1 Dates

- Assignment 5 project demonstrations in class: Wednesday, April 22, 2009
- Assignment 5 handin due: Wednesday, April 22nd

2 Introduction

Your previous soccer-playing robots required some notion of state in order to function, such as robot pose, a map, ball location, etc. State information, stored explicitly in variables, gave your controller a sense of context and immediate history about the robot's current situation. Decision making at any moment in time is informed by state information in order to deliberate and plan into the future or take immediate action towards some future benefit. State variables help our robots to be adaptive towards unexpected changes in the environment and flexible to new scenarios[1]. However, estimation of such variables comes at the cost of requiring greater computation, latency in decision making, and overall price tag for the robot.

Consider the case where you want to commercialize a soccer playing robot for a mass market[2]. Such a robot should not cost much money to purchase, on the order of the $150 Roomba Red or $250 Nintendo Wii. Further, every additional component added to the robot, such as memory, processing, sensing, and actuators, increasingly cuts into your profit margin. In such scenarios, reliance upon state variables costs you money[3] and thus, you might be inclined to minimize the number of these variables.

Using reactive control, your robot controller in this assignment will rely upon no state variables and simply react to what can be sensed at the current instance of time. In contrast to the deliberate intentionality of planning, reactive approaches rely upon **emergent behavior** where the "intelligence" of the robot is both a factor of its decision making, the properties of the environment, and the dynamics of controller-environment

[1]Learning aims to provide a greater level of adaptability such that the robot has long-term context and history towards learning from experience. It could roughly be thought of as never making the same mistake twice.

[2]Because we are in Rhode Island, let's say you are making a robot for Hasbro.

[3]"Money" in this case can be generalized to "cost" that limits practical implementation. So, even if you are not fiscally motivated, such costs limit your ability realizing your technical visions.

interaction. The Subsumption Architecture is one approach to reactive control that uses multiple robot controllers with a priority scheme. In particular, each robot controller is capable of independently producing some robot behavior (such as wandering or obstacle avoidance) given certain applicability conditions (or preconditions) are met (such as whether an obstacle is present). When their preconditions are met, higher priority controllers "subsume" lower priority controllers, whose decisions are then ignored.

3 Control Objectives

In the current assignment, you will implement a robot soccer controller that uses only the robot's on-board immediate sensing and perception. Specifically, you are only allowed to use the information provided in the **current time instant** by Player's bumper, ir, and blobfinding proxies. Timers, odometry, and variables that require history are not allowed in your client. With this information, you are tasked with implementing a Subsumption architecture. Towards this end, you must:

1. determine and implement an appropriate set of control behaviors for soccer
2. apply appropriate preconditions for each control behavior
3. determine and implement a priority scheme to coordinate the behaviors

Your robot will need to stay within the general field of play as given by the physical walls of the Roomba Lab and virtual walls placed around the pitch. Additionally, your robot should make reasonable attempts to avoid collisions with other robots, which will be marked with pink fiducials. We will definitely call player pushing penalties for this assignment!

Virtual walls (pictured to the right) are devices that emit infrared (IR) light that can be received by the iRobot Create and Roomba platforms. The distribution of space covered by the emitted IR forms a beam that can extend over 8 feet. These beams form an invisible barrier around the field that your robot should respect. Your control client will have access to the SmURV's IR sensing through the IR proxy.

Figure 1: An iRobot Virtual Wall device emits an IR signal that can be received by the SmURV and provided to your client by Player's IR proxy.

3.1 Sensing a Virtual Wall

All you have to do to sense a virtual wall is to create an IR proxy/interface[4] and access the boolean flag "wall hit" through a call to the function GetRange(5); the 5 indicates the index of the virtual wall sensor.

In order to actually receive IR information from your SmURV you need to configure the player server to provide IR data. This is done by adding a few characters to the player configuration file on the SmURV (which is located in `/mnt/hda1/rootcopy/home/rlab/projects/[blind]smurv/[blind]smurv.c` In the section

[4]See `http://playerstage.sourceforge.net/doc/Player-cvs/player/classPlayerCc_1_1IrProxy.html` for C++ clients

```
driver
(
    name "create"
    provides ["position2d:0" "bumper:0" "ircom:0"]
    port "/dev/ttyS0"
    safe 0
    alwayson 1
)
```

change the `provides` line to the following:

```
    provides ["position2d:0" "bumper:0" "ircom:0" "ir:0"]
```

After restarting your SmURV (or copying the configuration file into the virtual file system manually) you should have access to the IR sensors.

3.2 Behaviors and Prioritization

You will be asked to demonstrate (either through a demo or in your report) that each control behavior functions independently and properly. In addition, your independent behaviors will need to function together (without code modification) using a subsumption priority for coordination. You will also be asked to randomly (based on the opinion of the course staff) reprioritize your behaviors to evaluate emergent behavior properties.

4 Soccer Skills Challenges and Competition

The goal scoring challenges and inter-group soccer competition from Assignment 3 (Path Planning) will be used for this competition in addition to a modified navigation challenge. The navigation challenge will have the same format as Assignment 3 except that obstacles will be pink fiducials and the field boundaries will be given by the physical and virtual walls.

