Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

MEU44B12 Introduction to Autonomous Mobile Robotics

Practical Lab Report

Name: Shirish Senthil

Student ID: 21349979

Date of Completion: 04/04/2023

Date of Submission: 20/04/2023

Table of Contents

Abstract

This lab report explored several methodologies and statistically analyzed to see which robot navigation strategy is most suitable for the path set up. From the experiment, two iterations were run consisting of all the strategies, and the data was recorded. The three error parameters namely X-axis, Y-axis, and theta angle (robot pose) were recorded to decide which methodology was the most accurate. Two iterations of the 3 methodologies were run and the standard deviation from the origin was calculated as the error. The method with the lowest standard deviation is the preferred strategy. This was represented in a tabular form and the lowest error per corner was calculated to decide which methodology produced the most accuracy with high precision. The different types of unforced/random errors are also discussed to provide context.

Introduction

There is a deep requirement and rapid development in the implementation of autonomous capabilities in robots and systems. The applications of this AI technology expand from exploration using unmanned vehicles/rovers to underwater autonomous marine robots. Some commercial applications of autonomous robots are the “Roomba robot” which utilizes a navigation beacon, cameras for localization, and room mapping to allow for continuous cleaning of indoor spaces. [1] However, choosing and implementing which algorithm on the robot can be quite complex. This practical lab explores three methodologies: 1. Open-loop (No sensing), 2. Odometry only (Proprioceptive sensors only) and 3. Lidar + Odometry. A robot is used as a robot that is operated wirelessly. The primary objective of this lab report is to statistically evaluate the performance of these three methodologies to conclude which algorithm produces the least standard deviation from the end mark. This can be further used to justify the end result's variance in practical scenarios compared to the simulation conducted in Webots.

Experimental Setup

The robot is made to go around the 2*2-meter path shown below using the different algorithms:

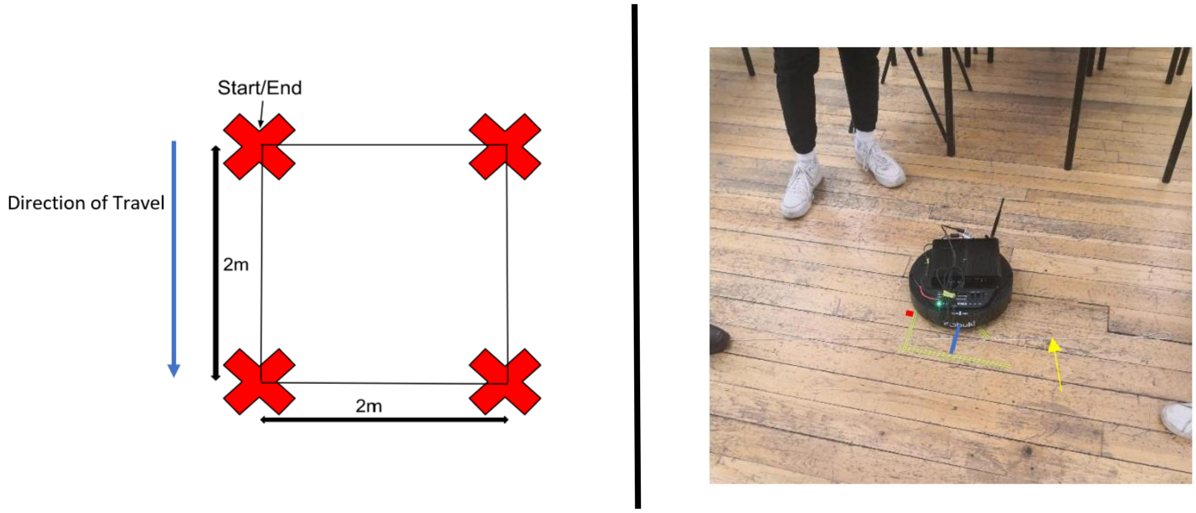

Fig 1. Left: The path to be taken by the robot. Right: Robot at the endpoint

The starting/End point is marked, and the robot travels downwards as stated in (***Fig 1, left)***. The red markers are the points at which the robot makes a 90° left turn until it runs the full loop and returns to the starting point. The distance of which the robot is away from the middle of the starting point is recorded, along the x and y coordinates. Further, the angle (theta) in which it deviates from the starting orientation is also recorded to perform a statistical analysis. In reference to the path figure above, the x and y axis is defined in ***Fig 2***.

