Robot Sensing And Navigation

Sensor Fusion- (LAB 4)

Srinivas peri

Introduction :

Sensor fusion is the process of merging data from multiple sensors such that to reduce the amount of uncertainty that may be involved in a robot navigation motion or task performing. In this lab we are using complementary filter method for our sensor fusion.

Sensors used in this lab for the data collection and analysis are IMU(inertial measuring unit) and GPS(global positioning system)

Magnetic Noise Reduction :

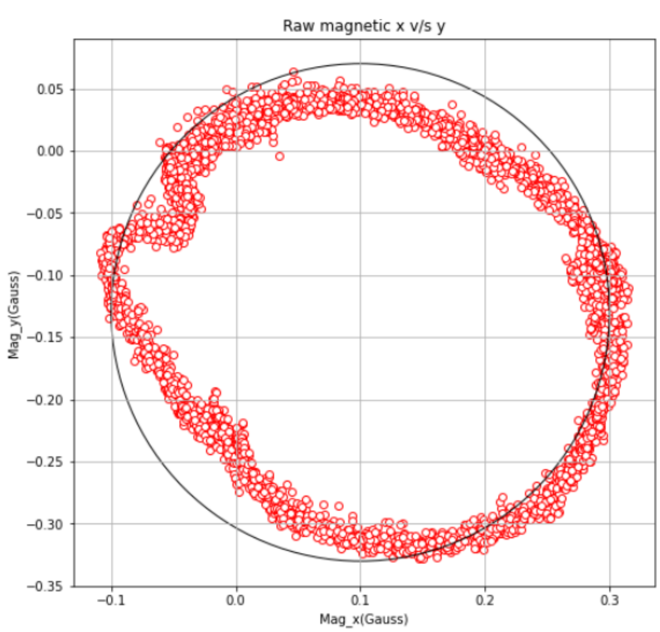

Fig 1 : Raw_magnetic_data

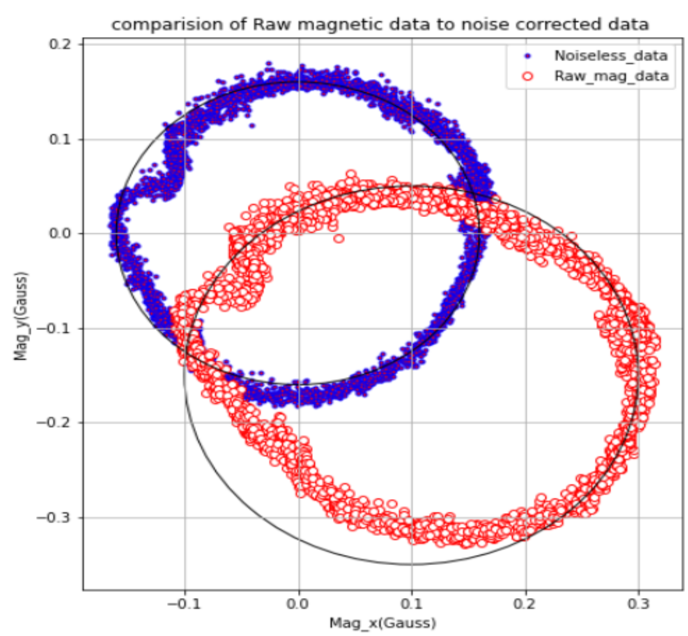

Fig 2 : Compared_magnetic_data

The idle magnetic field in x direction v/s magnetic field in y direction should be at the center (0,0) and should look as a circle. But due to certain noises which are caused by natural and human error we get distorted data. These errors are called

Hard iron and soft iron errors. We can eliminate them and generate calibrated data with the following approach.

Hard iron correction:

Hard-iron distortion is produced by materials that exhibit a constant, additive field to the earth's magnetic field, thereby generating a constant additive value to the output of each of the magnetometer axes. A speaker magnet, for example, will produce a hard-iron distortion. As long as the orientation and position of the magnet relative to the sensor is constant the field and associated offsets will also be constant. A hard-iron distortion can be visibly identified by an offset of the origin of the ideal circle from (0, 0).

From fig1 we can see how the data is shifted from the ideal center(0,0). To correct it we calculate the offset of each axis and subtract it from them. Considering X and Y to be our raw data from magnetic_x and magnetic_y.

X offset = [maximum(x) + minimum(x)]/2

Y offset = [maximum(y) + minimum(y)]/2

Soft iron correction:

soft-iron distortion is the result of material that influences, or distorts, a magnetic field—but does not necessarily generate a magnetic field itself, and is therefore not additive. While hard-iron distortion is constant regardless of orientation, the distortion produced by soft-iron materials is dependent upon the orientation of the material relative to the sensor and the magnetic field. This distortion can be identified when an ellipse with an angular rotation is observed. From Fig1 we can see how the data does not fit the circle and the ellipse is rotated. The procedure to eliminate the soft iron distortion is as follows.

