Lab Report #3 – Navigation and Obstacle Avoidance

Group #55: Alex Hale (260672475) and Xianyi Zhan (260672960)

ECSE 211 – Design Principles and Methods

October 5, 2017

Design Evaluation

Hardware

A simple three-wheeled robot was created using two motors, two wheels, one ball bearing, one ultrasonic sensor, one EV3 brick, and various connecting pieces (Figure 1). The ultrasonic sensor was mounted to the front of the robot, below the brick but above the wheels, about two centimetres off the floor.

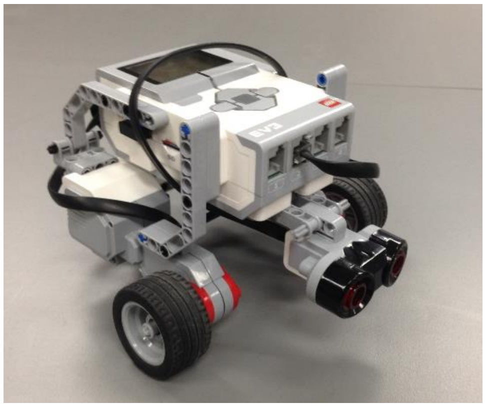

Figure 1 – Robot Hardware Design

Software

The software structure is outlined in Figure 2.

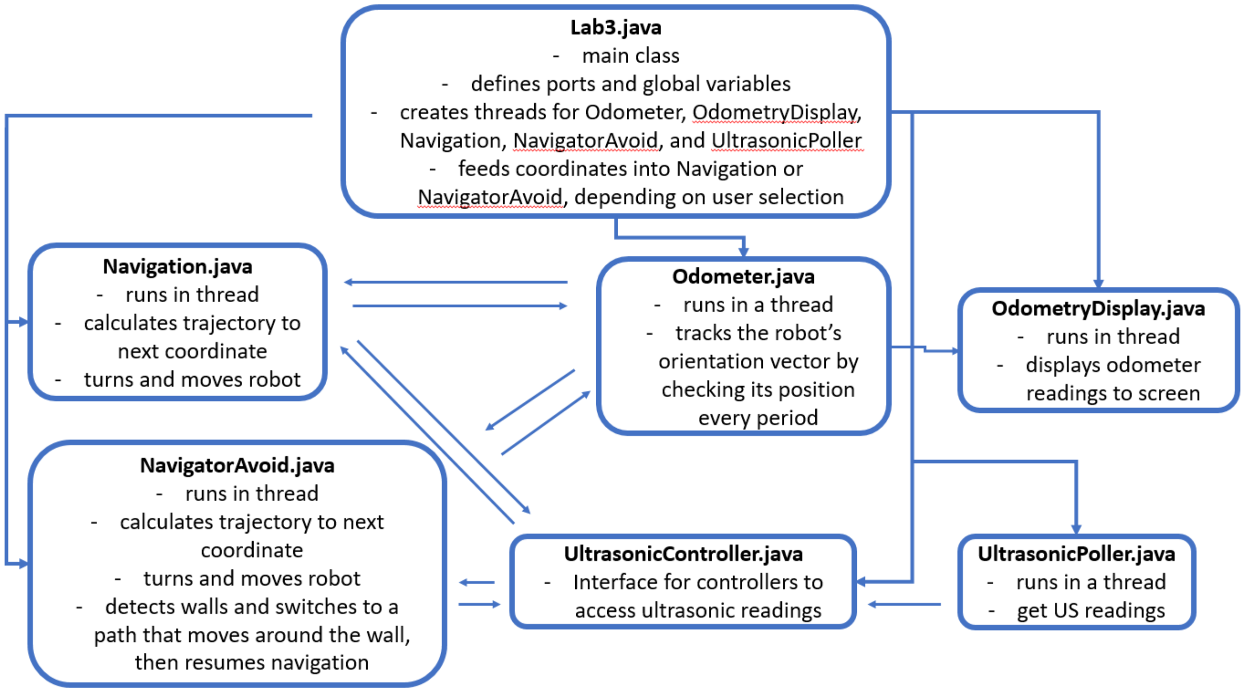

Figure 2 – Software Structure

Workflow (hours are a total of both lab partners)

1) Design robot based on given criteria (2 hours)
2) Write software based on and design criteria (10)
3) Test design on grid floor. Identify errors in hardware and software that cause the robot to fail the design criteria, fix them, repeat (20 hours)
4) Demo project to TA (0.5 hours)
5) Lab report (3 hours)

Test Data

NOTE: In each trial, the robot started with the centre of its wheel axle on top of the (0, 0) intersection. X_F and Y_F are the measured finishing positions with respect to the origin.