X, Y Coordinate axis and the angle measurement:

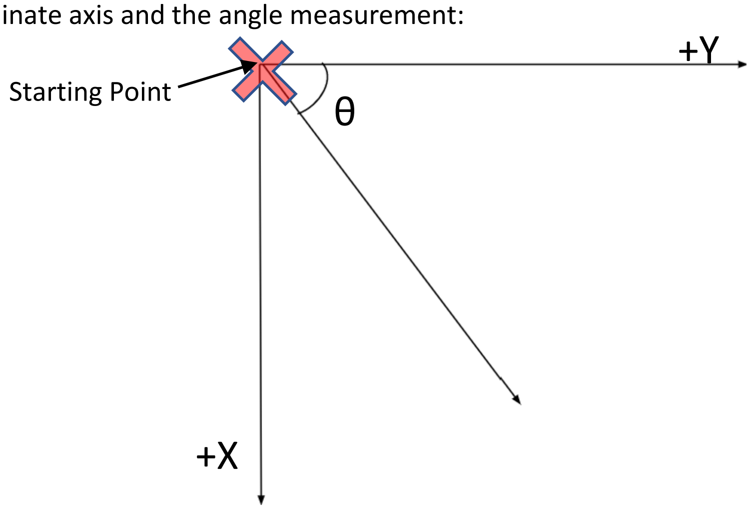

Fig 2. Coordinate system of the Robot

The relative position is recorded according to the coordinate system mentioned.

Measurement Tools

1. X-Axis: A thread was laid out in front of the Robot and was ensured to be parallel to the origin X-axis. Further, a ruler was used to measure the distance between the thread and the origin in mm.
2. Y-Axis: Like X-axis, the same thread is laid out to the side of the Robot and was ensured to be parallel to the origin Y-axis. Further, a ruler was used to measure the distance between the thread and the origin in mm.
3. The delta theta angle (final pose angle): It is calculated using the thread and a precise digital protractor. The thread is laid out in the direction that the robot is facing, and the angle is measured in degrees.

Methodology

For each methodology, a flowchart can be made to simplify the algorithm and give a better understanding as shown in **Fig 3**.

