CSCI 445 Lab 4 — Odometry Characterization

Prof. Ayanian, University of Southern California Spring 2017

In this lab we will investigate the quality of the odometry calculation of our robot.

1 Prerequisites

- Review odometry method for localization and the sensors used for it.
- Read about odometry error and some of its causes.
- Be familiar with plotting data using matplotlib (task 3 of programming assignment 1).
- Bring your laptop.

2 Turning Wheels Manually

Start streaming the encoder counts or computed odometry values. Then manually rotate the wheels while closely observing the readings. Now put the robot on the ground and manually push the robot to go forward, backward, and turn. In your tasksheet, briefly describe your general observation. Now consult the datasheet for the robot[1] to figure out why that is the case.

3 Rectangular Motion using Feed-forward Commands

In this part you will use simple time-based motor commands to let the robot move and observe the quality of the odometry computation by comparing the estimated state with the true state.

3.1 Simulation

3.1.1 Odometry Quality Analysis

In simulation, make the robot move on a rectangle of sides $1\,\mathrm{m}$ and $0.5\,\mathrm{m}$ at the speed of $100\,\mathrm{mm/s}$ while continuously collecting the locations estimated using robot's odometry. Now plot the ground-truth (robot's actual traversal path) and the odometry-based estimated locations in the same figure and compare. In the tasksheet, briefly describe your observation and analysis for why that is the case.

Hint: You are **not** asked to implement a controller to follow the shape. Instead, estimate the required time for the individual wheel commands manually until the robot roughly reaches the desired shape.

Hint: You can use your own code from Lab 3 or start of with the posted solution (`Lab3Solution.zip`).

Hint: In simulation, you can obtain the ground-truth of your robot's position by using `self.create.sim_get_position()`. It will return a 2-tuple containing (x, y) in meters.

Hint: You should use matplotlib to plot the data.

3.1.2 Different Encoder Update Rates

Remember from the Lab 3 that the default update rate of Create2's wheel encoders is every $15\,\mathrm{ms}$. In all previous tasks, the odometry calculations were done using the encoder counts updated at this default rate. Now, lower the odometry update rate to every $150\,\mathrm{ms}$ and observe the resulting localization data from the

[1] Robot's Datasheet: http://www.irobotweb.com/~/media/MainSite/PDFs/About/STEM/Create/iRobot_Roomba_600_Open_Interface_Spec.pdf?la=en

same rectangular motion as before. Plot the data versus the ground-truth and compare to the plot from task 3.1.1. In the task-sheet briefly describe your observation and the reasons why that is the case.

3.1.3 Different Speeds

Return the odometry update rate to its default rate at every $15\,\mathrm{ms}$. Now try increasing the speed of the wheels to $300\,\mathrm{mm/s}$ and do the same task as in 3.1.1. Compare the new plot to the plots from previous tasks and in your task-sheet briefly describe your observation.

3.2 Robot

3.2.1 — 3.2.3

Repeat all the tasks that you did in simulation in section 3.1 on the robot. For each task, compare the plots to the plots from simulation and in your task-sheet briefly describe your observation and the reasons why that is the case.

Hint: Please note that, for the rectangular ground-truth you need to manually measure and put tapes on the floor to indicate the target rectangular path.

3.2.4 Colliding into a Wall

Now, put the robot on a piece of cardboard provided and directly facing a wall. Have the robot move forward into the wall while monitoring the encoder readings and the collecting odometry data. Plot the odometry data and compare to the actual motion observed. In your task-sheet, briefly describe your analysis.

4 Circular Motion using Feed-forward Commands

Repeat the task in 3.1.2 on a robot, but this time for a circle. In this case, use a much slower update rate of $1\,\mathrm{Hz}$. Plot the data and explain your results.

Hint: Choose a small circle.