

# JMU 3D Team Description 2008

Gang Wang<sup>1</sup> , LaiZhihui<sup>1</sup> CaiHuaxu<sup>1</sup>, YangZhou<sup>1</sup>

<sup>1</sup>Compute Engineering College of JiMei University XiaMen City

FuJian China

[wangweb@sina.com](mailto:wangweb@sina.com)

**Abstract.** As our friendly organizing committee say in the 3d rule page of RoboCup 2007 Atlanta: This year we are really behind schedule! . However, thanks to the uni-koblenz University 's great work and all of the simulation 3d teams ' efforts, so that we can see the humanoid simulation 3D robot play on the soccer field, although it still has several bugs. This page is mainly introduce the JMU Humanoid Robot Soccer Simulation 3D League 's WorldModel, Positioning, Balance Control and some complex behavior of the attacker.

**Keywords:** humanoid worldmodel balance control climbing up

## 1 Introduction

JMU Simulation 3D Team was created in 2006, and join the RoboCup China Open 2006 and 2007. We also attended the RoboCup 2007 Latin American Open in remote way . During laster year's competition, we mainly focused on the humanoid robot's walk and stand up behavior under the previous version simulation server . And this Page is mainly describe the JMU3D agents' WorldModel, Positioning ,Balance, and Complex Behavior of the Attacker.

## 2 World model:

As we know, the rcserver3d-0.5.6's field is about 50M Length and 32M width . Agent can calculate their position every 0.02 second by analyzing the eyesight message and joint message received from the server . Now the robot can see 8 Reference Objects which includes 4 Flags and 4 Goal posts . Through the Polar Coordinate values of the 8 objects , we can calculate agent's absolute coordinate in the field. Now we are using the following regular:

If(Left\_FRP\_Force > Right\_FRP\_Force && body\_state!=Lie\_Down) we can caculate my\_pos\_z as:

My\_pos\_z=((Foot\_High+Shanks\_High\*cos(angle\_left\_leg\_5)\*cos(\*cos(angle\_left\_leg\_6)+ Thighs\_High\*cos(180-angle\_left\_leg\_4 - angle\_ left \_leg\_5 ) \*cos (angle\_left\_leg\_6) + Torso\_High \*cos(angle\_left\_ angle\_4 + angle\_left\_angle\_5 - angle\_left\_leg\_1)/2))

We can replace left\_angle with right\_angle when Left FRP Force < Right FRP Force.

After known the high of the camera ,we can calculate the distance from agent's feet to the reference object. e.g

My\_Horizontal\_distance\_to\_FLAG:

$$= \sqrt{\text{My\_distance\_to\_Left\_1\_Flag} * \text{My\_distance\_to\_Left\_1\_Flag} - \text{my\_pos\_z} * \text{my\_pos\_z}}$$

my\_pos\_x1=(dis\_left\_1\_flag\*dis\_left\_1\_flag- dis\_left\_2\_flag \* dis\_left\_2\_flag ) /96.0;

my\_pos\_y1=-(dis\_left\_1\_flag\*dis\_left\_1\_flag-is\_left\_2\_flag\*dis\_left\_2\_flag)/62.0;

$$\text{my\_pos\_x} = \sum_{i=1}^8 (\text{my\_pos\_xi})$$

$$\text{my\_pos\_y} = \sum_{i=1}^8 (\text{my\_pos\_yi})$$

The body direction also can be calculated base on agent's absolute position, and agent's body direction angle to the Flag or Goal Post. After working-out the agent's position and direction ,we can calculate the ball's absolute position in the field. So far the relative distance and angle to the ball is very important, and fortunately, it can be generated by the regular:

Distance\_to\_ball\_horizontal=distance\_to\_ball-my\_pos\_z

Rel\_dis2ball\_x=distance\_to\_ball\_horizontal\*sin(angle\_to\_ball\_ horizontal);

