Robocup 2008 – Soccer Simulation League
3D Soccer Simulation Team Description
IUST (IRAN)

Majid Yasari, Yousof Kanani, Arash Sahebolamri,
Vahid Garoosy, Naser Mozaiani , Majid Sadeghi Alavijeh

IUST Robotic Association Labs, Computer Engineering Department,
Iran University of Science and Technology, Tehran, Iran

{Majid.Yasari , Kanani.Yousef , Arash.Sahebolamri ,Vahid.Garoosi ,
NMozayani , Majid.Sadeghi.Alavijeh}@gmail.com

http://robotic.iust.ac.ir

Abstract. This essay is a quick overview of the agent made by the IUST 3D Soccer Simulation Team for participating in China Robocup Competitions 2008. After an introduction to the 3D Soccer Simulation League we will give a short description about our agent architecture and techniques that will be used in order to deal with several challenging problems in this field; balancing movements and making decision. We use some concepts such as Inverse kinematic, inverse pendulum, ZMP for making balanced movement and Fuzzy controller for reasoning and some advanced skills.

1 Introduction

Simulated environments are a commonly used method for researching artificial intelligence methods in physical multi-agent systems. Simulations are especially useful for two different types of problems:

1. To experiment with different sensors, actuators or morphologies of agents
2. To study team behavior with the set of given agents.

Additionally the connection between both types of problems is an interesting research Problem. 3D soccer simulation is a multi agent simulator that has been existed since 2004 in sphere mode that the principal's problem in it was corporation between agents but it underwent to major changes in RoboCup 2007 competitions and changed to humanoid robot simulation in which keeping motions balanced and stable, is the main problem. Because of the changes to the 3D simulator, we had to start our agent development from the scratch and try to find the how to perform basic skills such as walking, running, getting up, intercepting the ball, etc to implement our Agent's behavior. In the new server we face with 20 degrees of freedom (DOF) humanoid robot equipped with sensors such as Touch Sensor, Gyro Sensor, Collision Sensor, and Visual sensor.

This paper describes the main features and structure of our agent. After short description about our architecture, we explain some of important components of our agent such as World Model, Brain, and … .

2 Overview of 3D Soccer Simulation

3D soccer simulation developed last year to a more applicable approach for achieving the aim of RoboCup competitions. In the past years of competitions there have been matches between two teams that each one had 11 sphere agents, so the most important issue was to develop a team work between agents just like a 2D soccer simulation. The previous work is done by our team (Caspian) in the previous 3D soccer simulation server. There were low-level and High-level skills, which refer to all of agent's behavior independent from the others, like moving to a position and team work behaviors respectively; but from the Atlanta competitions sphere agents replaced by humanoid agents with 20 degrees of freedom. The humanoid robot was derived from the Fujitsu HOAP-2 robot model.

The humanoid robot has components that are assumed to be rigid bodies with a specific mass, and joints have been developed for connection between these components. Controlling of the robot is possible with the motors that apply torques on joints. The *Open dynamic engine* is used for simulation of humanoid robot where it's a high performance library for simulating rigid body dynamics. It computes dynamic and kinematic behaviors of multi-body robots with respect to torques of joints, gravity, friction, collision forces, etc.

There is a vision on the robot's torso that provides an unrestricted vision, in the other words all objects in the field are specified as points. All data is relative to the robot's position. Also server provides information from touch sensors on the feet, collision sensors and gyro rate sensors. Server simulates humanoid robot in each cycle, gives all sensory information to agents and receives information about torque of joints from agents, where each step is 0.02 seconds.

In this platform the main subject is to develop robot's skills like walking, getting up, kicking, etc. As it can be seen there has not been significant development on robots skills in 2007's competition. There is a major reason to this, teams were not familiar with the new 3D soccer simulation server and they didn't have enough time to develop high-quality skills for their robots. Nonetheless it is a big step to achieving high-capability humanoid robots. There is a big advantage of simulation; the development of humanoid robots can be done without wrestling with manufacturing humanoid robots and its costs and needs to high–tech in mechanics and electronics.

But it is expected that in the upcoming competition a great improvement on robot's skills will be achieved.

3 Agent Architecture

Behaviors of humanoid robots are based on the *agent architecture.* Robots need a good plan to do their tasks just the way that humans do. So designing agent architecture is the major step in development of humanoid robots. There are several decisions that can be made by robots respecting to input information. At interval of 0.02 seconds, the agent receives sensory information like vision, touching, etc. Hence, a good *insight* accomplishes from this information.

