Ri-one 2008 Team Description

Tatsuro Bito, Yuta Nari and Katsuari Kamei

Ritsumeikan University
1-1-1 Noji-higash, Kusatsu, Shiga 525-8577, Japan

ci012064@is.ritsumei.ac.jp
`http://spice.ci.ritsumei.ac.jp/~robocup/`

Abstract. Ri-one is a student team of extra curriculum activities in Ritsumeikan University. This paper describes the main feature of Ri-one 2008, the RoboCup soccer simulation 2D team. Especially, we mainly show the points that evolved from our previous team Ri-one 2007 [1] and also introduce several techniques of decision making.

1 Introduction

Ri-one is a student team of College of Information Science and Engineering, Ritsumeikan University. In our first participation in RoboCup 2006, we won the 3rd place. In RoboCup 2007, we won the 12th. Our main goal is ”to construct strong AI for soccer”, and in our opinion, the strong AI is that an agent have to perform the high-level decision making. In this paper, we introduce approaches to our goal. In Section 2, we explain our original World Model ”Power Field”. In Section 3 and 4, we describe ”Decision Making” and ”Action Strategies”, respectively. Finally, we summarize our ideas.

2 Power Field

Each agent have different view information. If there is this difference, agents can’t have same decision making because their decision making depends on the view information. We think that the difference of mutual decision making can be reduced, when agents blur the view information. Therefore, we developed original World Model, we called this Power Field (PF). In order to blur the view information, agent considers the opponent’s goal and teammates as a model which generates negative power, and the friendly goal and opponents as a model which generates positive power. As a result, the agents can blur the view information by making a pseudo field mountains and valleys as shown in Fig.1. These powers are calculated by six elements (agent position, direction of agent body, agent speed, last cycle the agents has been seen, distance from the goal, distance from the end of the field).

The judgement by PF is very useful for determining tactics because PF can be judged the field. However, when a certain amount of ambiguity is contained

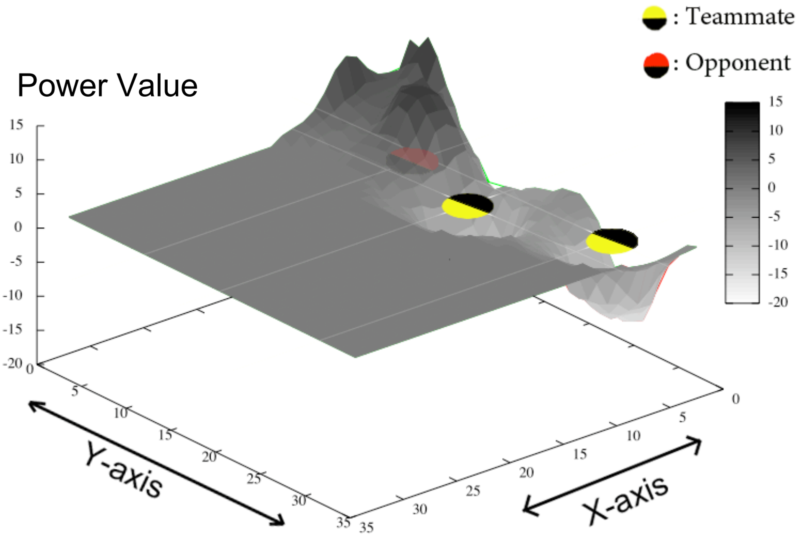

Fig. 1. Visualization of Power Field

in Power value, the judgment by PF is not so trustworthy. Therefore, the agents choose the best action by the actual view information after they extract candidate actions such as a pass by the judgment of PF at the beginning. As a result, PF is used as a means of deciding a strategic plan.

Although, when reliability of the agent's view infomation is too low, it is necessary to decrease a margin of error in the view information. To reduce this error, we use the revision of the view information by Communication. Fig.2 show on example of the revision of the view information by Communication.

In this situation, a teammate teaches reliability and opponent's position to a dribbler by using say command. The dribbler compares the reliability of opponent's position that he recognized (recognized position) with the reliability of opponent that he heard from teammate(real position), and then the dribbler gives a priority to the information of opponent's position that reliability is higher. By this method, the dribbler can get information which can be used PF.

