EECE5554 – Robot Sensing and Navigation

Lab-4

Introduction:

The primary focus of the third lab is to build a navigation stack using two different sensors – GPS & IMU, understand their relative strengths + drawbacks,. The driver was written to collect and parse the data from the sensors which were then published on the GPS and IMU topics. Once the data is collected using the driver it is then analyzed.

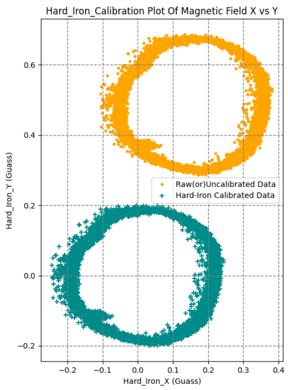
Data Collection:

The first set of data was collected for 5 – 10 minutes by going around in circles at ruggles with the nuance car. The second set of data was collected by going around the city of boston in a route with more than 10 turns.

Data Analysis:

In this section the two sets of data that were collected in the bag files 'data_going_in_circles.bag' and 'data_driving.bag' were analysed and plotted.

1. Estimate the Heading (Yaw)



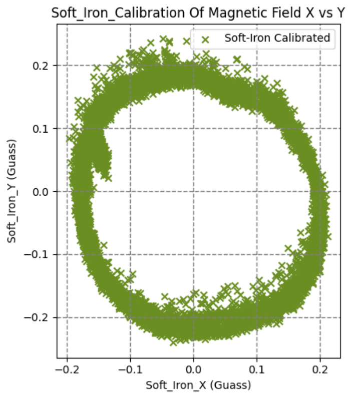


Fig 1: Magnetometer data Raw and Hard Iron Calibrated

Fig 2: Magnetometer data Soft Iron Calibrated

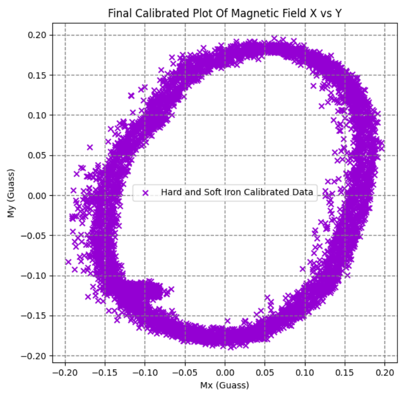


Fig 3: Final Calibrated Magnetometer data

Figure 1,2 and 3 shows the raw/uncalibrated magentometer data and correcting for Hard Iron and Soft Iron Distortions using the ‘Data_going_in_circles.bag’. This process involves translating the ellipse from its center to the origin. This is achieved by subsequently rotating the ellipse and scaling it to become a circle. The plots depicted above displays the all the calibrations and the raw data.