We need to get the length of the major axis by eyeballing or using an ellipse fit or get the maximum and minimum of x and y. That gives us the distance of the farthest point (x, y) from the center (0,0). The distance r can be obtained from

$$r = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Θ = arcsin(y1/r)

We use the rotation matrix to rotate the ellipse to bring it to position where major axis and minor axis are aligned to the x and y cartesian line.

$$R = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$$

V1 = Rv

Our V1 will be the product of rotation matrix R and vector components (v) of mag_x and mag_y.

After this rotation we will scale our date using **σ** which is called scale factor this converts our ellipse to a circle and is calculated by dividing minor axis (q) and major axis (r)

σ = (q/r)

Once scaling is completed a final rotation must be made to rotate the data back to their original position, thus compensating for the soft-iron distortion. In fig2 the blue circle is the calibrated data.

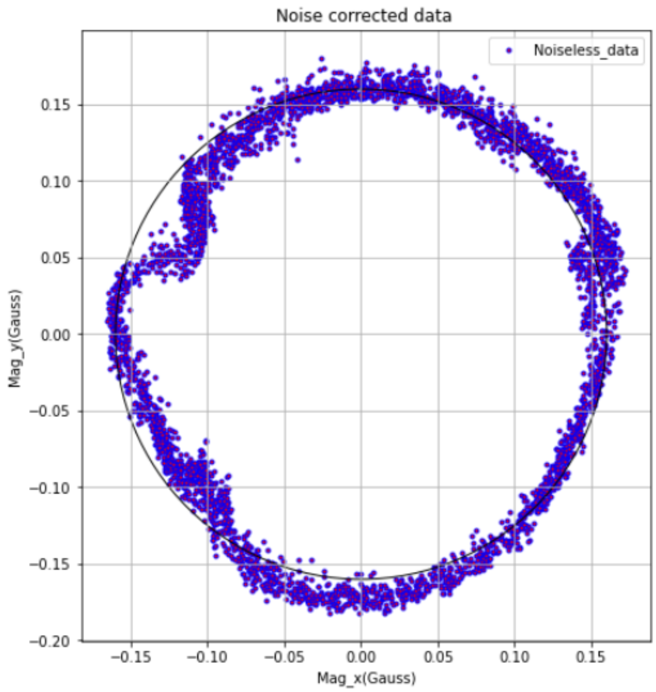

Fig 3 : Calibrated Magnetometer data

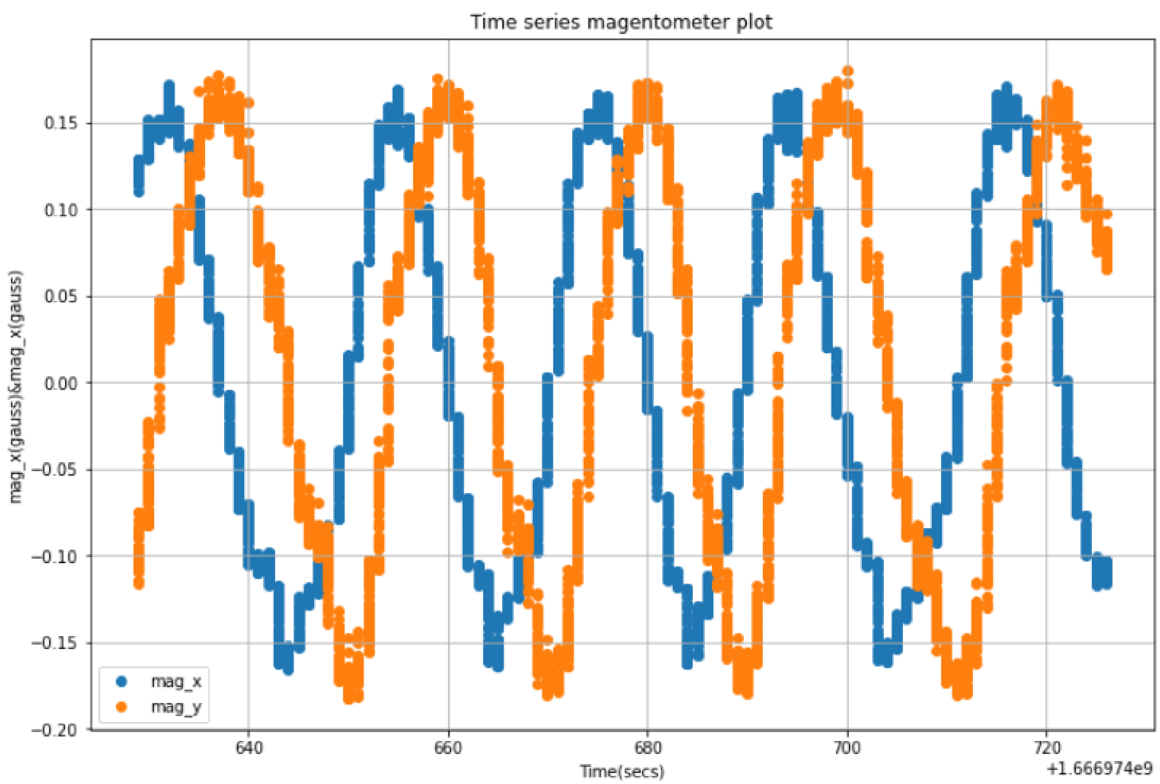

Fig4 : Time vs Magnetometer X and Y

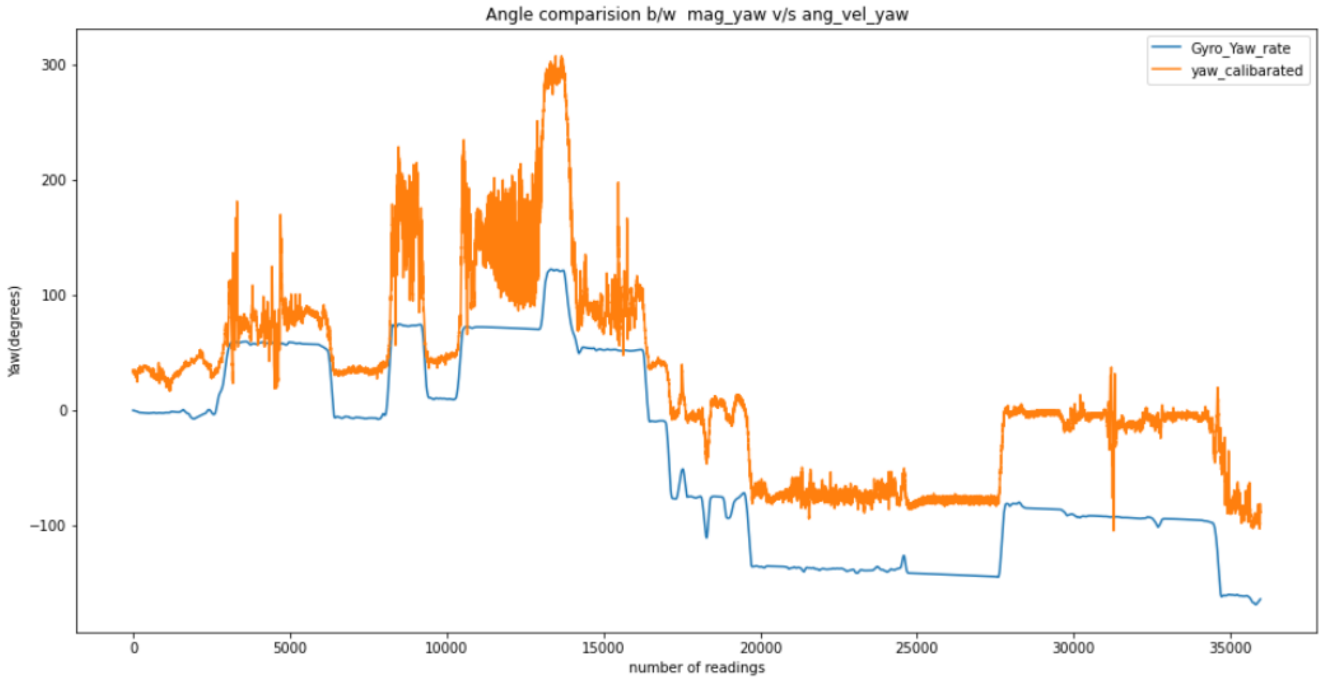

Fig 5 : Magnetometer Yaw & Yaw Integrated from Gyro together

The plot from Fig 5 shows that the yaw calculated from the magnetometer has an upward shift of approximately 20 radians whereas the yaw calculated from gyro starts from 0 radians.

