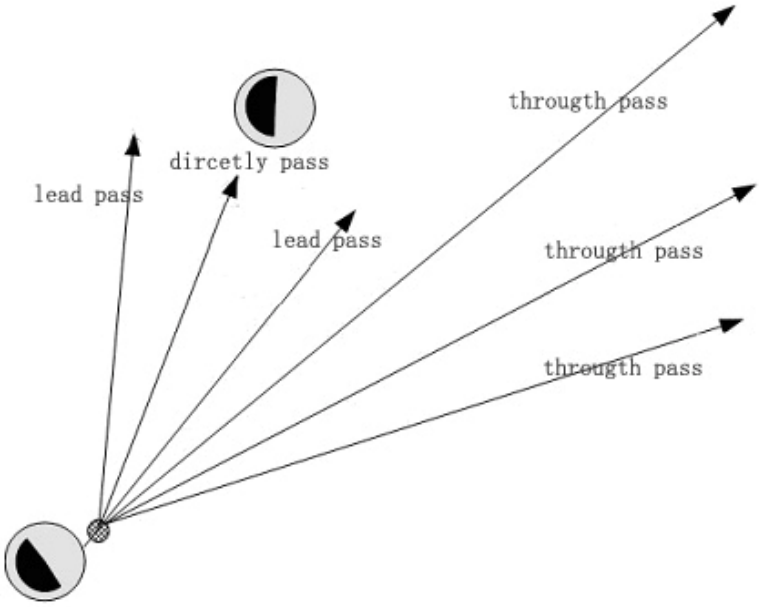


Fig.3 passing style

Generally speaking, as the situation that the ball holder player A is in the opponent's half field and the player is a attacking role, it will choose "two pass one" with the player B who has the highest success rate. And in another words, that is A will dash to the target location after passing the ball to B, and then player B will pass the ball to A again with choosing lead pass or through pass. But if the success rate is too low, player B will cancel the passing action to A, and it will choose to pass the ball to anther teammate or dribble according to the area it is in and its player type.

When player A who hold the ball is in the area near the enemy's end line，it generally choose to execute "cross" using directly pass or through pass, which is passing the ball to the teammate in front of enemy's penalty area. When the success rate of "cross" is too low, it will choose other strategy.

When player A is a organizer, (Choose center half-back or attacking midfielder as the organizer when the ball is in the middle side, and choose wing halfback or wing forward as the organizer when the ball is on the wing.), the attacking player in the near area (according to the regional division, with the same area or adjacent areas of A) should move action, the attacking player mentioned here refers to a teammate with no ball, and the organizer player A pass the ball to a location which has the highest success rate. And now it can choose one of the three passing styles according to the situation. And this strategy is also able to be used to the positioning ball strategy.

The local coordination mentioned above can implement using the basic action function and the player perceptual system function provided in agent2d.

3.3 Shooting

For improving the shooting, the selection for shooting-direction is optimized. In Agent2d, the scope of selection for shooting-direction is limited between the two goal posts. The goal line is divided from 8-10 parts; as a result, the shooting-direction can be dispersed. However, we find

that it is not prefect for all of shooting-distances. When the shooting-distance is very far, the change of shooting-direction will be not obvious. But when the shooting-distance is very close, the change of shooting-direction will be too large. The way we decide to do for optimizing is as follow. At first, we create a triangle by three positions' coordinate. One is the shooting player's coordinate which we mark as point A(x, y). Another two positions are the opponent's goal post's coordinates, which we mark as point B (m, n) and C (m, n'). So, we get the triangle △CAB. Secondly, we allow the player to shoot when the following judgment return true:

the max shooting-distance of the player≥ min(|AC|,|BC|)

Then we set the step length as 8° , so that we can divide ∠CAB to finite number of directions. If ∠CAB≤8° , the action of shooting will be prohibited (for thinking about the distance of shooting too far or the angle of shooting too small). Next, the player estimates the states of opponent players in next cycle by the situation of current cycle, in order to get the success rates of all shooting-directions, then, choose the shooting-directions as candidates whose success rate more than 70%. Last, for ever candidate, we mark the angle between the shooting-direction and the vector AC as α. And mark the angle between the shooting-direction and the vector AB as β. Then set a weight marked as w. We define the value of w as follow:

w=min(α，β)

We choose the shooting direction which has the minimum weight as the best one (not including the one whose weight is zero). As follow figure 4.