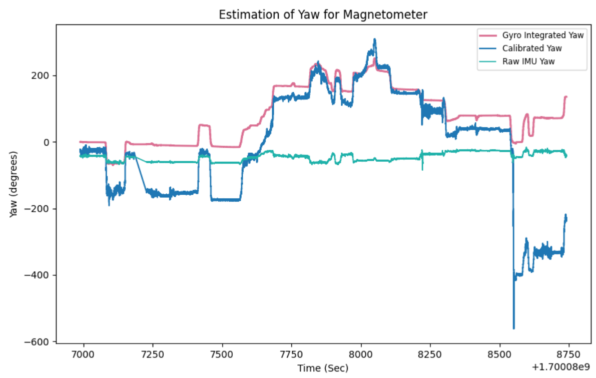


Fig 4: Estimstion of Yaw for Magnetometer

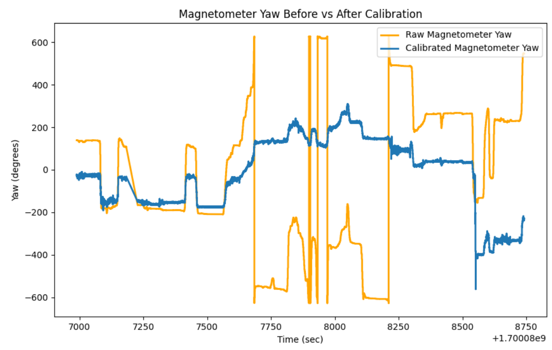


Fig 5: Yaw before and after Calibration

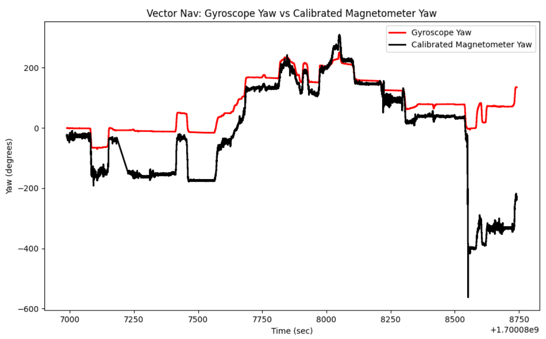


Fig 6: VectorNav Estimations

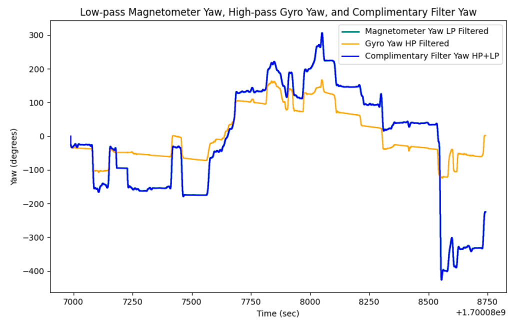


Fig 7: LPF, HPF and complementary Filters

Above plots depicts all the yaw estimations along with filters – Raw Magnetometer Yaw, Calibrated Magnetometer Yaw, Yaw from Gyro, Yaw from the IMU, LPF, HPF and Complementary Filter estimate of the Yaw.The Raw Magnetometer Data is calibrated by adjusting the hard and soft iron distortions and plotted.The ideal scenario would be to use both the magnetometer and gyro yaw by using sensor fusion and applying a filter on it to get the best of both worlds.

2. Estimate the Forward Velocity

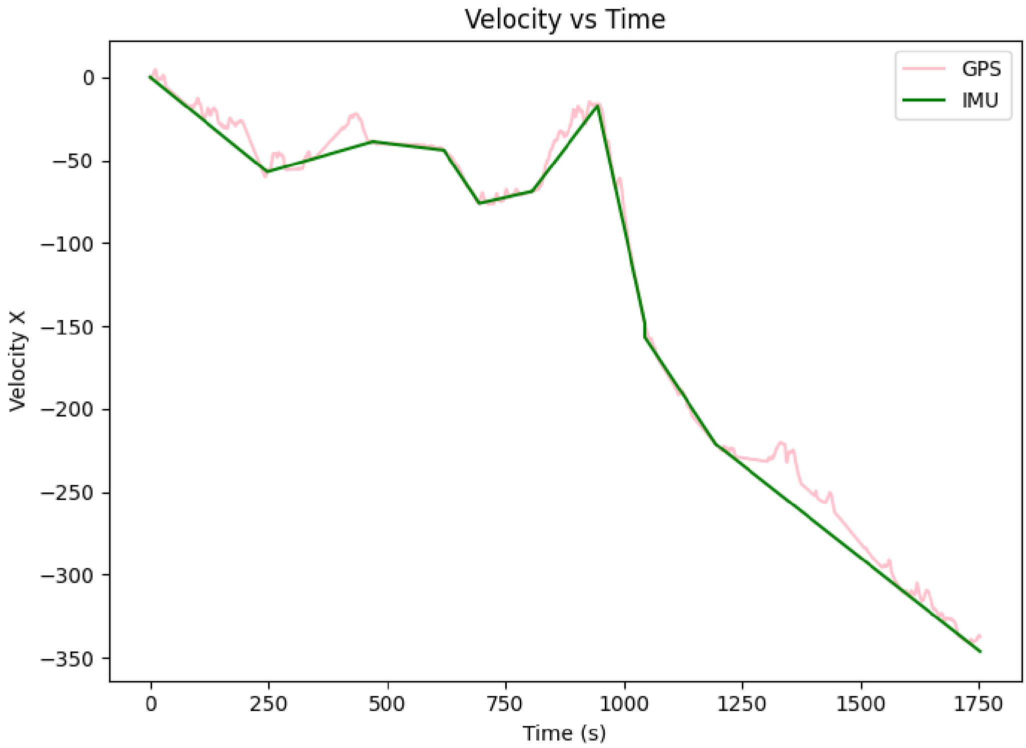


Figure 8: Velocity Estimates before calibration from the IMU and GPS

The plot is based on the data from driving around the boston under normal environment.The plot dpecits the points at zero as the car comes to a stop. Additionally, we could see the dip in the graph that is due to the car exhibiting deceleration at various turns.

