COMP 4500, Mobile Robotics I
Spring 2018
Prof. Yanco

Assignment 1

Out: Thursday, 25 January 2018
Due: Tuesday, 30 January 2018

Overview: In this assignment, you'll tell me about your robotics experience and one cool robot.

1. Write a paragraph or two describing your prior experience with robots (if any). Are you a robotics minor? What's your major? What classes have you already taken? If you have worked on a robot system, which one?
2. What do you most want to learn about this semester?
3. Do you have any concerns about the course?
4. Have you picked your lab partner? If so, who will you be working with this semester? (If not, don't worry.)
5. What email address do you want used for the class Google group?
6. Describe one cool robot that you have seen (1-2 paragraphs). Your description should include the application domain, the sensors used on the robot, and how autonomous the system is. Include a reference to the web link or other source where you saw the robot. You can include a photo of the robot, if you'd like. full suite of smart sensor

Your answers to the questions above should be turned in at the start of Tuesday's lab in hard copy. Make sure your name is on the assignment.

COMP 4500, Mobile Robotics I
Spring 2018, Prof. Yanco

Lab 1: Building a Mobile Robot Base

Out: Tuesday, 30 January 2018
Due: Tuesday, 6 February 2018, at the start of lab

Reading: The Art of LEGO Design by Fred Martin, on the course web site

Overview: This lab is focused on building a mobile base for your Wallaby, which will be used in future labs.

What to do: For the first labs, we will be using the Wallaby mounted on a mobile base. This base should have the ability to turn, which is most easily accomplished by building a chassis that has two rear wheels, each with separate motors, and some form of caster on the front (could be slides or caster wheels). Your base should also have a bumper its front, with touch sensors on the left and right sides of the bumper.

You can design your own base using materials in the lab (including Lego, K'Nex and Vex – but you need not feel limited to use only those materials). Plexiglas mounts are available for the modified servos, which will allow the servo to interface easily with Lego.

You can model your base after the Handy Bug 9719 described in Section 2.2.2 of the Martin book (pages 50 – 71), although you will need to adapt it to use the non-LEGO motors that we'll be using in the course. Ten copies of this book are available in the lab (and should stay in the lab). The parts are listed on pages 54 – 56, with the following corrections:

- p. 54, halfway down: 4 2x4 bricks listed (correctly) with incorrect picture (1x2 Technic beam). You need the 4 2x4 bricks, not the 1x2 Technic beam.
- p. 56, last item: 2 black rubber bands are listed. These are too small. Use 2 yellow rubber bands instead. (Better yet, as these will be hard to find in the bins, create a different bumper for your robot.)

You will need to modify the design to hold the Wallaby, but it shouldn't require too many changes. You will definitely need to modify for different touch sensors.

Regardless of your approach, you will need to have your mobile base completed before next Tuesday's lab. In our lab on Tuesday 2/6, we will be programming the robot, so it is important that your robot is ready to go. Your grade for this lab will be a binary grade, given on Tuesday 2/6 at the start of lab. If your robot is built, you will get full credit. If your robot is not built, you will get no credit. There is nothing to write up for this lab.

COMP 4500, Mobile Robotics I
Spring 2018, Prof. Yanco

Lab 2: Sense and Avoid (or Follow)

Out: Tuesday, 6 February 2018
Due: Tuesday, 13 February 2018

Overview: In this lab, you will program your robot to avoid obstacles by backing up and turning when one of the bump sensors is hit. Then you will modify your robot to include an IR sensor (the "top hat" sensor) so that your robot can follow a black line.

What to do in this lab:

0. Connect to your Wallaby

See the course website for the slides on how to connect to your Wallaby.

1. Obstacle Avoidance

For this lab, you'll need to modify your robot car to include a bumper, if you didn't already build one on the robot in Lab 1. A good design is to have touch sensors on either side of the front of your robot, then have a piece of Lego, plastic or other object connecting the two; this design allows for your robot to sense if it hits an object anywhere on the front of your robot, not just on one of the corners where your touch sensors are.

After modifying your car to include a bumper, write a program to make your robot move forward. Then modify it so that if your robot hits an obstacle on either side of the bumper, it will back up and then turn towards the open direction (i.e., if it hits on the right, it will back up and turn to the left, and vice versa). Be sure to comment your code appropriately.

2. Line Following

Now you will create a robot that will follow a black line on a white background. For this part of the lab, you will need to use the "top hat" sensors, which plug into the analog ports.

1. Experiment with the top hat sensors to determine what they read when placed over white paper and what they read over black lines. Is there a difference between shiny and flat surfaces? Choose ranges that would indicate black readings and white readings. Note: you may wish to have some dead zone between the two colors, depending upon your findings. Run the same tests in the line course that you'll be using in this lab.

