

Robogulf 3D Soccer Simulation Team Research Proposal

Moein Alinaghian, Seyed Salim Tabatabaei, Jawad Aref Alchikh, Amir Hossein Sadeghi

Ajman University of Science and Technology, UAE
robocup@linux-dubai.com
<http://robocup.linux-dubai.com>

Abstract. This paper is including two sections which our team has plan to research about them. First section is the usage of Martial Arts to make movement models for Humanoid Robots and providing much stable and smooth movements in the Simulation Server. Section two is about Joint's Angle Correction System to keep the robot balanced and making mathematical movement models.

1 Introduction

Humanoid is one of the most complex and attractive fields in robotics, and need a highly integrate team work. Unfortunately many teams can't participate in this league because of it's high expenses or lack of team member in a specific field like mechanics.

This year, by changing the 3D Soccer Simulation Server, participants have this chance to research on Humanoid Robots, too. It's an additional feature to high level behaviors, and the result can be used in Physical Humanoid Robots later, with less expense and without need to have a huge team contains a lot of members.

Regarding to the current server developing status, we may have some changes in our research areas, but we expected to have these majors.

1. Usage of Martial Arts in Humanoid Robots Movements
2. Joint's Angle Correction System to keep the robot balanced

Also, this project is already accepted as a Graduation Project by Ajman University.

2 Usage of Martial Arts in Humanoid Robots Movements

Most of the Humanoid Robots Movement Models are simulated from the normal human movement behaviors. But the human movements itself could be improved and become more stable using some sports like Martial Arts or Parkour.

They help to have a better balance, and some quicker actions, and some of them can be used in Humanoid Robot Movements, and the result is to have a better balance, and decrease of error effects.

2.1 Martial Arts Majors and Fields

Most of the fields of Martial Arts have two main approaches.

1. Increasing the muscles power
2. Learning and Using Special Techniques

The first aim is not useful for Humanoid Robots since they have motors with a constant power. To increase their motor power, motors should be changed!

The second aim is exactly what we need. Techniques help us to have more stability and balance and having higher speed and strength by using of more body parts together and use of result movement inertia with the same motors.

As an example, Wing Chun could be a good choice, since its most concentrate is on techniques and some of its movements like standing up can be used to create a Movement Model.

2.2 Process

In the first step, we are looking for the useful movements which can be used in the humanoid robots.

In the next step, we make the Base Movement Model of them, and will test them in the Simulator to compare the result with normal behaviors.

Next step is collecting all of the Movement Models and provide them as a research paper.

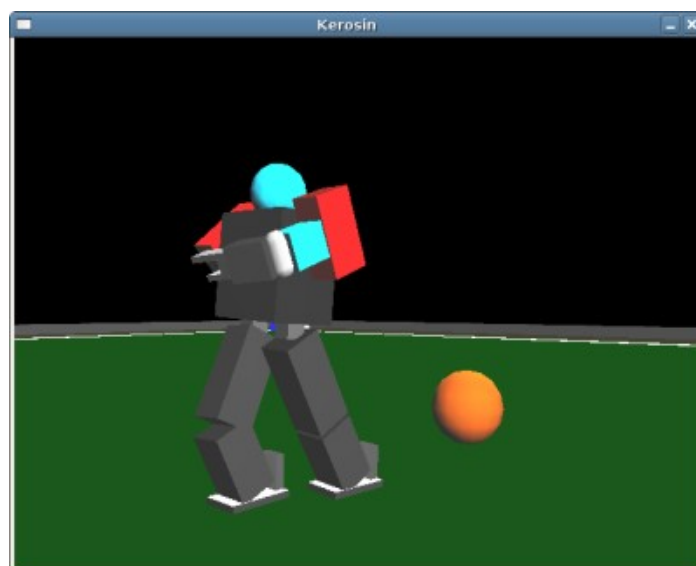


Fig. 1. "Gung Bo" Stance – Chinese Kung-Fu

3 Localising

The other thing that we are working on and is to be continued is a method to get precise location and direction of ourselves. The algorithm to be implemented is based on Kalman Filter Algorithm, but it's going to be modified to reduce its calculations (in case there are multi-agents of each team, it might have a huge overhead)

4 Joint's Angle Correction System to keep the robot balanced

To have an action, we should set the joint's angles in each step. Many times, the static rates make the robot unstable, because it's hard to calculate the exact angle of each joint for each step, and with decreasing the steps, errors also will be increased, and we get falls.

To solve this problem, we are working on a correction system. It's going to be a variation of Newton and quasi-Newton optimization algorithms.

It will help us to train our agents, and fix the angles and after this, robot will be balanced.

4.1 Example Usage

In the initialization, we should specify the Joints Relations. It's not needed to specify exactly the relations, and just grouping will be enough. The rest of calculations will be done using the neural network of this group which will be updated using the sensing information from the server, including touching and gyro sensors.

As an example, to define the relation for the joint “rleg_joint_5_6_2” of soccerbot, we should write the following code.

```
extern SoccerBot SB;
JointRelation* horGroup = new JointRelation(4);

horGroup.LoadNeural("horGroup.txt");
horGroup.Joints[0] = &SB.RLEG_5_6_2;
horGroup.Joints[1] = &SB.LLEG_5_6_2;
horGroup.Joints[2] = &SB.RLEG_2_3_2;
horGroup.Joints[3] = &SB.LLEG_2_3_2;
```

The angle of “rleg_joint_5_6_2” is directly related to these other joints. Also we should have some other classes for non-direct related joints. For example, “rarm_joint_1_2_2” is directly related to it's angle, but it's important to make the robot balanced.

Now, it can loads a free neural network, and after some steps it will update the neural network, and can save them in the same file. After some tries, we can use the result to fix the movements in robot steps.

By the way, it's just a simple view of this system to describe it, and it will be more complex, and will be dependent on the sensors which will be added later to the server.

4.2 Summery

You just need to specify the base angles for each steps, and have a trained neural network, and the rest will be done by correction system.

By adding some other functions, we can export the fixed base angles, and use them as a mathematical movement model to be used in physical humanoid robots, or Simulation Server without overhead process.

5 Refrences

1. Lee, S.-H.; Kim, J.; Park, F.C.; Kim, M.; Bobrow, J.E., Newton-Type Algorithms for Dynamics-Based Robot Movement Optimization
2. Joe Jones, Daniel Roth, Robot Programming : A Practical Guide to Behavior-Based Robotics
3. HRP-2 Humanoid Robot Website,
http://c110vh5n.securesites.net/global/ams/hrp_2.html