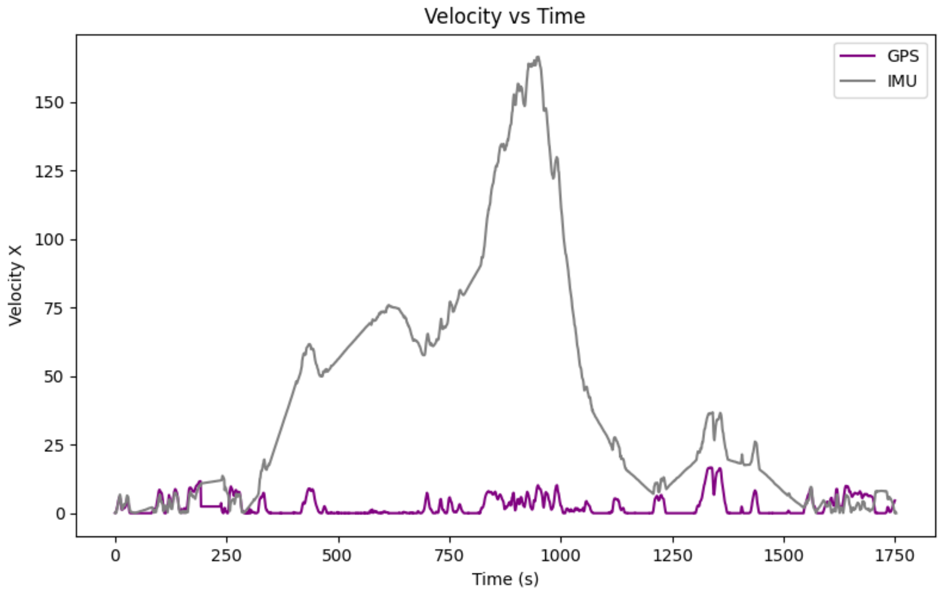


Figure 9: Velocity Estimate from IMU and GPS after Adjustments

3. Dead Reckoning

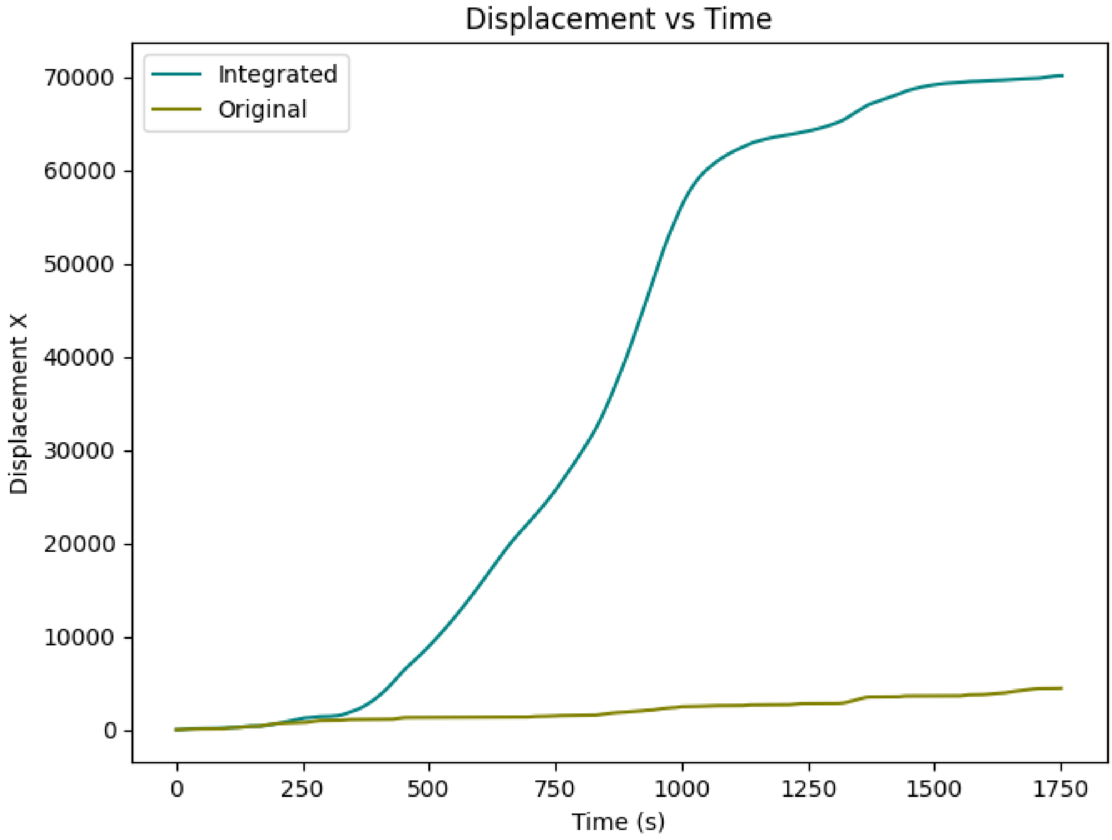


Figure 10: GPS and IMU Displacement

The plot above shows the displacement vs time graph from both GPS and IMU. This shows the absolute displacement of the vehicle is calculated by integrating the GPS and IMU velocities obtained in the previous step.

