ROBOPUB+: 2010 RoboCup 3D Soccer Simulation Team Description

Özlem Öztürk, Erdi Ölmezoğulları, Petek Yıldız, Semih Zenciroğlu

Dokuz Eylül University, Computer Engineering Department, İzmir, Turkey
ozlem.ozturk@cs.deu.edu.tr, erdi.olmez@gmail.com,
petekyildiz@gmail.com, semihzenciroglu@gmail.com

Abstract. This paper is about the developments made over the last year's RoboPub team and future thoughts about team's improvement for attendance in RoboCup 2010 3D Soccer Simulation League.

Keywords: 3D simulation, robot behavior, control panel.

1 Introduction

The RoboPub Team was founded in summer of 2008 and first attended RoboCup 3D Soccer Simulation League in 2009. The team was able to reach the 3rd round in this tournament. This year, our goal is to develop the last year's team into a more talented and challenging one. So, at first we wanted to learn the agent's behaviors and look them up from the Simspark Manual. Later we developed a tool to see how the joint movements affect the agent [1], [2].

This tool will be mentioned later. After the joints' effects on agent's movement are detected we decided to complete primitive motions until the Team Qualification Deadline. In later sections the development of these steps will be explained.

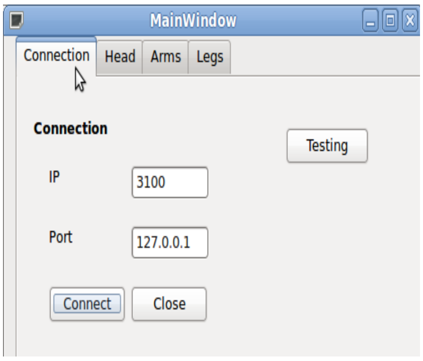
2 Robot Control Panel

The Qt framework that is a GUI development tool that works on the Linux environment is used while building this panel. The panel is designed to control fundamental robot acts and behaviors. The advanced robot acts are developed after joint angles' effects on robot behavior are watched here. The developed movements that are made by using this panel can be seen when it is implemented in Simspark server [3].

The panel has 4 sections that are;

- **Connection** : is provided server- client communication with TCP/IP.
- **Head** : controls robot's head. It can be moved left-right, down-up.
- **Arms** : controls robot's arms.
- **Legs** : controls robot's legs.

The sections are shown as below figures:

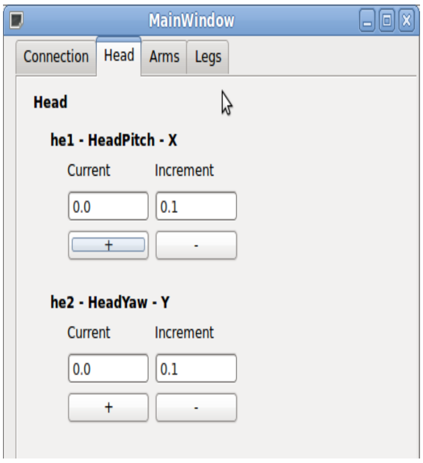


Connection

IP 3100

Port 127.0.0.1

Fig. 1. Robot Control Panel – Server Connection



Head

he1 - HeadPitch - X

Current 0.0 Increment 0.1

he2 - HeadYaw - Y

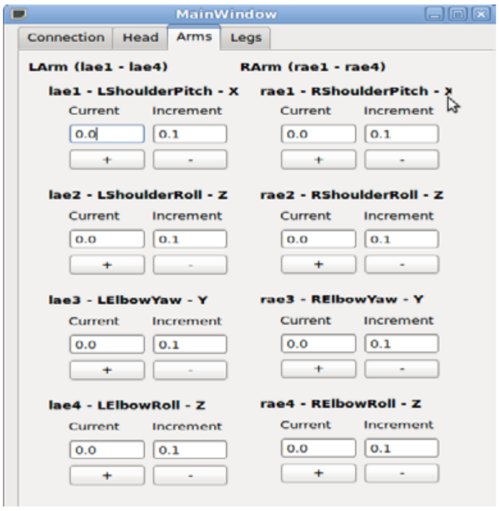
Current 0.0 Increment 0.1

Fig. 2. Robot Control Panel – Robot's Head joints

According to this control panel, robot acts can be generated and we can observe these acts at desired directions on the simulator

3 Developments

Last year's team's walking pattern was so unbalanced and not dynamic. So we made some configurations for removing this imbalance. Joint angles during walking pattern are improved by applying sinusoidal functions best fit. We have removed sudden frequency changes that may cause imbalance [4], [5], [7].



