LAB 4

Navigation with IMU and Magnetometer

Note: The bag files used for the analysis is taken from a different group (Sankara, Francis, Tejaswini, Damla) since there was an IMU orientation issue during my group’s data collection. Thanks to them. The used bag files are uploaded in gitlab repository.

The Driver used for data collection will be uploaded by Abinav.(https://gitlab.com/anantharaman.ab/eece5554)

MAGNETOMETER CALIBRATION

The magnetometer data from IMU sensor for car driving in circles is plotted as shown below.

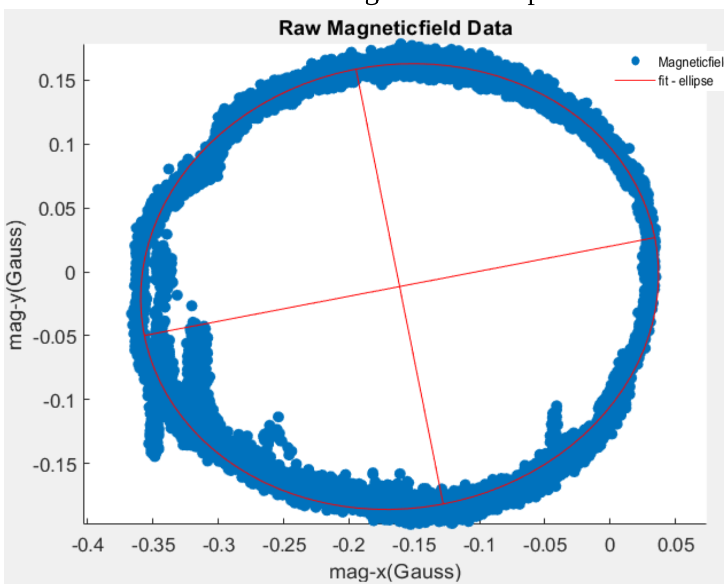

Fig 1: Magnetic field before correction – data driving in circles

As shown in the figure, the raw data obtained is affected by the hard iron and soft iron distortions. The magnetic field data with no distortions should have a circular shape centered at (0,0). Hard iron distortions are created by the materials near the environment of magnetometer that produces magnetic field. The permanent bias (offset) is caused by this hard iron effect. Soft iron distortions are caused by metals like Nickel and iron that influence or distorts the magnetic field. This causes the distortion of circular output of magnetic field.

Hence the data is calibrated using the following equation given in the VectorNav Library Reference Booklet.

$$\boldsymbol{m_c} = S_I(\tilde{\boldsymbol{m}} - \boldsymbol{b_{HI}})$$

$$\begin{bmatrix} m_{c_x} \\ m_{c_y} \\ m_{c_z} \end{bmatrix} = \begin{bmatrix} C_{00} & C_{01} & C_{02} \\ C_{10} & C_{11} & C_{12} \\ C_{20} & C_{21} & C_{22} \end{bmatrix} \begin{bmatrix} \tilde{m}_x - b_{H_0} \\ \tilde{m}_y - b_{H_1} \\ \tilde{m}_z - b_{H_2} \end{bmatrix}$$

Fig 2: Calibration Equation

The data obtained is corrected by removing the hard -iron and soft iron distortions using the following steps.

a. Hard Iron Correction – Subtracted the offsets (b_{HI}) from magnetometer raw data. The offset values considered are the center of an ellipse fitted in the raw magnetometer data shown in figure 1.

$$b_{HI} = [Xo, Yo]' = [-0.1604, 0.0197]'$$

b. Soft Iron Correction –

i. Rotate all the data points by an angle *theta* at which the major axis of fitted ellipse is making with the x-axis. Rotation matrix can be obtained using the equation below. *Theta* obtained is 0.1936 rad.

$$R = \begin{bmatrix} \cos(theta) & \sin(theta) \\ -\sin(theta) & \cos(theta) \end{bmatrix} \qquad (1)$$

ii. Scale the points in the rotated data points by using the matrix in equation 2. Scale factor can be obtained by using equation 3, where ‘b’ is the minor axis length and 'a' is the major axis length. Scale factor obtained is 0.8712.

$$R = \begin{bmatrix} sf & 0 \\ 0 & 1 \end{bmatrix} \qquad (2)$$

$$scale\ factor(sf) \ = \frac{b}{a} \qquad (3)$$

iii. Rotate back all the data points by an angle *theta.*

The time-series plots of magnetometer data before and after corrections are shown as below in figure 3.