5 Grading

Note: The soccer competitions will be held during class and the following hour (11-1pm). The TAs will announce available demo times, at which point you can choose one of those or (less preferably) schedule another time

Your grade for Assignment 5 will be determined by equal weighting of your group's implementation (50%) and your individual written report (50%). The weighted breakdown of grading factors for this assignment are as follows:

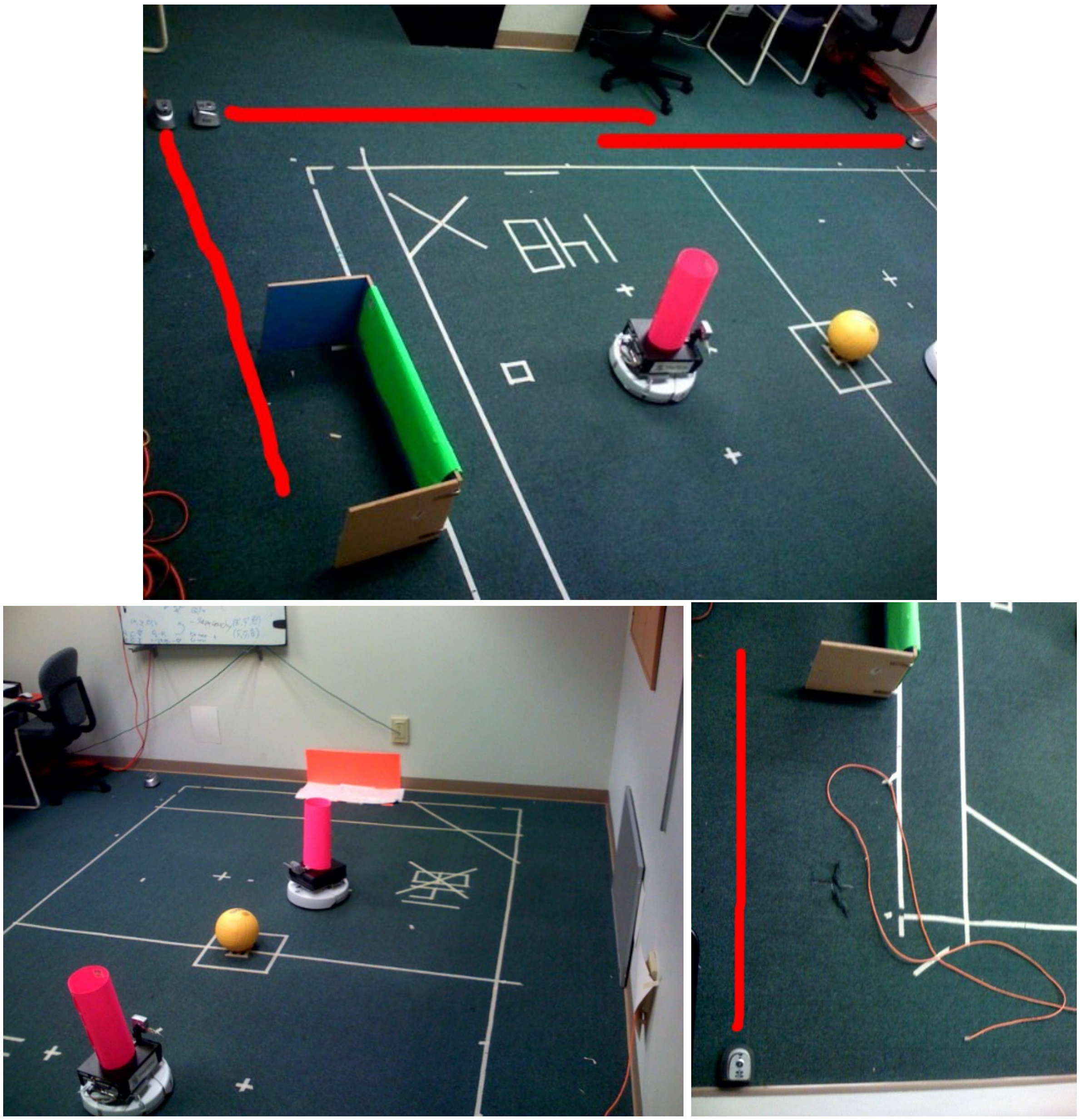

Figure 2: Snapshot of the “FC 148” robot soccer field surrounded by virtual infrared (roughly marked in red) and real walls. Your robot should properly react to both types of obstacles and other robots (with pink fiducials).

Project Implementation	
- Individual Behaviors → Does each of your behaviors properly function? → Is each of your behaviors completely reactive, without state variables? → Do your control behaviors allow for modularity and transparent reprioritization?	15%
- Behavior Choices → Do you have a sufficient set of behaviors for playing soccer? → Does your robot respect the environment (field of play) and avoid pushing other players? → Can your robot manipulate the ball and score goals?	15%
- Subsumption Priority → Does your robot properly prioritize its individual control behaviors? → Does each of your behaviors have appropriate preconditions to allow for coordination?	10%
- Soccer Proficiency → How well does your robot player soccer in the given environment?	5%
- Controller Robustness → Does your controller run without interruption?	5%
Written Report	
- Introduction and Problem Statement → What is your problem? → Why is it interesting?	7%
- Approach and Methods → What is your approach to the problem? → How did you implement your approach and algorithms? → Could someone reproduce your algorithms?	15%
- Experiments and Results → How did you validate your methods? → Describe your variables, controls, specific tests, and results from these test. → Could someone reproduce your results?	20%
- Conclusion and Discussion → What conclusions can be reached about your problem and approach? → What are the strengths of your approach? → What are the shortcomings of your approach?	8%