LAB-4

IMU & GPS Localization

EECE5554-Prof. Singh Shriman Raghav Srinivasan

Introduction

Positioning is one of the important challenges in transient environments and we use a selection of sensors like IMU's and GPSs to achieve it. Here, for this reason, we have used VN-100 IMU sensor to collect data of accelerometer, gyroscope and magnetometer and a BU-353S4 GPS sensor to collect the GPS positioning of the body through the filtration of GPGGA string. This report explores the utilization of sensor fusion for the accurate vehicle trajectory estimation by combining data from GPS and IMU sensors. Sensor fusion techniques are essential in robotics and autonomous systems to mitigate individual sensor limitations - such as GPS inaccuracies in urban areas due to PDOP or IMU drift over time. Various filtering methods, including high-pass filtering and low-pass filtering are applied to extract meaningful insights of noisy data.

Data Collection (Part-A)

The data was collected in two different ways – the first one was us collecting the data using one of our teammate's car (Thanks to Jonathan Bear) by going in a circular pathway around the Ruggles circle near the Centennial common of Northeastern University. The driver used was Jonathan Bear **(Gitlab ID-Bear.jo**)We went around the circular pathway about 4 times with the GPS sensor placed on top of the car and the IMU sensor strapped in front on the dashboard. The second set of data was collected as the car was driven through the city of Boston taking calculated turns and avoiding stoppage through the traffic as much as possible. Also, it was ensured that the start and end of the data collection ended in the same place thus making it easier to plot the trajectory of the path taken. To facilitate the data collection in both the scenarios, the drivers from LAB-1 and LAB-3 were used, and separate rosbag were collected.

Data Analysis (Part -B)

The analysis of the collected data has been performed entirely in MATLAB with the code being written in sectional format for easy execution and debugging.

Q1. How did you calibrate the magnetometer from the data collected? What were the sources of distortion present and how did you know?

The data collected by going in circles around the Ruggles circle is to be used to calibrate the magnetometer by removing the hard iron and soft iron errors.

Magnetometers are used to measure the strength and direction of magnetic field at their location, but since Earth's magnetic field Is not always uniform, there are found to be sources of distortions that affect the measurements of magnetometers such as:

i. Hard Iron error and correction
   Hard iron error or distortion is caused by materials that display a constant field to Earth's magnetic field, thus giving a constant additive value to the output of each magnetometer axis.

The graph in *figure 1* shows the Raw non-corrected magnetometer data.

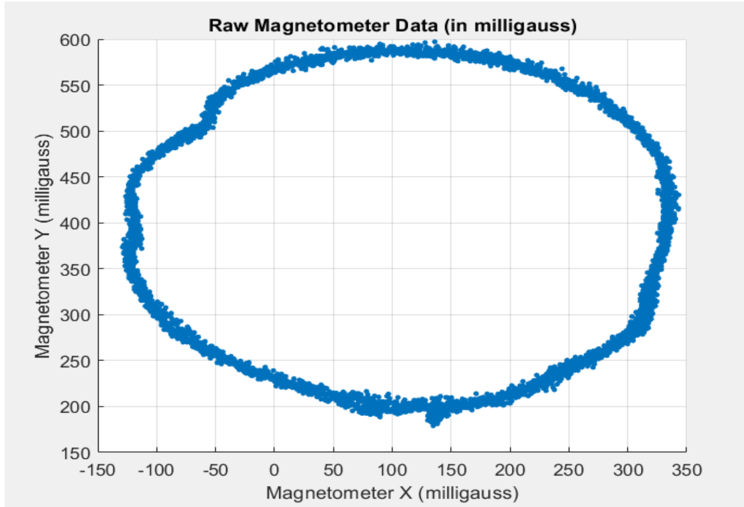

Figure-1 Raw Magnetometer Data

Hard iron distortion can be removed by removing the offset from the raw magnetometer readings and centering it at (0,0).

$$Xoffset = \frac{[\max(x) + \min(x)]}{2} \quad Yoffset = \frac{[\max(y) + \min(y)]}{2}$$

These offsets were then subtracted from raw magnetometer values to get the hard iron corrected magnetometer data.

