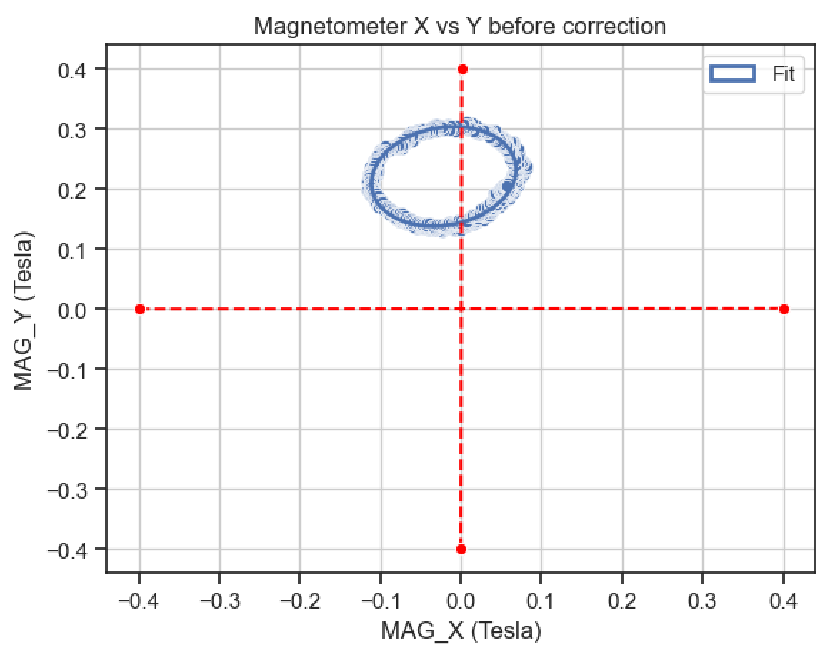
LAB 4: Navigation with IMU and GPS

Task: The goal of this lab is to build a navigation stack using two different sensors – GPS and IMU, to understand their relative strengths and drawbacks and get an introduction to sensor fusion

Procedure: Using the GPS and IMU sensors, firstly imu was mounted on the dashboard of Nuance Car and GPS's Magnetic back was placed at top of Car to enable the GPS to produce data, single dataset was collected, where first recorded for 30 secs with engine turned off to avoid any false error due to engine vibrations in imu since it's a very sensitive sensor, after that driving rounded circles were conduct to get data for calibration of magnetometer in imu, and then started to travel for roughly 4 Kms and returning to starting point to have reference path for dead reckoning.

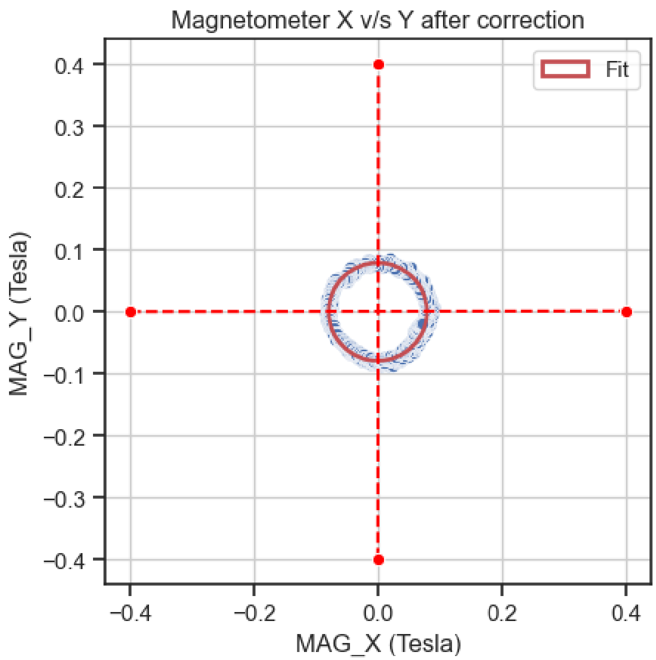
Calibration of Magnetometer: The intend of doing this is to compare the Yaw calculated by magnetometer to enable the comparison between yaw calculated by magnetometer, accelerometer, and gyroscope in imu since magnetometer measures the Earth's Magnetic field hence by taking that reference it is possible to improve the accuracy and reliability of yaw estimation in IMU where other sensor such as accelerometer and gyroscope get affected by error present in environment. To calculate the Yaw, we must take magnetic field at x axis and y axis and apply arctan2(magy/magx), but there are two types of errors that can affect the accuracy of magnetometer that are hard iron error and soft iron error. Where hard iron error is caused by magnetic sources in near IMU such as magnets, power supply, etc., this type of hard iron distortion will cause a permanent bias in the sensor output. Soft iron errors are caused due to deflections or alterations in existing magnetic field, where these distortions will stretch or distort the magnetic field depending on upon direction the field acts compared to sensor.