3 Decision Making

In the current rcssserver (rcssserver11.1.2), the agent view model is not effective compairing with the real soccer. In the real soccer, human players have a wide view with accuracy. However, in the simulation soccer, the agents can not get accurate view information with wide view. If the agents want to get accurate view information, they have to change their view width into narrow view width. In addition, receiving frequency of view information depends on the view width

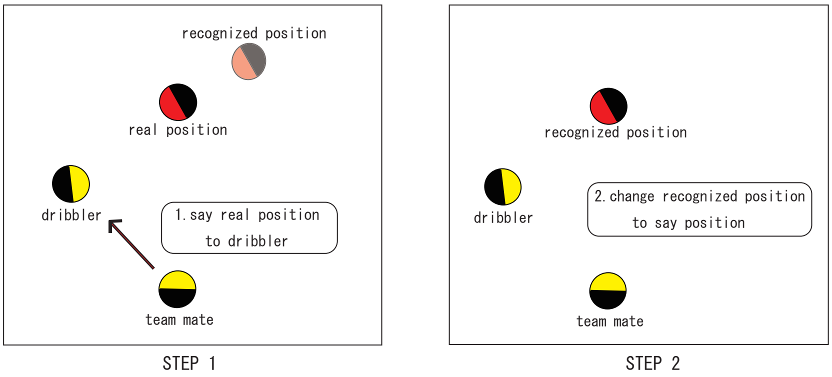

Fig. 2. The revision of the view information by Communication

and the view quality. Therefore, the agents usually perform their actions with insufficient view information. For higher-level decision making, to improve this weak point is essential. In order to solve this problem, we use two skills of See Synch and Communication in the following subsections.

3.1 See Synch

Sensing and acting is asynchronous in the current rcssserver because of different frequency of the received view information. To improve this complex, we use See Synch at all times in the simulation. See Synch is a method to synchronize sensing and acting. The method is composed loops of change of the view width as follows: narrow → narrow → normal (75ms+75ms+150ms=300ms) with view quality high [2]. By using this narrow-narrow-normal loop, the agents can receive view information three times in three cycles (one times per one cycle). As a result, the agents can decide their action after they received the view information in each cycle. In addition, to avoid breaking See Synch in simulation, our agents try again See Synch in free kick or goal kick.

3.2 Decision Making From Communication

Each agent perform decision making playing three basic roles: Forward, Midfielder and Defender. These three roles are seemed to be effective in simulation. However, these roles are not always effective in a specific situation. For deciding appropriate role in the specific situation, we use communication (decision-role-say) to decide the agent's role. Mainly, we use decision-role-say when the agent controls the ball. The reason is that the agent decides pass-receiver's role at the same time as the agent gives a pass for the pass-receiver. We use a short message (one word) for deciding the agent's role. The short messages corresponding the roles are shown in Table.1.

Table 1. Sample say messages and their roles

sample words	roles
w	Offensive Dribbler
k	Ball Keeper
i	Pass Waiter
g	Ball Shoter

In specific case as shown in Fig.3, the agent who controls the ball performs a through pass to a teammate with decision-role-say "w", this word indicates "Offensive Dribbler". The teammate who receive the message performs action playing the role of "Offensive Dribbler", the role is to perform the movement of a quick dribble for positively when an agent controls the ball. The role by decision-role-say is carried out till the say-receiver judges that he should quit a role by himself or he receives a say of quit role from a say-sender.

