Laboratory 4: Localization

ECSE 211: Design Principles and Methods

Prof. Ferrie, Prof. Lowther

Li Zhang (260743980)

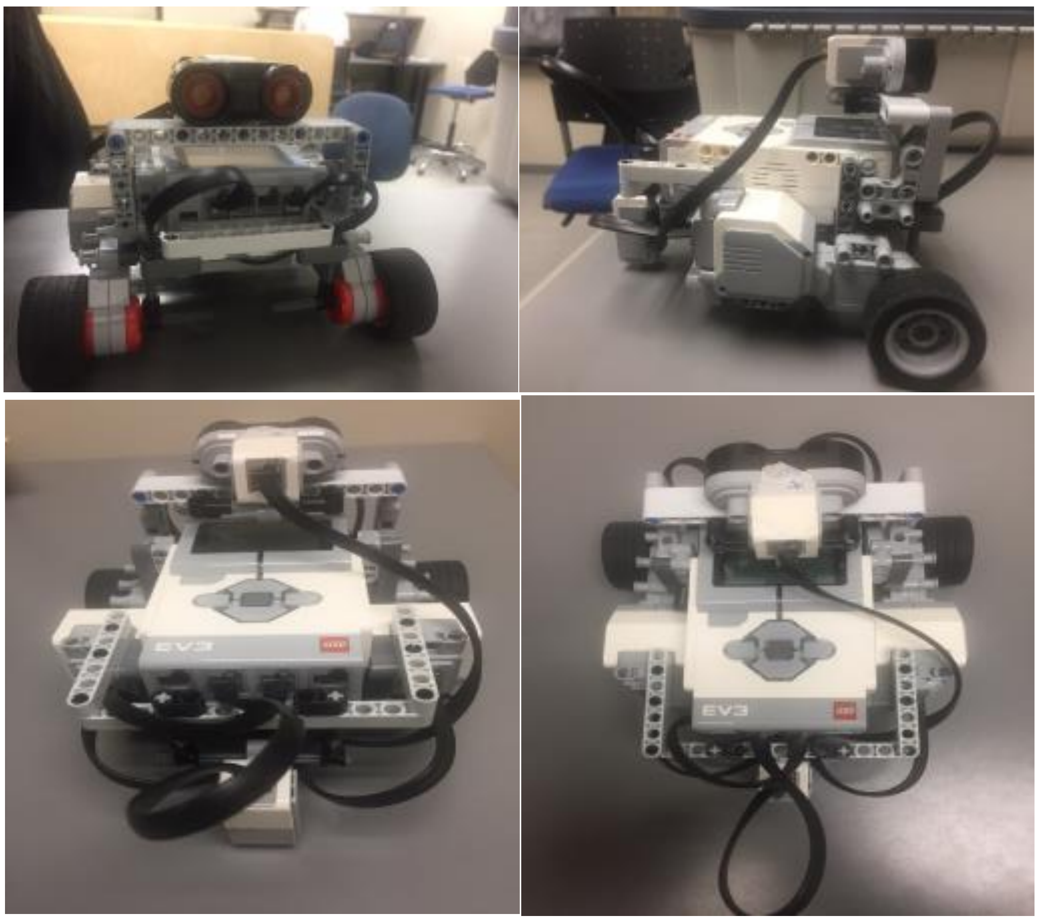
Marwan Chehade (260675988)

Group 10

I. Design evaluation

A. Hardware design

The hardware designed consists of the LeJOS EV3 Mindstorms Brick to which is attached two large motors in order to allow wheels to rotate and move the robot. Additionally, an ultrasonic (US) sensor is mounted to the front of the robot, above the center of rotation of the robot, and a light sensor to its back, approximately 15 centimeters away from its center of rotation (midway between the two wheels). These sensors are meant to be used in the aim of locating the walls and lines around and underneath the robot while it is localizing.