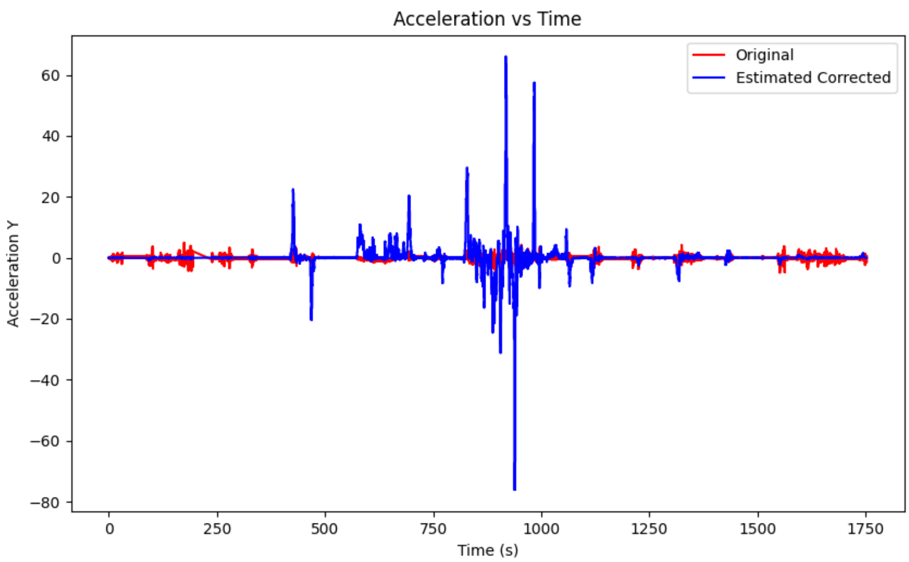


Figure 11: Acceleration vs Time

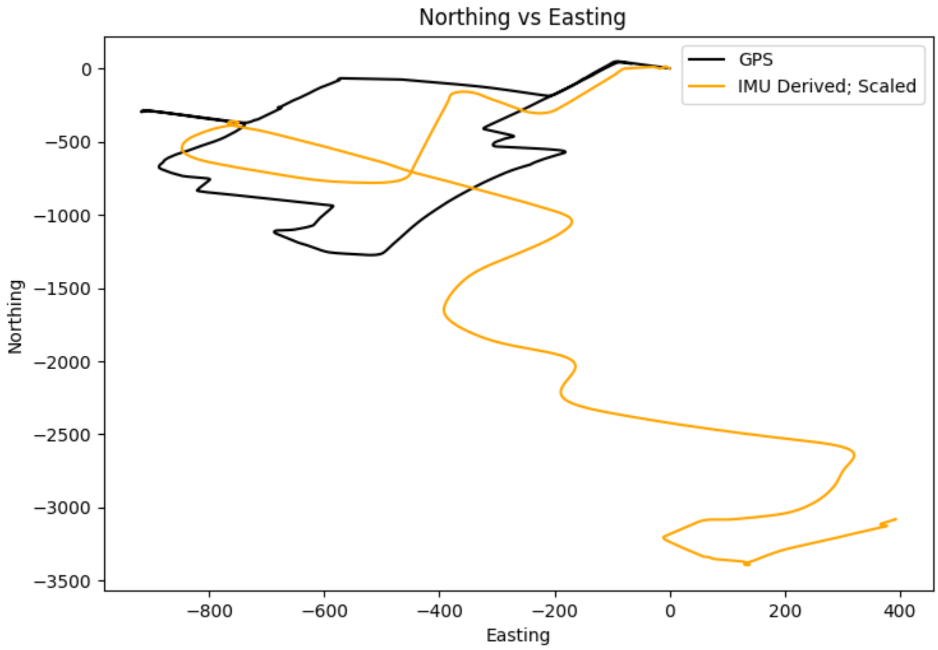


Figure 12: Trajectories

The easting and northing from GPS and the trajectory obtained by integrating the IMU Velocities and their components is shown in the plot. The IMU and GPS plots were made to coincide by applying a scaling factor to the IMU Values to match the GPS values. The scaling factor was found to be 0.09.

Questions :

1. **How did you calibrate the magnetometer from the data you collected? What were the sources of distortion present, and how do you know?**
   The Calibration of the magnetometer data consist of two steps: Hard-iron calibration and soft-iron calibration using scaling. The hard iron is caused by the presence of permanent magnets or ferromagnetic materials near the sensor. So during the hard iron calibration, we detect and correct the distortions. To determine the magnetic field range we calculate the minimum and maximum value along the x and y axes using the raw magnetometer data from MagField.magnetic_field.x and MagField.magnetic_field.y.

   hard_offset_x = (min_x +max_x)/2
   hard_offset_y = (min_y +max_y)/2

   These offset values are then subtracted from the raw values eliminating the hard-iron distortions. Following this we calculate the soft iron calibration where we calculate the major and minor axis of the ellipse using the calibrated data. The soft iron distortions are usually caused by other materials which can be magnetized. We calculate the length of the primary axis by using an elliptical fit. This gives us the distance of the farthest point from center. The distance r can be determined and the value of Theta is found using the arcsine function of the ratio fo the y component to the radius. A rotation matrix is used based on the theta value.Once the hard and soft iron calibration is done we apply a scaling to the calibrated data to address the non uniformity in sensitivity along the axes. The final data is also plotted.

2. **How did you use a complementary filter to develop a combined estimate of yaw? What components of the filter were present, and what cutoff frequency(ies) did you use?**

   In order to combine the magnetometer and gyroscope data we employ the complimentary filter to enhance the accuracy of the yaw angle estimation. This technique uses a first order filter consisting of a high pass filter for gyroscope data and a low pass filter for the accelerometer data.

   The yaw was calculated from the calibrated magnetometer values using the arctan(-y/x) formula. The data was then passed through a low pass filter where the sampling rate was 40 Hz for the calibrated data. The passband frequency was set at 0.1 Hz. Following this the angular velocity along the z axis was integrated and the integrated values were passed through a high pass filter which was sampled at 40 Hz and had a passband frequency of 0.0001 Hz. Further the low and high pass filters were combined and added to find the complementary filter.

3. **Which estimate or estimates for yaw would you trust for navigation? Why?**

   As shown in the above plots, it is visible that the complimentary filter is the most accurate for navigation as it combines the estimates from magnetometer and gyroscope. In case of individual sensor estimates there are multiple disadvantages like influence of magnetic field in magnetometer, drift over time in gyroscope and signal loss in GPS. But the estimate from combination of these sensors will help us overcome the disadvantages of these individual sensors.

