To: Dr. Berry

From: Team Moravec, Devon Adair and Hunter LaMantia

Date: 1/27/2019

Re: Homing: Hybrid Control Lab 4

**Introduction**

The purpose of this lab is to use two photoresistors on the robot to make it react to light. The robot’s photoresistors detect nearby light with more intensity than ambient light and with react depending on what mode it is in. In love mode the robot slowly homes in on the light, in aggressive mode the robot homes in quickly, in explorer mode the robot slowly moves away from the light, and in fear mode the robot quickly moves away from the light. There is also a docking mode, in which the robot follows a wall until seeing a light, at which point it homes in on the light, turns around to dock at it, and then returns to the wall.

**Methods**

We use state machines to let the robot determine how it should move. We kept our states from the previous lab, including following a right wall, following a left wall, following a hallway, has lost a left wall, has lost a right wall, is randomly wandering, or has found a new wall. We also added another state for light detection. The light detection state takes priority over all other states, but it can be interrupted by obstacle avoidance.

When the robot is in the state of detecting light, it checks what mode it is in. The modes are set manually before the robot runs. The modes include love, aggressive, explorer, fear, and docking. The first four modes all work very similarly. The robot moves straight forward by default. When light is detected the robot determines which sensor is detecting the most light, and it sets a value proportional to the value of that sensor. In love mode it subtracts that number from the closest motor’s speed, in aggressive mode it adds that number to the farthest motor’s speed, in fear mode it adds that number to the closest motor’s speed, and in explorer mode it subtracts that number from the farthest motor’s speed. By having this behavior, the robot can be defined as a Braitenberg vehicle.

In docking mode, the robot behaves like it did in previous lab, randomly wandering or following walls. The difference is that when light is detected while following a wall, it uses proportional control and the values obtained from the two light sensors to home in on the light. The robot tracks how many movements it takes to reach the light. Once the light sensor values are high enough, the robot will turn around to dock. After docking, the robot will move forward once for every movement it took to reach the light. Experimentally, we found that this was sufficient to get the robot back to following the wall.

We also implemented obstacle avoidance for when something gets between the robot and a source of light. We took our wall following behavior from the previous lab and removed the functionality for looking for the wall after losing it. This resulted in a behavior where the robot will move around an obstacle until it is past it, at which point it forgets about the obstacle.

*1. How reliable was the photoresistor at detecting the light in different environments, various distances and angles of incidence (head on, slightly left, slight right).*

The photoresistors are very sensitive to changes in ambient light. We had to readjust many of our constants and thresholds whenever the ambient light levels changed. We also found that the photoresistors work best when the light is head on. The left sensor is generally more reliable than the right one.

*2. How significant was the difference in photoresistor voltages for the left and right sides. How did you use this difference to extract directional information to move the robot toward the beacon?*

The left photoresistor had a wide range of values that we could expect to obtain, whereas the right photoresistor had a very tight range, resulting in the left photoresistor being much more reliable. The values were converted to a percentage-based scale that we made, so that we could compare the values directly and tell the robot which direction to move.

*3. How did you integrate the light sensors into the obstacle avoidance behavior?*

Our obstacle avoidance behavior is essentially wall following, except the robot breaks out of it when it finds that the light is no longer behind the obstacle.

*4. How reliable was the photoresistor at detecting different objects at various shapes, sizes and distances. Compare and contrast sensor data.*

N/A

*5. How significant was the difference in photoresistor voltages for the left and right sides. How did you use this difference to extract directional information to move the robot toward the beacon?*

same as #2

*6. How significant was the difference in sensor data based upon distance from the source? How did you use this difference to extract distance information to move the robot toward the beacon?*

The left photoresistor had a wide range of values that we could expect to obtain, whereas the right photoresistor had a very tight range. Therefore, the difference in voltages between them varied a lot based on the distance.

*7. What does the hybrid control architecture for your design look like? What was on the planning layer? Middle layer? Reactive layer?*

N/A

*8. What was your general strategy for planning the path back to the wall from the beacon?*

When light is detected while following a wall, the robot uses proportional control and the values obtained from the two light sensors to home in on the light. The robot tracks how many movements it takes to reach the light. Once the robot has reached the light, turned around, and docked, the robot will move forward once for every movement it took to reach the light. Experimentally, we found that this was sufficient to get the robot back to following the wall.

