ESE 650, Spring 2019 Assignment 2

Due on Friday February 19th, 2019,11:59PM

This assignment is split into two parts. The first part is worth 25 points and the second points is worth 75 points.

1 EKF (25 Points)

Type your answers in LaTeX in the colored boxes and submit the generated pdf on Canvas. **Show all your work for full credit**.

Problem 1 *Extended Kalman Filter Problem*

Consider a dynamical system defined by the following functions, g, and h corresponding to state update and measurement.

$$g\left(\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}\right) = \begin{pmatrix} x_1 + u_2 x_2 + u_1 x_1^2 \\ u_1 x_2 + u_2 \log(x_1) \end{pmatrix}$$

$h = (x_1/x_2)$

Compute expressions for the relevant Jacobian matrices, G_t and H_t, used in the Extended Kalman filter.

Compute the Kalman gain matrix K_t using the following parameter values.

$\hat{\mathbf{x}}_{t-1} = \begin{pmatrix} 3 \\ 4 \end{pmatrix}$, $\mathbf{u}_t = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$,

$\Sigma_{t-1} = \begin{pmatrix} 8 & 3 \\ 3 & 4 \end{pmatrix}$, $R_t = \begin{pmatrix} 5 & 1 \\ 1 & 3 \end{pmatrix}$, $Q_t = 10$

Note that you should compute Σ_t before computing K_t

2 Estimate Orientations (75 Points)

In this project, you will implement a Kalman filter to track three dimensional orientation. Given IMU sensor readings from gyroscopes and accelerometers, you will estimate the underlying 3D orientation by learning the appropriate model parameters from ground truth data given by a Vicon motion capture system.

This project mainly revolves around implementing the algorithm described in this paper A Quaternion-base Unscented Kalman Filter for Orientation Tracking.

Data is available here https://upenn.box.com/s/0sco8ey93itpdjdssrba2du50lita376

Instructions and Tips

1. You will find a set of IMU data and another set of data that gives the corresponding tracking information from the Vicon motion capture system. Download these files and be sure you can load and interpret the file formats.* The files are given as '.mat' files. Make sure you can load these into python first. (Hint - scipy.io.loadmat)
2. This will return a dictionary form. Please disregard the following keys and corresponding values: 'version', 'header', 'global'. The keys, 'cams', 'vals', 'rots', and 'ts' are the main data you need to use.

3. Note that the biases and scale factors of the IMU sensors are unknown, as well as the registration between the IMU coordinate system and the Vicon global coordinate system. You will have to figure them out.

4. You will write a function that computes orientation only based on gyro data, and another function that computes orientation only based on accelerometer data. Youshould check that each function works well before you try to integrate them into a single filter. This is important!

5. You will write a filter to process this data and track the orientation of the platform. You can try to use a Kalman filter, EKF or UKF to accomplish this. You will have to optimize over model parameters. You can compare your resulting orientation estimate with the "ground truth" estimate from the Vicon.

We will use ENIAC to submit assignments. After you have uploaded your submission to the ENIAC according to instructions in the above link, you may submit your estimate_rot.py **AND ALL FILES IT DEPENDS ON** for grading by running the following command on the ENIAC. For example, if estimate_rot depends on bias.py and motion.py then turn your assignment in using

turnin -c ese650 -p project1 estimate_rot.py, bias.py, motion.py