Lab 4: Localization report.

Group 22: **David Lavoie-Boutin, Daniel Macario.**

Data

Data obtained from the ultrasonic sensor localization. The positive y direction represents the angle 0, the values to the right of this axis are positive. So, for example, an angle of 10 means the robot was 10 degrees to the right from going north.

Measure #	Angle (Falling Edge) in degrees	Angle (Rising Edge) in degrees
1	9	9
2	3.1	-5.8
3	4.1	3.7
4	9.7	-3.6
5	6.8	7.4
6	5.2	3.7
7	8.2	4.8
8	0	0
9	1	2.1
10	3.6	-7.6

Mean and standard deviation tables, calculations can be found in a later section.

	Falling Edge	Rising Edge
Mean	5.1	1.4
Standard Deviation	3.3	5.5

Observations and Conclusions

- **Which of the two localization routines performed the best? Which performed the worst? What factors do you think contributed to the performance (or lack thereof) of each method?**

The mean calculation indicates that the rising edge ultrasonic localization algorithm performed significantly better than the falling edge alternative. In the case of the falling edge algorithm, the average angle of direction that the robot pointed to was just 1.4 degrees off the target, while the rising edge algorithm was off by 5.1 degrees on average. However, the metric that really allows us to define which algorithm performed the best is the standard deviation.

Notice how the standard deviation of the falling edge algorithm is 2.2 cm lower than the one presented by the rising edge algorithm. This suggests that falling edge localization was more consistent in terms of the direction the robot would point to after the algorithm, which translated to more reliable positioning. On the other hand, notice that the resulting form rising edge localization vary from -7.6 to 9. The proportionate number of negative and positive angles explains the close average to the target orientation, but it does not represent the average case of the algorithm given its large standard deviation.

The key difference between the two algorithms can be seen in how these recorded the angles used for the calculations needed to orient the robot. Falling edge localization recorded the angles of the odometer once the distance to the wall dropped below 50 cm; rising edge did the same but when the distance to the wall went above 50 cm. The fact that in the rising edge algorithm the robot kept moving away from the wall once it pinged, translates into a slower reaction time than the one presented by the falling edge algorithm, which could in turn result in flawed measurements of the distance to the wall.

Another important source of error that affected both algorithms resulted from stopping the robot each time it recorded the angle used for the orientation calculation. In the course of the algorithm the robot stopped twice because of this, which in turn resulted in 2 deceleration periods and two 2 acceleration periods. This significantly increases the possibility of wheel slip, which added to the ping delay of the rising edge algorithm resulted in a wide range of values.

- **Why does the light sensor provide a more accurate means of orienting the robot than the ultrasonic sensor?**

The light sensor is significantly more accurate than the ultrasonic sensor because the window of time between the moment it samples data and the moment it receives and processes it is very small. This is because the light sensor is fixed at a distance of 0.5 cm from the interface from which data is being collected (the ground), while the ultrasonic sensor is fixed at a varying distance of about 50 cm from the its interface (the walls). The ample window between sampling for information an receiving it leads to more significant sources of error from the environment, such as noise.

Also, given the stark contrast between the colour of the tiles and the gridlines on the plane, it is possible to adjust the light sensor so that it never fails to detect a gridline. Once this is achieved, the localization mechanism essentially depends on a properly calibrated odometer and implementing the right equations to orient itself north at the x and y coordinates of (0,0). In other words, there are less sources of error present when using the light sensor to localize the robot than when the ultrasonic sensor is used.

- **Propose a means of determining (approximately) the initial position of the robot using the ultrasonic sensor (Hint: Consider the minima of the ultrasonic sensor's readings as the robot rotates). Why is detecting minima with the ultrasonic sensor problematic?**

An option would be to initially position the robot parallel to the +y axis using falling edge ultrasonic sensor localization, and then measuring its distance to the walls located directly to its left and back. We can do achieve this by rotating the robot 180 degrees counter clock-wise and taking the ultrasonic sensor measurements at 90 and 180 degrees. This translates to obtaining the x and y coordinates relative to the origin of plane, which translates to the corner of the plane in which the robot is located. It is important to adjust for the distance from the sensor to the actual centre of the robot. See the diagram below for a clear representation:

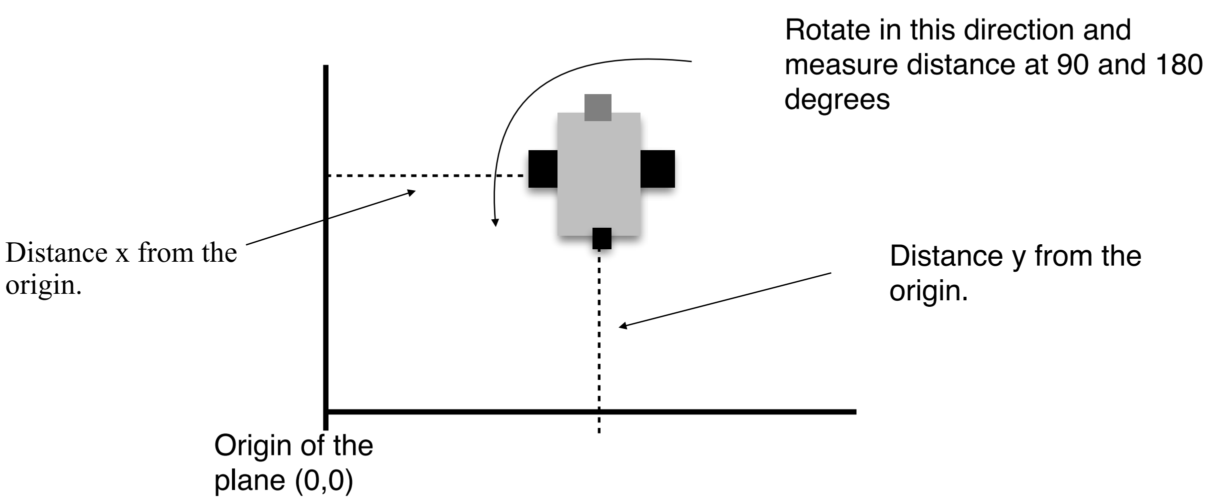

Figure 1. Diagram showing the proposed localization algorithm.

Error Calculations

Falling edge error calculations:

$$mean(angle) = \frac{9.0 + 3.1 + 4.1 + 9.7 + 6.8 + 5.2 + 8.2 + 0.0 + 1.0 + 3.6}{10} = 5.1$$

$$stdev(angle) = \sqrt{\frac{\begin{array}{l}(9.0-5.1)^2 + (3.1-5.1)^2 + (4.1-5.1)^2 + (9.7-5.1)^2 + (6.8-5.1)^2 + \\ (5.2-5.1)^2 + (8.2-5.1)^2 + (0.0-5.1)^2 + (1.0-5.1)^2 + (3.6-5.1)^2\end{array}}{9}} = 3.3$$

Rising edge error calculations:

$$mean(angle) = \frac{9.0 - 5.8 + 3.7 - 3.6 + 7.4 + 3.7 + 4.8 + 0.0 + 2.0 - 7.6}{10} = 1.4$$

$$stdev(angle) = \sqrt{\frac{\begin{array}{l}(9.0-1.4)^2 + (-5.8-1.4)^2 + (3.7-1.4)^2 + (-3.6-1.4)^2 + (7.4-1.4)^2 + \\ (3.7-1.4)^2 + (4.8-1.4)^2 + (0.0-1.4)^2 + (2.1-1.4)^2 + (-7.6-1.4)^2\end{array}}{9}} = 5.5$$

Future Improvements

1. Propose a way to avoid small errors more accurately than a clipping filter.

There are other types of filters that provide better accuracy for specific implementations. In an experiment such as this one, a moving average filter is likely to produce more accurate results. A moving average filter calculates the average of a specific window of measurements to minimize the size of the errors detected. It shifts the window of measurements as new ones are recorded.

2. Propose a sensor design that would result in a more accurate and reliable reading than an ultrasonic sensor.

The most significant problem we noticed with the ultrasonic sensor is the fact that the survey window is relatively large and not well defined; The values measured by the sensor varied from 10 cm to more than 50 cm (depending on the implementation). This large distances combined with the sampling delay of the ultrasonic sensor result in imprecise readings. A possible sensor that could solve this problem would minimize the delay of sampling and processing data, in this case the distance to the walls. This can be achieved by using a sensor based on light waves rather than sonic waves. Specifically, a laser range finder.

This type of sensor is able to calculate distances based on the time it takes to reflect a narrow beam of light off a surface back to the sensor. It would provide more reliable and accurate readings due to its narrow sampling window in terms of time and space, as well as reduce the possibility of false negatives because of controlled lighting conditions (the light fixtures in the lab are very reliable). This is an important point of contrast with the ultrasonic sensor, which is easily affected by ambient noise.

3. Propose another form of localization than rising-edge or falling-edge.

The rising and falling edge methods both force the robot to stop and turn in opposite directions. This can result in wheel slip and reduce the accuracy of the odometer. Instead of rotating, stopping, and re-rotating, the robot could execute a 360 degree turn in place and record the distance from the wall at every angle. Repeating this operation a few times and taking the average of the values at each angle would provide an accurate set of data. Then, using statistical analysis, the angles at which the robot breaks the 50 cm threshold could be calculated with great precision. Using this information, the robot could be placed parallel to the +y axis with a greater degree of precision than the one we would obtain from the falling or rising edge localization algorithms.