EE209AS (Fall 2018)
Computational Robotics

Prof. Ankur Mehta
mehtank@ucla.edu

Problem set 3
Due 2pm Thu. Nov. 1, 2018

Objectives

The goal of this lab is to explore Kalman Filtering (KF) to estimate the state of a simple two-wheeled robot. You will develop and implement a mathematical model of the robot sensor and actuator behavior and use it to evaluate a state estimation algorithm.

Deliverables

This project will require you to write code. Make sure the **well commented** code for this lab is committed and pushed, and submit a link to the repository. For some possible resources on git, see the previous pset.

You may work individually or in pairs on this assignment. Each person needs to submit their own solutions, but the team can submit common code. Indicate on your solutions who you worked with, and for each person identify 1) the specific contributions made by each, and 2) an aggregate percentage of the total work done.

Upload your solutions to gradescope.

Preliminaries

0(a). What is the link to your (fully commented) github repo for this pset?

0(b). Who did you collaborate with?

0(c). What other resources did you use?

0(d). What were the specific contributions of each team member?

0(e). What was the aggregate % contributions of each team member?

1 System definition

You will consider a 2 wheeled robot similar to the one shown in Fig. 1. It has two wheels of radius $r = 20mm$, separated by a distance $w = 85mm$. It drags a tail for stability.

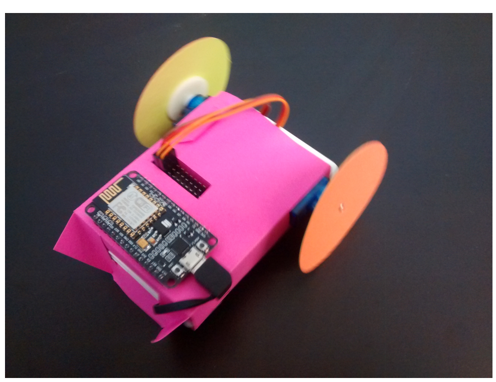

Figure 1: Two wheeled tank-drive robot

Each wheel is powered independently by a bi-directional continuous rotation servo—part number FS90R—with the angular velocity of the wheel controlled by a signal from the microcontroller. This allows the robot to drive forwards or backwards at variable speed, or turn with any turning radius. There may be slippage between the wheels and the floor; assume the resulting effective angular speed of the wheels is normal with a standard deviation of 5% of the max motor speed.

The robot has two laser range sensors—part number VL53L0X—and an inertial measurement unit (IMU)—part number MPU-9250. The output of these sensors will be a function of the positional state of the robot within its environment.

The robot will be driving within a rectangular environment of length $L = 750mm$ and width $W = 500mm$, consisting of 4 walls bounding an open space.

2 Mathematical setup

The state of your robot will consist of the 3DOF pose of the robot in 2D space. You may also want to include velocity terms. In addition, your state may include a gyro bias term for greater accuracy. Thus, your robot state will have at least 3 variables, but may have more.

Your control input will be the angular velocity values you prescribe for each wheel, for a total of 2 input variables.

Your sensor output from the laser range sensors will consist of the distance to a wall in a straight line in front of the robot and the distance to a wall in a straight line to the right of the robot. The IMU will return an absolute bearing indicating the angle of the robot with respect to magnetic north, and can also return an angular rate measurement. We will ignore the accelerometer on the IMU. Thus the robot system will produce 3 or 4 output values.

2(a). Define the system model. You will need to derive and present the mathematical equations describing this robot plant, including the appropriate sensor and actuator models and system dynamics. Be sure to clearly define and describe all variables and equations, and produce illustrative diagrams as necessary. Include realistic noise terms into the model as appropriate.

2(b). Create a Kalman Filter based state estimator to take the motor commands and sensor measurements and generate a state estimate. You will likely want to to implement an Extended Kalman Filter (EKF), but you could choose an Unscented Kalman Filter (UKF) instead. Be sure to explain which algorithm you chose and why, and generate the mathematical expressions.

3 Evaluation

You should implement and debug your algorithms in a simulated environment. You can choose to get the control inputs from a human driver, or implement pre-programmed behavior.

Within the simulation, you should implement models of your sensor and actuator response (especially including noise) to generate realistic sensor traces given the control inputs. Identify any numerical parameters that need to be set, and explain how you selected the values of those parameters. Generating these models may require finding information on hardware datasheets.

3(a). Run some examples that demonstrate the performance (in terms of accuracy and efficiency) of your computed state estimate over time as the robot is issued various commands. Consider both perfect knowledge of the initial state as well as no knowledge of the initial state. Clearly describe the experiments that were run, the data that was gathered, and the process by which you use that data to characterize the performance of your state estimator. Include figures; you may also refer to animations uploaded to your git repo.

3(b). Qualitatively describe some conclusions about the effectiveness of your state estimator for potential tasks your robot may encounter. How might you improve it?