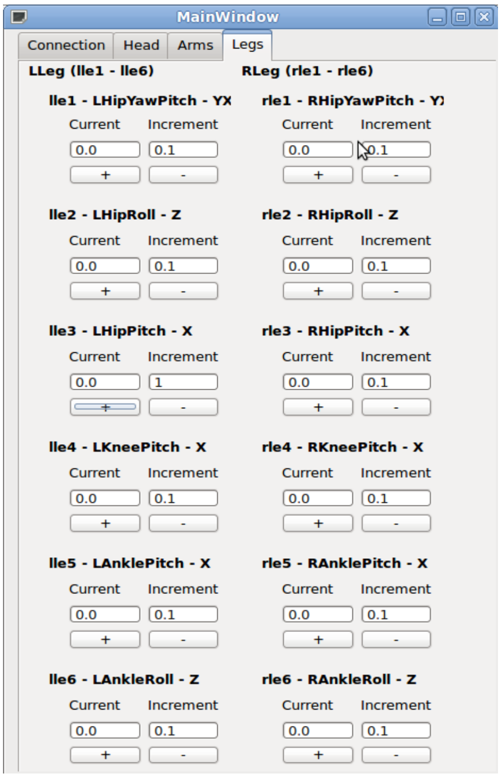
LArm (lae1 - lae4)

	Current	Increment
lae1 - LShoulderPitch - X	0.0	0.1
lae2 - LShoulderRoll - Z	0.0	0.1
lae3 - LElbowYaw - Y	0.0	0.1
lae4 - LElbowRoll - Z	0.0	0.1

RArm (rae1 - rae4)

	Current	Increment
rae1 - RShoulderPitch - X	0.0	0.1
rae2 - RShoulderRoll - Z	0.0	0.1
rae3 - RElbowYaw - Y	0.0	0.1
rae4 - RElbowRoll - Z	0.0	0.1

Fig. 3. Robot Control Panel - Robot's Arms joints



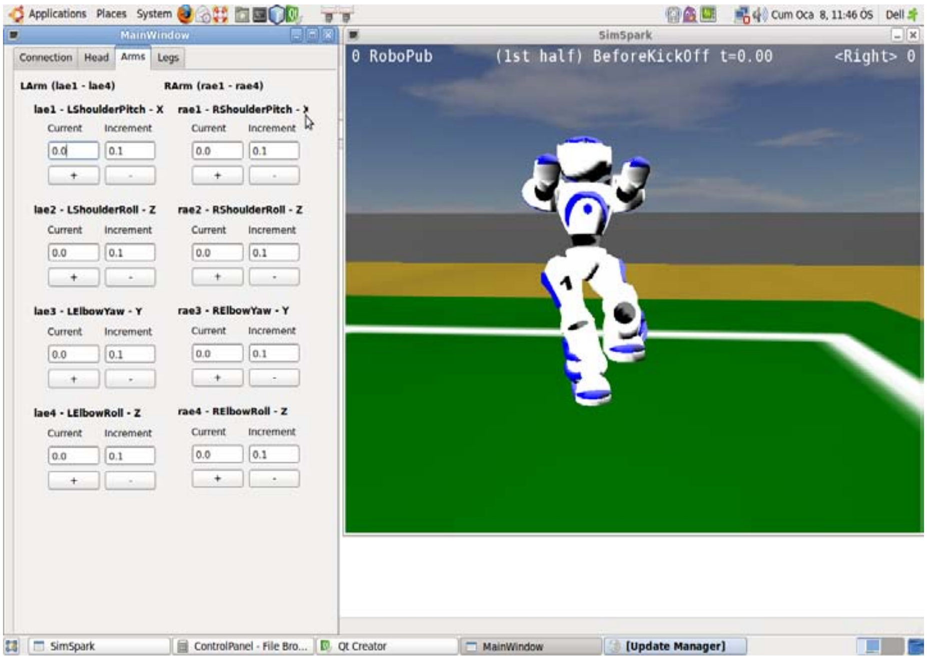
LLeg (lle1 - lle6)

	Current	Increment
lle1 - LHipYawPitch - YX	0.0	0.1
lle2 - LHipRoll - Z	0.0	0.1
lle3 - LHipPitch - X	0.0	1
lle4 - LKneePitch - X	0.0	0.1
lle5 - LAnklePitch - X	0.0	0.1
lle6 - LAnkleRoll - Z	0.0	0.1

RLeg (rle1 - rle6)

	Current	Increment
rle1 - RHipYawPitch - Y	0.0	0.1
rle2 - RHipRoll - Z	0.0	0.1
rle3 - RHipPitch - X	0.0	0.1
rle4 - RKneePitch - X	0.0	0.1
rle5 - RAnklePitch - X	0.0	0.1
rle6 - RAnkleRoll - Z	0.0	0.1