*9. How did the architecture respond to differences in robot start position or beacon location?*

If there is no beacon in sight, the robot starts out in either random wander or wall following, depending on if there is a wall present. If there is a beacon in sight, the robot goes straight to homing.

*10. How did the robot’s hybrid controller respond to dynamic changes in the environment (i.e. other robots and people) and compare this to purely deliberative control.*

Our obstacle avoidance treats all of those examples as obstacles and goes around them.

*11. Were there any challenges in implementing the homing routine?*

We just had to adjust our P controller values until it worked.

*12. What could you do to improve the robot homing?*

It already works very well, but switching from P control to PID control would make it more accurate.

*13. How did docking the robot modify the control architecture or algorithm?*

It stopped executing its regular movement code and drove back towards the wall until it found it.

**Results**

Our robot worked very well, with very few issues. Our behavior for returning to a wall after docking needs some tweaking. Since we tracked the number of movements moving towards the light, we have a problem where having many small course corrections leads to too many steps, causing the robot to overshoot and hit the wall on its return. However, we decided that this was acceptable because our wall following behavior allowed it to get back on track once the return function ended. To fix this problem we should have used our IR sensors to know when to stop moving into to the wall instead of setting a specific distance.

Although it wasn’t perfect, the robot did everything that it needed to do. This has likely been our most successful lab to date.

