UTUtd2006-3D
Team Description Paper

Nima Aghaeepour, Alireza Fotuhi Siahpirani, Saeed Masoudnia, and Soroosh Radpoor

Department of Computer Science
School of Mathematic, Statistics and Computer Science
College of Science, University of Tehran, Iran.
{nimaa, fotuhi, masoudnia, radpoor}@khayam.ut.ac.ir
http://www.fos.ut.ac.ir/robocup

Abstract. UTUtd-3D 2006 is based on experience of UTUtd-3D 2005 [1]. Last year the team worked on high level skills especially on positioning, ball handling, decision making[2]. As the server status for the 2006 competition is cleared recently, this paper mostly describes changes since 2005 in the architecture of the UTUtd-3d team and the planned changes for 2006 competition. Most important changes are: 1. Improvement of high level skills (shoot, pass, dribble). 2. Implementation of a new system to determine team's strategy and players positioning according to the way opponents play. We also describe our effort on a controller for the legged agents.

1 Introduction

UTUtd is the RoboCup team of school of mathematic, statistics and computer science, University of Tehran. This team has started its activity in 2D since 2000 year, and in the year 2004 when soccer simulation 3D competition became a part of RoboCup for the first time, this team started its work on 3D Soccer Simulation.

Soccer simulation is a Multi-Agent System. In Multi-Agent Systems several independent agents are defined in a world model and act autonomously based on the goal of team. So RoboCup Soccer Simulation is a very suitable test bed for implementation of intelligent Multi-Agent Systems.

RoboCup Soccer Simulation Server 3D (rcssserver3D) is a Multi-Agent simulation system for a physical agent in three dimensional environment and resembles to the real world much better than 2D. This server is based on SPADES [3], which is a middleware system and has unique properties; It is an event-based system that can be distributed among several machines. Rcssserver3D is relatively new and very different from 2D and considers real physical rules like friction and gravity. So calculation of some facts like prediction of position and velocity of ball and agent or interception are much more complex.

High level skills, positioning and decision making[1] were the most important issues that were studied last year. This year based on our previous years experiences we decided to work on a new idea to change the team's strategy in the

middle of the game. the new system will choose it's strategy between different predefined strategies.

Section 2 describes the affort to implement Dynamic Strategy Switching. We describe the skills which have been changed since last year in section 3. Other skills are described in [1]. Our new localization is described in Section 4. We although developed an agent base code for the legged agents, world modeling and some basic skills which are described in Section 5.

2 Dynamic Strategy Switcher

Dynamic Strategy Switcher (DSS) is our new approach to change agent's behavior according to the game state. Strategy switching consists of two parts: Containers that contain different strategies, team actions, agent behaviors and a brain to select from them. Currently we implemented the containers so that team can switch the strategy in the middle of a game. But the strategy selector is still under development. There are currently three containers, the most abstract and general strategies are in the first level container, the second level is a set of different team actions for each of the strategies, and the third one is the basic skill container. when the team's strategy changes team's formation and roles are changed according to the current strategy. Then each player finds its new role in the new team formation. The most important part is to coordinate agents' decisions so we can have a better team cooperation.

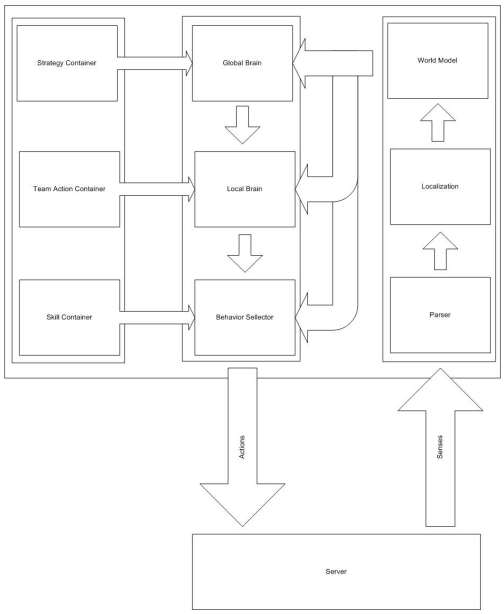

Fig. 1. Dynamic Strategy Switcher's Structure

3 Skills

In this section we describe the skills which have been changed since last year. Passing and Shooting haven't been modified since last year. Dribbling and kicking are described below.

3.1 Dribble

Dribble is one of the most important skills of an agent. Deciding how to dribble depends on several parameters like current team strategy (chosen by the global brain), team action, agent's role, agent's position, distance between nearest opponent and ball, opponent's empty regions. Currently we are developing a linear method for dribble desicion making. We are also trying to combine these parameters using different machine learning methods.