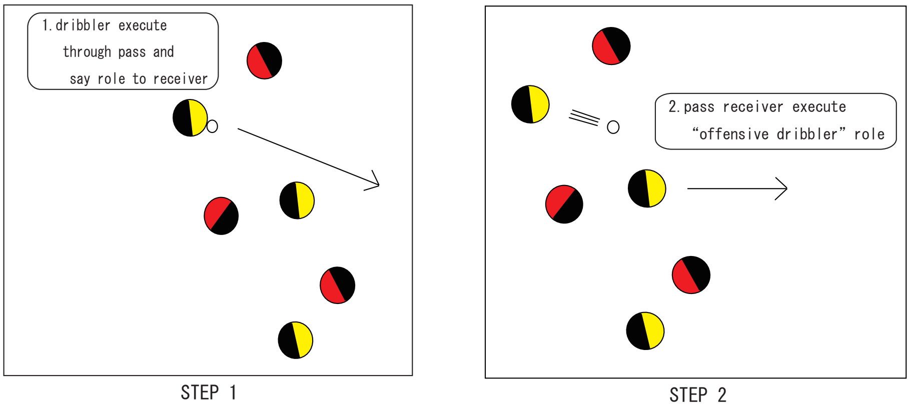

Fig. 3. Decision role by communication

4 Action Strategies

4.1 Dribble

In a meeting of past RoboCups, the dribble skill of the teams such as Brainstormers (the 1st place team of RoboCup 2007) and Wright Eagle (the 2nd place team of RoboCup 2007) is very good. It is rare that their team's dribbler is stolen a ball by other agents. In addition, "Neuro-Dribble" [3], which is designed by OPU hana (the 4th place team of RoboCup 2007) is very dynamic and flexible. In the soccer simulation, the dribble skill is the most important skill when an agent controls a ball. For keeping the ball in our possession, ”movement behavior” and ”decision of dribble route” are necessary techniques for dribble skill. Our dribble skill is focused on these two noteworthy points.

The movement behavior is the following three steps.

1. Determine the number of dash action.
2. Calculate the position to kick the ball.
3. Kick the ball or dash to ball.

In the first step, the number of dash is determined heuristically. The agent determines the number of dash n with distance from the agent to the closest opponent at the direction *direction* determined. In the second step, the agent calculates a maximal distance *dist* which the agent can move with n times dash. Moreover the agent calculates a vector v with *dist* and *direction*. Finally, if the agent can kick the ball , the agent kicks the ball with the vector v. If not, the agent dash to the ball. When the closest opponent exists in an agent kickable area, the agent gives a priority to hold the ball close to the agent's body rather than dash or kick the ball.

In the method of ”decision of dribble route”, we use Power Field. We decide dribble routes and nodes as shown in Fig.4.

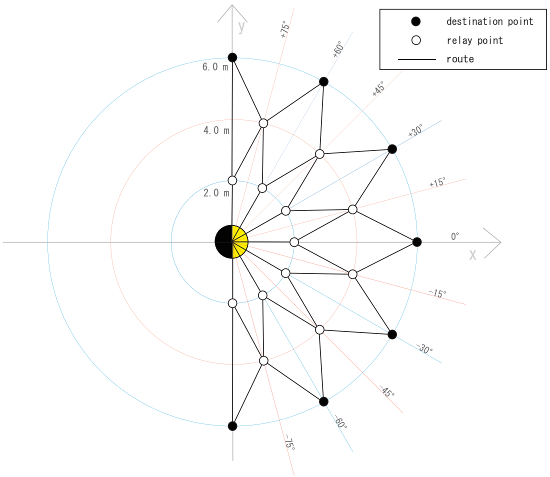

Fig. 4. Dribble rotes and nodes

After deciding routes and nodes, the agent carries out the following three steps.

1. Examine the power value of the each node's position.
2. Calculate the total cost about each route.

3. Determine the minimum route.

In the first step, the agent computes the power value *costs* of all node points. In the second step, the agent calculates the total *costs* of each route and determines priority of the route. The priority is determined by the following three points.

- The angle of dribble route is within the maximum angle that an agent can change direction at 1 cycle.
- Lower the priority of ± 90 degrees direction and the ± 60 degrees direction. Raise the priority of ± 30 degrees direction.
- When the position searched by the agent is out of a field, lower the priority of this route.

After calculating the total *costs* of each route, the agent determine the minimum route as an actual dribble route, and performs dribble action towards the destination position in Fig.4 of the actual route.

4.2 Pass and Shoot

We developed Gravity Model (GM) which gives a priority to some target points for a pass and a shoot. GM regards each agent and an object as a planet, and GM calculates the indicator showing the strength of connection between passer and receiver. GM considers the opponent's goal and teammates as a model which generates gravitation, and GM considers opponents and the friendly goal as a model which generates repulsion. The gravity value of an agent is given as follows:

$$GravityValue = X + Y \tag{1}$$

$$\begin{aligned} X = &\left\{(farcut - DisO)\frac{1}{lostcycleO^2}\right\} \\ &- \sum_{x=1}\left\{(farcut - DisT_x)\frac{35}{Dis\theta_{ot_x}}\frac{1}{lostcycleT_x^2}\right\} \end{aligned} \tag{2}$$

$$Y = \frac{2T_{PF}}{DisG^2}G \tag{3}$$

Where, *GravityValue*: The evaluation value in GM, *DisO*: Distance to the pass target, $DisT_1$: Distance to an one person (T_1) in the enemies who can intercept, *DisG*: Distance in a straight line between the pass target and goal center, θ_{ot_1}: The angle of pass target and T_1, G: The variable for compensation, T_{PF}: A pass target's force value acquired by PF, *farcut*: The maximum distance that can be passed, *nearcut*: The minimum distance that can be passed, *lostcycleO*: Cycle that lost the pass target and *lostcycle*T_1: Cycle that lost T_1. See Fig.5.

