FUT-K_3D Team Description Paper 2011

Masahito Yashiki, Kenta Miyajima, Kazutomi Sugihara, Kazumasa Ohkuma, and Teruya Yamanishi

Department of Management Information Science, Fukui University of Technology
Gakuen, Fukui 910–8505, Japan
{sugihara, ohkuma, yamanisi} @fukui-ut.ac.jp

Abstract. We describe concepts of movements for agent and unique tactics of team FUT-K on RoboCup Soccer 3D simulation here. In additions, the future work is mentioned.

1. Introduction

FUT-K that is mainly composed of undergraduate students of Fukui University of Technology in Japan has been organized since fall 2007. The purposes of our team are to grow knowledge and experience of the computer language and the information science through applying themselves to RoboCup Soccer 3D Simulation. Though almost members of our team are unskilled at programming yet, we believe that now our team is developing with getting advice from other teams.

In this paper, we introduce the improvements on our agent during this year as follows:

— Smooth and Omni-directional movements of the agent
— Coordinated Motion by Communication protocol
— Implementation of Probabilistic Behavior Selection

The details are explained in following sections.

2. Omni-directional Movement of the agent

Let us consider a movement for skew walking. For the movement of an oblique direction, our previous agent had to go forward after a change of direction. It has taken much time for the ambulatory movement. So we establish an Omni-directional movement of the agent by implementing skew movements using a superposition of the forward (backward) movement and the lateral one. That movement is realized by a function of θ which is defined as

$$\frac{M_{Forward(Backward)}}{M_{Lateral}} = \tan\theta, \quad (1)$$

where $M_{Forward(Backward)}$ and $M_{Lateral}$ are the forward (backward) movement and the

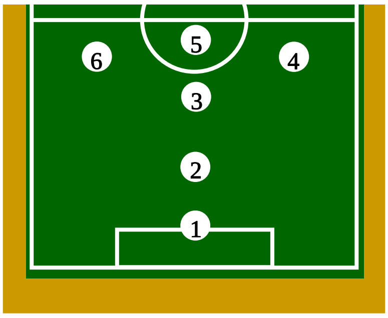

Fig. 1. Position of Agents
(1: Goal Keeper, 2: Sweeper, 3: Midfielder, 4~6: Offences).

lateral one, respectively. When agent is walking forward, θ is zero degree, and θ takes from -180° to 180°. For example, in the case that agent walks to the $\theta = 30^\circ$ direction, skew walking is realized by combining $1/2$ the forward movement and $\sqrt{3}/2$ lateral movement.

3. Coordinated Motion

Currently, we show grouping of each agent in Fig.1. Here, three agents who are in first alignment play the role of offence. However, unfortunately we cannot develop supple pass movement because of limitations of the joint of 3D agent. Accordingly, keeping a ball, the agent only attacks the goal of enemy by dribbling, and then our agents of friend crowd around one place of the ball since coordinated motions were not implemented. So, we introduce two-way communication between the three offence agents in the position of top line and one midfielder who is in behind offence agent in order to implement coordinated motions of tactical change in positions of four agents. Here, we develop new communication protocol by using two functions of Say-Effector and Hear-Perceptor.

3.1 Communication protocol

The agent can get the relative coordinate of objects on the soccer field (e.g. each flag, agents of friend and/or enemy, and the ball) from server by every 0.02 seconds. But the function of Hear-Perceptor receives any message at every 0.04 seconds. In addition, the agent cannot get all messages transmitted from agents at the same time due to collision between messages and to be limitations of Hear-Perceptor. So, we suggest the method which each agent transmits one message in turn as follows:

STEP1: The i-th agent transmits a message,

STEP2: After i+1-th agent confirms to receive the message, its i+1-th agent transmits a completion massage on the reception,

STEP3: After i-th agent confirms its completion massage, the agent "i" stops own

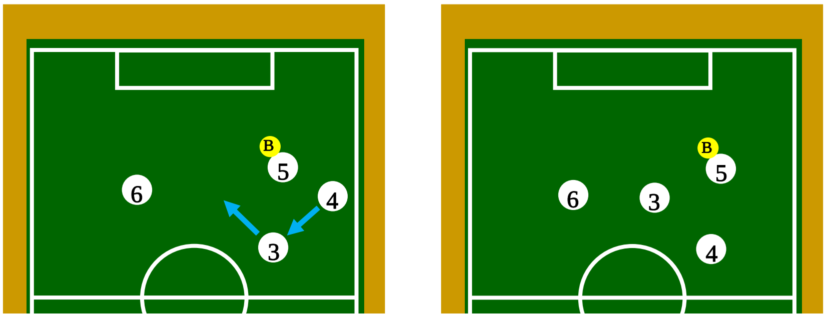

Fig. 2. Changing Positions(Left：Before Right：After) and Object of B is Ball.

transmission of the message,

STEP4: After i-th agent confirms to stop the transmission of the message, it increases one to “i” and returns to STEP1.

