91.549 Mobile Robotics, Fall 2004
Prof. Yanco

Lab 3: Behavior-Based Control

Out: Thursday, 30 September 2004
Due: Thursday, 7 October 2004, at start of class

Overview: In this lab, we'll explore behavior-based control methods: subsumption and behavior combinations using fuzzy logic. There are two methods of using behaviors: sequences and blends.

Lab 2 used sequencing, both with the finite state machine based control. Subsumption is also a sequencing method. Subsumption will allow us to create a behavior-based robot in which higher level behaviors will take precedence over lower ones.

To use blending, you need only have multiple behaviors active simultaneously. These are usually in the same state, but could be in different states. In this lab, you will use fuzzy rules to combine behaviors.

Part I: Subsumption in Pyro

Read the beginning of the Pyro Module on Behavior Based Control (linked on the course web page). For this lab, we'll just be looking at the section on vertical behaviors. Try running the Subsumption.py in simulation (and on the robot if you'd like).

Instead of Exercise 1 as described in the Pyro module, extend Subsumption.py to include wall following. Think carefully about how to order the Wander, Avoid, and WallFollow behaviors. Which should have the highest priority? Which should have the lowest?

Part II: Investigating Fuzzy Rules and BBWander.py

Read the rest of the behavior based module. Fuzzy rules will allow us to combine behaviors based upon a statement's degree of truth. For example, if you are checking if the robot is close to the wall, the fuzzy value will be larger when the robot is close to the wall and smaller when the robot is far from the wall.

Now load the BBWander.py program (the code is also attached to this lab). Notice that the code for a behavior based brain is different than the code for a direct control brain. You'll write classes of Behaviors and States (notice that you're importing from pyro.brain.behaviors – a good thing to do is to look at the code in this file to see the internals).

The combination using the fuzzy rules can be found in the update method of the Avoid class. There are 4 fuzzy rules. Write up an explanation of how these rules are working to hand in with your lab.

While running BBWander.py in simulation, select brain from the view menu. You will be able to watch the fuzzy rules and how much each contributes to the overall behavior of the robot.

Part III: Writing Behavior Based Code Using Fuzzy Rules

In Lab 1, you wrote a direct control brain that could follow walls on the left and the right. Now you will write a behavior based brain that uses fuzzy rules to follow walls. Break your wall following code into rules for following on a side and staying safe in front. You should also be able to relocate a lost wall, as you did in Lab 1.

Does the behavior of your behavior based wall follower differ significantly from your direct control wall follower? Explain your answer.

What to turn in:

- For Part I, turn in your code and show me your code running on a robot.
- For Part II, turn in your written explanations of the fuzzy rules in BBWander.,
- For Part III, turn in your commented code for the behavior based wall follower, and a discussion of the differences between your direct control wall follower and your behavior based wall follower. You also need to show me your system working on the robot.