Furthermore, we supposed the opponent satisfying the following two conditions can intercept a pass.

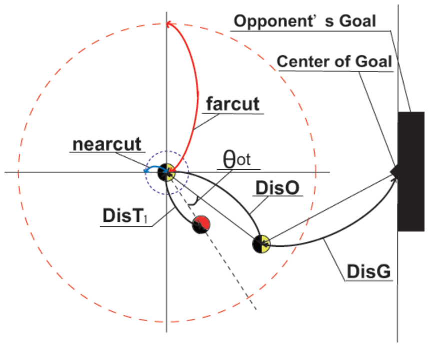

Fig. 5.

- The difference between $DisO$ and $DisT_x$ must be less than 10.0.
- θ_{ot_1} must be less than 50.0.

The agent passes a ball to the teammate which has the highest Gravity Value. Similarly, when a pass target is changed to a shoot course, this shoot can be evaluated if Y is disregarded.

4.3 Rundown

Sometimes, a certain intercepting by two or more teammates might be required. We made Rundown for such a situation. Rundown is to surround an opponent and to intercept a ball by three teammates. Rundown begins from surrounding an opponent with a ball by teammates. Actually, rundown is started when a ball approaches the center of triangle made by the three teammates near the ball. For instance, the rundown is executed in case as shown in Fig. 6 by PF. In that case, the teammates move so that they make minus power of PF around the ball.

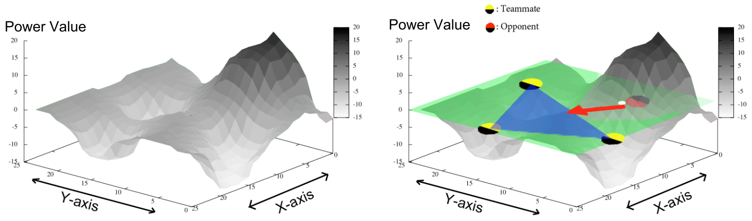

Fig. 6. Example of Rundown situation

4.4 Dribble interrupt

Recently, as the number of teams with the powerful dribble such as Brainstormers, OPU hana, etc. has increased, the technology that stops a dribble certainly is indispensable. Dribble interrupt starts from approaching an opponent's dribble course by two teammates. In order to stop the powerful dribble, one moves to the same direction as the opponent movement, and another moves between the goal and the opponent. To complete the former movement, the agent predicts the position of the ball after 2 seconds, and then moves there. To complete the better movement, the agent moves to a position where he can obstruct opponent's shoot. In addition, that position is calculated by GM.

4.5 Search of a mark position

In defense, marking a specific opponent might be better rather than protecting by formation. Brainstormers with powerful mark technology stops the attack by Man Mark(the mark by one for one), in addition raises the ball possession rate in the middle stage, and advances the game advantageously. So, Man Mark's validity is clear, therefore our team also adopts Man Mark. Basically, a teammate playing the role of "Marker" who is the nearest the positive power place around the goal realizes the mark by moving so that a positive power place is negated. Moreover, sometimes it might be better to obstruct a centering according to the position of the ball. In that case, the agent calculates Gravity Value of the ball and the mark target by GM, and he moves to the position which the minimum Gravity Value.

5 Conclusion

First, we proposed our creative World Model "Power Field." Secondly, we described Communication and See Synch for decision making and the action strategies after the decision making. We spent a lot of work in constructing the high-level decision making architecture because the high-level decision making is essential for the agents to have a consistent intention. Our techniques such as Power Field, Communication and See Synch are useful for the decision making and the action strategies. We believe that we can approach to " Agents have a consistent intention " by using these methods.

References

1. Shinya Otsuki, Tatsuhiko Katayama, Kouji Ohnaka, Tomohiro Iwasaki, Ryo Ishihara. Ri-one 2007 Team Description, 2007.
2. Hidehisa Akiyama, RoboCup Soccer Simulation 2D League Victory Guide. page 244-256, 2006.
3. Tomoharu Nakashima and Yukio Shoji, Team Description of OPU hana 2D 2007, 2007.