EECE 5554 Robotics Sensing and Navigation
Lab-4 Report

Note : Driver and launch files are present in Chris's and Matt's git as the team used their driver and launch files .

Inroduction :

Sensor fusion is the process of combining data from multiple sensors to improve the accuracy and reliability of the measurement or estimation. In many applications, no single sensor can provide all the required information, and multiple sensors must be used to obtain a complete understanding of the environment or the state of a system. In this lab ,GPS and IMU sensors are used for improving the navigation stack .This lab is split up into two parts :
1.Driving in circles for magnetometer calibration
2.Driving around the city to perform estimation of heading , velocity and perform dead reckoning

Part 1 : Magnetic Calibration

For this task we drove around the Ruggles circle for 5 times putting the IMU in the positive x direction and calibrated the IMU.
Magnetometers are used to measure the strength of a magnetic field. This can be accomplished in several ways, but a method based on magnetoresistivity is commonly used in MEMS-based magnetometers.Normally we calculate the heading angle by taking the arc tan function of yaw in y wrt x direction and this should result in a circle centered around (0,0).Unfortunately there are hard iron and soft iron distortions present in nature .Hard iron distortions is present due to objects that generate their own magnetic field which would result in a measurement offset .Soft iron distortions are present due to ferromagnetic materials which bend the magnetic fields and change the orientation of the circle.

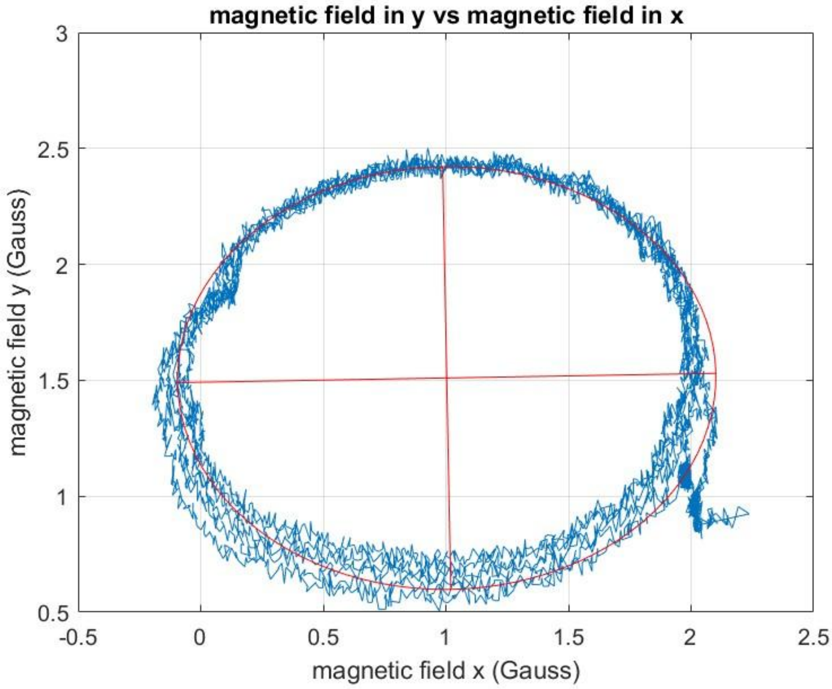


Figure 1: Raw magnetometer readings

From the figure it is evident that there are soft and hard iron distortions as the centre is not at (0,0) and the circle is squished into an ellipse .Hard iron distortions can be present due to speaker in the car and electronic devices . Soft iron distortions could be due to metal body of the car and near the car as we drove around Ruggles .

To solve for both the distortions and make calculations easier I used the fit ellipse function and got the offsets , bank angle and calculated the scale . After using the following equations I was able to calculate the hard and soft iron offsets by translation,rotation and scaling of data

$$m_c = S_I(m^{\sim} - b_{HI})$$

.

Where mc = Corrected magnetometer readings

Si(Scaling matrix *Rotation matrix) =scale which is obtained by taking $\sigma = \frac{q}{r}$ where q is the length of short axis and r is the length of long axis .This is then multiplied by rotation matrix R

R is given by $R = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$ .In our case , we haven't changed the heading mode so I accounted for declination by calculating it from the fit ellipse and subtracting it from theta m-bHi : This the difference between the original reading and the hard offsets obtained using ellipsefit.X0_in and ellipsefit.Y0_in;