Vision sensor obtains clear information about objects as points that consist of distance and angel to the robot's coordinate system that is attached to its torso. Three flags on the field's corners are sufficient to realize the position of every object on the field on a universal Cartesian coordinate system. *World model* information is easily populated in this way. Then *brain* of robot makes a decision, like moving to a specified position, kicking ball to a desired direction, diving to left or right, etc. For doing tasks, initial and final situations play important roles. Brain must define a procedure from initial to final position. This procedure consists of different *skills* like "1st get up, 2nd walk towards ball, 3rd kick the ball". 20 degrees of freedom humanoid robots have non-linear dynamics that is very hard to control, even for stable walking. So these skills should be defined for agent. The specified cycle of limbs' motion can make different skills; these motions can be explained by *limbs behavior*. This information about joint are finally transferred to server by *connection*. The next cycle comes and some changes take place in the world model so situation changes; now there are two ways to go, first one is continuing the previous procedure, and second is making some other decision.

We believe that the development of modules that are briefly introduced here has the most important role in the humanoid 3D soccer simulation. These are our modules:

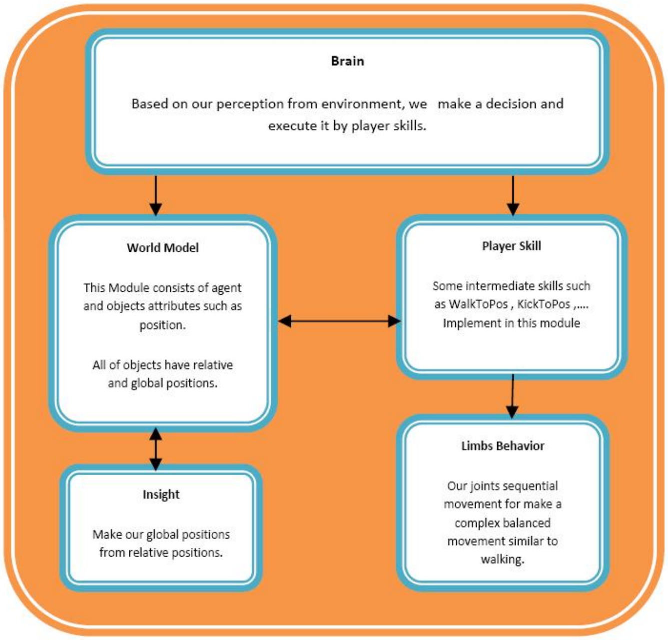

Fig. 1 Our Logical Agent Diagram

3.1 Brain

The Brain module is the main part of agent architecture that is categorized in high-level behavior. It should be considered that the brain is independent from the physical attributes, in this level we look at our robot as a sphere (similar to the sphere model in the old soccer simulation sever).

An example is given here for better describing of the brain behavior:

We assume that the ball is between our humanoids' robot and opponent team's robot, in this state, first, the brain of robot make a decision and ordains to *walk* towards the ball, if our robot reaches to the ball by using walking earlier than the opponent robot then the brain of robot makes a new decision again and ordains to kick the ball towards the goal, but if the opponent's robot reaches the ball earlier than our robot, the brain of robot makes a new decision according to this situation, for example in this situation brain of robot ordains to move to the new position, and intercept the ball from the opponent.

3.2 World Model

It is essential for an agent to have a world model in order to depict self information, field information and ball information; therefore it can perform intelligently in each situation. The world model is updated every cycle by incoming strings from the server which in turn is parsed by our parser. The agent can utilize this world model to make the best decision at that moment. Two important modules of world model are described below.

3.2.1 Localization

One of the main functions of world model is localization. Using the flags, we can find our position on the field, and once we know where we are we can easily calculate the absolute positions of other objects on the field.

3.2.2 Center of Mass (CoM)

Center of mass like localization is one of the most important functions of world model. According to the subject that will be described in *Player Skills* section completely, center of mass is very important for equilibrium of agent.

3.3 Player Skills

In the pitch, agents need to do some tasks. First and most important skill of biped humanoid robots is walking. Research on biped humanoid robots is currently one of the most exciting topics in the field of robotics and there are many ongoing projects [2, 3, 4]. After defining skills for humanoid robot, agents must be trained to find better ways to accomplish a desired skill. Skills developed in this project are: walking, turning, kicking, getting up, falling down (for goalkeeper only) that are described blow.

3.3.1 Walking Pattern Generation

From the viewpoint of control and walking pattern generation these works can be classified into two categories. The first group requires the precise knowledge of robot dynamics including mass, location of center of mass and inertia of each link to prepare walking patterns. Therefore, it mainly relies on the accuracy of the models [2, 3, and 5]. This group is named as the ZMP based approach since they often use the zero-moment point (ZMP) for pattern generation and walking control. The resultant force of gravity and inertia force are called the 'Total Inertia Force'. ZMP is the point where the line of action of the Total Inertia Force intersects with the ground surface [6].

