Principles of Robot Autonomy I
Homework 5
Due Tuesday, December 3rd (5:00pm)

Starter code for this homework will all be through Google Colab. Relevant links are listed for each problem.

You will submit your homework to Gradescope. Your submission will consist of a single pdf with your answers for written questions and relevant plots from code.

Your submission must be typeset in LaTeX.

Problem 1: Kalman Filter for 2D Landmark Localization

Objective: Implement a Kalman Filter to estimate the 2D global positions of four landmarks in an environment with a known robot pose.

Setup: For this problem, we will use a Google Colab notebook linked here:
https://drive.google.com/file/d/1W9Ordl8-KPC5Xh2L5CzBtKnWHUQ4i4Je/view?usp=sharing.
Please make a copy of this notebook to your drive to make changes.

Introduction: In this problem, we'll consider a robot navigating through an environment with the discrete-time TurtleBot Kinematics Model:

$$\begin{aligned} x^r_{t+1} &= x^r_t + V_t \cos(\theta^r_t)\Delta t, \\ y^r_{t+1} &= y^r_t + V_t \sin(\theta^r_t)\Delta t, \\ \theta^r_{t+1} &= \theta^r_t + \omega_t \Delta t. \end{aligned} \tag{1}$$

where the robot pose is known, and is defined by the state vector $\mathbf{x}^r_t = \begin{bmatrix} x^r_t & y^r_t & \theta^r_t \end{bmatrix}$. As the robot navigates through its environment, it receives noisy measurements of the global 2D positions of four landmarks in the environment. These landmarks are stationary objects in the environment, and their state vector is defined as follows:

$$\mathbf{x}^m_t = \begin{bmatrix} x^{m1}_t & y^{m1}_t & x^{m2}_t & y^{m2}_t & x^{m3}_t & y^{m3}_t & x^{m4}_t & y^{m4}_t \end{bmatrix}^T \tag{2}$$

The measurements have an associated measurement uncertainty of $R = 0.25I$.

(i) In this problem, the state vector we aim to estimate is $\mathbf{x}^m_t$. Assuming our linear Kalman Filter dynamics function takes the form $\mathbf{x}^m_{t+1} = A\mathbf{x}^m_t$, **what is matrix A**?

(ii) Assuming our linear Kalman Filter measurement function takes the form $\mathbf{z}_t = C\mathbf{x}^m_t$, where $\mathbf{z}_t$ is the measurement vector of 2D global landmark positions, **what is matrix C**?

(iii) Using the matrices determined in Parts (i) and (ii), implement the linear Kalman Filter on the 2D global landmark positions in the associated notebook. **Include all resulting plots in your writeup.**

(iv) **Briefly answer the following questions:**

(a) What is the value of the Process Noise, Q? What assumption are we considering regarding the landmarks to obtain this value of Q?

(b) What are the main assumptions considered on the system and noise, for Kalman Filters in general?

Problem 2: Extended Kalman Filter for Robot Localization

Objective: Implement an Extended Kalman Filter to estimate the pose of the robot given 2D *relative* position measurements of four landmarks in the environment.

Setup: For this problem, we will use a Google Colab notebook linked here:
https://drive.google.com/file/d/18lM_-A3d_OEhrJPyVfdOlzk7kYyPgiy5/view?usp=sharing.
Please make a copy of this notebook to your drive to make changes.

Introduction: In this problem, we'll consider a robot navigating through an environment with uncertain discrete-time TurtleBot Kinematics Model:

$$\begin{aligned} x^r_{t+1} &= x^r_t + V_t \cos(\theta^r_t)\Delta t + w^x_t, \\ y^r_{t+1} &= y^r_t + V_t \sin(\theta^r_t)\Delta t + w^y_t, \\ \theta^r_{t+1} &= \theta^r_t + \omega_t \Delta t + w^\theta_t. \end{aligned} \tag{3}$$

where the robot pose is unknown, and is defined by the state vector $\mathbf{x}^r_t = \begin{bmatrix} x^r_t & y^r_t & \theta^r_t \end{bmatrix}$. The noise vector $\mathbf{w}_t = \begin{bmatrix} w^x_t & w^y_t & w^\theta_t \end{bmatrix}^T$ is drawn from a zero mean Gaussian distribution $\mathbf{w}_t \sim \mathcal{N}(0, Q)$, where $Q = 0.1\Delta t^2 I$. The global ground truth positions of these landmarks are known. These landmarks are stationary objects in the environment, and their state vector is defined as in Problem 1, Eq. (2). As the robot navigates through its environment, it receives noisy measurements of the 2D positions of four landmarks in the environment relative to the robot's current pose. The measurement equation for landmark 1 is as follows:

$$\mathbf{z}^{m1}_t = \begin{bmatrix} \hat{x}^{m1}_t \\ \hat{y}^{m1}_t \end{bmatrix} = \begin{bmatrix} \cos(\theta^r_t) & \sin(\theta^r_t) \\ -\sin(\theta^r_t) & \cos(\theta^r_t) \end{bmatrix} \left(\begin{bmatrix} x^{m1}_t \\ y^{m1}_t \end{bmatrix} - \begin{bmatrix} x^r_t \\ y^r_t \end{bmatrix} \right) + \mathbf{v}^{m1}_t, \tag{4}$$

where the measurements have an associated measurement uncertainty of $\mathbf{v}^m_t \sim \mathcal{N}(0, R)$, where $R = 0.25I$. The full measurement vector of all landmarks is represented as:

$$\mathbf{z}^m_t = \begin{bmatrix} \mathbf{z}^{m1}_t \\ \mathbf{z}^{m2}_t \\ \mathbf{z}^{m3}_t \\ \mathbf{z}^{m4}_t \end{bmatrix} \tag{5}$$

(i) In this problem, the state vector we aim to estimate is $\mathbf{x}^r_t$. **What is the state Jacobian of the dynamics function, $G(\mathbf{x}^r_t)$, for our EKF?**

(ii) In this problem, the state vector we aim to estimate is $\mathbf{x}^r_t$. **What is the Jacobian of the measurement function, $H(\mathbf{x}^r_t)$, for our EKF?**

(iii) Using the matrices determined in Parts (i) and (ii), implement the Extended Kalman Filter on the 2D relative landmark positions in the associated notebook. **Include all resulting plots in your writeup.**

(iv) EKF considers first order-linear approximations which are tedious to calculate and not generalizable. **Can you think of scenarios for the TurtleBot where this approximation will fail / be inaccurate? Describe any one strategy that could produce more accurate/generalizable solutions?** (**Hint:** One method is to consider other types of filter to model nonlinearities. Feel free to describe any other valid strategy.)

Problem 3: Particle Filter for Robot Localization

Objective: Implement a Particle Filter to estimate the pose of the robot given 2D *relative* position measurements of four landmarks in the environment.

Setup: For this problem, we will use a Google Colab notebook linked here:
https://colab.research.google.com/drive/1pv7wMAk-juGhooIc9pLA8CN3kB0iyJwV?usp=sharing.
Please make a copy of this notebook to your drive to make changes.

Introduction: In this problem, we'll consider the same setup as Problem 2, with kinematics described by Eq. (3), and measurement model described by Eq. (4).

(i) Implement the Particle Filter on the 2D relative landmark positions in the associated notebook. Remember to carefully read the code that is provided! **Include all resulting plots in your writeup**.

(ii) **How do the plots for this Particle Filter compare to the plots from the EKF in problem 2**?

(iii) **Answer the following True/False questions:**

(a) Particle filters depend on the type of measurement and dynamics models.

(b) Particle filters are deterministic.

(c) For unimodal distributions, it is better to use Kalman Filters instead of Particle Filters.

Problem 4: EKF Slam

Objective: Implement an Extended Kalman Filter to do Simultaneous Localization and Mapping (SLAM) of the robot's pose and four 2D marker positions given 2D *relative* position measurements of four landmarks in the environment.

Setup: For this problem, we will use a Google Colab notebook linked here:
https://drive.google.com/file/d/15oI_Bc27CSltGWjdZx58HQWQWd-NELti/view?usp=sharing.
Please make a copy of this notebook to your drive to make changes.

Introduction: In this problem, we'll consider a robot navigating through an environment with the uncertain discrete-time TurtleBot Kinematics Model in Problem 2, Eq. (3), where the robot pose is unknown. The global ground truth positions of the landmarks are now also unknown. These landmarks are stationary objects in the environment, and their state vector is defined as in Problem 1, Eq. (2). As the robot navigates through its environment, it receives noisy measurements of the 2D positions of four landmarks in the environment relative to the robot's current pose. The measurement equation for landmark 1 is the same as in Problem 2, Eq. (4), and likewise the full measurement vector of all landmarks is represented as Eq. (5). In this problem, we aim to estimate the concatenated robot and landmark state vector:

$$\mathbf{x}_t = \begin{bmatrix} \mathbf{x}_t^r \\ \mathbf{x}_t^m \end{bmatrix}. \tag{6}$$

With a 3D robot pose and four 2D landmark positions, our state vector now has 11 state variables.

(i) In this problem, the state vector we aim to estimate is $\mathbf{x}_t$. **What is our dynamics function state Jacobian $G(\mathbf{x}_t)$ for our EKF?** (An easier notation for this would be to use block matrix notation).

(ii) In this problem, the state vector we aim to estimate is $\mathbf{x}_t$. **What is our measurement function Jacobian $H(\mathbf{x}_t)$ for our EKF?**

(iii) Using the matrices determined in Parts (i) and (ii), implement the Extended Kalman Filter for SLAM in the associated notebook. **Include all resulting plots in your writeup.**

(iv) **Briefly explain why the estimated landmark positions and robot trajectories have higher uncertainty compared to problems 1 and 2, respectively.**