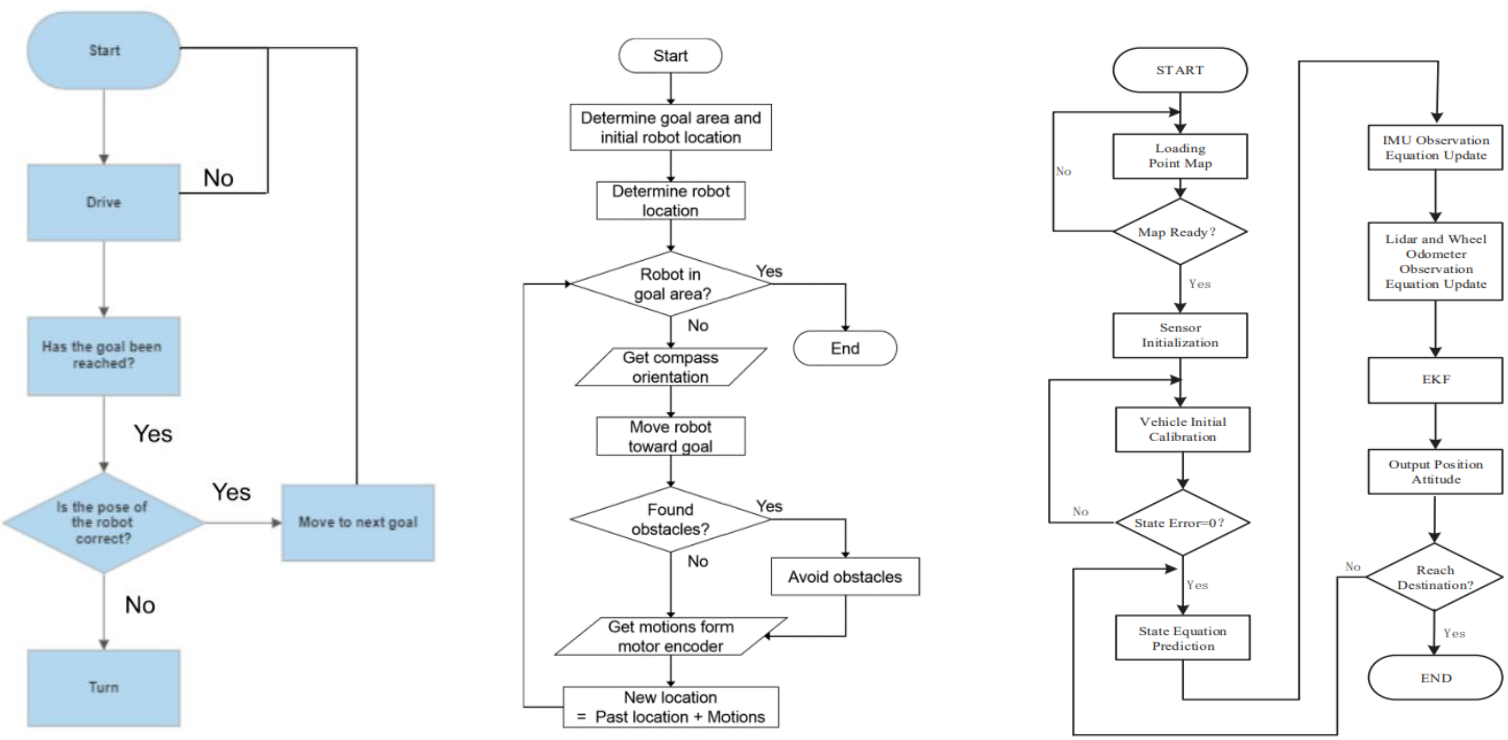

Fig 3. Flowchart of Different Methodologies. Left: Open Loop, Middle: Odometry (IMU Proprioceptive sensor), Right: Odometry + Lidar. [2][3]

The three methodologies explained:

1. Strategy 1: The first flowchart in (Fig.3, left) shows the basic process that was conducted in the first iteration. The flowchart is self-explanatory as it involves driving to the goal in the other corner, and if it has reached it checks the pose of the robot to the next goal. It makes a 90 If the pose is also correct, then it proceeds to the next goal till it reaches the starting point.

2. Strategy In the second flowchart/methodology, the use of odometry principles in the form of an Inertial measurement unit (IMU) sensor. The principle of odometry involves the calculation of the change in position over time and reroutes itself if it is not following the path trajectory. There is a form of self-correction that can be seen as the robot traverses the path. Therefore, this process only uses proprioceptive sensors.

3. The third methodology is a combination of two prominent techniques. It involves a lidar sensor that maps the room for the robot's navigation and the odometry technique for traversing the path trajectory with a rerouting mechanism. This results in a combination of both Proprioceptive (IMU) and exteroceptive (Lidar) sensors. This gives an optimal autonomous robot that can traverse.

Results

Table 1. Error calculation in iteration #1

	x error avg.**(mm)**	**y error** avg. **(mm)**	**theta error** avg. **(°)**
Strategy 1	150.1	-115.85	13.6
Strategy 2	6	-6.45	0.46
Strategy 3	19.45	-0.2	1.25

In Iteration #1 from **Table 1**, it can be seen that there is a strong variance when comparing all the 3 strategies. Strategy 1 shows the highest magnitude of standard deviation error which is expected as methodology #1 has no implementation of sensors. The inability of the robot to reroute itself and follow the path trajectory based on real-time data through sensors leads to high errors on the X and Y axis. Furthermore, the final pose angle calculated in theta is also higher than the other methodologies.

In the comparison of strategies 2 and 3, it can be seen that both produce lower levels of accuracy. An initial evaluation states that strategy 2 has better X-axis and theta angle accuracy. This initially suggests that strategy 2 might be the more accurate, however, the integration of Lidar and IMU sensors should indicate more precision and accuracy. Based on this Iteration #2 is done and analyzed in **Table 2**.

Table 2. Error calculation in iteration #2

	x std.error (mm)	**y std.error (mm)**	**theta std.error (°)**
Strategy 1	267.114	-115.850	13.600
Strategy 2	42.976	63.210	5.998
Strategy 3	27.793	34.470	5.068

Upon further analysis of strategies 2 and 3 in iteration #2, it can be seen that strategy 3 indeed has a lesser standard deviation error than strategy 2. This is expected as the combination of exteroceptive and proprioceptive leads to an effective robot with more precision and accuracy.