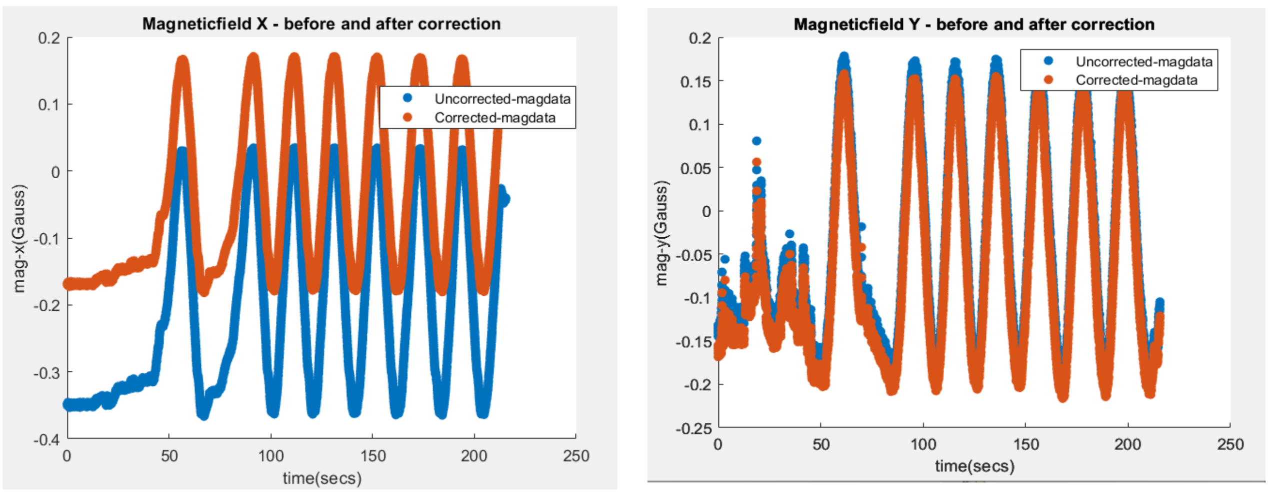

Fig 3: Magnetic field before and after correction – time series.

The corrected magnetic field data based on the steps discussed above are plotted as shown in the figure 4 below.

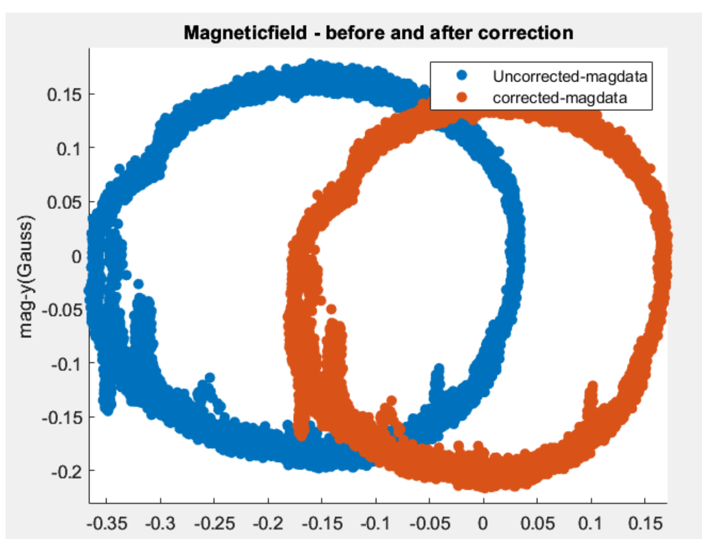

Fig 4: Magnetic field before and after correction – data driving in circles

The calibration parameters obtained are

$$SI = \begin{bmatrix} 0.8760 & 0.0243 \\ 0.0243 & 0.9952 \end{bmatrix}$$

$$b_{HI} = \begin{bmatrix} -0.1604 \\ 0.0197 \end{bmatrix}$$

The calibration parameters obtained in from the driving data in circles is used for the calibration of data obtained from Boston tour based on the equation in figure 1. The values obtained before and after calibration is plotted as shown in figure 5.