For Our data set the error which was present in IMU was both hard and soft iron error.

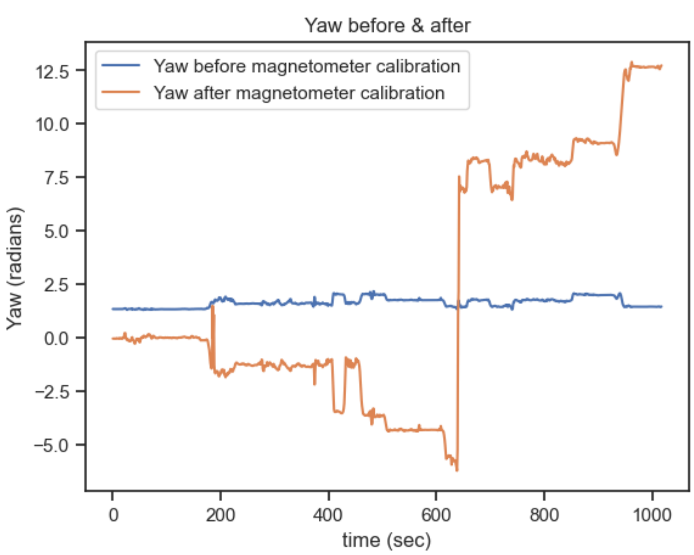


Here it is noticeable that data collected have both error hard and soft iron error so we must conduct calibration so we can have reliable data to begin with.

Calibration: To calibrate the Magnetometer we can begin with first dividing the dataset to the range at which the vehicle was getting rotation now we can take the data of magnetic field at x and y axis and zip them up, to fit ellipse at them, where output will give center, width, height, phi, where, we can subtract the center of ellipse to get the data at center by doing so we can remove the bias which is present in data this cause to remove hard iron error from data. Soft iron error we can remove that by getting phi got from ellipse fitting and we can rotate all the point in data to phi as rotation matrix, which enable the point to rotate and get align straight in terms of ellipse fit, and then we take ratio of width/height which is known as scale factor and use that to shrink the largest axis in ellipse to rescale it as circle and by doing that soft iron error get eliminate usual reason for soft iron error is magnetic field get affect and get elongated at certain axis hence by doing so we can eliminate this error.



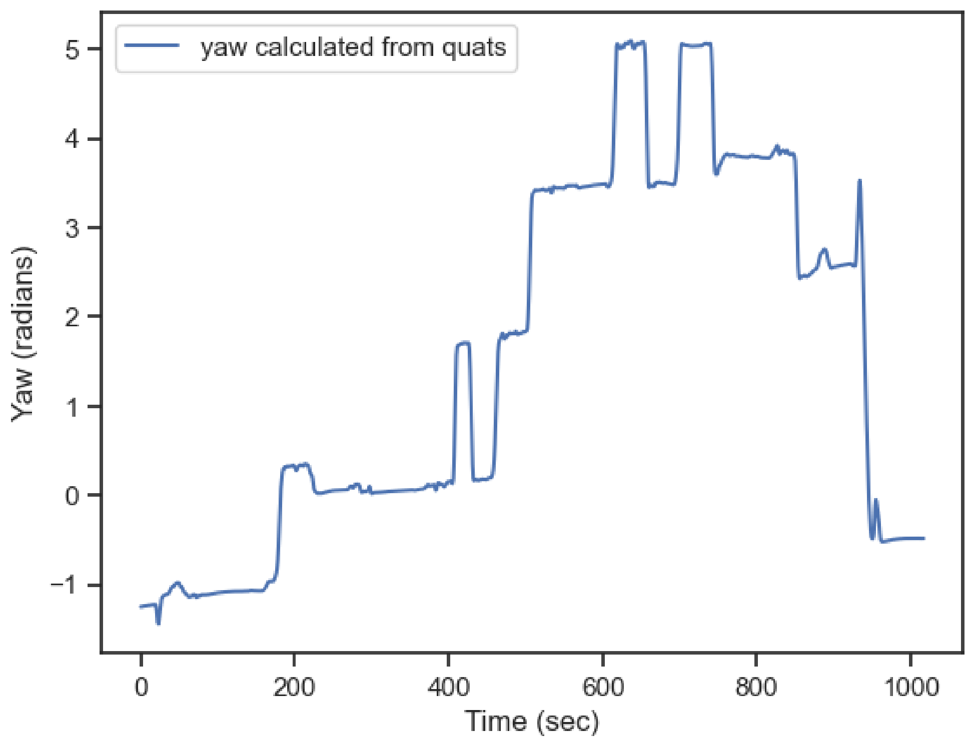
Based on calibration, yaw calculation conducted using arctan2(y/x) result which is achieved are:



‘2) Complementary filter was developed using yaw generated by arctan2(y/x) where y and x are magnetic field at x and y axis where cutoff frequency used 0.05 with Nyquist frequency which is 0.5*frequency of signal and order of two with cutoff frequency of 0.05/Nyquist frequency hence that left with 0.005 hz of cutoff frequency at lowpass filter. Second part is high-pass filter with cutoff frequency of 0.0001 and order of 2 by applying on yaw calculated by integrating gyroscope Z axis [angular velocity at z] and integrating using cumulative trapezoid integration which result in Yaw and for complementary filter I used alpha as 0.05 using complementary yaw = alpha * lowpass_yaw_mag + (1 - alpha) * (highpass_yaw_integrated).

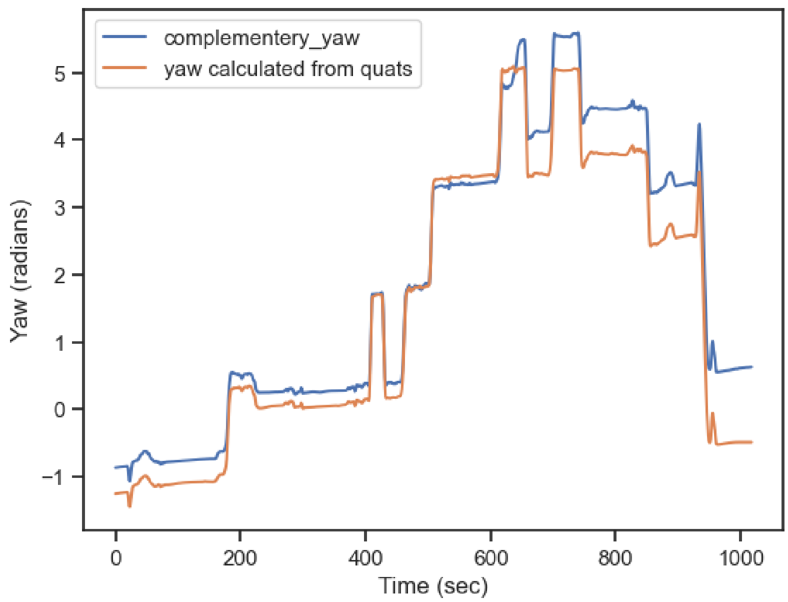
Here for the first reference, there is calculated Yaw which is calculated from quaternions obtained from IMU driver and based on this formula: yaw quaternions = np. arctan2(2*(x*y + z*w), 1-2*(y**2 + z**2)).