$$magX\ corrected = magX\ data - Xoffset$$
$$magY\ corrected = magY\ data - Yoffset$$

ii. <u>Soft iron error and correction</u>

Soft iron error or distortion occurs when any material such as ferromagnetic ones in the influence of magnetic field alters the magnetic field but does not create any magnetic field on its own. This error is usually affected by material's orientation relative to sensor and magnetic field. This error can be found when an ellipse with an angular rotation Is made. To eliminate this error, first we need to estimate the length of the primary axis using elliptical fit. Then, we need to determine the max and min of x and y and determine the distance r which is given by:

$$r = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}$$

After this we eliminate the soft iron distortion by multiplying the hard iron corrected mag values with a scaling factor which is the ratio of length of major to minor axis.

$$scaling\ factor = \frac{scaleX}{scaleY}$$

$$scaleX = \max(abs(magX\ corrected))$$
$$scaleY = \max(abs(magY\ corrected))$$

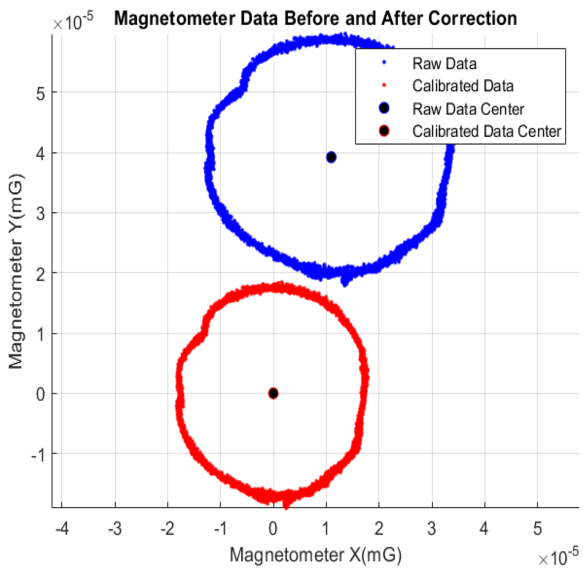

Figure-2

Figure-2 shows the corrected data of magnetometer that was taken by going in circles.

Q2. 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?

To determine the combined yaw estimate, the calibrated data from the previous step of magnetometer calibration was used and then acted upon the dataset that was collected by going around the streets of Boston. The steps for the calculation of combined estimate of yaw is given as follows:

I. The raw yaw angle is calculated from magnetic field x and magnetic field y values of the magnetometer present in the IMU by :

$$raw\ yaw = \arctan\left(\frac{mag\mathrm{Y}}{mag\mathrm{X}}\right)$$

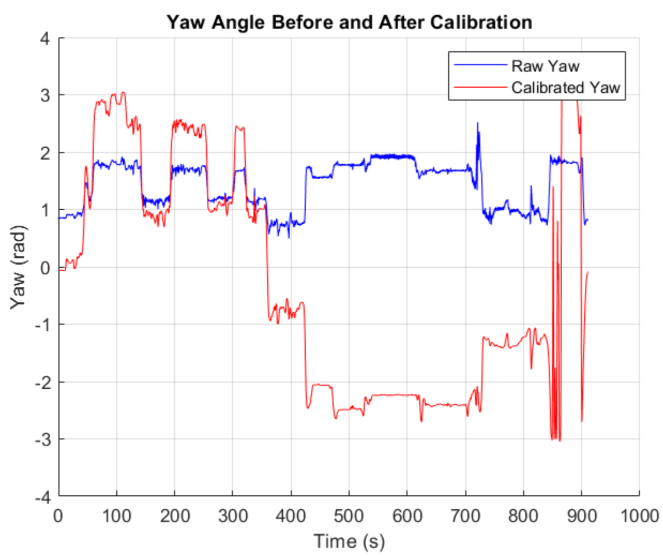

Figure-3

II. The corrected yaw angle calculated as per the given formula:

$$calibrated\ yaw = \arctan\left(\frac{mag\text{Y}corrected}{mag\text{X}corrected}\right)$$

And then plotted as shown in *figure-3*.