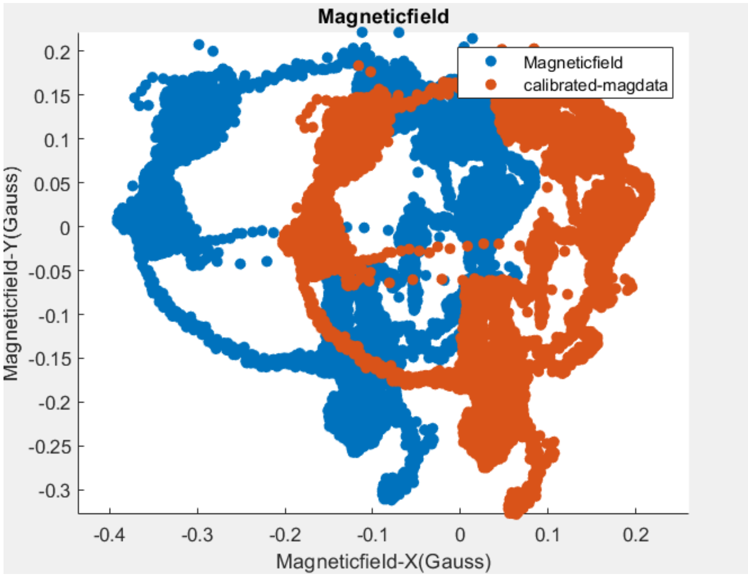

Fig 5: Magnetic field before and after correction – Boston mini tour

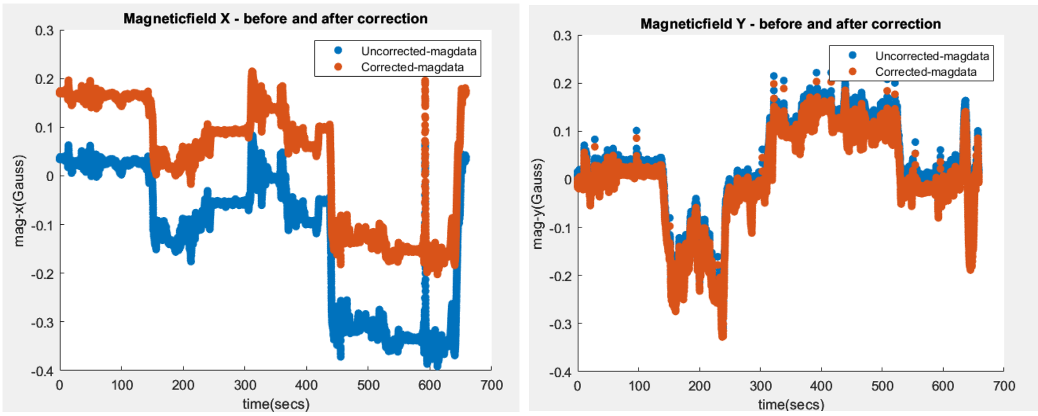

Fig 6: Magnetic field before and after correction – Boston mini tour time series.

ESTIMATE THE HEADING

The yaw angle is obtained from the raw magnetometer data and calibrated magnetometer data using the equation shown below and plotted in figure

$$\text{Yaw from magnetometer} = \tan^{-1}\left(-\frac{mag_y}{mag_x}\right) \qquad (4)$$

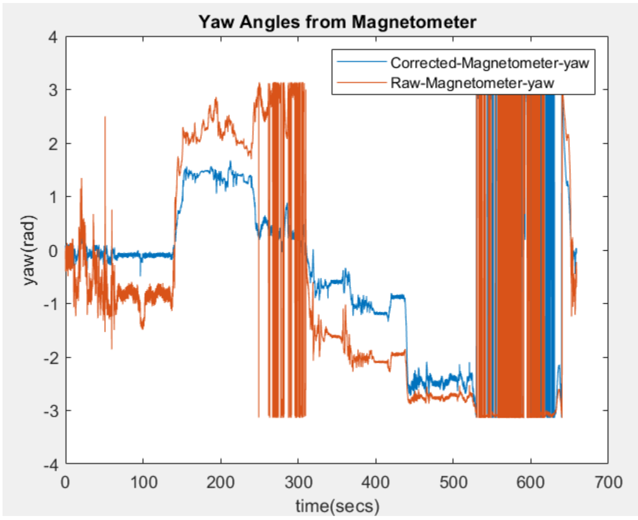

Fig 7: Yaw from Magnetometer – Before and After Correction

Angular velocity in z direction obtained from gyroscope is integrated to get the yaw from gyroscope. Corrected Magnetometer yaw at t =0 is used as the yaw from gyroscope at t = 0.