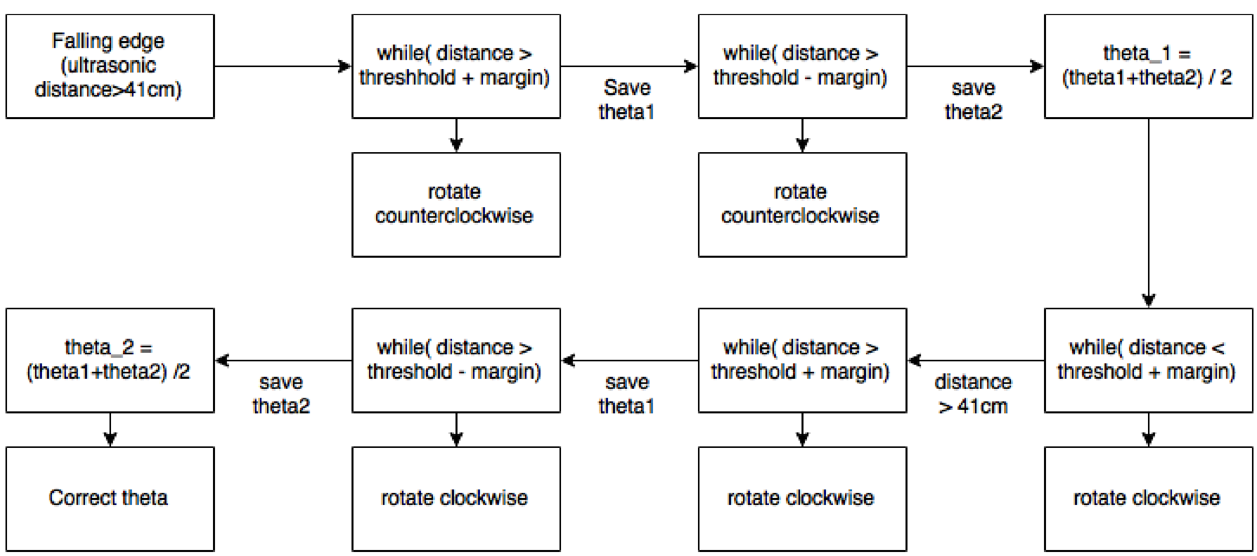


B. Software design

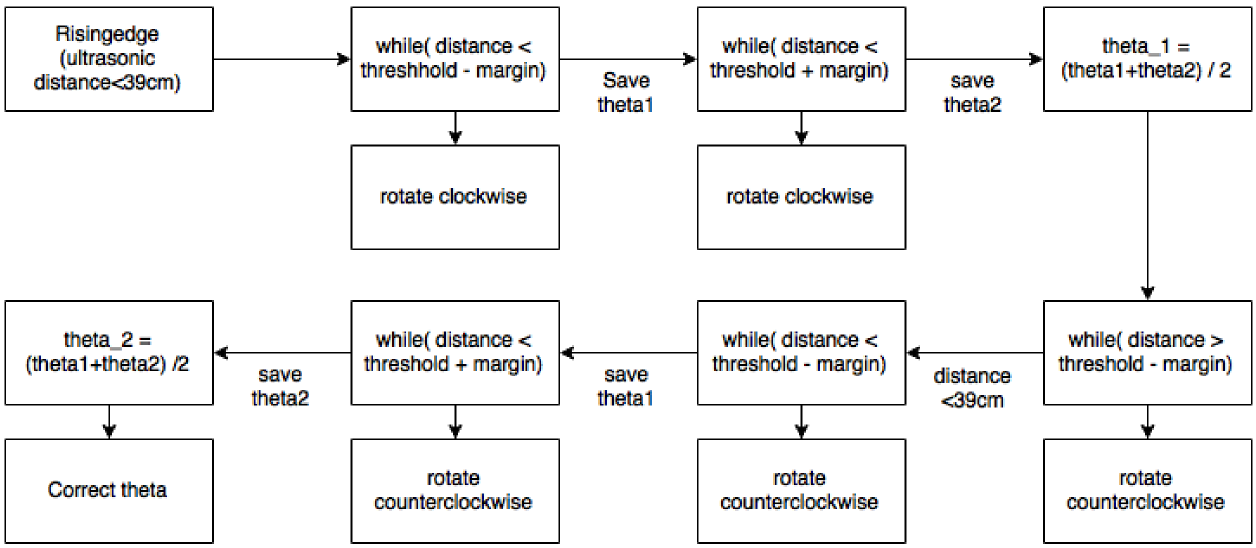
1- Ultrasonic Localizer

The ultrasonic localizer method makes the robot rotate on itself until it detects a significant increase or decrease of the distance returned by the sensor, which would signify that it is turning from the wall to the open space, or the opposite. A threshold is also set up so that that robot can detect when it is turning towards or away from a wall. It then saves the angles and using those angles, it can determine the correct angle that the robot is facing. Two different ways of detecting the change of direction can then be implemented.

a. *Falling Edge:* The robot starts facing the direction opposite to the wall. As it rotates counter clockwise, it detects that the ultrasonic sensor values have dropped below the sum of the threshold and the specified margin of error (US value< threshold+margin). A first value for the angle theta ϴ is noted. As it continues its rotation, the US value then drops underneath the value of [threshold-margin] and a second value for theta is acquired. Then, an average of these first two values generate Θ_1. The robot then stops and rotates clockwise, going through the same process to obtain Θ_2. An angle $\Delta\Theta$ is calculated and added to the current angle read by the odometer. This allows the robot to know its current heading and can thus adjust itself to turn towards 0˚.



b. *Rising Edge:* In the same way that the falling edge method is executed, the rising edge implementation rotates the robot rotating clockwise and averages the angles at which the sensor values exceed [threshold-margin] first then [threshold+margin]. It repeats this procedure by rotating counter clockwise and calculates its current heading to correct it.



2- Light Localizer