Contrary, there is the second group which uses limited knowledge of dynamics e.g. location of total center of mass, total angular momentum, etc. Since the controller knows little about the system structure, this approach much relies on a feedback control. It's called the inverted pendulum approach, since they frequently use an inverted pendulum model .Using the first method is suitable for usual pattern walking, but it suffers with situation like a walking on stepping-stones. A novel walking pattern generation was introduced that allows arbitrary foot placements as a mixture of the ZMP based and the inverted pendulum based approaches [7].

In the walking pattern generation the main idea is obtaining ZMP in the polygon off agent's feet. If this event occurs then the agent's walking becomes stable. First statically walking is described as follow: if there is no big changes in the joint velocity, so the inertial forced can be neglected in the calculation, so just the gravity force affects in the robot's kinematics. The question is what positions are stable? For answering this question one needs to check gravity forces that acts on each body, and their reaction that applies from ground surface to robot's feet. The simplest approach can be treated: just one force acts on the center of mass (CoM) of agents that it's equal to agent's weight. If it intersects ground surface at point which is in the polygon off robot's feet, agent is stable, else this is an unstable position. But statically walking is very slow, so this procedure is not desirable.

Dynamic walking not only is influenced by gravity force, but also by inertial forces that becomes from bodies acceleration (linear or angular acceleration) and applied to CoM, so here the ZPM is major too.

Here, an approach develops for a simple inverse pendulum and then extends for multi-body agent. Identifying of ZPM has a straightforward calculation for inverse pendulum respect to its mass and CoM acceleration in each direction, i.e. ZPM could be found if the position and acceleration of CoM is known. But it's very confusing; agent's behavior must be planed in order to locate ZPM in specific area, but ZPM can be calculated if the behavior is known!

But there is a key. This problem generation of walking pattern can be an inverse problem of this, i.e. finding the CoM trajectory in order to attain a specific ZPM trajectory. This is shown in fig.2. The ZMP of the desired CoM pattern is called 'Desired ZMP'.

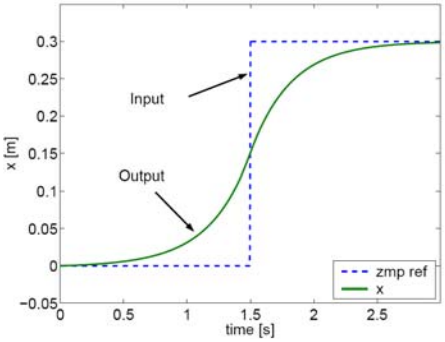

Fig. 2 Desired ZMP and essential CoM to provide this change.

As it can be seen in fig. 2 that desired ZMP has a step change in 1.5s but CoM must starts moving *before* this. A controller called "Preview control" utilizes future information was first proposed by Sheridan in 1966 [8].

A robust Fuzzy PID controller is developed to make good real-time generation of CoM's position. A preview controller must be adjusted with its preview time. Schematic of controller is shown in the fig. 3 Where P specifies the desired ZMP.

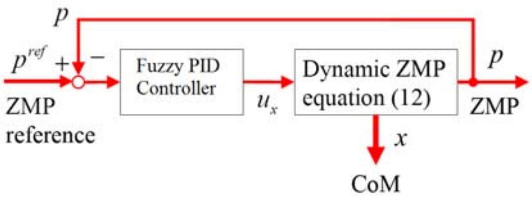

Fig.3 Close loop controller for following the desired ZMP.

But behavior of multi-body agent is not guaranteed by justifying this controller that is extended for lump mass. But as a simpler implementation CoM can be considered in a specific location of multi-body robot. But some tuning must be applied in the preview control system such as changing preview time of future information.

So walking path is generated with respect to desired ZMP, here is an example that shows how desired ZMP can be generated. When a robot wants to walk, in each step it has one of its feet as a support foot, so ZMP must be in the area of those foot, i.e. ZMP must be moved toward that foot as fig.4 shows.

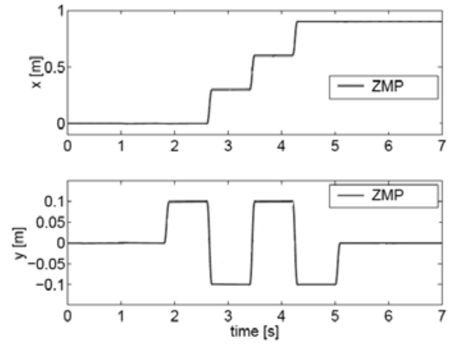

Fig.4 Example for Desired ZMP

Walking is not complete yet, the robot needs to know that how it should change its limbs. With a known position of CoM, changes from current position to position that provides new Com position is not known. First from a look up table the next position is found, but many situations between these two positions can occur, for choosing the trajectory of joint in the interval of current and next position “inverse kinematics” is used.

