Fundamentals of Mobile Robots

Exercise 2

- **Deadline**: 09.02.2024 (before the exercise class)
- **One** report per group.
- Only the provided template of exercise is acceptable.
- Please use the Teams channel to ask your questions.

Problem 1:

Consider a robot working in 1D discrete space ($0 < x < +10$) with a wall located at $x = 10$ (See Figure 1). Assume that the motion of the robot can be described by $x_{k+1} = x_k + v_k + w$ where x_{k+1} is the next position of the robot after applying the control action, v_k is given velocity as control input, and w denotes the external disturbance. A LiDAR has been attached to the robot in order to measure its distance (d_k) to the wall according to $d_k = 10 - x_k + n$ where n denotes the noise. At the current time, the robot position belief is $\sim N(0, 0.01^2)$. The external disturbance and the sensor noise have a normal distribution of $\sim N(0,\ 0.1^2)$ and $\sim N(0,\ 0.03^2)$, respectively. A template MATLAB code is provided, where you can insert your code.

Note: In this exercise, please assume that X is a continuous variable.

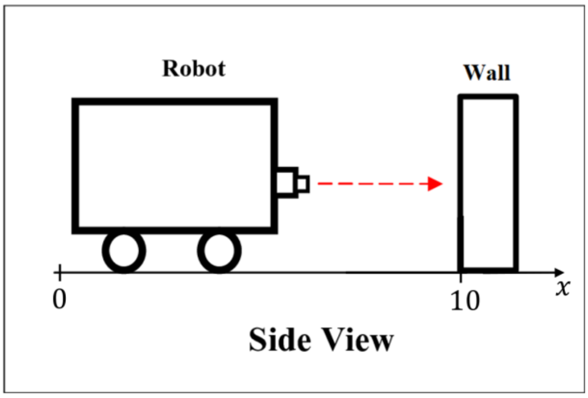


***Figure 1** Differential Drive Robot*

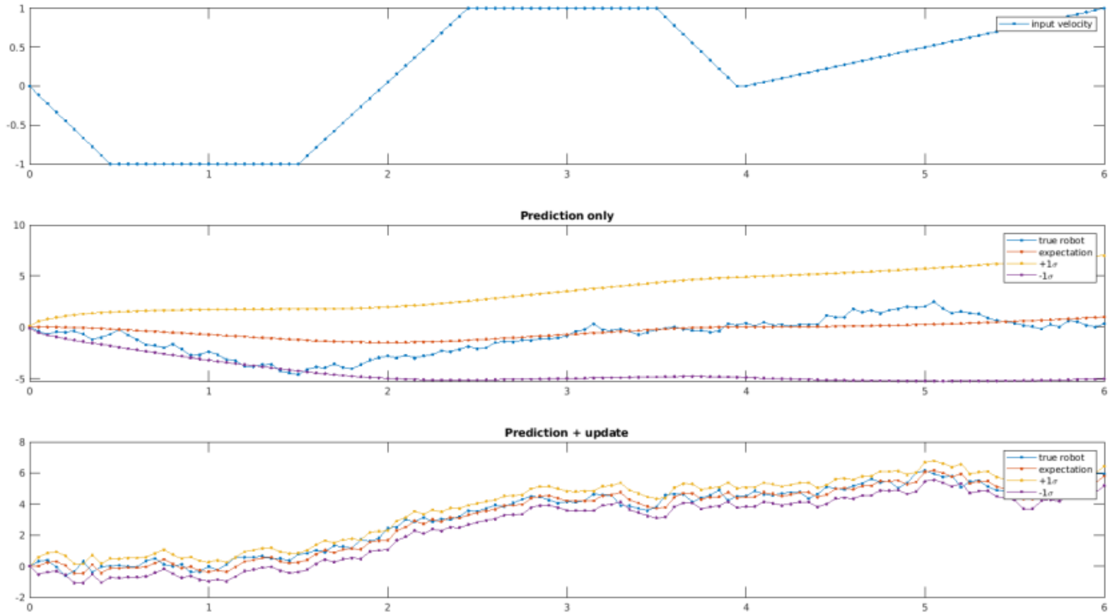
1. **Prediction only**: Calculate the mean and variance of the position ($\hat{x}$ and P) given only the odometry (speed v_k) inputs (no sensor inputs). Plot the true robot trajectory, your prediction mean, prediction mean +/- 1σ versus time: the velocity values, time steps, and other related parameters are provided in the Matlab template.