The light localizer software involves making the robot travel forward to detect the x axis then backup, turn to 90 and do the same to detect the y axis. Now that it is placed at a position close enough to the origin (0,0), it starts rotating counter clockwise and records the angle at every line met (negative y axis, positive x-axis, positive y-axis and negative x-axis). Using these values, the robot can determine its current x and y position as well as a ΔΘ, which is the theta correction it needs to correct its angle. Consequently, it can travel to the origin and correct its theta adequately.

II. Test Data

Table 1 Measurements made after 10 trials

	Falling Edge			Rising Edge		
Trials #	Ultrasonic localization ϴ (˚)	Final ϴ (˚)	ϵ (cm)	Ultrasonic localization ϴ (˚)	Final ϴ (˚)	ϵ (cm)
1	5	4	1.5	4	2	2.2
2	3	2	1.5	7	5	4
3	5	4	2	8	7	2
4	3	2	1.7	7	10	3.7
5	4	3	1.3	10	8	4.3
6	6	4	2.4	15	12	2.5
7	3	3	4.2	5	3	1.7
8	2	1	2	6	4	3
9	4	3	1.8	12	6	5.2
10	7	5	2.8	9	5	1.6

III. Test Analysis

For each test, compute the mean and standard deviation of the ultrasonic angle error, Euclidean distance error, and final angle error. Be sure to show general formulas and sample calculations.

$$Mean\ \mu = \frac{\sum error}{trials}$$

$$Standard\ deviation\ \sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(error_i - \mu)^2}$$

where N = number of trials

Table 2 Mean and standard deviation for theta and the Euclidean error after 10 trials

	Falling Edge			Rising Edge		
	Ultrasonic ϴ (˚)	Final ϴ (˚)	ϵ (cm)	Ultrasonic ϴ (˚)	Final ϴ (˚)	ϵ (cm)
Mean	4.2	3.1	2.12	8.3	5.5	3.02
Std Dev	1.469694	1.1357817	0.813388	3.163858	2.2472205	1.164302

IV. Observations and Conclusions

1. *Which of the two localization routines performed the best?*

Comparison of light and ultrasonic localization: The light localization routine performed better as the final theta obtained was (approximately) always corrected by getting closer to 0 both for the falling edge and rising edge methods. This routine also showed high precision when it comes to its final position (low Euclidean error) However, for the ultrasonic localizer, the angle at the end was not always as close to 0 as it should be specifically for the rising edge.