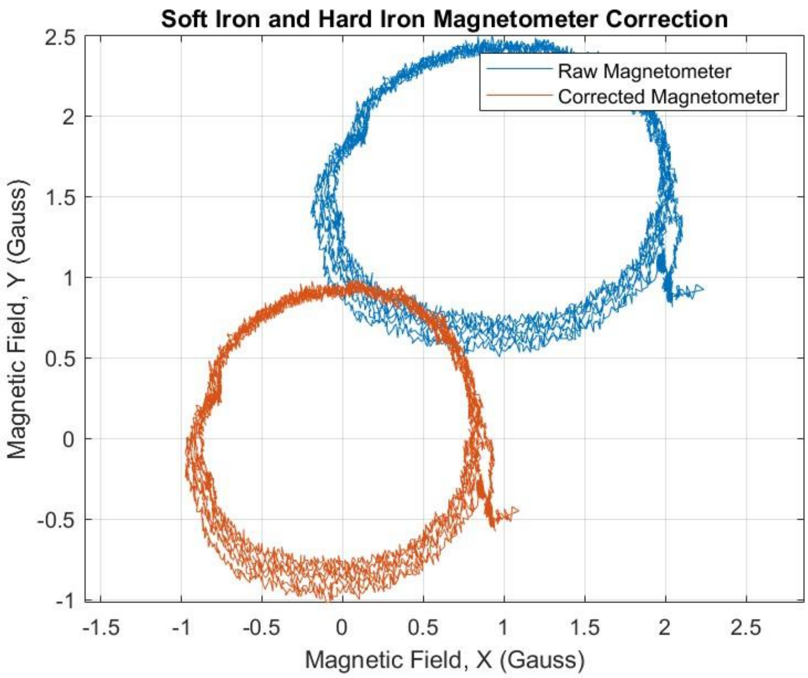


Figure 2 : Raw and Calibrated Magnetometer readings

The above plot shows the raw scattered plot(orange circle) to be translated to a circle translated, rotated and scaled with origin as shifted which is the desired plot after correcting the hard and soft iron errors.The radius of the magnetometer is between 0.4 and 0.5 which is equal to Earth's magnetic field strength at that point of time.

Part 2 : Driving data Yaw estimation

Using the soft and hard iron distortions obtained from driving in circles , we have to calculate the corrected yaw for magnetometer for driving in the city .The magnetometer system errors have been dealt but there are external magnetometer sources of noise which is why we pass the signal through low pass filter .This would add lag and make the system less responsive .Another option is to fuse the magnetometer with the gyroscope readings .When the magnetometer shows change in reading, it can be confirmed with the gyroscope if it came from the sensor or due to the noises.
Integrating the measured yaw angular rate from the gyro, we receive another estimate for the yaw of the system.Gyroscopes have bias and high frequency noises which are mitigated by passing it through a high pass filter .The advantage of the gyro is for short-term precision and error correction while the advantage of the magnetometer is long-term accuracy.So, sensor fusion is done by using a complimentary filter by adding the low pass magnetometer yaw with high pass gyroscope yaw .

In order to use a complementary filter to find a combined estimate of yaw the following steps were followed:

1. First the yaw is calculated from the magnetic x and y values using the arctan2(-y/x). Where the y and x are the corrected Magnetic_Y and Magnetic _X.
2. Now this corrected magnetic yaw is unwrapped and passed through a low pass filter using, y = lowpass(x,fpass,fs) where x is the corrected magnetic yaw that has been sampled at a rate of 40 hertz. fpass is the passband frequency of the filter which was set to 0.0001 hz
4. The Angvel_z which is the angular velocity in z is integrated and the integrated value is passed through the high pass filter which has been sampled at a rate of 40hz and the pass band frequency of the filter is set to 0.0225.
5. Once the lpf and hpf are obtained, the sum of both lpf and hpf has given the complementary filter output.
All the graphs of LPF, HPF, CPF are plotted and plotted in the fig-5