III. The corrected yaw angle is then unwrapped and then the unwrapped yaw angle is integrated using a custom integration function which integrates based on gyroscope's z angle that we obtain from the IMU given by the formula:

$$yaw\ angle = \int angular\ velocity\ z\ .\ d(time\ vector\ imu)\ \approx custom\_integration$$

Figure-4 shows the plot between the gyroscope's yaw angle value and the calibrated yaw angle value calculated from the previous calculations. It is observed that the values are almost similar denoting its accuracy.

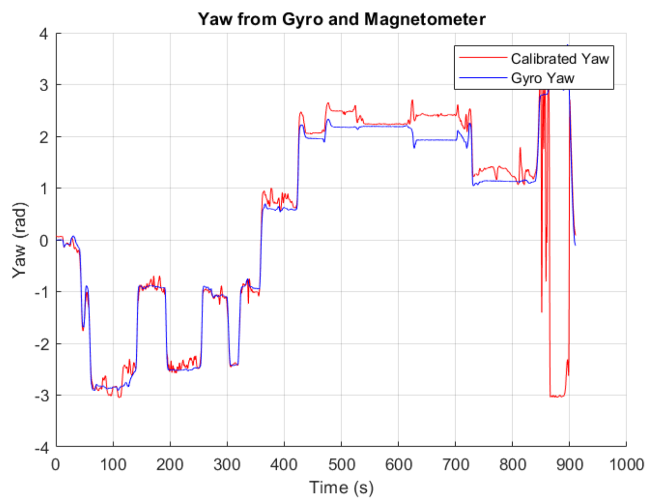

Figure-4

IV. This yaw angle is then passed through the low pass filter with a cut-off frequency of 0.15 and then through a high pass filter with the cut-off frequency of 2. This can be observed in Figure-5 where the responses of each filter are plotted for comparison. It is here observed that the complementary filter produces the most promising signal response.

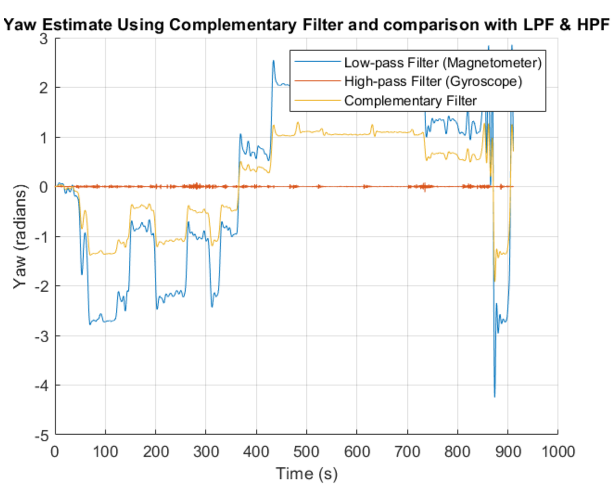

Figure-5

V. After these calculations, the angle is through the complementary filter and plotted graphically and compared with the original yaw from the imu as shown in *Figure-6*.

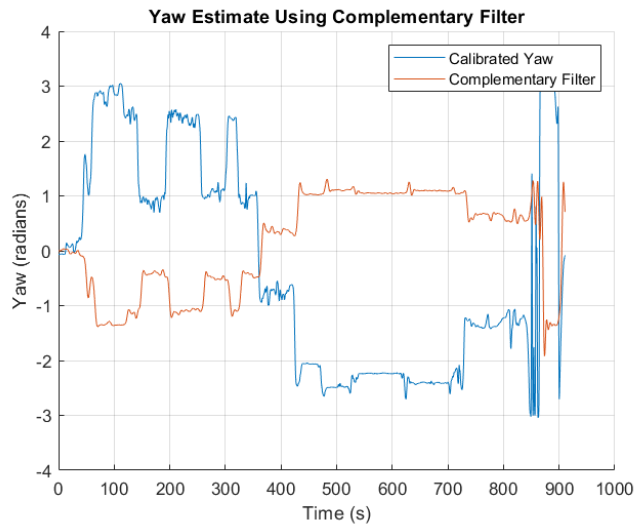

Figure-6

Q3. Which estimate or estimates for yaw would you trust for navigation? Why?

The below reasons culminate the fact that the yaw from the complementary output as the better for navigation than the yaw computed from the IMU as seen in Figure-6.