Note: variance is the square of standard deviation. Usually, the notation we use for standard deviation is σ and for variance is σ^2. In Matlab, you can use $var(x)$ command. In python, the equivalent command is

$$numpy.var(x,\ ddof = 1).$$

2. **Prediction + Measurement**: Assume that the localization system at each step receives odometry values (v_k) and measurement (d_k). Plot the true robot trajectory, your prediction mean, prediction mean +/- 1σ versus time.
3. Discuss your understanding of the two simulations you have done.

Example plots:



Problem 2:

We will use the ROS bag data in *Template/Q2* directory. Assume that the given data in that ROS bag are mean values. (Check slide #7 in the PowerPoint file of lecture 3.) You will simulate the robot from the same initial condition ([0 0 0]) 100 times.

dt = 0.1 sec; speed of wheels in the rosbag are in rpm; the radius of wheels is 0.1m; the distance between the wheels is d = 0.4 m.

1. Complete the provided python template. In the python code, the following calculations happen. All you need to do is just complete the template.

- Initiate 3 vectors for x, y, ψ. . Each vector has 100 elements: each for one instance of simulation. To be more specific, it means that you have 100 robots that move from the same initial point, and because of the noise, they have different paths as they move.
- Simulate the particles (each particle is like one independent robot) according to the following 4-parameter dead-reckoning uncertainty model: [a1, a2, a3, a4] = [20, 6, 25, 8]
- Plot the robot path in the XY plane. Note: to keep the plot clean, plot only the particles at every 700-sample time.

Problem 3 (EKF):

Consider a differential robot moving in an environment where there are three landmarks. Wheel angular velocities (in rad/sec) are provided in ROS bag file in *Template/Q3* directory. The given bag file is approximately 40 seconds long with 5 segments. The first and the last segments are 5 seconds long and the robot has no motion. The middle 3 segments are 10 seconds long, and the robot moves with constant linear and angular velocities which are different for each segment. Wheel radius is $r = 0.1m$, and distance between two wheels is $d = 0.4m$. Considering the following information, answer the questions:

Odometry errors (noticed that the noise distributions are assumed constant)

$$e_{rot1} = e_{rot2} = e_{trans} = N(0{,}0.25)$$

The topic (/Landmark_dist) gives you the distance—range—of the robot with respect to three sensors located as follows. The noise of the three sensors is normally distributed with zero mean and following covariance:

$$R_k = diag(\sigma_{L0}^2, \sigma_{L1}^2, \sigma_{L2}^2) = diag(0.05{,}0.05{,}0.05)$$

Landmark locations:

$$[X, Y, Z, Roll, Pitch, Yaw]$$

$$L_0 = [7.3, -4.5, 0, 0, 0, 0]$$

$$L_1 = [-5.5, 8.3{,}0{,}0{,}0{,}0]$$

$$L_2 = [-7.5, -6.3, 0{,}0{,}0{,}0]$$

1. Complete the template. In the python template, the following calculations happen.

- Create and linearize robot motion and sensor measurement models.
- Perform robot state prediction steps until a new sensor measurement is received. Perform an update step when the new sensor message is received.
- Plot the trajectory of the robot.

2. (Optional) Plot the variance of each state, x, y, ψ. . Discuss your understanding of these figures. (What the magnitude of each variance has to say; What happens when we receive or do not receive updates; etc.)