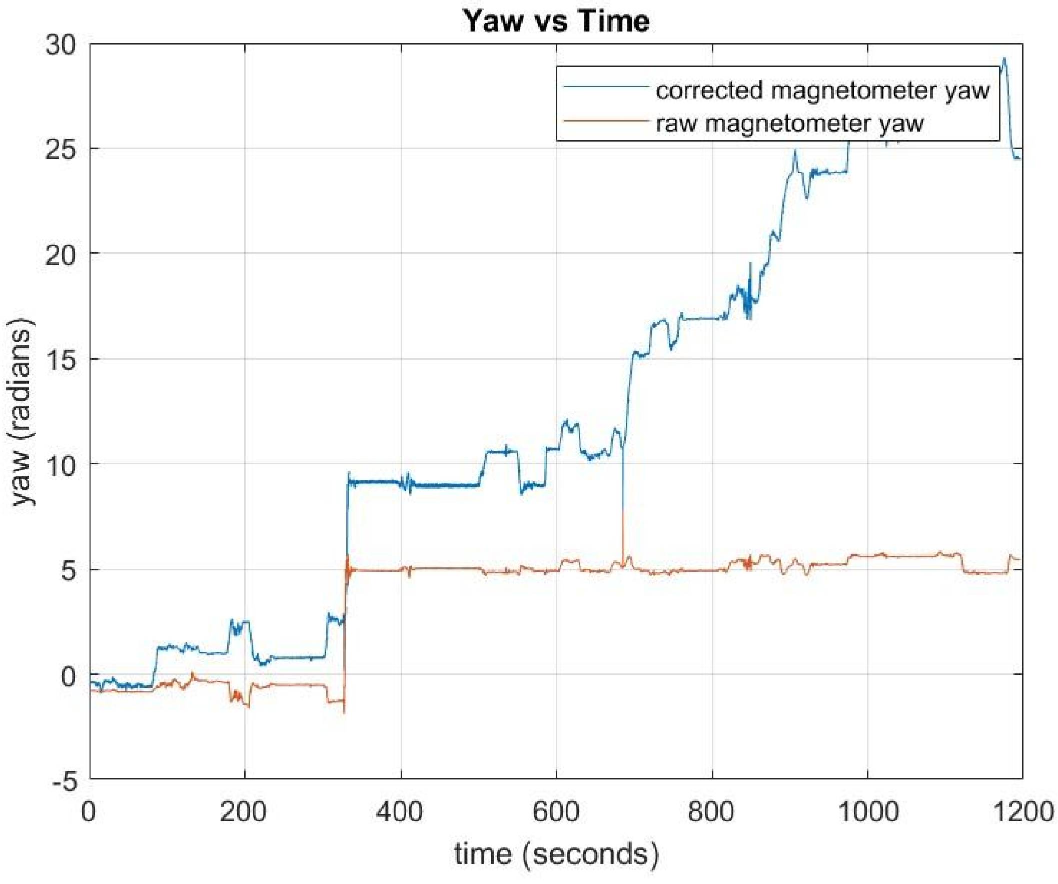


Figure 3 : Raw and Corrected yaw from Magnetometer

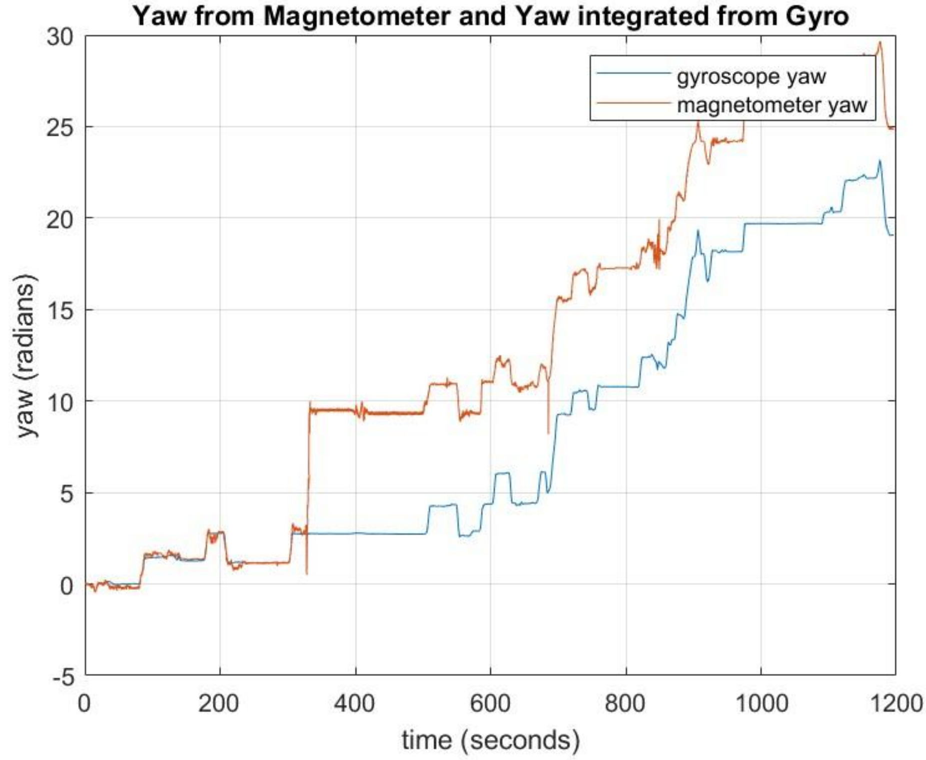


Figure 4 : Corrected yaw vs Gyroscope yaw

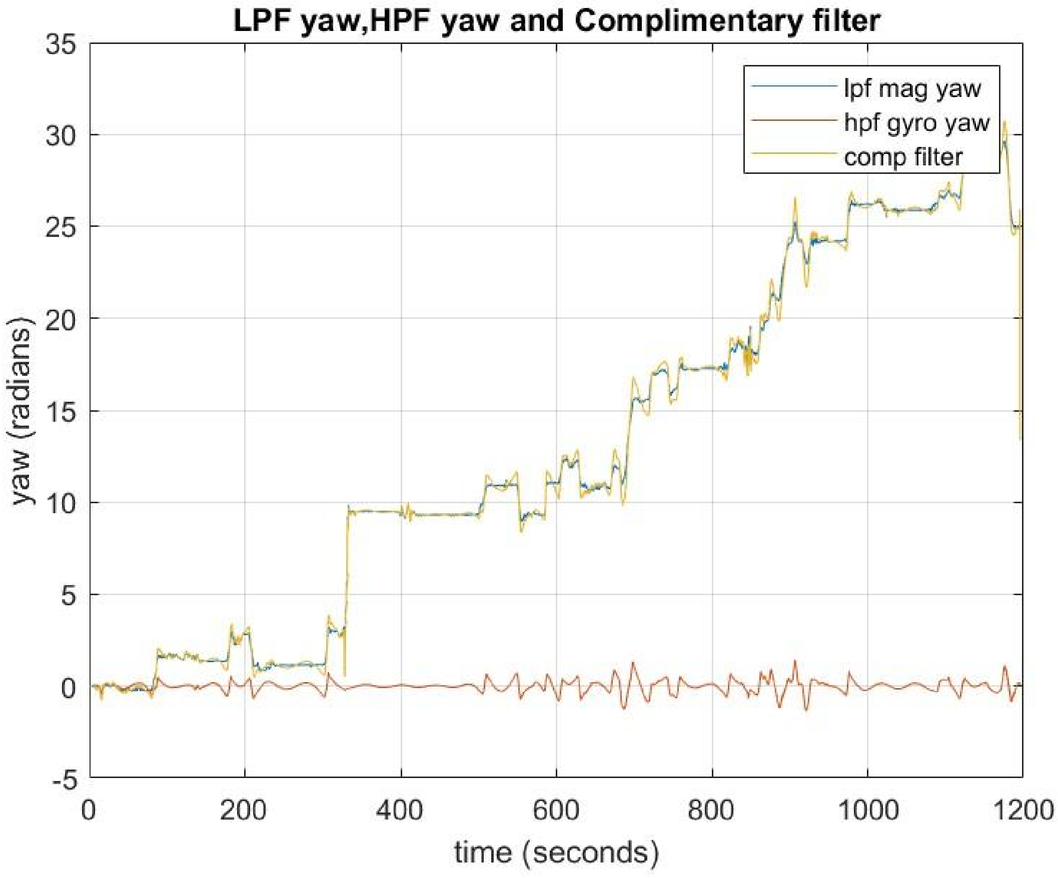


Figure 5 : LPF ,HPF and Complementary filter

I would trust the complimentary filter for my navigation as it is a combination of high pass and low pass filter and it makes use of the advantage of magnetometer and gyroscope such as short term precision for gyroscope and long term accuracy for magnetometer. Magnetometer also provides an initial bias estimate .The complimentary filter graph is similar to that of IMU yaw graph .

Part 3 : Forward Velocity Estimation

The IMU is placed in the positive x direction and the forward acceleration is calculated using the accelerometer . Integrating the forward acceleration using cumtrapz function gives us the forward velocity . It can be seen from fig 6 that there is a positive bias present in the acceleration of IMU .Integration only makes it amplified hence giving rise to a plot given in fig 6.Velocity from gps is obtained by taking the distance (utm easting and northing) /time (gps)

For adjustment : The primary modification made involved subtracting the biases from the data. To accomplish this, the regions of the graph with no motion were identified, and the mean of these regions was calculated. This mean was then subtracted from the remaining data, and the process was repeated for all stationary regions in the acceleration plot. Once the biases were removed, the acceleration data was integrated to determine the velocity data obtained from the accelerometer.