Following plot is the yaw obtained by VectorNav:

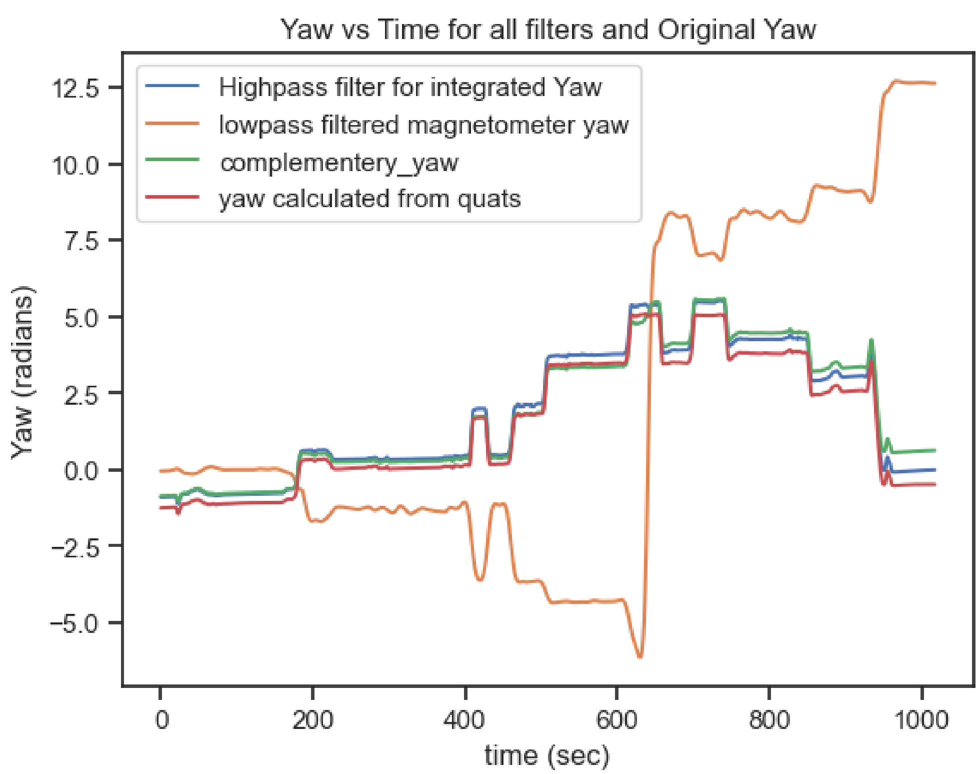


following was obtained from VectorNav IMU, and based on above result it is possible to compare the result from complementary filter to estimate yaw calculated from magnetometer and yaw from gyroscope, here low pass going to remove high frequencies components and stabilized the signal.

Whereas the gyroscope calculated yaw passed through a high pass filter which removes drift from data where it removes low frequency from data, and this stabilize the signal and generate result combined both to get value near to yaw calculated from Quat’s yaw.

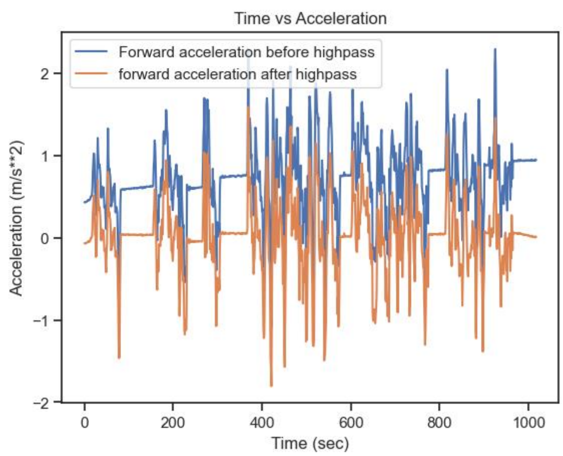


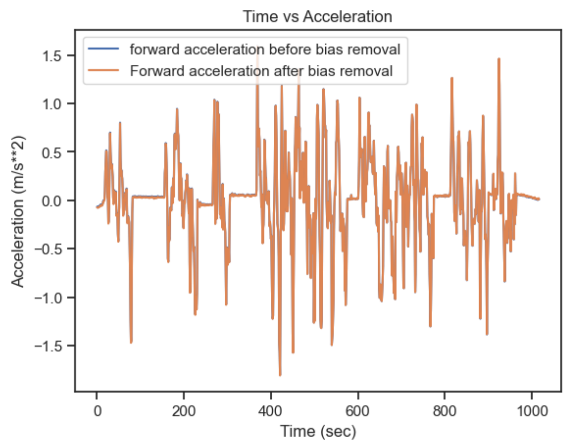
As the above estimation it is noticeable that Complementary filter has been designed appropriately to estimate the yaw in compared to original yaw obtained by IMU hence its conclusive that complementary filter can be used for further analysis. Comparing the architecture of complementary filter by plotting lowpass, high pass, complementary filter with original yaw.