4. **What adjustments did you make to the forward velocity estimate, and why?**

   The forward velocity can be calculated by using the accelerometer data from the IMU. We adjust the forward velocity estimate by removing the values where the vehicle was stationary. This is to avoid the increase in bias for forward velocity. First we eliminate any biases present in the data and then calculate the mean of the subset of the accelerometer data. We then center it to zero. The forward velocity was then corrected by integrating the x-axis acceleration. This improved the accuracy of the forward velocity .

5. **What discrepancies are present in the velocity estimate between accel and GPS. Why?**

   We could observe the velocity change with respect to time for GPS and IMU from the plots above. Although the sensors provide a good idea of the velocity they are susceptible to noise and drift over time. This results in inaccurate plots and measurements. In case of GPS, the velocity estimates are precise over longer time periods but the signal can be impacted by multiple factors like mutlipath interference, signal reflection etc. But the data from both accelerometer and GPS can be combined to provide a more accurate estimates.

6. **Compute $\omega\dot{X}$ and compare it to $y\ddot{y}$(obs). How well do they agree? If there is a difference, what is it due to?**

   An inconsistency was noted in the dead reckoning system's output, prompting a detailed examination of factors impacting its accuracy. Various potential sources of disparity were investigated, encompassing sensor noise, integration errors, modeling assumptions, calibration issues, and external forces. To rectify these disparities, strategies such as the incorporation of advanced filtering techniques to mitigate sensor noise, enhancement of integration algorithms, reevaluation and adjustment of modeling assumptions, meticulous sensor calibration, and accounting for external influences were explored. This thorough analysis lays the groundwork for refining the dead reckoning algorithm, aiming to enhance its predictive capabilities to better align with the actual dynamics of the vehicle.

7. **Estimate the trajectory of the vehicle (xe,xn) from inertial data and compare with GPS. (adjust heading so that the first straight line from both are oriented in the same direction). Report any scaling factor used for comparing the tracks.**

   The plots for the trajectory using the GPS and IMU data by using the UTM data and velocity estimates are plotted above. In order to achieve the closest match of both the plots the scaling factor was set at .

8. **Given the specifications of the VectorNav, how long would you expect that it is able to navigate without a position fix? For what period of time did your GPS and IMU estimates of position match closely? (within 2 m) Did the stated performance for dead reckoning match actual measurements? Why or why not?**

   We could see from the trajectory plot that the estimated position closely matched the actual position for some time from the initial point. But then the IMU plot started to deviate slightly from the GPS. The stated performance for dead reckoning did not match the actual measurements due to various factors such as sensor noises, drift, vehicular movements etc.

9. Estimate xc and explain your calculations

9. Estimating X_c

Let

$r = (X_c, 0, 0) \Rightarrow$ Const

$\omega = (0, 0, \omega)$

$v = V + (\omega \times r)$

$\ddot{x} = \dot{v} + \omega \times r$
$= \ddot{x} + \dot{\omega} \times r + \omega \dot{x} + \omega(\omega \times r)$

$\omega = \omega \hat{k} \rightarrow$ z axis
$r = x_c \hat{i} \rightarrow$ x axis

$\ddot{x} = \ddot{x} \hat{i}$
$\dot{\omega} = \dot{\omega} \hat{k}$
$\dot{x} = \dot{x} \hat{i}$

$\ddot{x}\hat{k} = \ddot{x}\hat{i} + \dot{\omega} x_c \hat{j} + \omega \dot{x} \hat{j} - \omega^2 x_c \hat{i}$

there is no $\hat{j}$ component in $\ddot{x}$

$\dot{\omega} x_c + \omega \dot{x} = 0$

$x_c = \frac{-\omega \dot{x}}{\dot{\omega}} = -\frac{\omega \dot{x}}{a_x / r'}$

$x_c = -\frac{\omega r' \dot{x}}{a_x}$

where
$\omega \rightarrow$ ang. vel z
$\dot{x} \rightarrow$ linear vel x
$a_x \rightarrow$ linear acc x
$r' \rightarrow$ Instantaneous center of curvature

X_c Estimate:

$X_c = -\frac{\omega r' \dot{x}}{\ddot{x}}$ //