There was a scaling difference too so scale factor was calculated and multiplied to get the modified velocity.The peaks and troughs nearly align although it might need a better filter to match .The main reason for the issue was the accumulation of errors resulting from integrating a noisy acceleration signal. Moreover, the route taken was not flat, which might have added some gravitational components to the forward acceleration in the pitch angle. Despite several attempts to correct the scale, such as accounting for pitch variation, integrating at intermediate positions, and trying other methods, no better outcome was achieved.

.

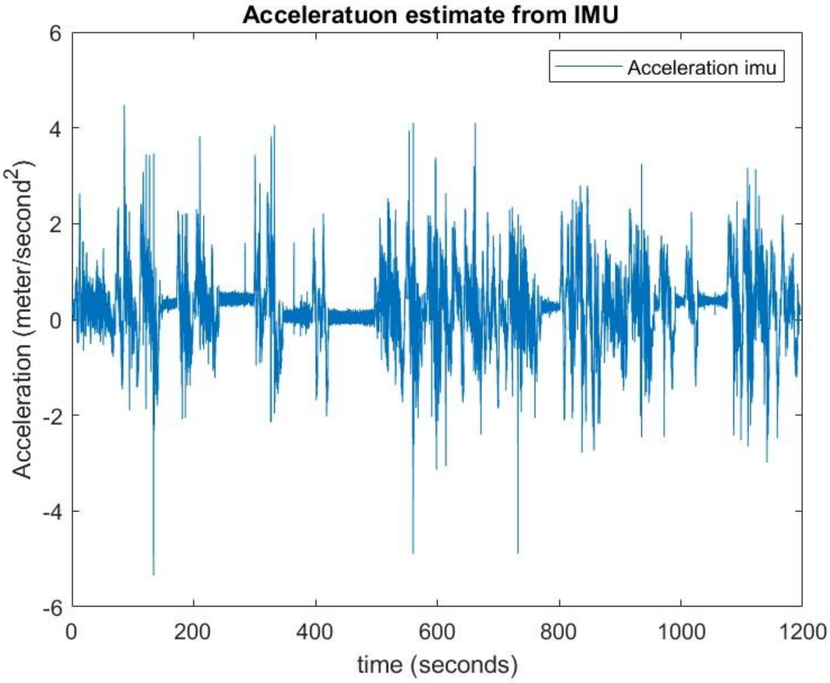


Figure 6 : Acceleration x vs time

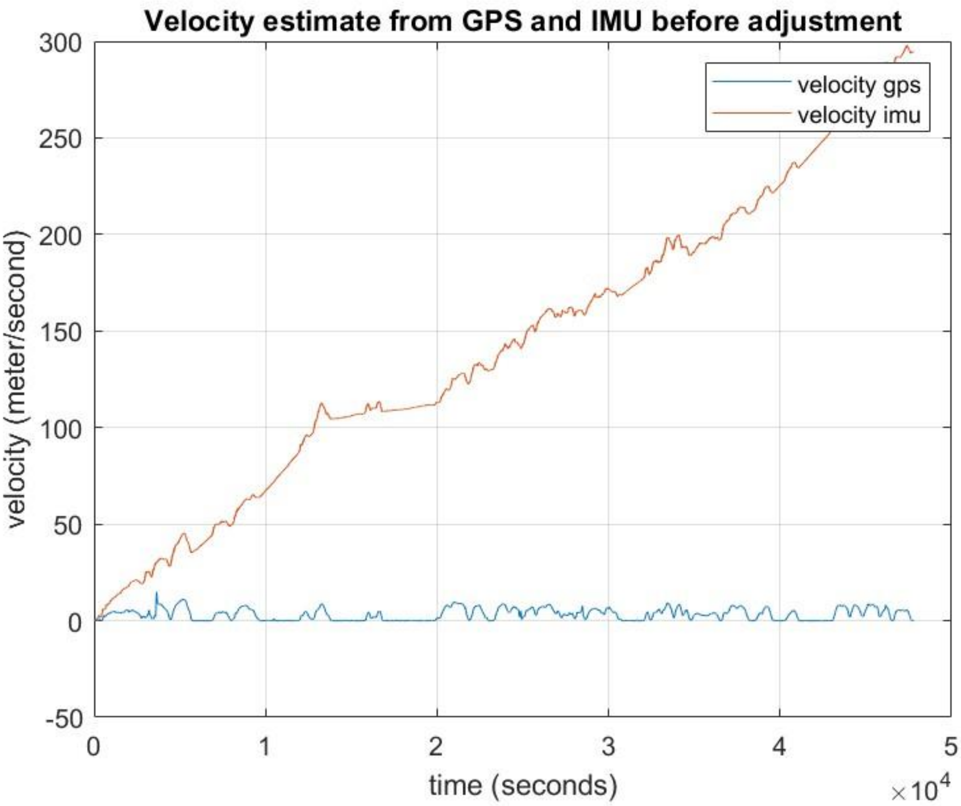


Figure 7 : IMU vs GPS velocity before adjustment

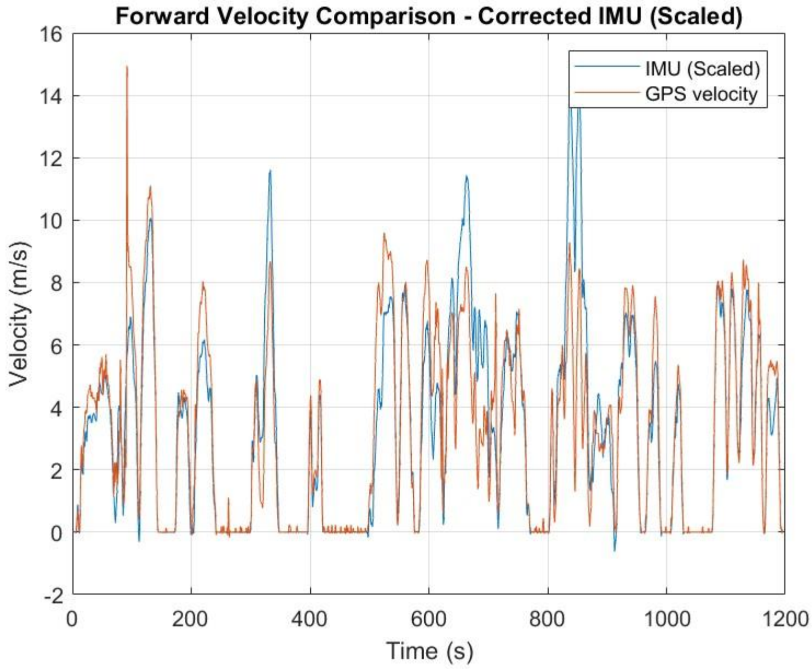


Figure 8 : IMU vs GPS velocity after adjustment

Part 4 : Dead Reckoning with IMU