- The complementary filter output is obtained as the sum of the high pass and low pass filters and smoothened out and it is more evident that only the complementary filter fuses the data from the gyroscope as well as the magnetometer providing us with a reliable yaw angle.
- When the trajectory paths were estimated in the end by considering both gyro integrated yaw as well the complementary filter yaw for velocity estimation and path estimation separately, promising results were observed in the complementary filter output in comparison to others.

Q4. What adjustments did you make to the forward velocity estimate and why?

The forward velocity is estimated by obtaining velocities from IMU sensor's accelerometer as well as the GPS sensor. The following steps were followed in the determination of forward velocity:

i. For obtaining the velocity from the IMU data, the value from the accelerometer acceleration was directly integrated to obtain the velocity from the IMU. But, due to the presence of negative bias, there was need to offset the negative bias of the forward acceleration X of the IMU. The offset was calculated by first initializing a stationary index of timestamps to zero for referencing and then taking the mean of forward acceleration of that modified timestamps and subtracting it from the original acceleration values and plotted as seen in *Figure-7*.

$$Accelration\ offset = mean\big(forward\ accelration(stationary\ index)\big)$$

$$Corrected\ Accelration = Forward\ Accelration - Accelration\ offset$$

$$forward\ velocity\ imu = \int Corrected\ Accelration.d(Timestamps\ for\ driving)$$

$$forward\ velocity\ imu = cumtrapz(driving\ timestamps, driving\ corrected\ accelration)$$

ii. For obtaining the velocity from the GPS data, a different method was used. First the difference between the latitude and longitude was calculated in radians

$$dlat = deg2rad(latitude(i) - latitude(i-1)$$
$$dlong = deg2rad(longitude(i) - longitude(i-1)$$

Then, using the Havesine formula method, the great circular distance between the 2 points was calculated and then the final value was multiplied with the Earth's radius to give the actual distance between 2 points.

$$a = \sin^2\left(\frac{dlat}{2}\right) + \cos\big(latitude(i-1)\big).\cos(latitide(i).\sin^2\left(\frac{dlong}{2}\right)$$

And final value,

$$c = 2.\tan 2(\sqrt{a}, \sqrt{1-a})$$
$$distance = R(637100) * c$$

Similarly difference of time were also calculated and then, velocity was calculated by dividing the distance by the difference time.

$$dlifference\ time = gps\ time(i) - gps\ time(i-1)$$

$$driving\ gps\ velocity = \frac{distance}{difference\ time}$$

iii. After calculating both the velocities, it was plotted as shown in *Figure 8* for reference. As seen in Figure-7, there is a positive bias observed in the IMU velocity after the offset correction while the gps velocity is seen to be accurate and hovering around near the x axis line.