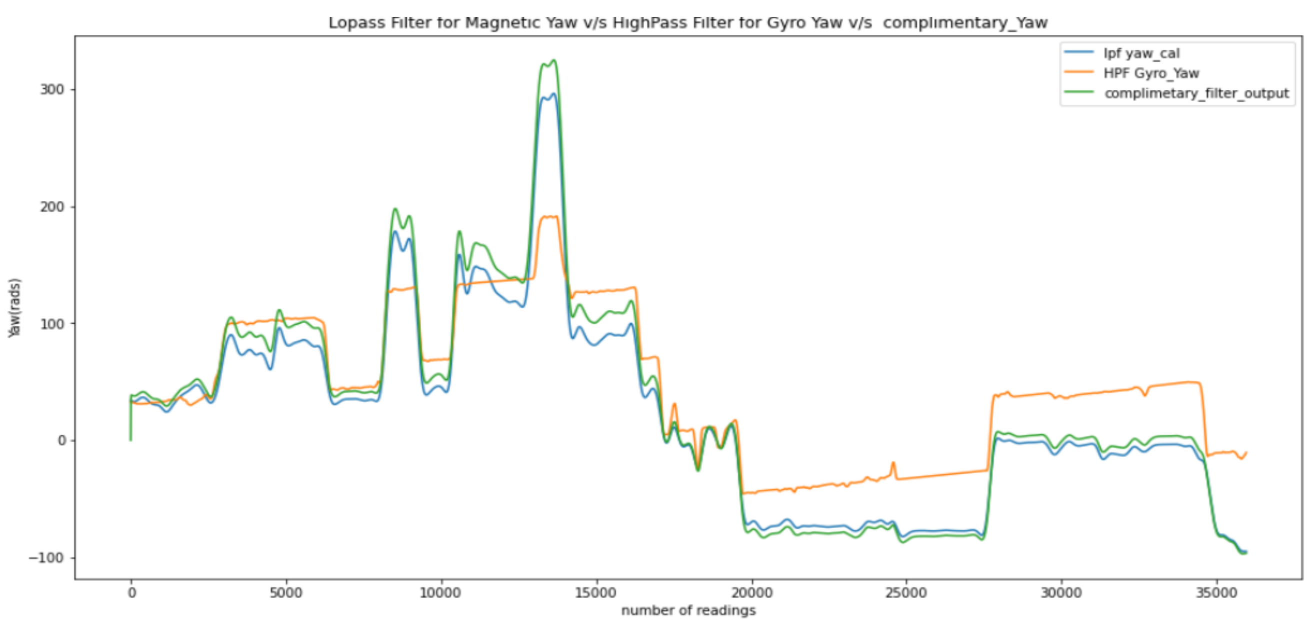

Fig 6 : LPF, HPF, and CF

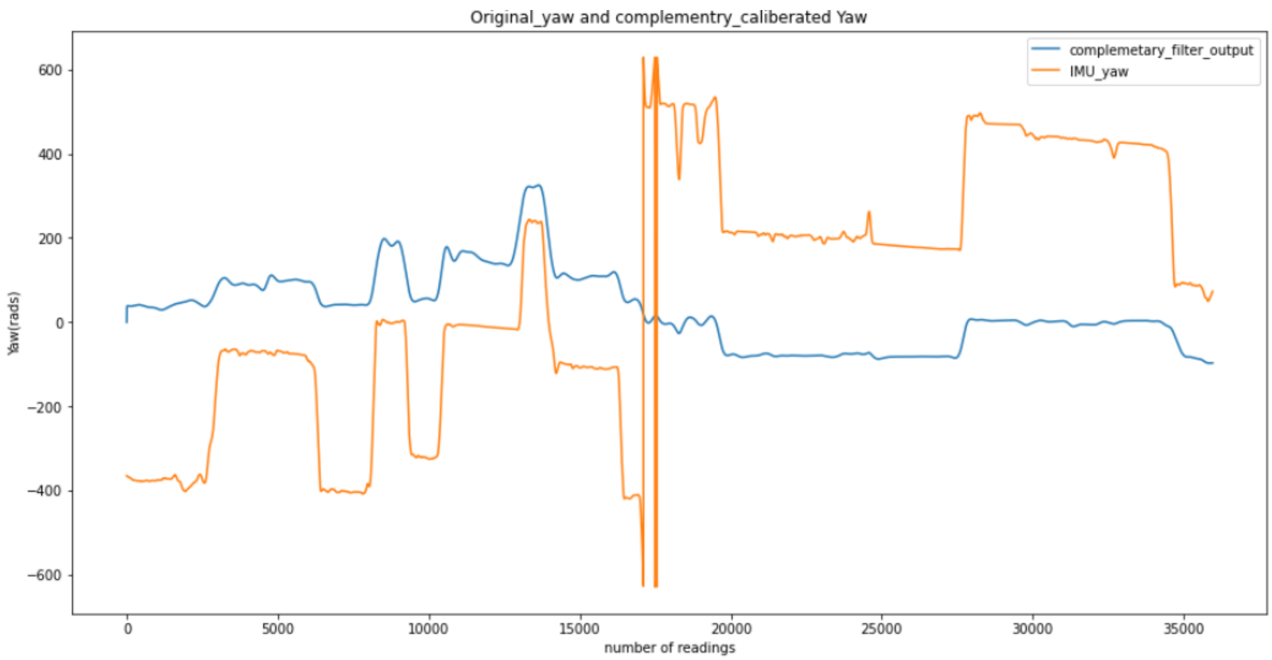

Fig 7 : Yaw from the Complementary filter & Yaw angle computed by the IMU

Complementary filter is the combination of low pass filtered calibrated magnetic yaw and high pass filtered gyro yaw. The imu generated yaw has data noise in the region 17000 to 17500 which in case of the complementary filter output is negligible as the combined filter data compensated for the spike in distortion and reduced it near zero. The use of butterworth 3rd order filters for high pass with 0.0001 cutoff frequency and low pass with 0.01 cutoff frequency gave me the best possible outputs with reduced noise.

Complementary Filter

$\theta k\mathbf{+1} = \alpha\ \mathbf{*}\theta k + \omega\mathbf{\Delta}S + \mathbf{(1-}\alpha\mathbf{)}\ \theta accel$

We can calculate alpha (weighting factor)by getting the standard deviation of the time as the date is being collected at that intervals and is denoted by tau(**τ**) and sampling frequency fs and converting frequency to time we get dt = 0.025

$\alpha = \boldsymbol{\tau}/(\boldsymbol{\tau}+dt)$

Estimates of forward velocities:

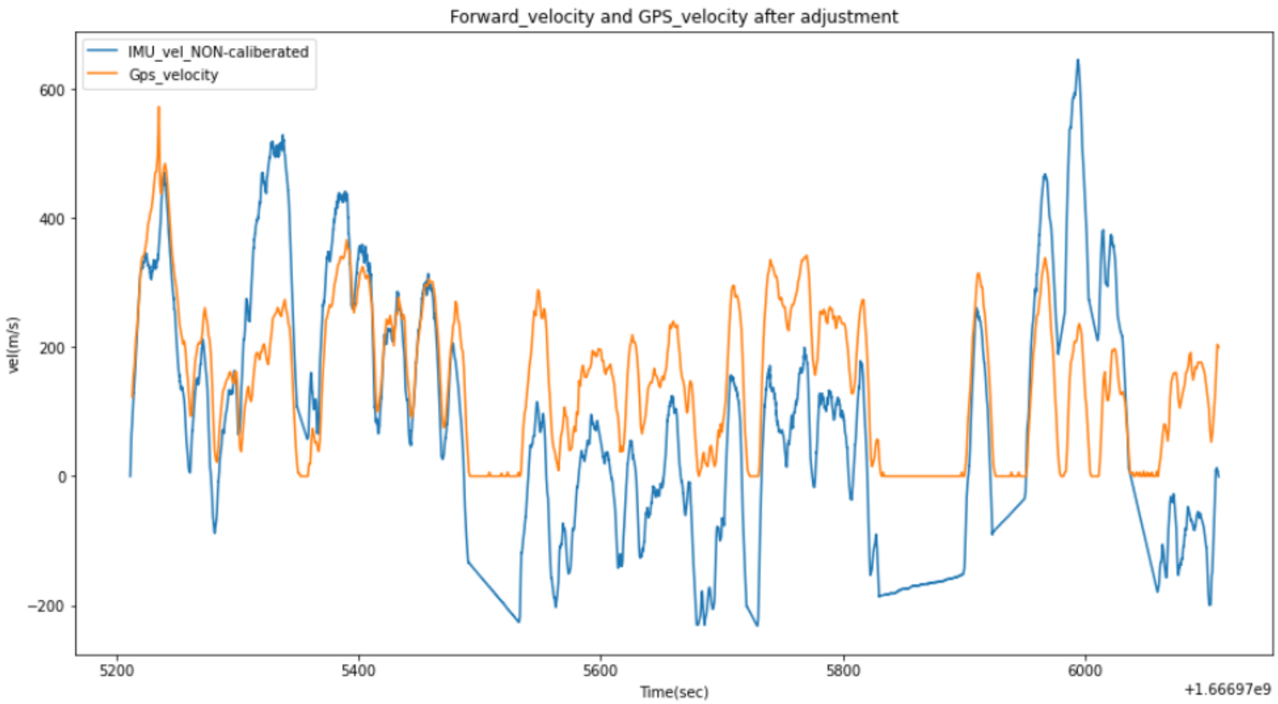

Fig 8 : Velocity from IMU and GPS **before** IMU velocity adjustment

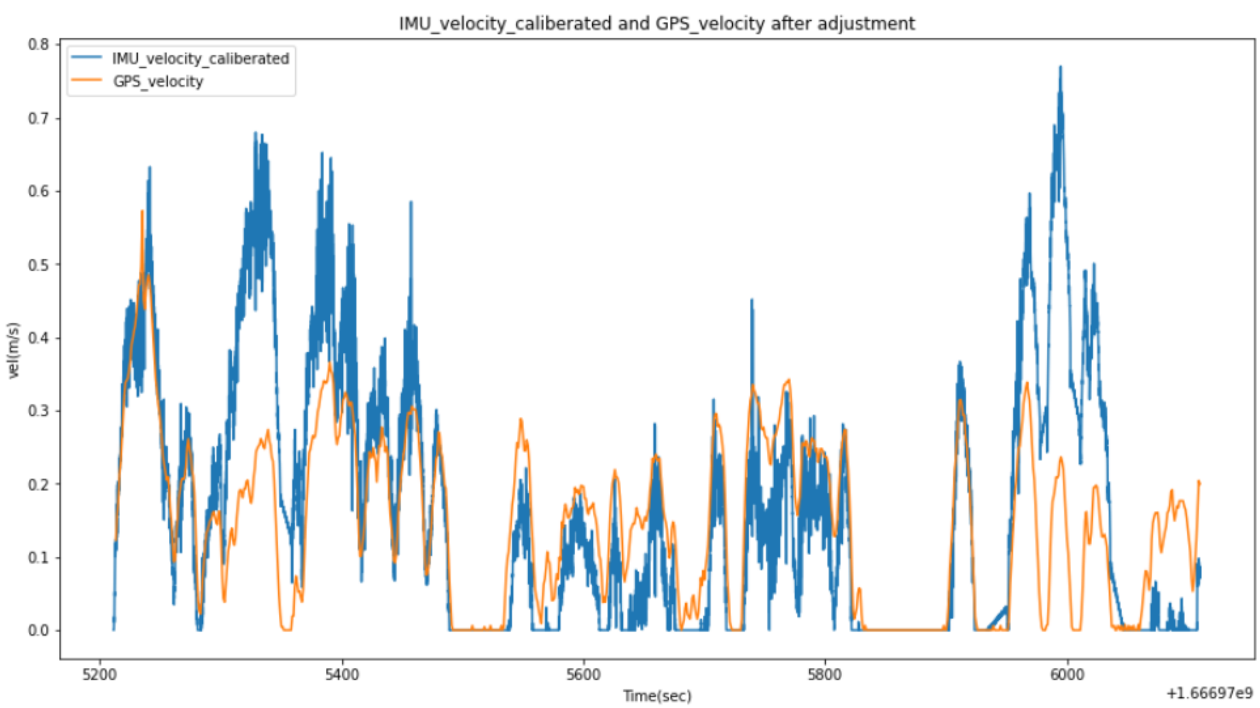

Fig 9 : Velocity from IMU and GPS **after** IMU velocity adjustment

Integrating acceleration gives us velocity which is the case for imu but calculating the distance between one to another point and differentiating them gives us velocity from gps. But there exists a problem with the imu velocity calculation. It goes to negative velocities when there is a stop or sudden halt. This is due to signal stops or sudden braking of the vehicle which causes a jerk in motion. When started again it reagins to the positive axis as shown in Fig 8.

To match IMU velocity with gps velocity we need to remove the jerk from the velocity. We can calculate jerk by differentiating acceleration as jerk is change in acceleration w.r.t time.

Jerk = da/dt

Forward velocity = ∫(Acceleration - jerk)

And then making whatever the negative values to zero we will have the output without jerk and this is represented in Fig 9.

Dead Reckoning :