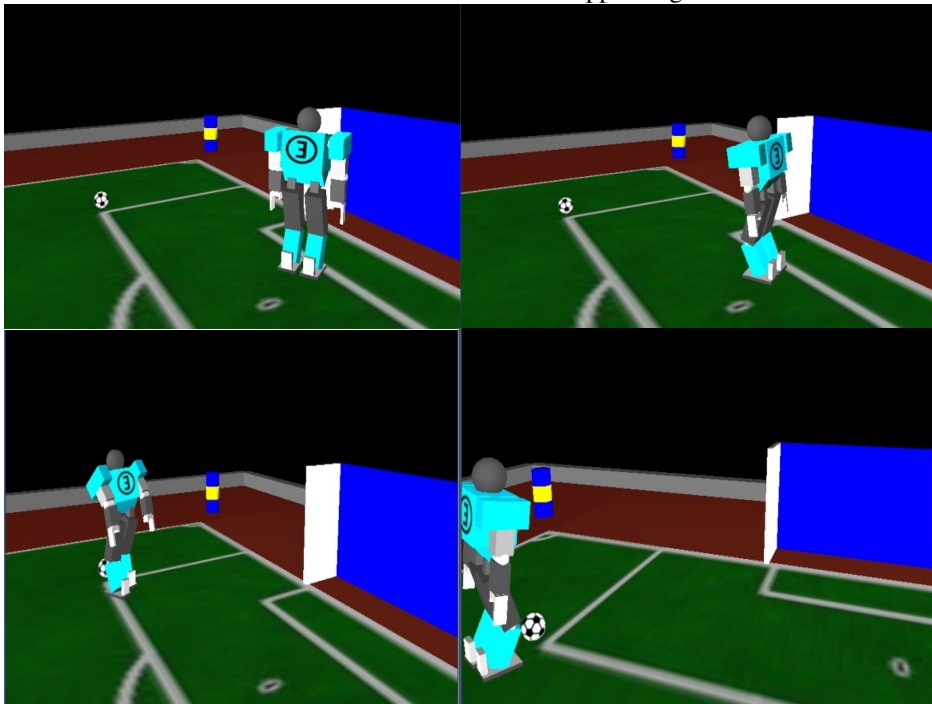
Rel\_dis2ball\_y=distance\_to\_ball\_horizontal\*cos(angle\_to\_ball\_ horizontal);

### 3 Balance Control

The Balance Control is the base skills for other motion such as walk, turn, kick, etc. Currently, agent can feel its balance just through the FRP value. If the message from server didn't include the FRP value, it indicate that the agent is flying in the sky, and we can know that is unstable, and the agent should take some measures, such as stop currently motion, or climb up. If the FRP force effect on the front of feet for a short time, we can know that it maybe falling down forward, so we have to rotate agent's hands up to save more time when it climb up.