2. If you are trying to follow a black line on a white background, discuss algorithmic differences between using one top hat sensor and using two top hat sensors.

3. Now you will write the program that will allow your robot to follow a line. You may choose whether to use one top hat or two. Please discuss in your lab report why you chose the number you did. Choose good mount locations for your sensor(s)

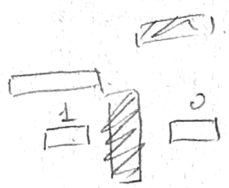
and attach them to the robot, using a fairly temporary mounting method such as double sided tape. Write the code to follow a black line. Turn in your commented code.

4. Was your robot able to traverse the entire course? If not, what part did it fail on? Why?

What needs to be turned in and demonstrated:

<u>Lab Report:</u> For your lab report, print out your code for both parts and attach answers to each of the questions raised above. Each team only needs to submit one lab report; however, I strongly encourage you to write the report together so that both members of the team participate in all aspects of the lab. Your lab report is due at the start of class on Tuesday, 13 February – printed out. Print it at home, print it in a computer lab on campus – just do it before class starts.

<u>Demonstration</u>: Demonstrate your line following robot for me and the class at the start of class on Tuesday, 13 February (or during class on Tuesday, 6 February, if you're done on the first day of the lab).



dif = 1 - 0

>0 left

<0 right

COMP 4500, Mobile Robotics I
Spring 2018, Prof. Yanco

Lab 3: Braitenberg Vehicles, Emergence, Meta-Sensing and Randomness

<u>Out:</u> Tuesday, 13 February 2018
<u>Due:</u> Tuesday, 27 February 2018, at the start of class

The questions in Part I should be answered individually and turned in with your lab report next Thursday. Each team will turn in one lab report (written jointly) that has each person's individual answers to the questions in Part I attached.

Part I: Individual Questions

Question 1:
Describe your line following robot from Lab 2 as a Braitenberg vehicle, based upon your reading of Chapters 1-4 of Braitenberg's Vehicles book.

Question 2:
Do you think that Braitenberg's method really can describe complicated robot behaviors? Why or why not? (No more than 1-2 paragraphs is needed to answer this question for this assignment, although one could write much more on the topic.)

Part II: Lab

In Lab 2, you designed code that would avoid obstacles and follow lines. Now you will be creating robots that are attracted to light.

For this lab, you may use the robot you built for Lab 1 and 2 or you may choose to redesign the robot with a different drive train. (Section 4.5 of Fred's book has a nice presentation of how to build with Lego. I highly recommend that you read it. I also handed out his Art of Lego article in class a few weeks ago.) However, all exercises can be completed with your existing robot. I leave this choice to you.

You only need to write one lab report for your group. I would prefer that your work in lab and your lab reports be collaborative work. One partner should not be writing the programs or the lab report alone. All of the questions below should be answered in your lab report.

For any exercises that ask you to write code, please turn in the code with your lab report. Please comment your code appropriately.

1 *A. Light sensors*

Write a short program that prints the value of two light sensors (to be distributed in class), which you should mount to either side of the front of your robot. Experiment with how the sensor values change as you move your robot around the room.

Answer the following questions in your lab report (from Martin 2.4.5):

1. How do the light sensor readings change as the amount of light increases?
2. Do the two light sensors typically give the same reading, or do they vary by the angle toward the nearest light source?
3. What are the maximum and minimum readings you can obtain? What are typical readings from the ambient light sources in the room?
4. Do both sensors seem to provide the same reading for the same amount of light, or are the readings from one shifted with respect to the other?

B. Shielding light sensors

Light sensors will be much more directional if you shield them using a small piece of tubing or some electrical tape, which will help you to find and approach a light source. Design shields fo the two light sensors on your robot. With your new shielding, what are the readings when the sensor is pointed directly at a light? How do the readings change as it turns away from the light source (10°, 20°, 45°, 90°)?

C. Normalizing light readings to motor commands

In order for your robot to function like a Braitenberg vehicle, you will write a program that uses the signals from the light sensors to control the robot's motors. To make your robot seek out a light, you will need to convert your light sensors' readings to a value where the darkest value makes the motor stop and the lightest makes it turn quickly. Write and test the function. Turn in the code.

1 *D. Light-seeking* 2 programs

Using your normalizing function, write a program that turns your robot into a light-seeing Braitenberg vehicle. Test it. Does it work? If not, what could be done to improve it?

1 Rewrite your program to avoid sources of light. Describe its behavior in different situations.

E. Light and touch sensitivity

1 Add your bumper behaviors from Lab 2 to the light-seeking robot code. Your robot should now seek out light while it will back up and turn if it hits something with its bump sensors.