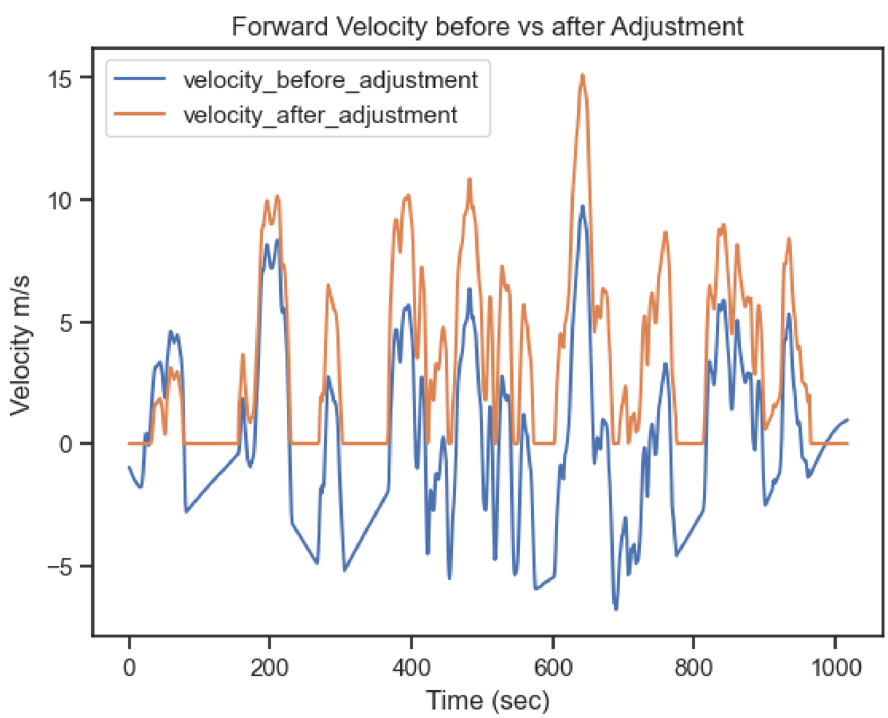
3)Based on the result which are obtained from Magnetometer, Gyro, Complementary filter and yaw calculated by Quaternions, it is fair to conclude that for navigation purposes complementary filter based yaw can use, here the best pick would be yaw calculated by Quaternions but out of 3 output most optimal and closest would be output of complementary filter since error is lesser compare to other output also attributions and information of both components at different proportions are present.

As part of this Lab we consider as forward acceleration is considered as X axis as front of car hence based on that this plot is shown to estimate the acceleration here high pass filter has been implemented to make signal to a bit stable and then implemented detrend the signal to remove the bias of system which sets the data at center to have lesser error in data get before integration.





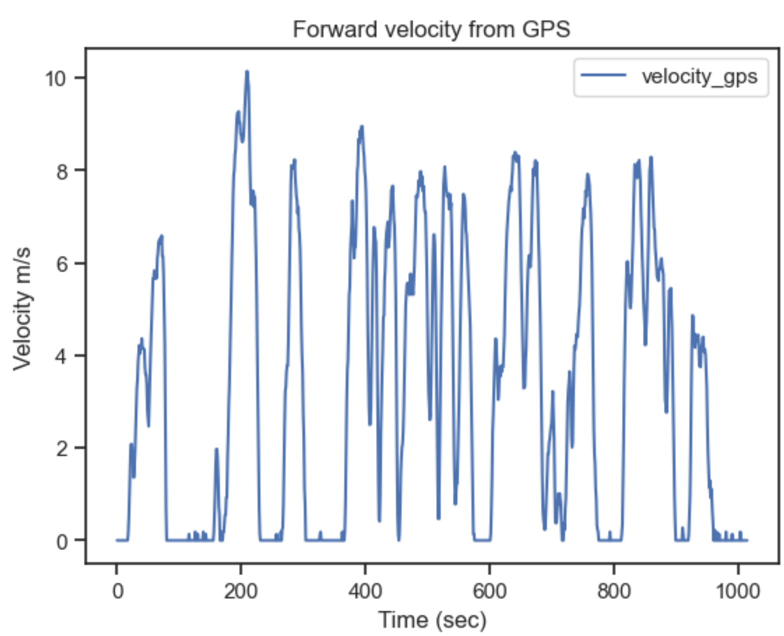
Based on optimized signal, integrating the Linear acceleration with time supplies following result:



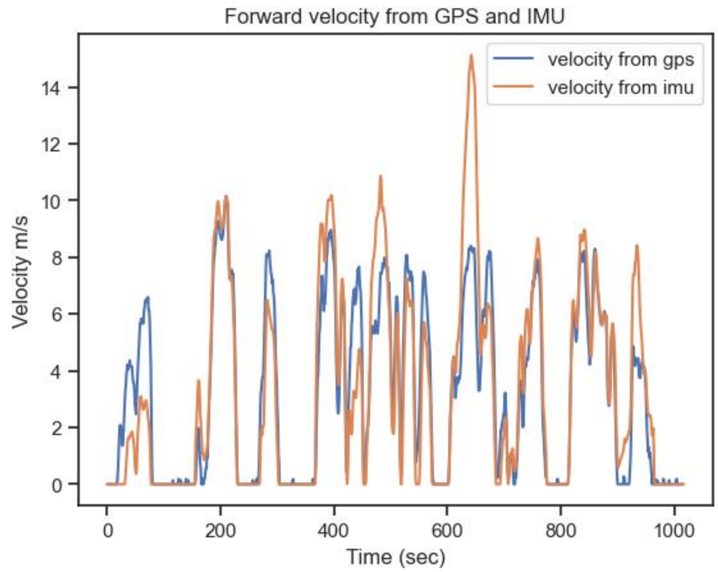
4) It is noticeable that since the accelerometer is sensitive sensor and vibrations can cause error to it but based on above error correction method, it is able to possible to remove bias but one factor which cause trouble is negative acceleration. The case at which when vehicle need to apply sudden brake in traffic at that point acceleration tend to go in negative rate hence when integration is applied that values converges to negative velocities so to eliminate, numeric adjustment need such as cases where velocities go negative for that point zero those output for that particular timeframe and also need select the minimum point at which the velocities get negative the minimum of that value is added to particular windows for example 0 to 200 secs time frame there is spot of negative velocity so that need to adjust and the value that got shifted also need to add back to original window at rectify this error.

5) Based on the Output of velocity from IMU and GPS it is noticeable that there are some dissimilarities are present in output, for this few parameter could be contributing factor, firstly accelerometer in IMU is really sensitive and hence chances of getting error increase really high, which cause sensor to record error and bias shift into it. GPS velocities are calculated by change northing and easting changes and the rate at which it produces data is slow compared to IMU and hence when computing there are lesser chances of having errors to it. Since car is moving on road while collecting data it is noticeable that vehicle had to apply brakes multiple times and at that point the acceleration get negative value since the rate at which imu recording data is really high and when it record negative value at instantaneous time and later on taking integration provide negative velocities windows so to remove that error, pointwise bias negative value elimination is needed to conduct and doing so somewhat generalize the data which shoes some changes.

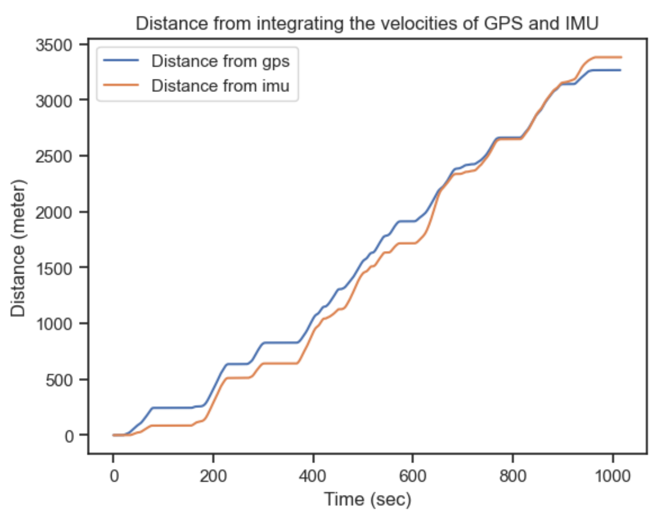
Now considering the GPS velocity, for that easting and northing data which obtained from GPS need to take differentiate with time to obtained velocities where we take consecutive difference distance with change in time supplies the velocities.