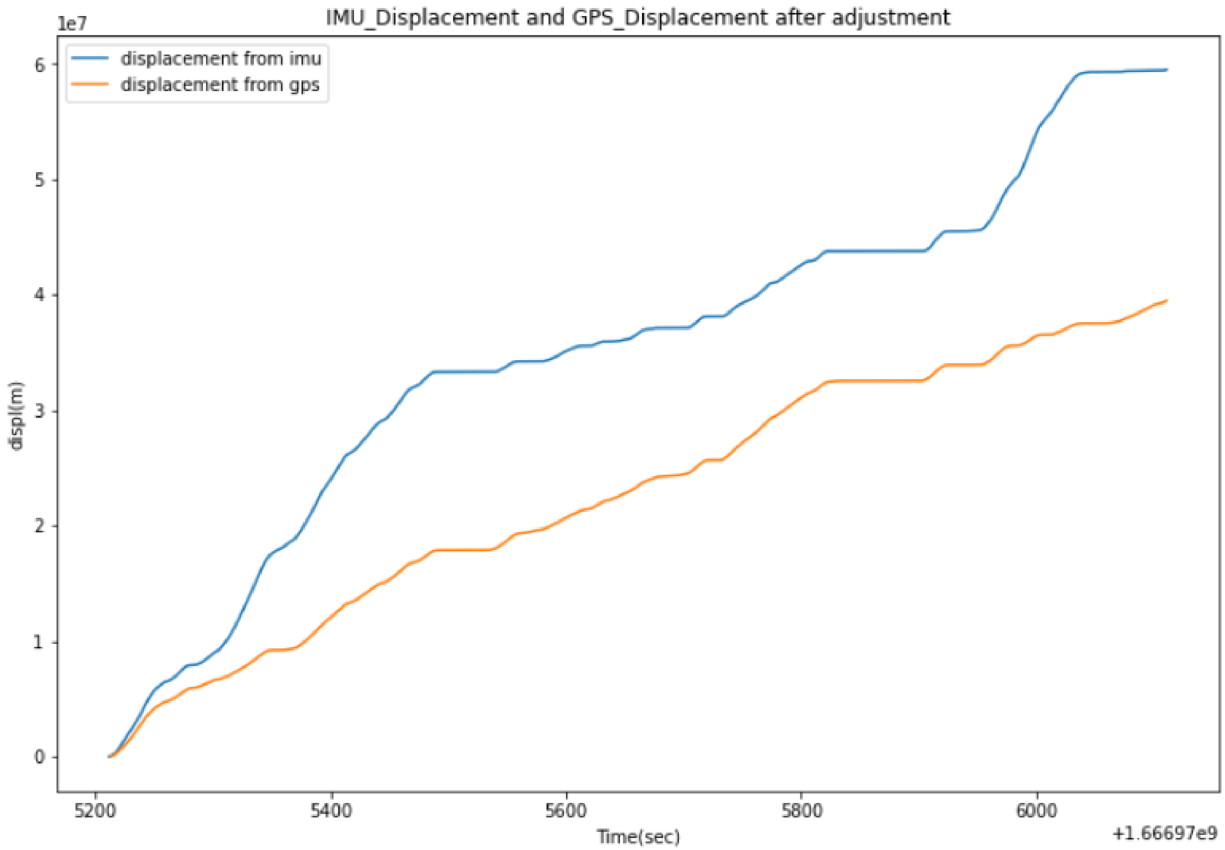

Fig 10 : Displacement from IMU and GPS **after** IMU velocity adjustment

The imu displacement with respect to gps displacement is about 1.5 to 2 meters with a little to reduce the error. This is caused due to the integrated carry forward of the error as the forward velocity has no data remaining of the jerk values. Hence the difference.