Here the communication is continuously repeated until end of the games and the initial value “i” is set as one. After STEP4, if the all agents participating in communications finish the transmission and reception of the message, number of “i” is initialized to one. This algorithm is inspired by a token passing mechanism of access control method for network systems

3.2 Tactical Changes in Positions

As agents can communicate with each other by exchanging the message, we attempt to make the code on a coordinated motion of soccer agents. So, the nearest agent from the ball performs to approach to the ball, and if that keeps the ball, it begins the movement of dribbling or kicks the ball. Other agents begin the coordinated motion keeping the formation described in advance. In addition, if the agent keeping the ball goes into other agent’s location shown in Fig. 2 (Left), the rest starts the position change corresponding to the location on agent keeping the ball.

STEP1: The agent transmits a message on the information of all agent’s location to other agents (for example, the message is transmitted by 5th agent in Fig. 2),

STEP2: Other agents confirm to update the information of the location by STEP 1 and compare own present location with updated one. If the present location laps over the other agent’s one, the agent of original position changes own position to midfielder and then transmits a message of the change (the message is transmitted by 4th agent in Fig. 2),

STEP3: Current midfielder agent confirms the change by STEP 2, and moves the forward(empty) position brought by the change (the message is transmitted by 3^{th} agent in Fig. 2).

Tactical change in positions is carried out by such a procedure. The result is shown in Fig. 2 (Right).

4. Probabilistic Behavior Selection

In the previous version, our team's agents were implemented an individual behavior in any situation. Consequently, there were some situations to be in a deadlock in a game. Our agents are implemented the function called "Probabilistic Behavior Selection" that even if they encounter the identical situation, they stochastically select one action from multiple pre-defined behavior. Here, we assume the probabilities of each behavior are set before a game. In addition, depending on the result of Probabilistic Behavior Selection, we also consider a method to update the probabilities during the game.

4.1 implementation of Probabilistic Behavior Selection

Before the implementation, we define the behavior to be selected. There are three types: "Passing", "Turing-aside" and "Breakthrough" and then the following eight options on these types.

- ✓ Passing : Direction(Right or Left)
- ✓ Evasion : Direction(Right or Left), Speed(Quick or Slow)
- ✓ Breakthrough : Kicking or Dribbling

When a specific condition(e.g. one enemy's agent approaches our agent) was satisfied, the agents keeping a ball stochastically selects one from these options.

4.2 Judgment of game situation

In order to perform the function Probabilistic Behavior Selection, we make our agent discriminate the situations in a game: that is, situations are described by five factors, "Area of our agent keeping a ball", "Distance between our agent keeping a ball and the nearest enemy's agent", "Distance to the nearest friend's agent", "Elapsed time" and "Point Spread". Our attempt makes a distinction between 384 patterns in a game.

4.3 Setting probabilities of each behavior

The probabilities of each behavior under each situation were calculated by following.

Table 1. Calculated Probabilities to each behavior by Neural Network

Behavior	Probabilities (%)
Passing to left	9.54
Passing to right	13.50
Quickly turning-aside(Left)	16.69
Slowly turning-aside(Left)	8.34
Quickly turning-aside(Right)	9.63
Slowly turning-aside(Right)	7.46
Breakthrough “Kicking”	20.38
Breakthrough “Dribbling”	14.41

Table 2. Criteria of “Success” on behavior

Behavior	Criteria of “Success”
Passing	After 2 seconds of passing, at least, one of the friend’s agents is nearer to the ball than any enemy’s agent.
Turning-aside Breakthrough “Dribbling”	Keeping the behavior to be selected in five seconds
Breakthrough “Kicking”	After 2 seconds of kicking, at least, one of the friend’s agents is nearer to the ball than any enemy’s agent. Or the moving distance is more than 4. (The distance of 4 is equal to the diameter of the center circle.)

STEP1: We pick up some situations at random as samples. By Analytic Hierarchy Process(AHP), the probabilities to select each behavior are given in each sample situation.

STEP2: Using the three layer neural network, the probabilities in the remaining situations are estimated based on the probabilities obtained in STEP1.

Table 1 shows an example of the estimated probabilities in STEP 2.