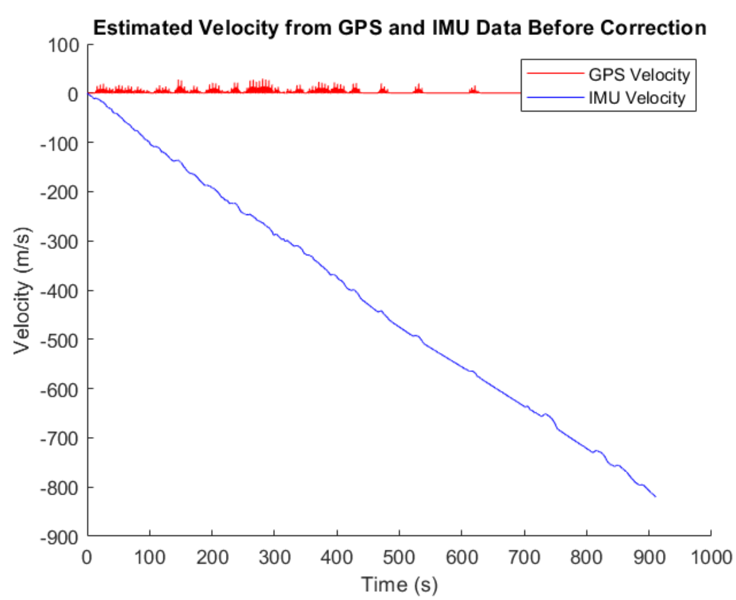

Figure-7

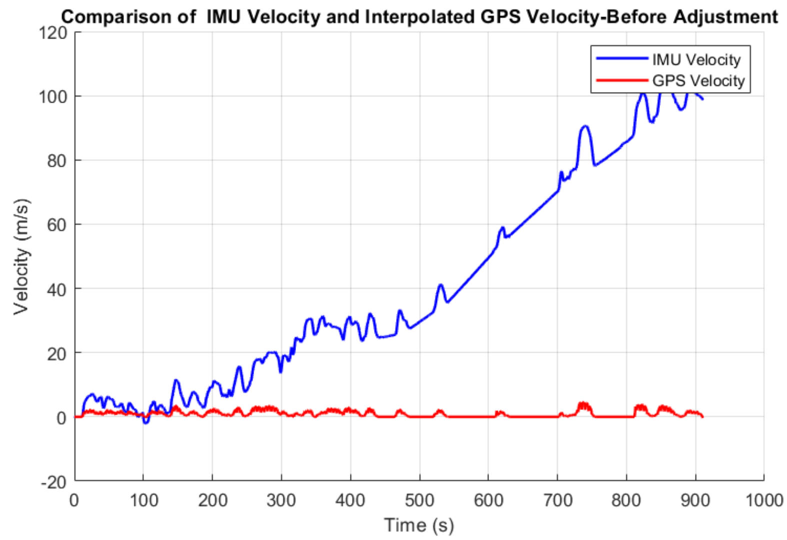

Figure-8

iv. This positive bias was rectified using a high pass filter and passing the calculated imu velocity through a high pass filter of cut off frequency 0.1 and plotted as seen in the Figure 9. From this figure, it can be observed that the IMU and GPS velocities are cutoff only in positive part since there was another offset to detrend the negative values and filtered it out and providing a promising response.

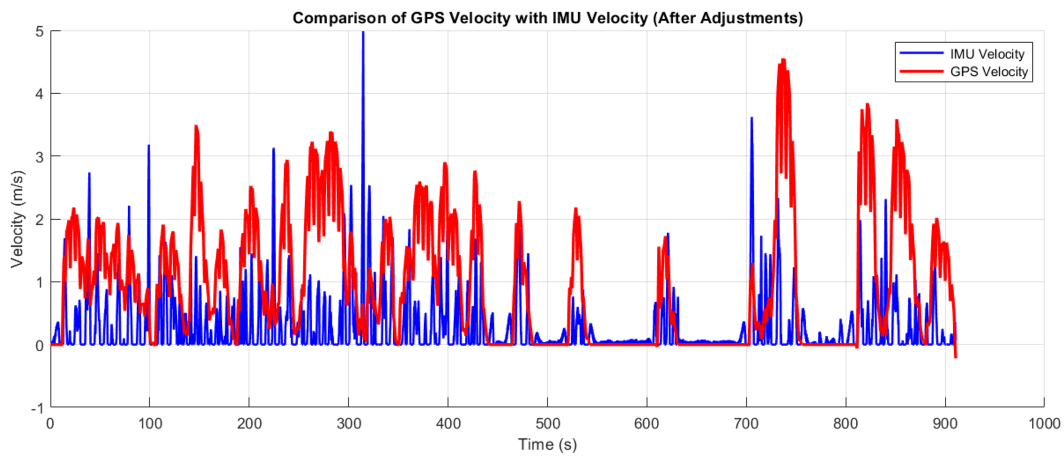

Figure-9

Q5. What discrepancies are present in the velocity estimate between accel and GPS. Why?

The discrepancies that were observed in the velocity obtained from the IMU acceleration is the presence of negative bias which is not at all observed in the GPS velocities. This is because the velocity estimate from GPS acceleration signifies the perfect zero velocity points at points wherever there is a halt or stop due to traffic. The velocity estimate from IMU shows some error and bias which then required a high pass filtering.

Q6 Compute ωX and compare it to $y^{\cdot\cdot}$obs . How well do they agree? If there is a difference, what is it due to?