Does the bump sensor code accomplish the goal of getting your robot to work its way around obstacles? Explain why or why not. If not, improve your code so that your robot does a better job of avoiding obstacles while it is seeking out light.

This lab is modified from Fred Martin's Robotic Explorations book.

COMP 4500, Mobile Robotics I
Spring 2018, Prof. Yanco

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.

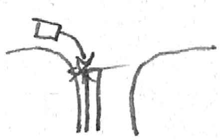
You can have 1 or 2 sensors to use during this lab. The sensor's range is 4 to 30 inches. Low values indicate a large distance and high values indicate a distance approximating 4 inches. If something is closer than 4 inches, the values will go low again due to the optics of the sensor.

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.

COMP 4500, Mobile Robotics I
Spring 2018, Prof. Yanco

Lab 5: Servo Motors

Out: Tuesday, 6 March 2018
Due: Tuesday, 20 March 2018, at the start of class

Overview:

In this lab, you'll use the IR distance sensor ("ET") and a servo motor to create a robot that drives towards open space. You'll create a mount for your servo motor on the front of your robot, then attach the distance sensor to the servo. This design will allow you to turn the distance sensor to point in different directions, so that you can find free space.

Using servo motors:

The servo pins on the Wallaby are labeled -, + and S. Black goes to – (black is usually ground in electronics), red to + (red is usually power in electronics), and yellow to S (signal).

To enable your servos, you need to call

```
enable_servos();
```

To disable your servos (do this when you're not using them anymore to save power – when servos are enabled, they are using power to maintain their position):

```
disable_servos();
```

To change a servo position, call

```
set_servo_position(<servo_num>, <servo_position>);
```

where `servo_num` should be 0-3, depending upon the port your motor has been plugged into. Values for `servo_position` should range between 0 and 2047, and give you about 180 degrees of movement. Note: Servos may run up against their stops at low or high position values. Giving a servo such a position command will use your battery's power at an alarming rate.

```
get_servo_position(<servo_num>);
```

returns an `int` corresponding to the position at which that servo is set.

If your servos are acting really strangely (e.g., twitching), your battery power is probably low.

There's a bin of attachments for the servo horn in the center of the room, with screws to attach. Pick a piece to screw onto the servo horn. To this piece, you can attach a piece of Lego using the sticky tabs. Then you can attach the sonar to the servo.

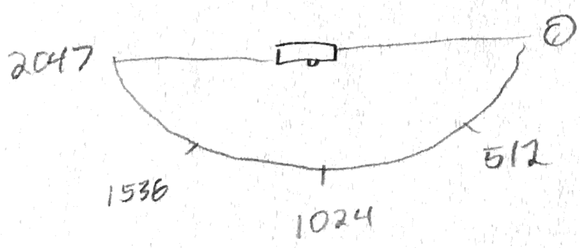
What to do:

For this lab, you should have the servo motor turn the distance sensor to 5 different positions spaced evenly through the servo's range, taking readings at each one. Then move your robot in the direction that had the farthest readings. (Break ties by picking the straightest route.) This will lead your robot into open space. It should also have the emergent behavior of turning your robot into a corridor follower in the hallways.

You may need to insert a short sleep between taking distance readings at each position, if you find that you're reading the prior position's IR return (you'll notice odd behavior if this is happening).

What to turn in:

Turn in a copy of your code (commented, of course) at the start of class on Tuesday, 20 March. You should also show your robot's behavior to Zhao in today's lab (if you finish during class – it's a pretty short lab) or at the start of class on Tuesday, 20 March.



COMP 4500, Mobile Robotics I
Spring 2018, Prof. Yanco

Lab 6: Vision

<u>Out:</u> Tuesday, 20 March 2018
<u>Due</u>: Tuesday, 3 April 2018

Overview:

In this lab, you'll use the camera to track colored objects.

Using the camera:

See the vision slides on the course web site.

What to do:

For this lab, you should write a program that will allow your robot to search for an object of a particular color (a bin of colored balls and brightly colored objects are in the lab for your convenience). Your robot should be able to track the object once it is found, in a similar fashion to your light seeking robot. If the object moves away, the robot should move closer to it (hint: look at the size of the tracked blob).

Once you can track an object and then keep it near the robot, you have some freedom for the lab, which spans two lab periods. Make the robot do something interesting with vision. For example, you might have your robot find a particular color ball, then kick it away after getting near it. Or maybe you try to gather all of the same colored objects into some location. Your choice. Be creative.

What to turn in:

Turn in a copy of your code (commented, of course) and a description of what you decided to do for the second part of the lab (can be in a comment at the start of the code instead of a separate page) at the start of lab on Tuesday, 3 April. You should also show your robot's behavior to me and to the class in the lab on that day – we'll have a set of mini-demos. You can also make a movie of your robot's behavior if you'd like.

image : .midx , .midy
range