Integration of above velocity estimates gives us the displacement from GPS and IMU as shown in Fig 9 .It can be seen that IMU plot drifts away due to amplification of bias.

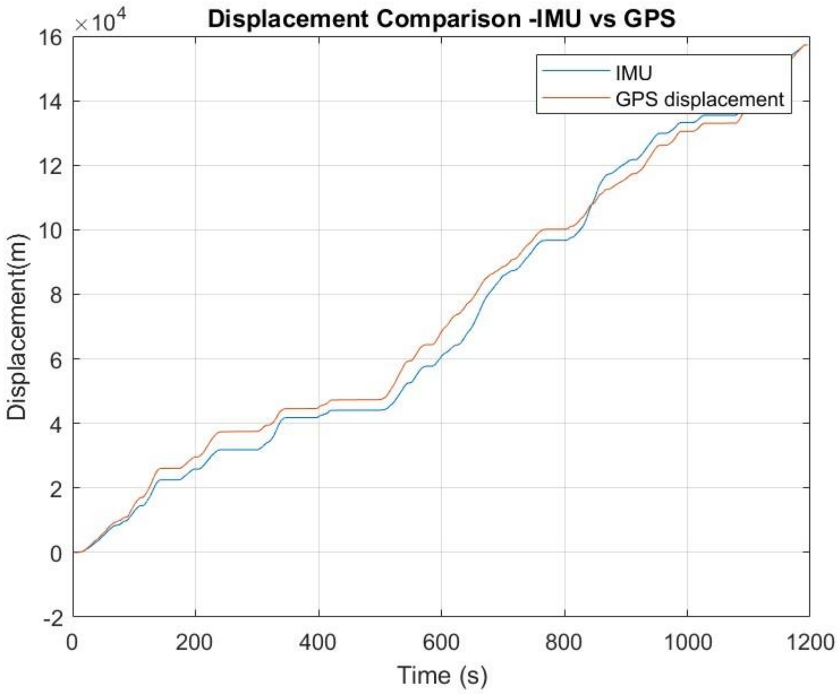


Figure 9 : Displacement IMU vs GPS

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

These equations of motion simplify the motion of the vehicle to a 2D plane. The vehicle's center of mass is located at position (X,Y,0) and rotates about the center of mass at a rate of (0,0,ω). The position of the inertial sensor is (x,y,0), and its position relative to the vehicle's center of mass is (xc,0,0). Assuming there is no skidding and negligible offset, the observed forward acceleration of the IMU is the same as the vehicle's forward acceleration. The forward velocity of the IMU previously calculated can be multiplied by the yaw-rate measured by the gyro and compared to the observed lateral acceleration measured by the IMU.