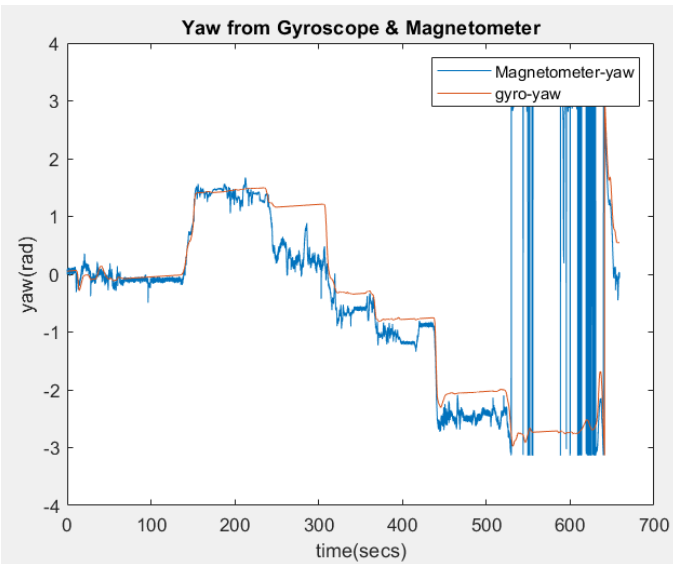

Fig 8: Yaw from Magnetometer – Before and After Correction

The magnetometer readings provide better indication of static values (long-term values). And as shown in the figure 8, the data appears to be noisy. To eliminate short-term fluctuations and preserve long-term readings, a low pass filter can be applied to magnetometer readings. Inbuilt MATLAB low pass filter with a pass band value of 0.001 and sampling frequency of 40 Hz is used.

Gyroscopes works well under moving conditions and when the rotational velocities are high. Hence gyroscope can be trusted whenever there is a significant change of the angles during short term readings. Hence a high pass filter can be performed on gyro data to preserve short term readings from Gyroscope. Inbuilt MATLAB high pass filter with a pass band value of 0.01 and sampling frequency of 40 Hz is used.

The obtained low pass and high pass filter values are fused by complementary filter (*cf*) based on the question below.

$$cf = a * LPF + (1 - a) * HPF \qquad (5)$$

A weightage (a) = 0.992 is used for the complementary filter.

The values obtained using low pass filter, high pass filter and complementary filter is plotted as shown in the figure 9.

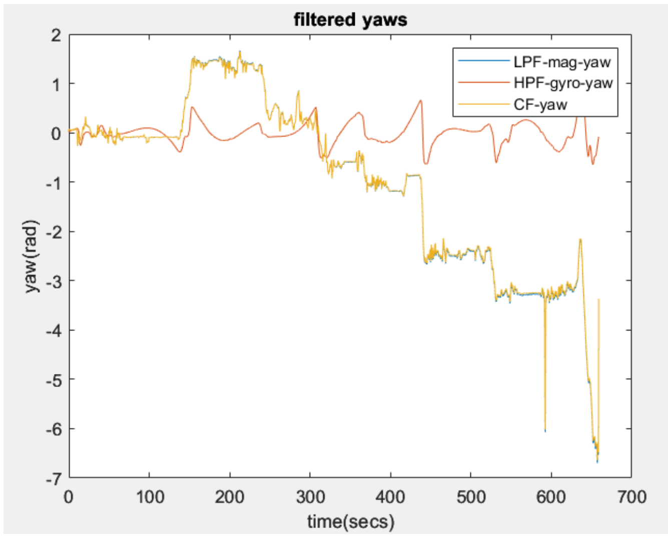

Fig 9: Filtered Yaw (LPF , HPF and CF)

The yaw obtained from complementary filter is compared with that of yaw obtained from IMU and is plotted in the figure below.