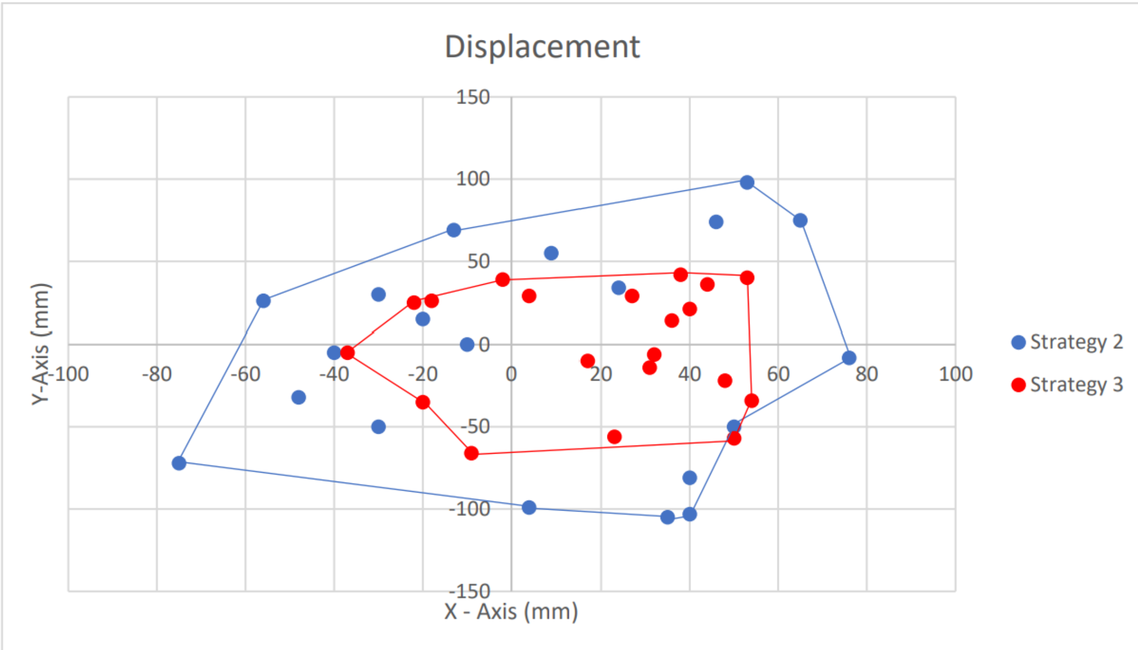

Fig 4. Comparing error values of Strategy 2 vs 3

When comparing all the error data of Strategy 2 and 3 in **Fig 4**, it can be seen that Strategy 2 with IMU has a wider range and spread of data leading to his inconsistencies. Whereas, Strategy 3 has a closer spread of error and is more consistent. Moreover, 3rd methodology data points are clustered better than 2 around x=40mm and y=10mm, indicating a higher precision, and are closer to the origin (0,0), indicating high accuracy. However, there seems to be a high variance in the data as the error between different iterations is high, especially when compared to Webots that run the simulation of the robot where more likely to reach exactly back to the origin (0,0).

There can be several factors as to why both Iterations can have different results from the simulated bot:

1. Webots software doesn't account for the random errors hence the final error that is measured is highly likely to be more than the robot simulation on Webots.
2. The precise placement of the robot on the origin (0,0) may not always be possible due to human error. Additionally, the robot may not be oriented in a perfectly straight direction, which can result in accumulated errors as it moves along its intended path.
3. Technical errors during odometry where the wheel distance is not correctly measured or the lidar scanner fails to map the environment properly can lead to an increase in the final error.
4. Human errors while measuring the final deviation using rules and protractors can lead to discrepancies in the values.
5. Errors such as vibrations or manufacturing defects to inaccurate readings by proprioceptive (IMU) and exteroceptive (Lidar) sensors.

The average error per corner of all the strategies can be given based on iteration #2 as presents a fairly accurate representation of the robot methodologies.

Table 3. Error/corner of all 3 methodologies

Error/corner	x std.error (mm)	y std.error (mm)	theta std.error (°)
Strategy 1	66.779	-28.963	3.400
Strategy 2	10.744	15.803	1.500
Strategy 3	6.948	8.618	1.267

Conclusion

From evaluation, it is concluded that strategy 3 is the most accurate as it poses rerouting and self-adjusting capabilities. This is followed by strategy 2 and then 1. There were several considerations in unforced/random errors, therefore is room for improvement in obtaining the data more accurately. Strategy 3 resulted in the least error/corner of **6.948 mm** on the x-axis**, 8.618 mm** on the y-axis, and **1.267°** deviation per corner. Therefore, a combination of both proprioceptive (IMU) and exteroceptive (Lidar) sensors can make the robot more precise and accurate rather than just having a proprioceptive sensor or no sensor at all. Also, strategy 3 produced more consistent data as compared to other methodologies. The robot that is simulated on Webots may not always reflect the practical application, as there can be random errors that are unaccounted for in the software and mentioned in this report. An overall suggestion is that more iterations can be performed to determine the exact errors from these individual methodologies to justify these conclusions.

References

1. Zhang, T., Li, Q., Zhang, Cs. et al. Current trends in the development of intelligent unmanned autonomous systems. Frontiers Inf Technol Electronic Eng 18, 68–85 (2017). https://doi.org/10.1631/FITEE.1601650
2. Atsuzawa, K.; Nilwong, S.; Hossain, D.; Kaneko, S.; Capi, G. Robot Navigation in Outdoor Environments Using Odometry and Convolutional Neural Network. In Proceedings of the IEEJ International Workshop on Sensing, Actuation, Motion Control, and Optimization (SAMCON), Chiba, Japan, 4–6 March 2019.
3. Zhang, S. et al. (2019) “Lidar-IMU and wheel odometer based autonomous vehicle localization system,” 2019 Chinese Control And Decision Conference (CCDC) [Preprint]. Available at: https://doi.org/10.1109/ccdc.2019.8832695.