Comparison of falling edge+light localization and rising edge+light localization: It is important to note that if we compare the different routines consisting of the two types of US localization followed by the light localization the results obtained show that falling edge performs much better than rising edge and the final result also reflects this difference.

2. *Which performed the worst?*

The ultrasonic localization routine did not perform as well as the light localization one. The angle obtained was not as close to 0 degrees and the robot's heading depended on the starting position of the robot which shows an inconsistency in its performance.

3. *Was the final angle impacted by the initial ultrasonic angle?*

The final angle is not noticeably impacted by the initial ultrasonic angle and this can be explained by the fact that the $\Delta\Theta$ calculated is based on the angle at which the black lines are crossed and thus having a large ultrasonic angle error leads to a larger correction and thus the final angle ends up being relatively small. However, it is important to note that the rising edge method did not perform as good as the falling edge and this can explain the final angles which are do not regularly fall into the [0;5] degrees range.

4. *What factors do you think contributed to the performance of each method?*

The falling edge method performed better than the rising edge method as the mean and standard deviation for the ultrasonic angle, final angle and the Euclidean distance errors were smaller for the falling edge (respectively $1.47<3.16$, $1.14<2.25$ and $0.81<1.16$).

This can be due to the fact that for the rising edge, the ultrasonic sensor values can sometimes largely change even if the robot is not facing an empty space yet while rotating. The US sensor value increases by a large amount and the robot thus considers that it should rotate in the other way and record the angle it is at. The final result obtained is affected and further away from what it should be.

5. How do changing light conditions impact the light localization?

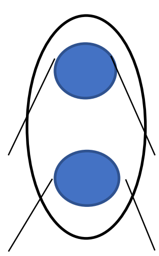
The light conditions can impact the light localization because ambient light for example can make all the light values increase and thus values for which the robot thinks it is over a clear tile, it is actually crossing a black line and does not detect it. This can be improved by implementing a differentiation method where the software takes the difference between consecutive values from the light sensor in order to assume a black line is detected or not. Therefore, if the US values go from 13 to 6 in a closed room and from 20 to 14 in ambient light, if the software is programed to consider a difference between values of 5 or more, it will detect black lines in both cases. But if it only detects a black line when the sensor values are less than 10, the robot will not detect any black lines in ambient light. The light localization routine can then fail if lines are not detected adequately.

V. Further Improvements

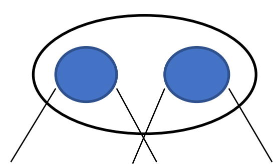
1. Propose a way to minimize errors in the ultrasonic sensor.

To minimize errors in the ultrasonic sensor, one can use an average of values given by the sensor in order for the robot to respond to it. An average allows to have more reasonable values if the robot is rotating slow enough, thus any error would be compensated by using the average of a few values.

A filter is also important and leads to suppressing outliers or very large numbers. And mounting the ultrasonic vertically instead of horizontally which can lead to picking up distances from a smaller range thus adding precision for the robot's movements.



Vertical Sensor



Horizontal Sensor

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

If a robot has two ultrasonic sensors, at 90 degrees from each other, such that one is facing forward, and one at its left (so one at 0 degrees and one at 270 degrees), it can rotate until both sensors read the same distance. this would work if the robot is initially placed along the 45 degrees angle line. Once the robot has approximately the same distance from both sensors, it can then be assumed that the robot is facing the 270 degrees, and can thus turn 90 degrees clockwise to turn to 0 degrees.

3. *Discuss how and when light localization could be used outside of the corner to correct Odometry errors, e.g. having navigated to the middle of a larger floor*

Light localization can be used outside of the corner by implementing a method that allows it to locate every black line it crosses and correct its x and y values adequately. Consequently, if it is travelling in a straight-line as opposed to a diagonal (this can be determined considering its actual coordinates and the waypoint it is headed to), it will modify its x and y positions to be multiples of the length of a tile when it detects a black line. Whenever it reaches a waypoint, it can also rotate by 360˚ and detect the angles at which it crosses black lines so that it can correct its theta and heading in a similar manner to what was done in the light localization routine.

Additionally, the robot can move to one of the corners of the tile and implement a light localization routine to correct its x, y and theta.