UNSW COMP3431: Robot Architectures S2 2007

Assignment #1

Ball Tracking and Life Saver Challenge

Due: Start of Lab, Week 6 (1pm, 29 August 2007)

1 Overview

This goal of this assignment is to give you broad familiarity with writing behaviours in the rUNSWift architecture. In particular, in this assignment you will write behaviours to track and chase a normal orange RoboCup ball.

Students will perform this assignment in small groups of 2 or 3. You are requested to work closely together on the project, developing the code together (rather than using the "divide and conquer" technique of splitting the project into parts, developing the parts separately, and then recombining).

1.1 Use of current rUNSWift software

You already have access to the source code of a complete soccer playing robot, including behaviours to complete most of this assignment. For this assignment you are free to use any base-station code, any C++ code on the robot (accessed through the VisionLink module) and the basic python behaviour files on the robot listed here:

```
PyLib/*
PyCode/Action.py
PyCode/Behaviour.py
PyCode/Constant.py
PyCode/Debug.py
PyCode/Global.py
PyCode/hFrameReset.py
PyCode/hMath.py
PyCode/hPauseMotion.py
PyCode/hPWalk.py
PyCode/hWalk.py
PyCode/hPSensor.py
PyCode/Indicator.py
PyCode/Packet.py
PyCode/pDoNothing.py
PyCode/pInitial.py
PyCode/pReady.py
PyCode/pSet.py
```

Note that parts of other files may be called by those files, and in particular Global.frameReset() calls HelpTrack.determineBallSource(). You should not add further references to other files, and you should look at the above code path to check that it does what you want if you use the global ball location variables.

You are encouraged to read the other python files, but please, no direct copying.

1.2 Other Requirements

Students are asked to avoid the use of classes. In particular, students should not use classes for singleton objects. If you don't know what a singleton object is, avoid using classes entirely. You are encouraged to use python modules for encapsulation if necessary.

All behaviours should work both with and without other robots on the field, and with the robot wearing either colour uniform.

1.3 Deliverables

Parts 2 and 4 of this assignment require you to develop behaviours on the robot. These behaviours will be handed in as follows: Before the start of the lab when the assignment is due, each group should have checked into the Subversion repository (`https://roborouter.cse.unsw.edu.au/svn/comp3431/users/`) a branch containing their code. This branch should use Subversion correctly so that `svn diff` returns a reasonably sized output. Each group should also have a checked out working copy of their code with no local modifications on one of the lab machines. The code in that working copy should be compiled and ready for installation on a memory stick. The compiled code should have been tested on the robot before the class starts.

During the lab we will make memory sticks from those working copies. The solutions will be compared and graded. Part 2 requires the robot to stand still and only move its head. Part 4 requires the robot to chase the ball. Each of the behaviours should have its own python module usable by the `spip` script. You are not required to modify the standard 'ready' or 'set' states.

The report in part 5 should be an electronic file checked in to the same branch as your code. It should be in an easily readable format, *e.g.* plain text, html, PDF or postscript.

2 Ball Tracking (5 marks - graded in lab)

The first part of this assignment is to write a behaviour that tracks the ball. The robot should stand still. If the robot cannot see the ball, the robot's head should move smoothly about looking for the ball. If the robot can see the ball, then the robot should look at the ball.

The robot should track the ball in such as way as to be able to calculate the distance to the ball. The robot's head should track smoothly without jerks or oscillation (small oscillations caused by Sony's low level control are allowed, but should be damped where possible). The robot should track the ball with the ball in as many locations as possible relative to the robot. The robot's head should not touch the ball.

Students are required to make the robot track the ball using at least two different techniques. Their grade in this section will be on the quality of one of the techniques chosen by the group before the lab. A comparison of the techniques is part of part 5 of the assignment.

3 Ball Searching

In this part of the assignment, students have to use the robot's head and legs in co-ordination to find the ball. This part of this assignment will not be graded separately, but is a recommended step on the way to the next part of the assignment.

There are two scenarios that need to be considered here. In the first, initially the ball is held off the field. The robot should move about looking for the ball. The ball will then be placed on the field. The robot should find the ball and start tracking it as soon as possible.

In the second scenario, the ball is already on the field. It is then moved slightly (either quickly or with some occlusion so that tracking is disrupted). The robot should re-acquire the ball as soon as possible. A single behaviour should work for both scenarios.

4 Ball Time Trial (5 marks - graded in lab)

In this part of the assignment, the robot must do more than track the ball – it must walk towards the ball and touch it with its front legs. The goal is to reach the ball as fast as possible. Again there are two scenarios; one where the ball has been unseen for a while, and one where the ball was seen recently.

5 Report (5 marks - handed in at start of lab)

For this section each group should write a short report. This report should describe the two ball tracking techniques implemented in part 2. It should describe the strengths and weaknesses of each technique.

The report should also describe the solution to part 4. What are the strengths and weaknesses of your solution?