3.2 Kick

Kicking as a basic skill affects passing, shooting and dribbling. To have a safe pass or dribble we need to know where the ball hits the ground. We can physically calculate it for the first time. But first bounce can't always be used for the target because the ball's speed may be too much and we can't intercept the ball at the first bounce (as shown in Fig. 2). Because of the complexity of the computations required to predict the ball's movement after hitting the ground we used a neural network which will estimate where and when the ball will hit the ground after n'th bounce. We also need to calculate a suitable power and angle to control the ball's speed at the target. There is another neural network which estimates the optimum power and angle using the output of the function which calculates all of the bounce points. (as shown in Fig. 3)

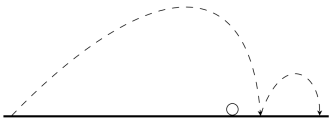

Fig. 2. Agent can't intercept the ball when speed is too much

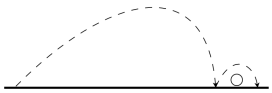

Fig. 3. Ball velocity is reduced after first bounce so the ball can be intercepted

4 Localization

The current method of decision making for the interceptor is relied on objects' positions and velocities; Hence precision of an agent's perception barely affects agent's behavior during the game. The relation between agent's decisions in higher levels and agent perception of environment causes the agent to become unstable in the case of receiving noisy data. In order to reach to a stable point it is required to apply an efficient and accurate method of localization. So far many methods have been implemented by other teams for localization. According to the method which was mentioned in[4] position of the player is determined by means of statistical calculations and regarding to the error's distribution function. Other methods of localization discussed in [5], use the information from different parts of the field i.e. different Flags to specify agent's position. Our new method of localization is a combination of methods mentioned in [5]. This new method decreases the vision error considerably.

5 Controller for the legged agent

When the rcssserver3D-0.4 released we started working on legged agents. We extended our agent's base-code and parser to work with the new agent. Lots of changes are required for the basic skills. Currently we are able to move the agent's joints without falling.

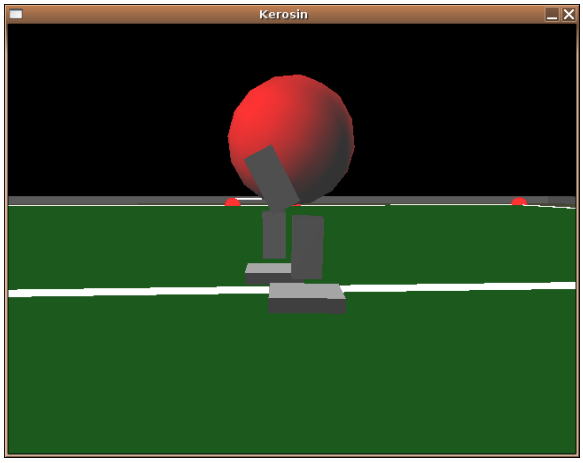

Fig. 4. Moving the agent's joints

There are some design issues that hardend agents movement. When an agent rises one of it's legges, because of the distance between it's mass center and it's other leg, the agent can't keep it's balance and will fall. Raising one leg is

required for walking because of the infinite friction. These are possible ways to ease agents movement:
- Decreasing the distance between agent's legges. Current distance is too much and causes the agent to fall when it tries to raise one of it's legges.
- Increasing contact area between agent's legges and ground.
- Adding arms to the agents which will allow them to keep their balance while rising one of their legges.

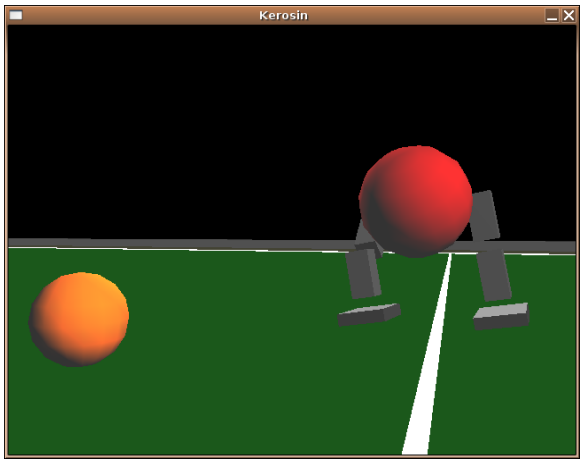

Fig. 5. Falling Agent

6 Future Works

The following is a brief list of our activities for development in the year 2005:

- As the legged agents are going to be used in 2007 3D soccer simulation competition, we are working on our basic skills. Currently we are trying to speed up agent's movements to avoid falling.
- Implementing the Global and local brains to selecting between available strategies and using SayEffector/HearPerceptor to syncronize decisions between all agents
- Customizing our positioning to handle new offside rule.

References

1. Dashti, H.T., Aghaeepour, N., Asadi, S., Bastani, M., Delafkar, Z., Disfani, F., Mam Ghaderi, S., Kamali, Sh., Pashami, S., Fotuhi, A., UTUtd2006-3D Team Description Paper (2005), http://www.fos.ut.ac.ir/robocup/download.htm

2. Dashti, H.T., Aghaeepour, N., Asadi, S., Bastani, M., Delafkar, Z., Disfani, F., Mam Ghaderi, S., Kamali, Sh., Pashami, S., Fotuhi, A., Dynamic Positioning based on Voronoi Cells(DPVC), Springer's Lecture Notes in Computer Science, http://www.fos.ut.ac.ir/robocup/download.htm
3. Riley, P., Riley, G.: SPADES A Distributed Agent Simulation Environment with Software-in-the-Loop Execution. In: Winter Simulation onference Proceedings (2003) 817–825
4. Kyrylov, V.,Brokenshire, D.,Hou, E., Optimizing Precision of Self-Localization in Simulated Soccer Agents
5. De Boer, R., kok, J., The Incremental Development of a Synthetic Multi-Agent System: The UvA Thrilearn 2001 Robotic Soccer Simulation Team.