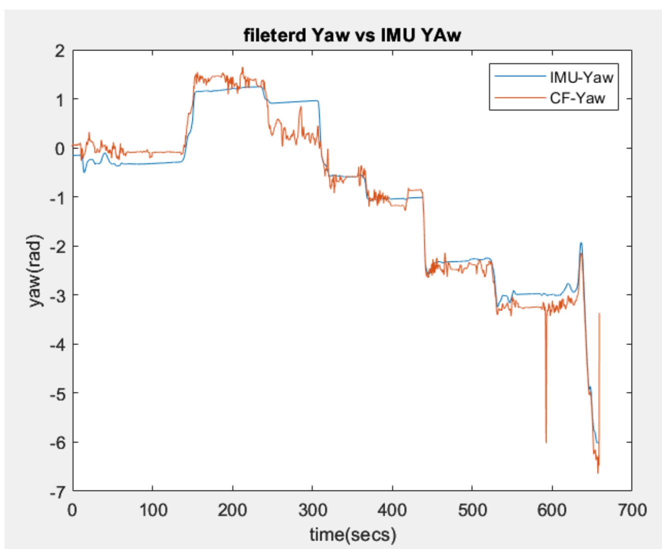

Fig 10: Complementary filtered yaw vs IMU Yaw

As shown in the figure 10, the yaw values obtained from complementary filter is following similar trend as of the yaw obtained directly from IMU. However, there is slight offset between both readings. As explained earlier, the magnetometer provides a good indication of long term readings and gyroscope provides a food indication of short term readings. These values passed through low pass filter and high pass filter is fused through a complementary filter will provide a better indication of true yaw than yaw obtained directly from IMU. Hence the values obtained from complementary filter can be considered for navigation.

ESTIMATE THE FORWARD VELOCITY

The forward acceleration (in x axis of IMU) is integrated to obtain the forward velocity and is compared with the velocity calculated from GPS data as shown in in the figure 11.

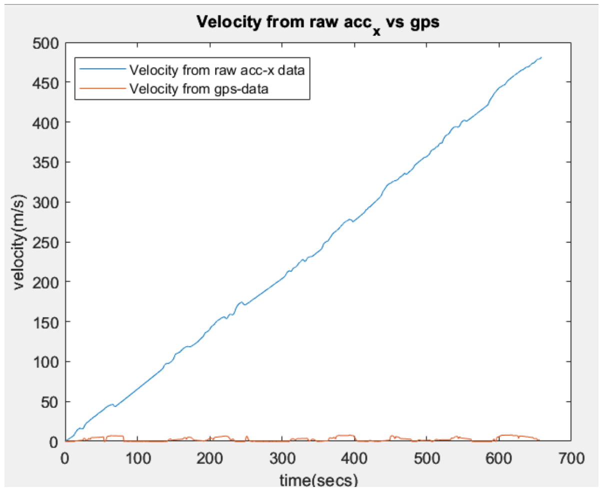

Fig 11: Forward velocity from raw acceleration vs GPS data

As show in the figure, the forward velocity obtained from raw accelerometer is deviated a lot from that obtained using GPS data. This is because the raw accelerometer data has bias and noises associated with it. The accelerometer bias is changing every time as well the accelerometer goes to zero and goes up or down. The change in bias can be due to the following reasons.

- Bias instability
- When car is stopped, the ground's inclination was not the same as the time when car was started.
- The car stopped with a huge jerk & this caused the IMU to bump a little in its leveling
- The vibrations of the car can be different when it is stationary on a different surface

These error values and noises are added up together while integration to obtain a high error. The GPS & IMU velocity should match each other when the car is stationary. At these stationary points, the accelerometer acceleration data should also be zero. Following steps are used to correct the raw acceleration data.

- Remove noises by passing data through a low pass filter. A cut off frequency of 0.001 Hz is used.
- First the data points with constant acceleration are found using the ‘diff’ function in MATLAB.
- A dynamic code in MATLAB is ran to check if the GPS velocity corresponding to these data points are zero.
- If the GPS velocity is zero at these data points and acceleration values at these data points are non-zero, then the non-zero value is considered as the bias for that period. This bias, calculated from the mean of acceleration datapoints,is subtracted from the acceleration values till the car stops next time. i.e a bias is calculated every time when car stopped and is subtracted from the subsequent acceleration points when the car is moving.

The raw acceleration, low pass filtered acceleration and bias removed acceleration obtained is shown in the figure 12.