4.4 Update of the probabilities during a game

As it is impossible to get enough information of enemy’s teams, the probabilities set before a game is not necessarily proper for playing. Therefore, we try to update the probabilities of each behavior in a situation, based on the result of Probabilistic

Behavior Selection. The results are expressed by three states, “Success”, “Failure” and “Behavioral incapacitation”. The conditions of “Success” on each behavior are defined as shown in Table 2. According to the states, during a game, the probabilities are updated as follows:

- ✓ Success : Increasing the probabilities of selected behavior and decreasing the ones of the remaining by constant values.
- ✓ Failure : Decreasing the probabilities of selected behavior and increasing he ones of the remaining by constant values.
- ✓ Behavioral incapacitation: Handling in the similar way to “Failure”, but the constant values is less than “Failure”.

4.5 Experiment

To verify the validity of our method, we attempted the following experiment. Our agents except “goalkeeper” and “sweeper” are implemented Probabilistic Behavior Selection. Our team, contained the implemented agents, played against three teams(“A”, “B” and “C”), possessing different skills and abilities according to the format of RoboCup 2010. Table 3 shows the result of 100 games with each team. From this results, there are not significant outcome about these indexes. On the other hand, we notice that updating the probabilities is more effective than Probabilistic Behavior Selection without the update in a game. It is estimated that the given probabilities before a game is inappropriate in this experiment.

Table 3. Results of experiment

Opponent	Victory or defeat	Goal For	Goal Against
A	6－68－26 (8－60－32)	18(18)	121(104)
B	14－22－64 (15－21－64)	20(18)	31(25)
C	69－ 1－30 (79－ 2－19)	133(150)	1(3)

#1 Victory or defeat: Won－Lost－Tied
#2 ( ) means the result of non-implemented Probabilistic Behavior Selection

Moreover, the frequency of updating the probabilities for each agent is tabulated in Tables 4-6. Here ③-⑥ means the number of our agents shown in Figs. 1 and 2. It can be said that the frequency of updating the probabilities are different by positions, opponents. That implies that it is possible to examine the feature of opponents from the frequency of updating for each agent. As mentioned before, since the given probabilities before a game may be inappropriate, we expect that the modification on the probabilities will generate more significant results. Furthermore, the strategies of our agents will be improved by playing with the other opponents.

Table 4. The frequency of updating the probabilities about games with team A

Number	Success	Failure	Incapacitication	Subtotal
③	118	263	339	720
④	103	315	329	747
⑤	217	481	445	1143
⑥	105	331	366	802

Table 5. The frequency of updating the probabilities about games with team B

Number	Success	Failure	Incapacitication	Subtotal
③	107	238	337	682
④	107	270	365	742
⑤	150	415	454	1019
⑥	117	283	347	747

Table 6. The frequency of updating the probabilities about games with team C

Number	Success	Failure	Incapacitication	Subtotal
③	64	133	176	373
④	67	173	215	455
⑤	158	396	341	895
⑥	108	276	342	726

5. Conclusions and Future Works

We implemented the Omni-directional movement as the basic movement necessary for soccer games. In addition, the coordinated motion was realized by transmitting and receiving messages using the two-way communication between agents. But this method may increase times according to increase number of agents. Also, if agents rapidly act tactical changes of locations from the coordinated motion, we should consider more stable walking.

For the autonomous decision, we challenged to implement Probabilistic Behavior Selection and then update of the probabilities of each behavior during the game, because it is expected to improvement the strategies of our agents. However, we could not find a significant outcome from the results of games with three teams possessing different skills. The cause of the problem can be integrated into three factors, “Few elements to describe situations in a game”, “Unstable movement of agents” and “Scanty update of the probabilities of each behavior".

In the future, we will want to consider improving on how to update of the probability of selection. For example, in order to increase the frequency of updating probability of selection, if it is a similar game situation, an agent continuously updates the probabilities of each behavior.

Acknowledgements

This work is supported in part by research grants from Fukui University of Technology and a Grant-in-Aid for Scientific Research(C-21500962) from the Japan Society for the Promotion of Science.

References

1. Chang, M., Mitsui, T., Asai, H., Sugiyama, M.: Neural–Oscillator–based Algorithm for Humanoid Robot Locomotion. In: Ann. Rep. of Gifu Prefecture Research Institute of Information Technology 5, pp. 77–79 (2003)

2. Nakamura, A., Yamanishi, T.: A development of simple gait methods for 3D agents. In: The 71th National Convention of IPSJ, pp. 2-87 – 2-89 (2009).

3. van Dijk, S., Polani, D: Principled Construction of Perception and Action Skills for Bold Hearts 3D 2009, In: the Team description paper of 3D simulation league in RoboCup 2009 Graz (2009).