

Lab 4: Wall Following

Out: Tuesday, 27 February 2018

Due: Tuesday, 6 March 2018, at the beginning of class

Overview: In this lab, you'll use the optical rangefinder sensor ("ET") to build a wall-following robot.

You can have 1 or 2 sensors to use during this lab. The sensor's range is 4 to 30 inches. Low values indicate a large distance and high values indicate a distance approximating 4 inches. If something is closer than 4 inches, the values will go low again due to the optics of the sensor.

Mount the sensor on one side of your robot (you can continue to use the your robot base or redesign your robot to your liking). If you mount on the left, your robot will follow walls to its left; if you mount on the right, your robot will follow walls to its right. (If you'd like, you may mount sensors on both sides, and then choose to follow whichever side is near a wall. Or you can experiment with two sensors on a single side of the robot, one near the front and the other near the back.)

Since the sensor values invert when something is closer than 4 inches, it can be helpful to mount the sensor towards the center of your robot, so that you won't experience the inversion problem.

There are two ways to write a wall following program, described below. You should implement both.

1. The first is to have two states: one if the sensor reading indicates that the robot is too close to the wall and the other that the robot is too far. In the too close state, the robot should turn away from the wall. In the too far state, the robot would turn towards the wall. You may select the distance at which you'd like to follow the wall. I'd recommend that you pick a distance a bit away from the 4" minimum, since you'll get readings that increase from this point whether you get closer or farther away. Implement this solution. Turn in your code.
2. The second way to implement wall following code is to have three states: too close, just right, and too far. Modify your code to include the third state, the ideal distance from the wall. In this state, you'll drive forward. Experiment with the width of the "just right" range. What works best? Turn in your code.

Answer the following questions in your lab report, stapled to the code from the two strategies above (one set of answers, written collaboratively):

1. How do the two programs compare?
2. Which one does a better job following the wall?
3. How far can your robot travel down the hall (okay to estimate this distance)?
4. What does it do when there are doorways with open doors?
5. What does it do with recessed doors?

At the start of next week's class, you will show me your wall following robot in the hallway, demonstrating the method that you think works best.