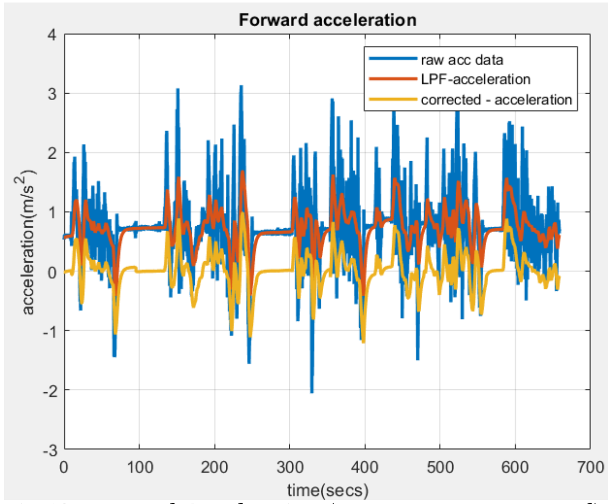

Fig 12: Forward Acceleration (raw vs LPF vs corrected)

The corrected forward acceleration is integrated again to obtain forward velocity and is compared with velocity obtained from GPS data.

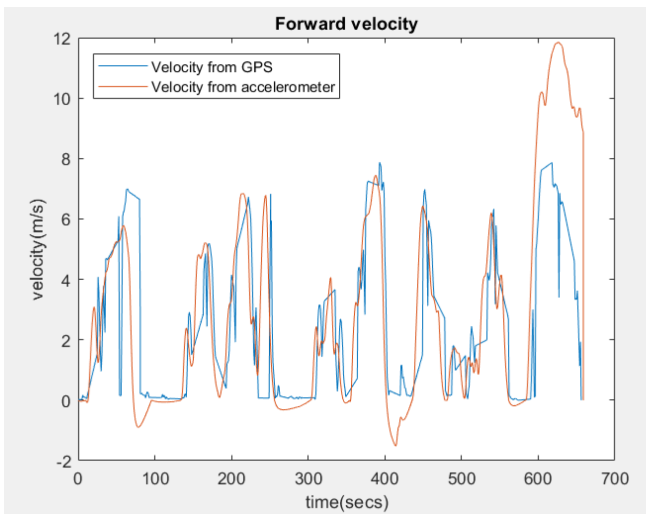

Fig 13: Forward velocity from corrected acceleration vs GPS data

However, there are still some discrepancies in the velocity estimate from acceleration data and GPS data. One reason could be that proposed bias calculations based on mean method couldn't account some stops at time t ~80 secs, t~400secs, t~600 secs. Hence some points are acceleration and deceleration points are missed and this added up to velocity errors during integration. There might be different linear/non-linear relationship to calculate the bias which can be used. The fact that IMU data are obtained at 40 Hz and GPS data is obtained at a frequency less than 1 Hz might also be reason for this discrepancy.

DEAD RECKONING

The forward velocities obtained from GPS data and corrected acceleration is integrated to obtain displacement.

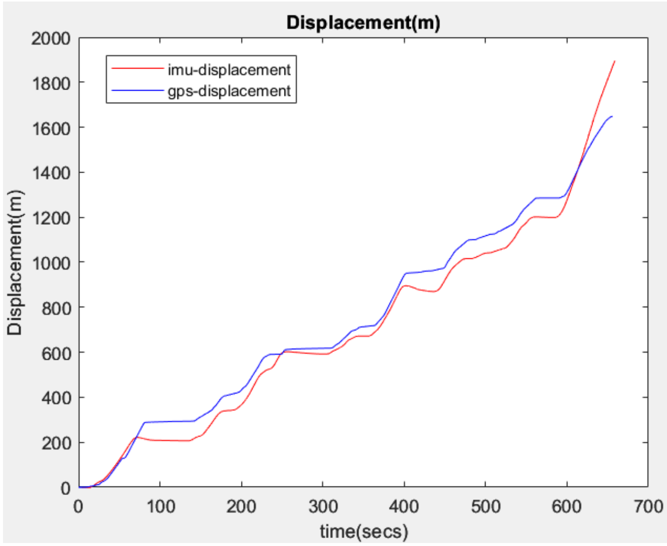

Fig 14: Displacements obtained from forward velocity

The values obtained from w*X_dot is compared with acceleration of y axis from IMU as shown in figure 15.