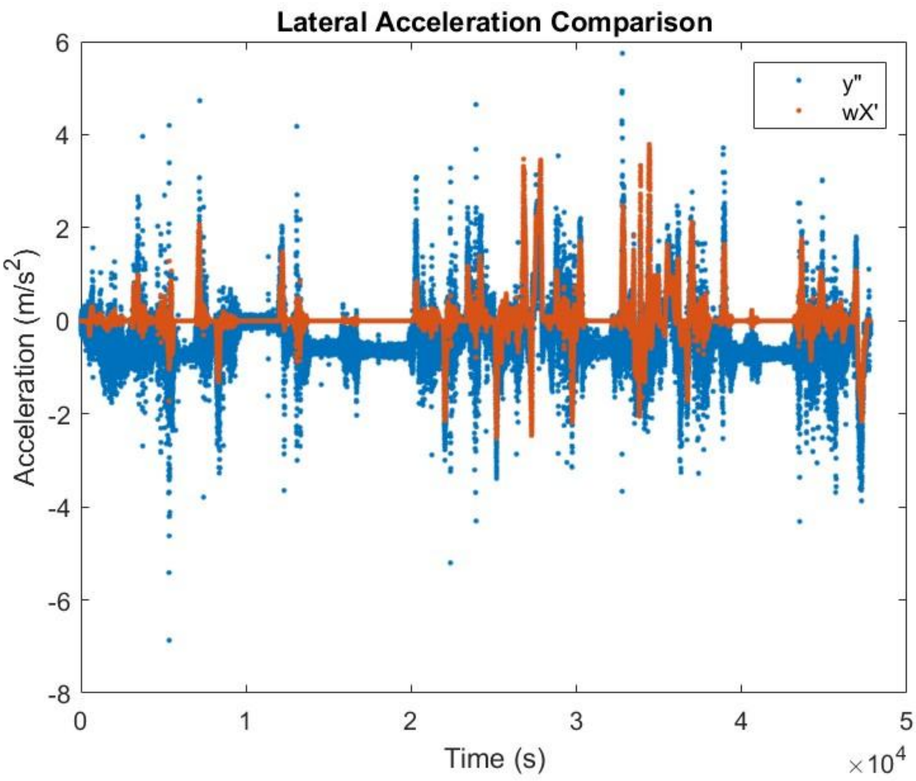
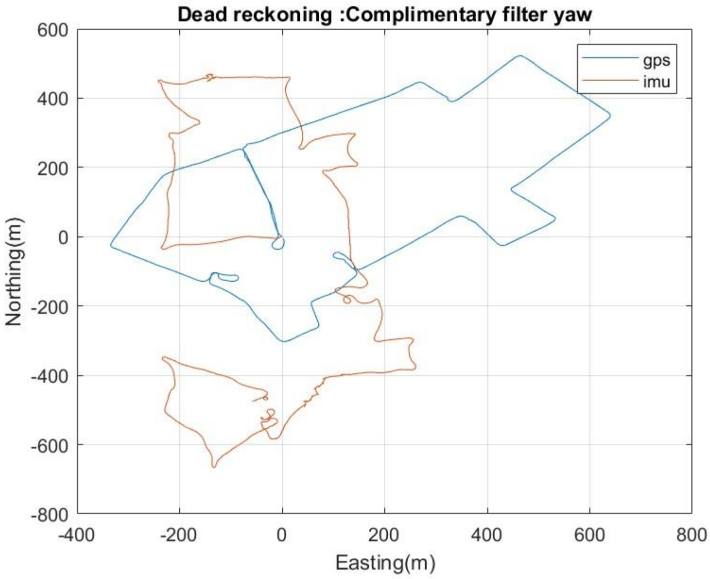


Figure 10 : wx vs Y

The spikes are time aligned except for lateral acceleration bias which is present due to the gyroscope .This could be due to the assumption of 2d operation field and ignoring the IMU's offset by taking(xc=0).I have checked it with complimentary filter and corrected magnetometer readings.