**Appendix**

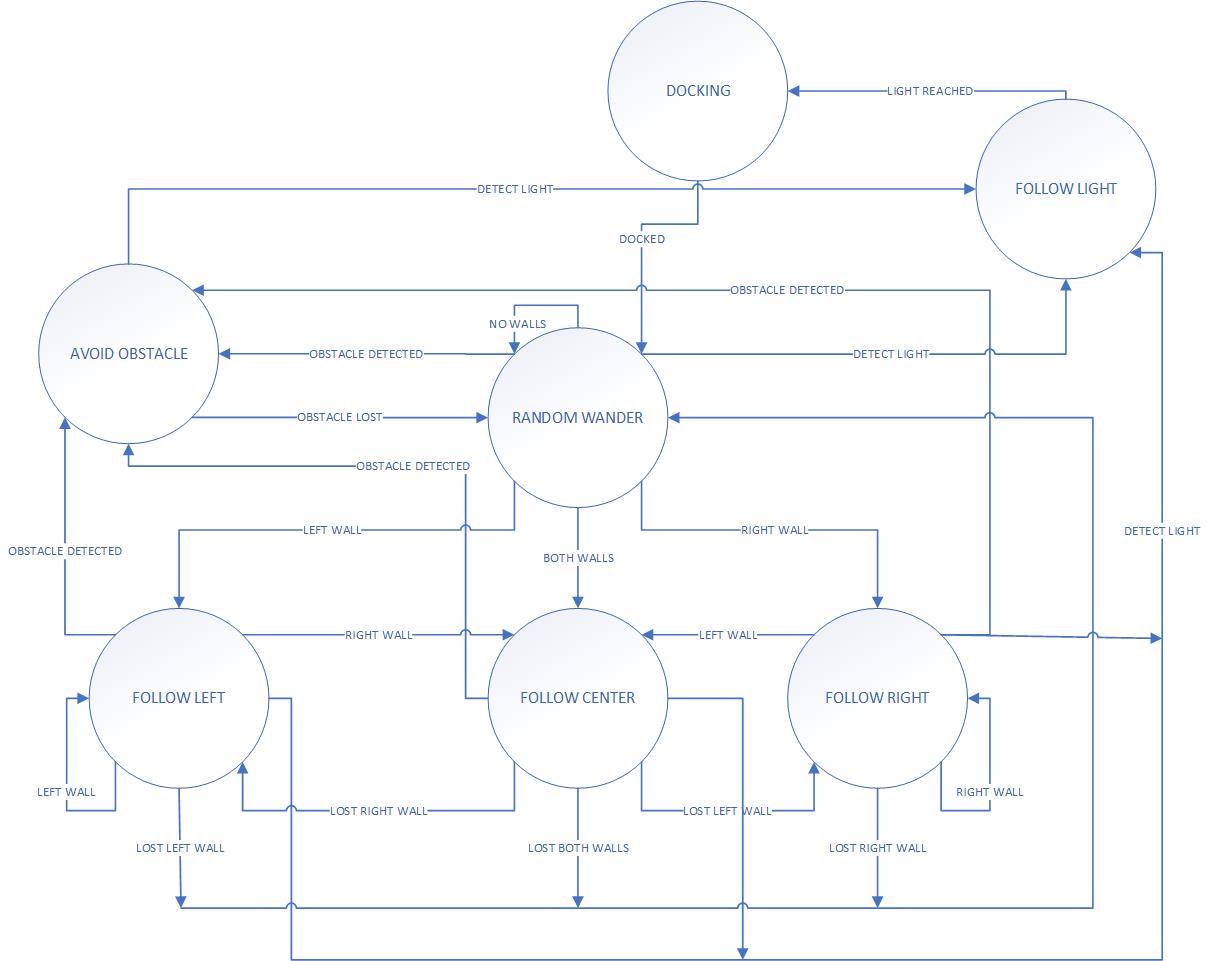


Figure 1: State Machine with wall following, obstacle avoidance, random wander, and light following

Table 1: State Transitions for the State machine

|  |  |  |  |
| --- | --- | --- | --- |
| **State** | **Input** | **Output** | **Next State** |
| Random Wander | No Walls | No Walls | Random Wander |
| Random Wander | Obstacle Lost | Obstacle Detected | Avoid Obstacle |
| Random Wander | Lost Right Wall | Right Wall | Follow Right |
| Random Wander | Lost Left Wall | Left Wall | Follow Left |
| Random Wander | Lost Both Walls | Both Walls | Follow Center |
| Random Wander | Light Reached | Light Detected | Follow Light |
| Avoid Obstacle | Obstacle Detected | Obstacle Lost | Random Wander |
| Avoid Obstacle |  | Light Detected | Follow Light |
| Follow Left | Left Wall | Left Wall | Follow Left |
| Follow Left | Left Wall | Lost Left Wall | Random Wander |
| Follow Left | Lost Right Wall | Right Wall | Follow Center |
| Follow Left |  | Obstacle Detected | Avoid Obstacle |
| Follow Left |  | Light Detected | Follow Light |
| Follow Right | Right Wall | Right Wall | Follow Right |
| Follow Right | Right Wall | Lost Right Wall | Random Wander |
| Follow Right | Lost Left Wall | Left Wall | Follow Center |
| Follow Right |  | Obstacle Detected | Avoid Obstacle |
| Follow Right |  | Light Detected | Follow Light |
| Follow Center | Both Walls | Lost Both Walls | Random Wander |
| Follow Center | Right Wall | Lost Right Wall | Follow Left |
| Follow Center | Left Wall | Lost Left Wall | Follow Right |
| Follow Center |  | Obstacle Detected | Avoid Obstacle |
| Follow Center |  | Light Detected | Follow Light |
| Follow Light | Light Detected | Light Reached | Random Wander |
| Docking | Light Reached | Docked | Random Wander |

Table 2: Photoresistor values in different conditions

|  |  |  |
| --- | --- | --- |
| Conditions | Left Photoresistor (V) | Right Photoresistor (V) |
| Ambient light on the table | 870 | 925 |
| Ambient light under the table | 610 | 770 |
| Sensor covered | 170 | 120 |
| In front of a flashlight or cell phone light | 1010 | 1015 |

Table 3: Photoresistor values in different conditions and different lengths

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | Left Photoresistor (V) | | | Right Photoresistor (V) | | |
| Environment | Distance (in) | Angle of Incidence  -45 deg | Angle of Incidence 0 deg | Angle of Incidence +45 deg | Angle of Incidence  -45 deg | Angle of Incidence 0 deg | Angle of Incidence +45 deg |
| On the table | 6 | 940 | 915 | 905 | 935 | 945 | 965 |
| On the table | 12 | 895 | 890 | 900 | 925 | 930 | 945 |
| On the table | 18 | 880 | 880 | 895 | 925 | 925 | 940 |
| On the table | 24 | 870 | 875 | 890 | 925 | 923 | 935 |
| On the table | 30 | 870 | 875 | 890 | 920 | 920 | 935 |
| Under the table | 6 | 920 | 875 | 815 | 895 | 925 | 945 |
| Under the table | 12 | 845 | 795 | 765 | 885 | 880 | 885 |
| Under the table | 18 | 805 | 735 | 730 | 875 | 845 | 850 |
| Under the table | 24 | 785 | 695 | 710 | 875 | 825 | 825 |
| Under the table | 30 | 775 | 670 | 690 | 870 | 810 | 810 |