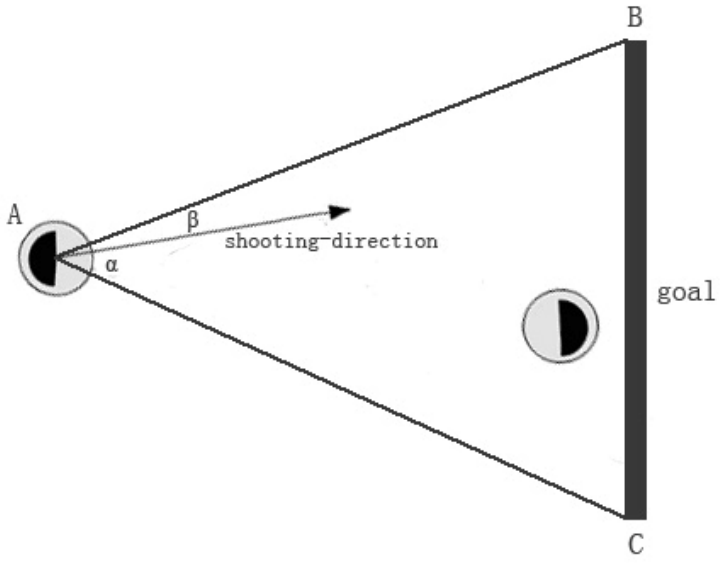
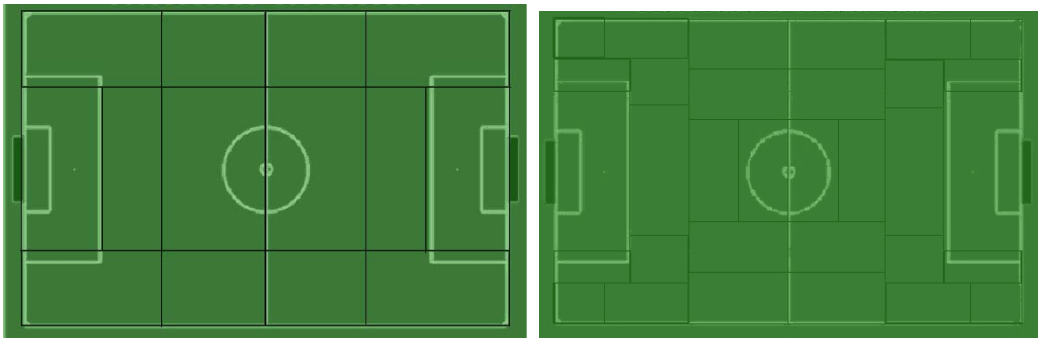


Fig.4 shooting

3.4 Field Partition

For making strategy more accurately, in agent2d, the field is divided into a lot of areas. We do some further refinement base on the original. We narrow the scope of the original area, expand some new areas and refine the area of midfield and wing. So we can avoid the accuracy being reduced by the too large partition of field. As follow figure 5.



A. the original partition B. the optimized partition

Fig. 5 field partition

4 Special game mode strategies

In the Robocup game, besides the normal pattern(play_on), there are many special game modes，such as kick_in (throw), free_kick (Kick / free kick), corner_kick (corner), goal_kick (goal kick), goalie_free_kick, back_pass, ind_free_kick, penalty_kick. We develop the offensive and counterattack strategy correspondingly against different game modes.

In the low layer of agent 2D, after entering special game mode, we simply judge whether with ball or without ball, then, if with ball, judge our goalkeeper whether is activated or not, if he is activated, execute the basic movement function of goalkeeper. If not, the player execute Kick-off simply, the others move simply. For example, the player who is the nearest to the ball, kick the ball to the player who is the nearest to goal without care the distribution of the other players and judge the one whether catch the ball suitably or not. So, we design some advanced strategy against the above situation. Firstly, consider the situation of the game into account, don't kick the ball casually. If you find a teammate from the ball closer to the ball is not suitable, including pass intercepted and get the ball but control it so that lose the ball etc. So player take a look at the field environment, consider pass the ball to distant teammate or pass the ball until the teammate move to the right place. Even in some case, we can do some proper return; make the ball under our control, which can find out some better chance.

In goal_kick pattern, to get more efficiency at the goalkeeper kick-off, we decide Kick-off direction by judging the proportion of both players. At the same time, we also strengthen the offensive by making use of fixed mobile of few players.

5 Conclusions

To sum up, though we are improving the base, our focus is on setting up and modifying high level strategies. Through the experiment and repeated practice, it is witnessed that the improvements lift the ability of our team. In the future, we are going to apply some intelligent algorithm into the overall coordination in our team to enhance the flexibility of our team and the awareness of offense and defense, thereby perfecting our entire team.

References

[1] Federico B.D and Alessandro R.D. Three Approaches to the Coordination of Multiagent Systems [J] Universita degli Studi di Parma, 2003

[2] Stone. P. Layered Learning in Multi-Agent Systems: [PhD Thesis]. Carnegie Mellon

University, 1998
[3] Wu Bo, Wu Min, Cao Weihua.Behavior-based Dual Dynamic Agent Architecture [J]. Journal of Central South University of Technology, March, 2003
[4] Mao Chen, Klaus Dorer. Users Manual RoboCup Soccer Server [S]. The RoboCup Federation, 2003