Navigation Test

Test results are shown in Figure 3.

Trial	$\mathbf{X_F}$ (cm)	$\mathbf{Y_F}$ (cm)	$\mathbf{X_{odometer}}$ (cm)	$\mathbf{Y_{odometer}}$ (cm)	**Euclidean Error (ϵ)**
1	61.45	-0.33	60.99	0.01	0.57
2	58.19	1.89	61.00	0.00	3.39
3	60.92	-1.11	60.99	0.00	1.11
4	58.43	1.97	60.99	-0.01	3.24
5	60.33	-0.28	60.98	0.00	0.71
6	57.53	1.12	60.98	0.01	3.62
7	61.15	-0.31	60.99	0.00	0.35
8	59.03	1.83	61.00	0.01	2.68
9	59.79	-0.41	60.99	0.00	1.27
10	57.82	1.37	60.99	0.00	3.45

Figure 3 – Test Results

Test Analysis

$$Mean\ of\ error\ (\mu) = \frac{sum\ of\ all\ errors}{number\ of\ trials\ (N)} = \frac{20.39}{10} = 2.04$$

$$Standard\ deviation\ of\ error = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\varepsilon_i - \mu)^2} = \sqrt{\frac{1}{10}(16.41)} = 1.28$$

The error present is due to a combination of the odometer and the navigator. The odometer introduces a very small amount of error due to the accuracy of the sensor that reads the number of wheel rotations, so the odometer readings vary by a very small amount ($\pm 0.01 \text{ cm}$). The navigator introduces error because it is open-loop (i.e. uncorrected). The navigator relies on odometer readings to calculate how much to turn and how far to travel, compounding any existing error.

Most of the error is due to the physical traits of the robot. The wheels slip slightly when starting and stopping, the wheel axle is flexible, the wheels compress slightly, and the starting position and direction of the robot is not the same for each trial. These sources of error are much more significant than the error introduced from the odometer and navigator.

Observations and Conclusions

The navigation controller converts given coordinates to a distance from the origin and, using the current reading of the odometer, calculates the distance and change in angle to that coordinate. After ensuring that the robot will turn the shortest distance necessary, the robot turns to face the next coordinate. Finally, the robot moves forward the calculated distance to the next coordinate. The process is then repeated for the next coordinate.

The navigation controller moves the robot to within ~2 cm of the destination on average. Most of that error occurs due to physical characteristics of the robot, and not the navigator controller. The robot does not oscillate at all on the destination, because no correction feedback is made.

Since the main source of error in navigation is the robot's physical characteristics, increasing the speed would make the navigation less accurate. However, this error increase would not be significant so long as the acceleration of the motors is kept low. The acceleration is the main source of wheel slip, so increasing the robot's top speed while keeping acceleration the same would not make too much of a difference.

Further Improvements

The error due to hardware could be reduced by making the robot more structurally stable and by better characterizing the robot's physical features. If the wheel axle were less flexible, the robot would rotate and drive more precisely. If the robot's physical features (e.g. axle flex, wheel compression, difference in motor speeds) were characterized better, adjustments could be made in both hardware and software to correct them.

If a light sensor were added to the hardware design, then large adjustments could be made in software to reduce error. A light sensor could detect when the robot crosses gridlines, compare them to the current odometer position, and adjust the odometer accordingly. A light sensor could also be used upon arrival at the destination to explore the surrounding area and finish the run on top of the exact grid intersection. If two light sensors were used, navigation could become even more accurate – as was outlined in detail in the previous report, the robot could measure the distance travelled between the moments the two sensors detected a gridline, and calculate a course correction to adjust the odometer.