For estimation of trajectory ,Northing and Easting is calculated using imu velocity * sin (heading) and imu velocity * cos(heading) which gives rise to ve and vn which are integrated to give xe and xn .this is compared with the Northing and Easting values from the plot

.Scaling factor of 0.75 is used .Dead Reckoning of IMU was close to GPS but it was flipped and scaled .As we were able to remove the bias in the acceleration of IMU there will always be an error in the dead reckoning path . Although the shape is recognisable it can be seen that the imu's values are drifting away from that of gps values ans there is a scale change .



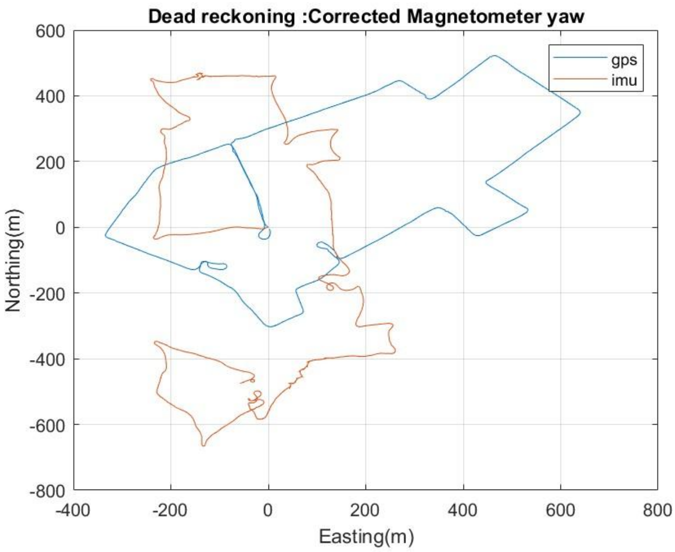


Figure 11 : Dead Reckoning of IMU and GPS

Using the following calculations in the figure it can be estimated that xc =1.04m offset .

$$v_{sensor}^{U} = v_{car}^{U} + \omega \times \rho_{sensor}^{R}$$

$$a_{sensor}^{U} = a_{car}^{U} + \dot{\omega} \times \rho_{sensor}^{R} + \omega \times (\omega \times \rho_{sensor}^{R})$$

$${}_{T}^{R}R^{T}\, a_{sensor}^{R} = a_{car}^{U} + \dot{\omega} \times \rho_{sensor}^{R} + \omega \times (\omega \times \rho_{sensor}^{R})$$

$${}_{T}^{R}R^{T} \begin{bmatrix} a_{imux} \\ a_{imuy} \\ 0 \end{bmatrix} = \begin{bmatrix} a_{utmx} \\ a_{utmy} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \dot{\omega} \end{bmatrix} \times \begin{bmatrix} x_c \\ 0 \\ 0 \end{bmatrix}^{R} + \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} \times (\begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} \times \begin{bmatrix} x_c \\ 0 \\ 0 \end{bmatrix}^{R})$$

$$\begin{bmatrix} a_{imux} \\ a_{imuy} \\ 0 \end{bmatrix}^{R} = {}_{U}^{R}R \begin{bmatrix} a_{utmx} \\ a_{utmy} \\ 0 \end{bmatrix}^{U} + \begin{bmatrix} 0 \\ \dot{\omega} x_c \\ 0 \end{bmatrix}^{R} + \begin{bmatrix} -\omega^2 x_c \\ 0 \\ 0 \end{bmatrix}^{R}$$

$$\begin{bmatrix} a_{imux} \\ a_{imuy} \\ 0 \end{bmatrix}^{R} - {}_{U}^{R}R \begin{bmatrix} a_{utmx} \\ a_{utmy} \\ 0 \end{bmatrix}^{U} = \begin{bmatrix} -\omega^2 \\ \dot{\omega} \\ 0 \end{bmatrix}^{R} x_c$$

Discussion :

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)?

Based on the specifications of the VectorNav VN 100 IMU, it should be able to navigate without a position fix for up to several minutes, depending on the conditions. Specifically, the VN 100 IMU has a drift rate of 0.01 degrees per second, which means that after 100 seconds, it will have drifted by one degree. The actual time it can navigate without a position fix will depend on various factors such as the accuracy of the initial position fix, the level of external disturbances, and the desired level of accuracy.From the graph it can be estimated that there is a position match for almost 100 seconds .