After computing the velocity estimates, we need to estimate the vehicle orientation and velocity throughout the drive, in order to estimate the trajectory. First we compare the connection between linear acceleration in Y(Y'') , yaw rate about the z axis(ωX) and velocity in X direction(X')

The ωX and y'' obs were obtained using the formula :

$$\ddot{x}_{obs} = \ddot{X} - \omega\dot{Y} - \omega^2 x_c$$
$$\ddot{y}_{obs} = \ddot{Y} + \omega\dot{X} + \dot{\omega} x_c$$

As given in the document, the notation taken for the position of the center of mass of the vehicle and its rotation rate about the center of mass CM (0,0, ω). The position of inertial sensor in space is denoted by (x,y,0) and its position in the vehicle frame as (X_c, 0,0) .With assumptions Y =0 (no lateral skidding) and X_c = 0, these equations simplify to

$$\ddot{y}obs = \omega\dot{X}$$

$$X = cumtrapz(driving\ timestamps, forward\ accelration\ corrected)$$
$$\omega\dot{X} = angular\ velocityZ.X$$

By applying these relations, and substituting the necessary values , Y'' was calculated, but noise was significantly observed in the Y'' obs therefore a low pass filter with a cut off frequency of 0.2 of order 4 was applied to reduced the noise characteristics and then it was plotted as observed in Figure-10.

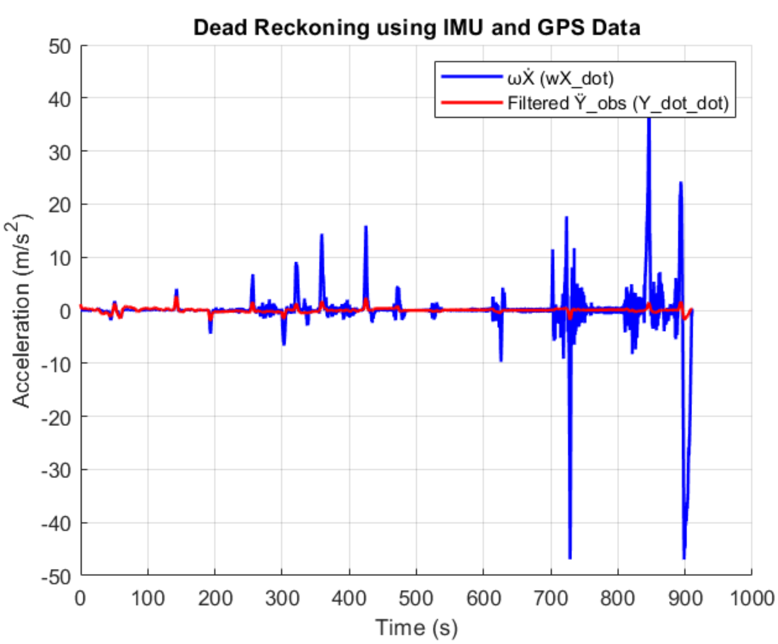

Figure-10

Q7. Estimate the trajectory of the vehicle (xe,xn) from inertial data and compare it with GPS. (adjust heading so that the first straight line from both is oriented in the same direction). Report any scaling factor used for comparing the tracks

For the estimation of trajectory, the IMU's forward velocity was decomposed further to Easting and Northing components using the aligned yaw.

$$Ve = velocity * \cos\,(aligned\ yaw)$$
$$Vn = velocity * \cos\,(aligned\ yaw)$$

Also, the headings of gps as well as IMU's were matched together with gps heading going by the formula:

$$gps\ heading\ = \arctan((northings\ (i) - northing(i-1), eastings(i) - eastings(i-1))$$
$$imu\ heading\ = aligned\ yaw$$

Finally, to determine the easting displacement and northing displacement both the velocity components were integrated with respect to the time of driving dataset as a vector.

$$Xe = \int Ve.\,d(time\ driving) \approx cumtrapz(time\ driving, Ve)$$
$$Xn = \int Vn.\,d(time\ driving) \approx cumtrapz(time\ driving, Vn)$$

These GPS and IMU paths were converted into relative coordinates to better visualize the trajectory's shape and orientation.