3.3.2 Kick to Specified Position

In this sub problem we want to shoot the ball to a specified position. For solving this problem we designed a fuzzy controller with six rules that is shown below:

Fig. 5 Fuzzy Controller structure

According to the shown figure our controller consists of three input variables that are described below.

Table 1. Our Fuzzy Controller Variables

Variable	Description
BallTargetDist	Ball distance from specified position
DiffAngle	Different Angle between Robot To Ball Vector and Ball to destination Vector
RobotBallDist	Robot Distance from the ball

Our Fuzzification step is shown below:

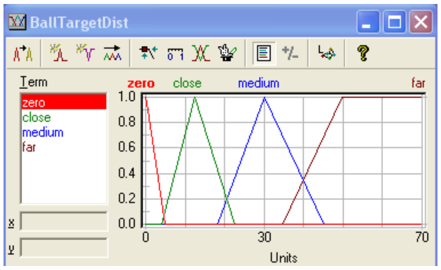

Fig. 6 *BallTarget* membership function

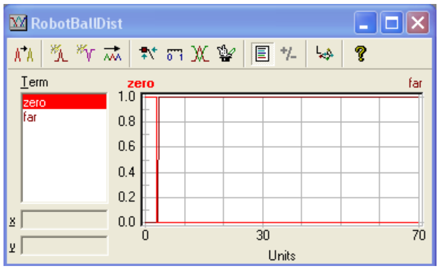

Fig. 7 *RobotBallDist* membership function

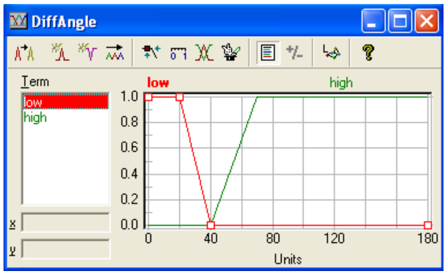

Fig. 8 *DiffAngle* membership function

Output of our fuzzy controller for this module is a floating number that describes the appropriate power for kicking the ball to the desired position that is specified by the upper level.

4 Future Works

To reach adaptive solutions in dynamic environments which conditions varies time to time, we need adaptive methods that can be combined with our current solutions and improve them as more as possible. Using r*einforcement learning* and *genetic algorithms* in our solution frameworks are our main goals in the future.

References

1. B. Slamet and M. Pfingsthorn, Manifold SLAM: a Multi-Agent Simultaneous Localization and Mapping System for the RoboCup Rescue Virtual Robots Competition, Master's thesis, Universities van Amsterdam, December 2006.

2. Nishiwaki, K., Sugihara, T., Kagami, S., Kanehiro, F., Inaba, M., and Inoue, H., "Design and Development of Research Platform for Perception-Action Integration in Humanoid Robot: H6," Proc. Int. Conference on Intelligent Robots and Systems, pp.1559– 1564, 2000.

3. Gienger, M., et.al, "Toward the Design of a Biped Jogging Robot," Proc. of the 2001 ICRA, pp.4140– 4145, 2001.

4. Inoue,H., Tachi,S., Nakamura,Y., Hirai,K., Ohyu,N., Hirai,S., Tanie,K., Yokoi,K. and Hirukawa,H., Overview of Humanoid Robotics Project of METI, *Proc. of the 32nd ISR*, April, 2001.

5. Huang, Q., Yokoi, K., Kajita, S., Kaneko, K., Arai, H., Koyachi, N., Tanie, K., "Planning Walking Patterns for a Biped Robot," IEEE Trans. on Robotics and Automation, Vol.17, No.3, June 2001.

6. Toru Takenaka, "The Control System for the Honda humanoid robot", Oxford University press, age and ageing 2006; 35-s2:ii24-ii26

7. Shuuji KAJITA, Fumio KANEHIRO, Kenji KANEKO, Kiyoshi FUJIWARA, Kensuke HARADA, Kazuhito YOKOI and Hirohisa HIRUKAWA, "Biped walking Pattern Generation by using preview control of zero moment point" ,IEEE , Internationl Conference of robotic and automation, September 2003.

8. Sheridan, T.B, "Three Models of Preview Control," IEEE Transaction on Human Factors in Electronics, 7-2, 1966.