Fig. 4. Robot Control Panel - Robot's Arms joints



LArm (lae1 - lae4)

	Current	Increment
lae1 - LShoulderPitch - X	0.0	0.1
lae2 - LShoulderRoll - Z	0.0	0.1
lae3 - LElbowYaw - Y	0.0	0.1
lae4 - LElbowRoll - Z	0.0	0.1

RArm (rae1 - rae4)

	Current	Increment
rae1 - RShoulderPitch - X	0.0	0.1
rae2 - RShoulderRoll - Z	0.0	0.1
rae3 - RElbowYaw - Y	0.0	0.1
rae4 - RElbowRoll - Z	0.0	0.1

```
0 RoboPub        (1st half) BeforeKickOff t=0.00        <Right> 0
```

Fig. 5. Control Panel with SimSpark Simulator

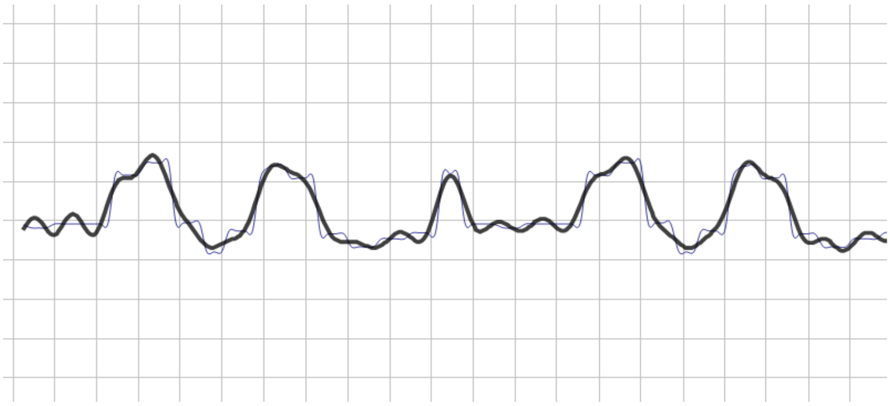


Fig. 6. Sinusoidal Function Graph for 3rd joint of right leg [6]

Also the get up pattern is updated. In addition giving example for being not dynamic, when the agent had fallen down, the walking pattern was continuing and our team was late for reacting the events that are happening during the game. This should be handled [7].

Also, we can detect coordinates of the ball and the agents that are in the vision angle. That is calculated by using the static flag positions. Some distance calculations are done as distance between agent and the ball, agent and opponent goal or two agents which are necessary for developing the playing strategy.

4 Developments

As it’s mentioned previously, the movement of our agent is developed from the last year’s team and it still doesn’t work dynamically. We will be studying on making our application in some kind of dynamic way. This is important for reacting the events that are happening in the game in the real time. Without being dynamic these possibly belated reactions may not be meaningful and may ruin all playing strategy.

On the other hand, of course there is ‘the decision side’ of the game. As we can detect the coordinates of ball, the opponent agents and our agents that are in the visual angle of the agent, we can decide what to do.

As soon as our agent has got the ball, our aim is heading directly to the opponent’s goal quickly and more stable. When it gets into the appropriate range for shooting the ball, it will try to shoot and score. If it falls down when headed to opponent side, the other agent (the one that is not goalkeeper) will head for the loose ball for earning it when it sees it’s teammate has fallen down and lost the ball.

When the opponent has the ball, the one that is the nearest from this opponent agent will be headed for stealing the ball from it. Again, if it falls down, the other will make its move for the ball.

When the ball is in loose state, again in the same manner the nearest agent will make its move for gathering the loose ball.

Last year our team was newly founded and so it was our first attendance. We didn’t win a good position at the end of the tournament, maybe because we were inexperienced. This year we expect to correct and develop the defective and insufficient parts from last year until the beginning of the tournament and win a better a position.

5 References

1. http://simspark.sourceforge.net/wiki/index.php/Users_Manual
2. http://wiki.robocup.org/wiki/Main_Page
3. http://qt.nokia.com
4. Shuuji Kajita, Fumio Kanehiro, Kenji Kaneko, Kiyoshi Fujiwara, Kazuhito Yokoi and Hirohisa Hirukawa, "A Realtime Pattern Generator for Biped Walking", Proceedings of the 2002 IEEE International Conference on Robotics & Automation Washington DC, May 2002
5. Patrick Lam, "Walking Algorithm for Small Humanoid", Master of Computer Science Center for Image Technology and Robotics Department of Computer Science University of Auckland, New Zealand, 2002
6. http://www.uniphiz.com/findgraph/
7. Robopub 2009 Robocup 3D Soccer Simulation TDP