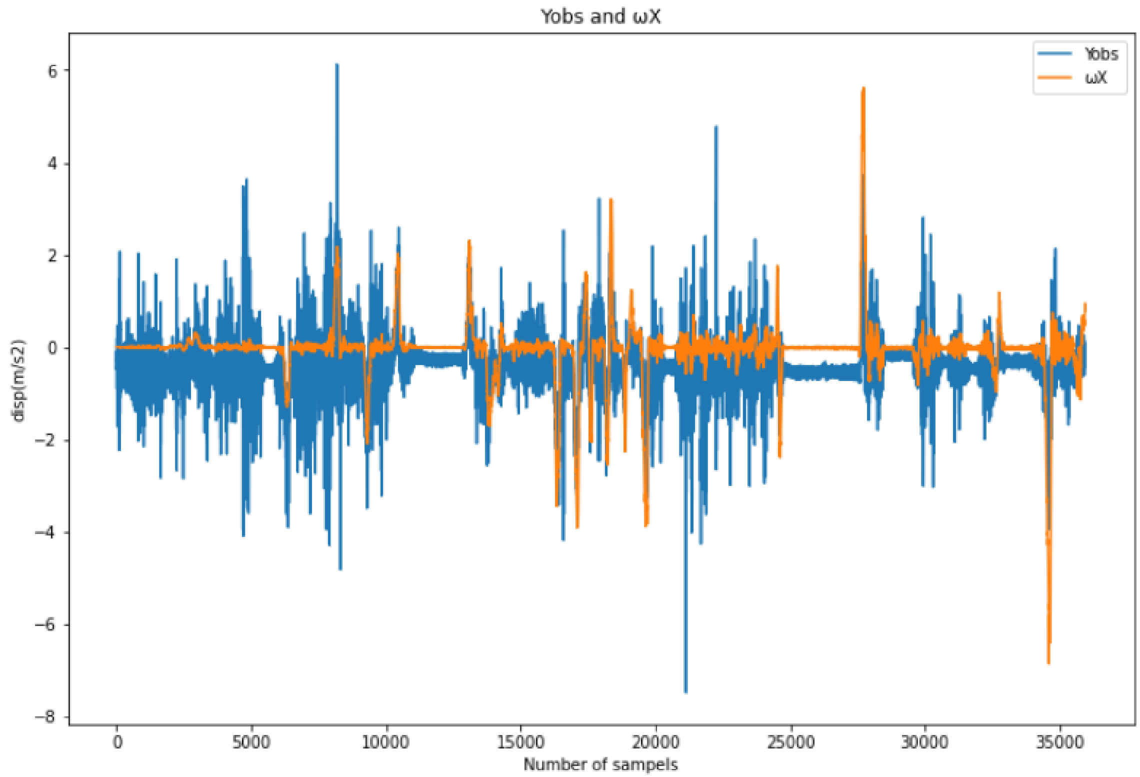

Fig 11 : Velocity from IMU and GPS **after** IMU velocity adjustment

ωX is the the product of velocity obtained from linear acceleration in x direction to angular velocity in z direction. And $\ddot{y}_{observed}$ linear acceleration in y direction.From fig 11 we can see that the noise in $\ddot{y}_{observed}$ is much greater than to that of the ωX. This is because when integrating the error also gets integrated with the original data causing more noise. But for ωX the X velocity obtained when multiplied with ω will make the angular rotation compensation and reduces the error .

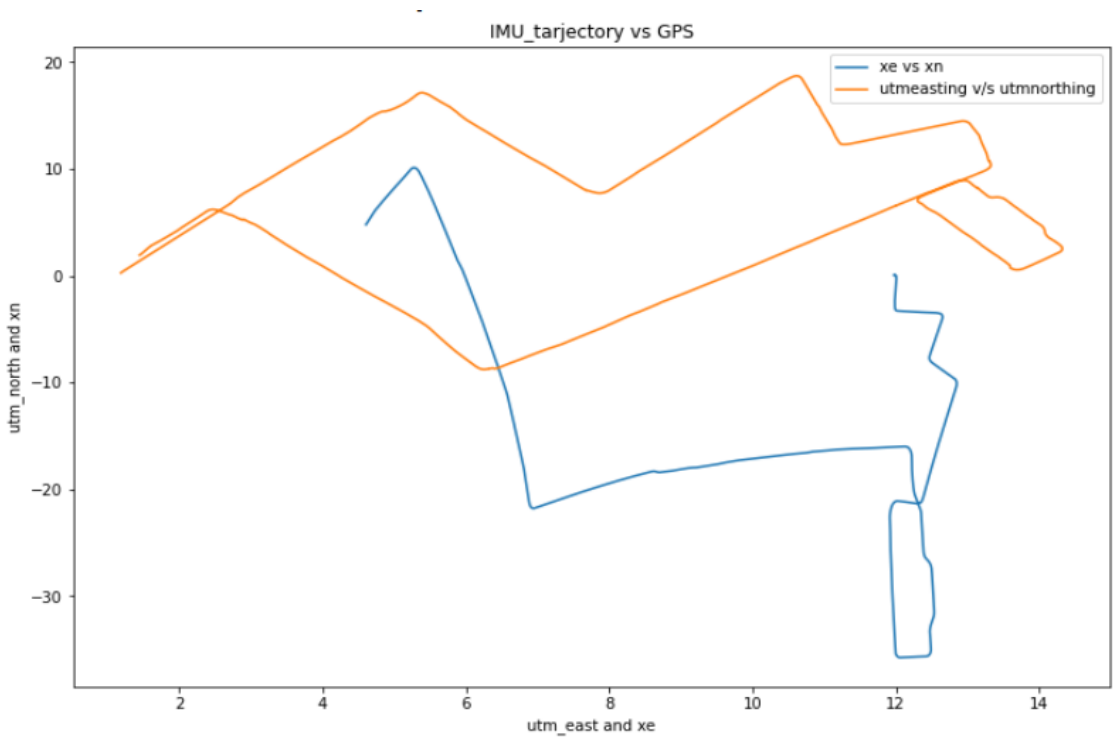

Fig 12 : Trajectory from IMU and GPS

Scaling factor on the x-axis **xe** and y-axis **xn** is 10^4 and rotated about -108 radians.

3Q. As we have calculated multiple yaw angles the outcome I got from my data calibration brings me to use low pass filter yaw , yaw calibrated from magnetometer or complementary filter yaw. And the best fit is based on the outcome with comparison to the gps data. For me the yaw calibrated with magnetic field data and complementary filter yaw gave nearly equal outcomes. With the LPF the probability of missing data is high. But in the other two methods yaw from the magnetometer is less prone to errors as it is calculated with other 2 axis roll and pitch than with complementary filter yaw.

9Q. With the given and observed statistics of vectornav imu it is possible to navigate for a certain distance without any discrepancies but not possible to entirely depend on it without a position fix. On fine tuning the parameters to a much precise value achieving dead reckoning without a position fix is certainly a possibility for the future. This is the same in my case.

Driver used for Data collection and data from , German Gonzalez

****