Differential Drive Mobile Robot Simulation and Dead Reckoning

Ameer Hamzah and Faizan Aziz

May 24, 2024

Abstract

The aim of this lab is to understand the fundamental method of dead-reckoning which is used for the mobile robot localization. This approach is very common because the robot uses only its internal sensors to localize itself. The implementation was done using the provided simulation and the issues and results are given in the later part of this document.

1 Introduction

Briefly describe what was done in the Pre-Lab, the Lab session, and the Post-Lab. Then describe the organization of the report.

This report is organized as follows: Section 2 describes how to simulate the robot by implementing the `DifferentialDriveSimulatedRobot` class inheriting from the `SimulatedRobot` base class. This section also explains how to use the class to simulate the trajectories requested in the lab. Section 3 details the dead reckoning localization method of a differential drive mobile robot. The technique is implemented in the `DR_3DOFDifferentialDrive` class as a child class of the `Localization` class providing an implementation for their virtual methods. Finally, section 4 is devoted to the description of the problems found during the lab.

2 Simulation

First, we did the simulation part. That was very difficult. There were certain symbols which we did not know how to use. But we did it because we like simulation. Here I will specify the certain functions that we used in the simulation after understanding them. In the simulation, the first class was *SimulatedRobot*. In this class, the a general robot simulation is defined irrespective of the type of the robot (aerial robot, mobile robot or a legged robot). This class is then extended using another child class *DifferentialDriveSimulatedRobot*. Some of the methods and attributes of *SimulatedRobot* are over-ridden.

3 Dead-Reckoning

Dead-reckoning is all about using the encoder values to find the pose of the robot. This technique is widely used in robotics but prone to errors.

4 Issues

Describe here any potential problems you found and how you solved it.

5 Conclusions

References

[1] Pere Ridao. "*Probabilistic Robot Localization and Mapping Python Library Documentation.*"

[2] Ameer Hamzah. *"Center of Robotics Excellence, 2nd Edition"*. Journal. 1994.