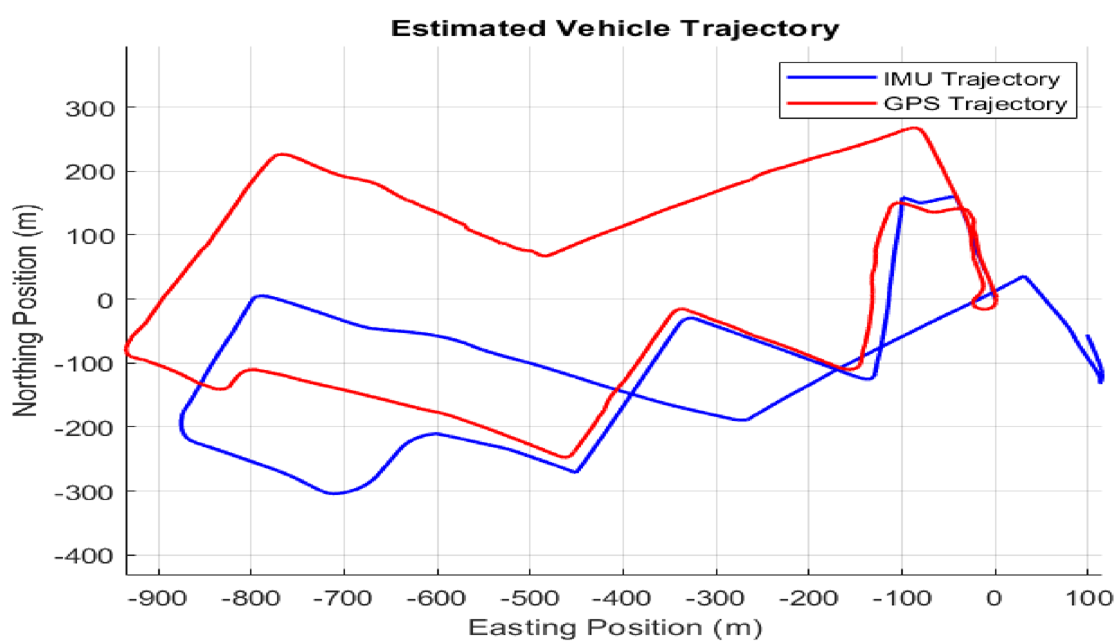

Figure-11

Figure-11 depicts the trajectory that has been estimated from the GPS and IMU values.

Q8 . Given the specifications of the VectorNav, how long would you expect that it is able to navigate without a position fix? For what period 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?

Given the specifications, I don't really expect the VectorNav IMU to perform much better than it has given results in this case. My judgement is because this IMU does not have any feedback mechanisms through which the errors and bias occurring could have been reduced while in data collection. Although GPS and IMU estimates of positions match closely at some points for some amount of time, we can notice that if position fix is applied when observe almost similar results.

Q9. Estimate xc and explain your calculations (bonus up to 100%)

The Xc value is estimated --Xc: **0.2785 meters**

$\ddot{x}_{obs} = \ddot{X} - \omega^2 x_c \rightarrow (1) \Rightarrow \ddot{X} = \ddot{x}_{obs} + \omega^2 X_c$

$\ddot{y}_{obs} = \omega \dot{X} + \dot{\omega} X_c \rightarrow (2) \Rightarrow \dot{X} = \dfrac{\ddot{y}_{obs} - \dot{\omega} X_c}{\omega}$

$\Rightarrow$ Differentiating x_c

$\rightarrow \ddot{x}_c = \dfrac{[\dddot{y}_{obs} - \ddot{\omega} X_c]\,\omega - \dot{\omega}(\ddot{y}_{obs} - \dot{\omega} X_c)}{\omega^2}$

$\rightarrow$ Comparing with $\ddot{x}$ from (1)

$\rightarrow \ddot{X}_{obs} + \omega^2 x_c = \dfrac{(\dddot{y}_{obs} - \ddot{\omega} X_c)\,\omega - \dot{\omega}(\ddot{y}_{obs} - \dot{\omega}^2 X_c)}{\omega^2}$

$\rightarrow \omega\ddot{\omega} x_c + \omega^4 x_c - \dot{\omega}^2 x_c = \omega \dddot{y}_{obs} - \dot{\omega}\ddot{y}_{obs} - \omega^2 \ddot{x}_{obs}$

$\Rightarrow x_c = \dfrac{\omega \dddot{y}_{obs} - \dot{\omega}\ddot{y}_{obs} - \omega^2 \ddot{x}_{obs}}{\ddot{\omega}\omega - \dot{\omega}^2 + \omega^4}$