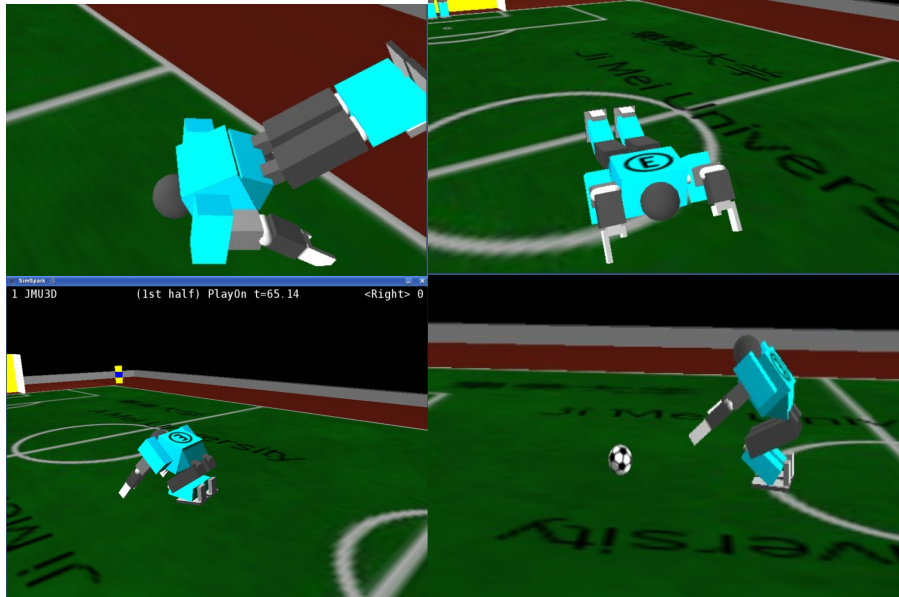
### 4 Integration Behavior

After get a balance pose, agent have to do a lot of action, currently we just have two agents each team, so the passing and formation skill is no yet developed. So far, the JMU3D Attacker can implement the following actions: turn left/right, walk forward, move body left/right, turn around ball, kick ball, and climb up. These actions can ensure the agent have a good performance in the field. The following picture will show you how does JMU3D attacker find ball and kick ball into the opposite goal.

