Ain Shams University, Faculty of Engineering

Mechatronics and Automation Program

MCT 431: Autonomous Systems

Milestone 4

Deadline: 20 May, 2023 @11:59 PM

1 Milestone Objective

The objective of this milestone is to implement the Kalman filter approach to our Turtlebot3 mobile robot in order to filter the odometry data that is published by our robot.

2 Requirement 1

2.1 Description

1. In Gazebo the odometry data that is published by our Turtlebot3 is fairly accurate. In order to simulate a noisy data that resembles that obtained from a real sensor, you are required to create a python script that subscribes to the odometry data of the Turtlebot3 and adds a random noise to the subscribed odometry data. After the addition of the noise to the original odometry data, you should start publishing the new noisy data. Later you will be applying the kalman filter approach to the noisy data you created.

2.2 Submission

1. For this requirement your going to submit the package you created along with any scripts or files you created in the package.

2. You are required to submit plots and graphs showing the difference before and after the addition of noise on your states and comment on these results.

2.3 Hints & Tips

- Refer to the recording and code of Lab 8 and section 6 to guide you through out your implementation.

- In the case of our mobile robot the three states that your going to add noise to are the X, Y & Theta.

- The package "matplotlib" should aid you in creating the required graphs and plots.

3 Requirement 2

3.1 Description

You are required to implement the Kalman filter approach to filter the noisy data you created in the previous requirement.

1. For simplicity you are required to only implement the kalman filter approach on the state "X" **Only**.

2. You are **Not** required to integrate the previous milestones with this milestone. Launch and run your scripts on a Turtlebot3 in an empty environment.

3. You are required to test your kalman filter on a moving Turtlebot3. For simplicity publish a constant velocity command of 0.2 m/s linear velocity to your Turtlebot3 and use this constant velocity in your code when calculating the state space representation.

3.2 Submission

1. For this requirement your going to submit a video showing Gazebo running in the background with Turtlebot3 launched. You should open a new terminal and run both your scripts. The turtlebot3 should start moving and the kalman filter should start printing the kalman gain.

2. You are required to submit plots and graphs showing the difference between the predicated states, the noisy states and the filtered states and comment on these results.

3. You are required to submit a report briefly explaining the idea of Kalman filter and how it works.

3.3 Hints & Tips

Refer to the recording and code of Lab 8 and section 6 to guide you through out your implementation.

4 Rules and Final submission

1. Cheating in any way is not accepted, any team that copies the Milestone from another team, both teams get a **Zero**.

2. A late submission will result in grade deduction.

3. You are requested to send the entire package you created as a zip file including any codes, folders and files you create inside the package.

4. You are required to send the screenshots you took for each requirement. The screenshots should have a clear view of the terminal (including the lines written), and the name of the computer must be clear and unique for each Team.

5. Send your zipped package, videos and screenshots to the following email: auto.systems.submissions@gmail.com. In the email subject write "Milestone 4 submission". In the body of the email mention your names, IDs and team number. It is required that only **One** person from the team submits the Milestone.