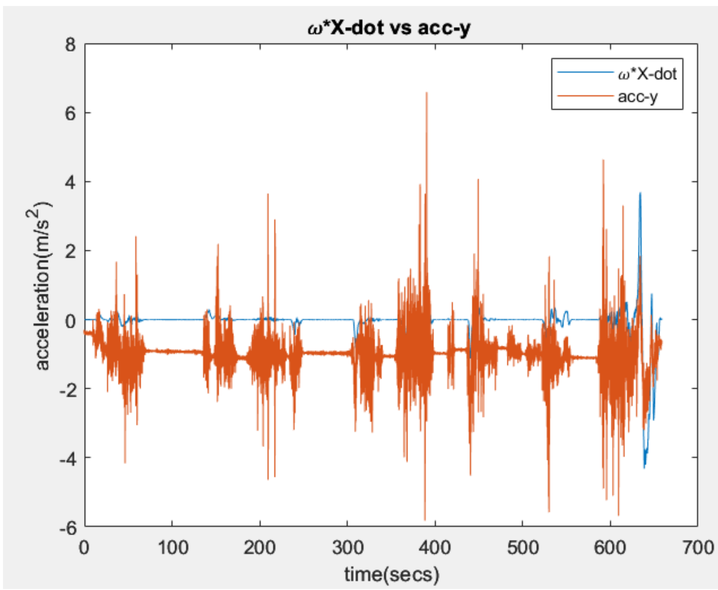

Fig 15: w*X_dot vs acceleration(IMU)in y axis

The values follow a similar pattern. However, there is difference. This might be because the acceleration in y-axis is very noisy and has a bias. This bias seems to drift as well with respect to time. Noises can be removed using a low pass filter and bias can be found to reduce it from subsequent values.

The trajectory of the vehicle is estimated as shown in figure 16. The heading of the IMU trajectory is adjusted by adjusting the slope of first straight line. Even though the start points are same, and heading is corrected, the end points of GPS and IMU trajectory is not coinciding. The path is almost consistent however the straight line distances are not same and appears to be scaled differently.

The difference in trajectory should be due to the difference in velocities calculated from GPS and accelerometer as shown in figure 13. The discrepancies in velocity estimated from GPS and accelerometer due to the reasons explained previously is getting added up upon integration to obtain the position. As shown the figure 13, the large discrepancy between the velocities is happening towards the end causing the displacement calculated from IMU data to be high. The GPS and IMU measurements are matching closely (2m) during the start period. However the errors after this time period, causing the trajectory estimated from IMU to drift more. Dead reckoning might be more reliable with a position fix in between intervals.

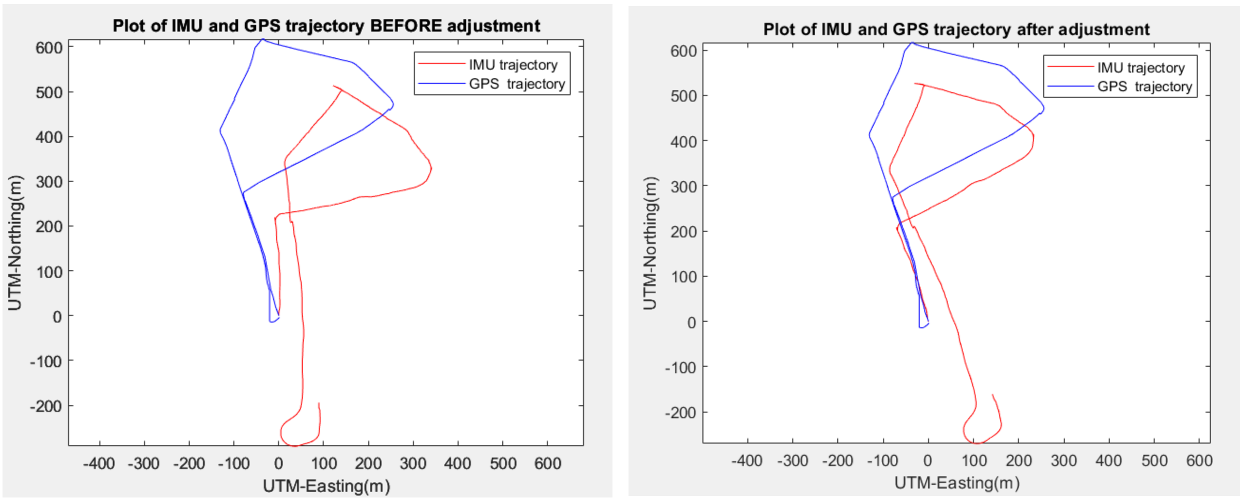

Fig 16: Estimated trajectory from IMU and GPS