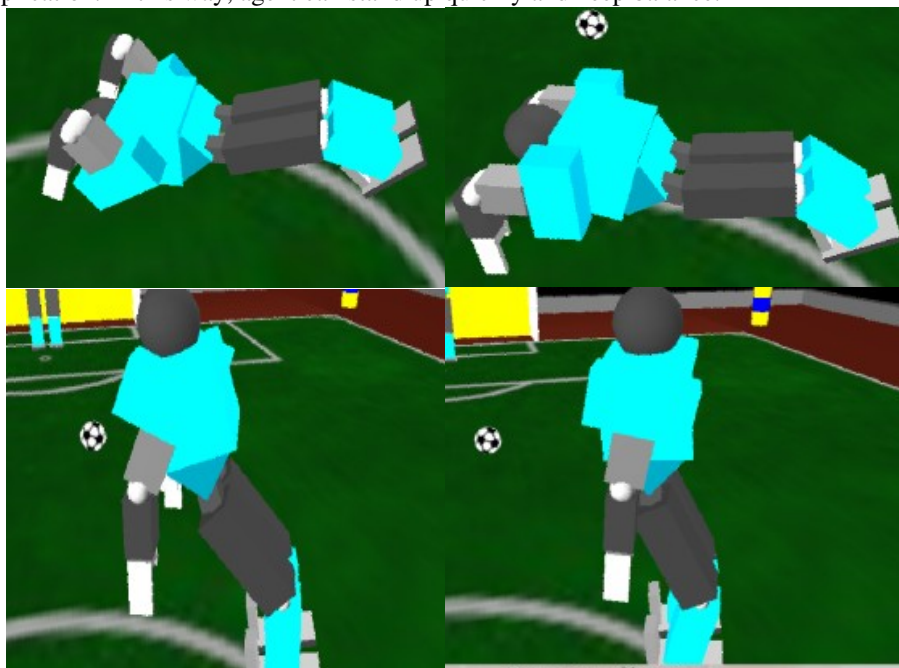


## 5 Feature

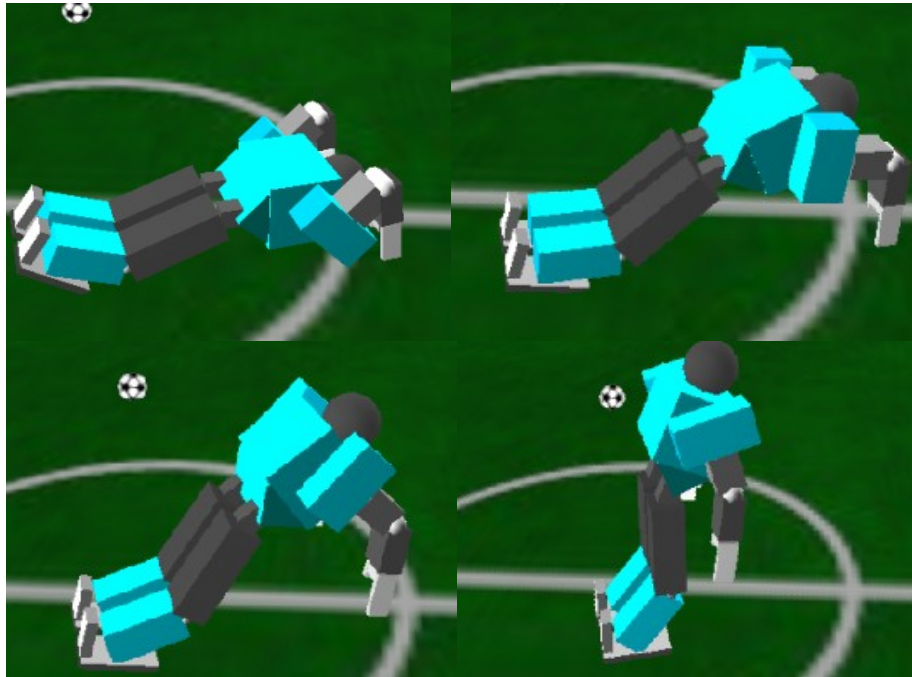
Although JMU3D is young, it still have some features. One feature is its climbing up action. As we know, if agent is lying down on the ground, it may have two situations, one situation is facing up to the sky, another is facing down to the field. And that cause we have to climb up by two different ways. And we know a human can climb up more stably when his hands are in front of his feet. In JMU3D, we can make the agent reverse 180 degree when the first situation is happen, by rotate both hands. And then we can climb up by gather up our body and make the barycenter locate in the center of agent's feet, then climb up slowly. In that case agent can stand up quickly even when lying on the ball and can get more chance of controlling ball.



Owing to our hard work, the climbing up action is becoming easier and faster. When facing up to the sky, agent rotate both hands. And then we can climb up by forcing application. In this way, agent can stand up quickly and keep balance.



When facing down to the field, agent can use the same method to stand up.



JMU3D has another feature which is its kick skills, agent can kick ball quickly without stand up, and can prevent kick to field directly, so it can get a perfect kick result. So far JMU3D attacker can kick in twenty balls averagely during each game, without enemy.



## 6 Conclusion:

During the pass half year, we have get a great progress in Humanoid Simulation 3D, mostly in the base skills and worldmodel construction. In the next period we'll focus on the runing action and catching ball action of goalie. We will also research on the simulation server especially on the ode problem, to make the server run more smoothly. For making sim3D more popular, we decide to build a monitor spark under MS Operating System environment. We believe that under the OC/TC members' great work and all the Robocup simulation 3D teams' effort, the Sim3D will get more progress in the coming year.

## References:

1. Hirchika Inoue: Dynamically-Stable Motion Planning for Humanoid Robots[J] Auton Robots 2002
2. Goldberg K Y, Raibert M H, Conditions for symmetric running in single—and double-support[C]//Trans IEEE Conference on Robotics and Autommation, 1997.
3. R . Murphy . MIROSOT97-Robotics Competition Corner[J] . Robo-tics and Autonomous Systems 1998
4. Fujitsu Automation Co., Ltd.: MIINIATURE HUMANOIID ROBOT 「HOAP-2」 IINSTRUCTION MANUAL
5. M. Eaton and T. J. Davitt, "Evolutionary control of bipedal locomotion in a high degree-of-freedom humanoid robot: first steps," Artificial Life and Robotics, Vol. 11, No. 1, pp. 112-115.
6. T. Yanase, T. Iba. Evolutionary Motion Design for Humanoid Robots, GECCO'06, July 8–12, 2006, Seattle, Washington, USA