Plotting the result to compare both the velocities obtain from GPS and IMU are shown below:



Dead reckoning: To compare the distance obtained by Imu and GPS it is necessary to take integration of both velocities obtain by gps and imu which supply below:

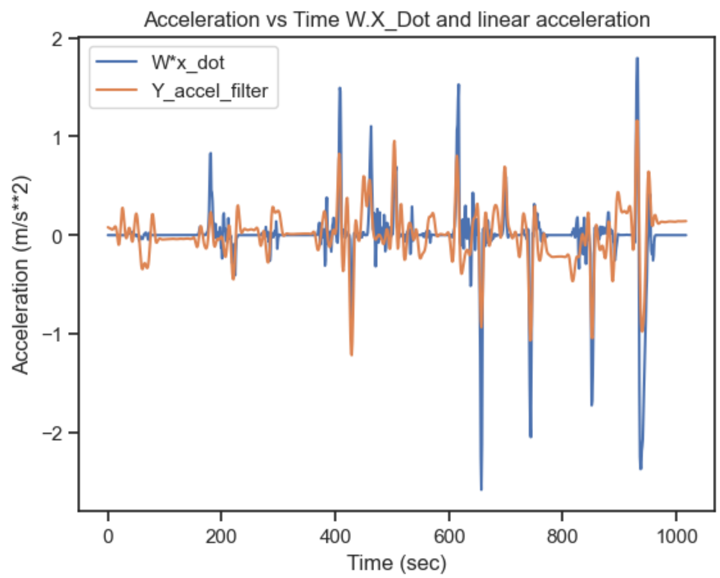


6)It is noticeable that distance obtained by both sensors are quite similar and based on this further analysis can be conducted using the following data.

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

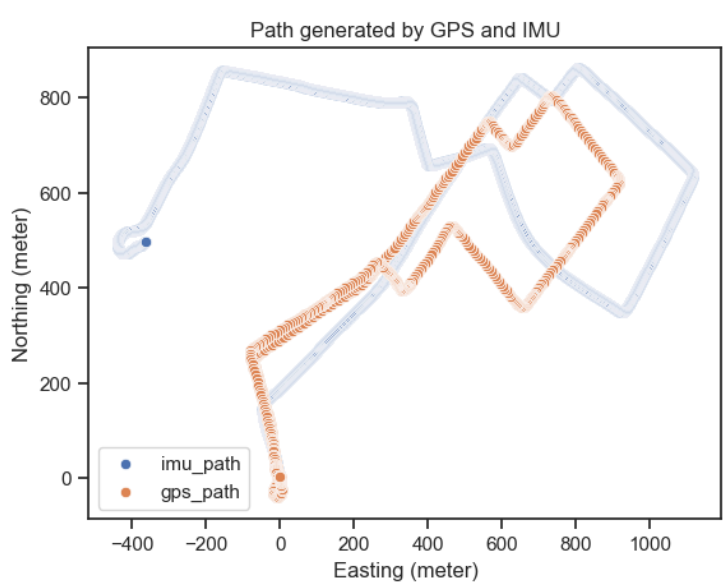
As This formula represents the data observed by IMU and X dot dot is the acceleration of car in x axis and omega represent the yaw of vehicle and Y dot is forward velocity in y axis here xc is imu distance from center of mass of car but for this analysis, going to consider 0 as imu center of mass is same as car's.

Here since cars cannot skidding sideways so Y dot is going to be 0 and since there is no velocity in Y axis then there will be no acceleration hence due to that Y_dot _dot=omega(w)*X_dot should give the same similar plots.



Based on the above figure it is noticeable that fairly the value is similar with for both with some error present in it since we used velocity which is obtained by acceleration in x axis it is already noted that the error in velocity is adjusted to reduce the noise in the system but even after adjustment there are still presence of error can been seen which get carried in while plotting although compare to Acceleration in Y axis it still reasonable.

7)for this Have to use scaling of 1.2 for both easting and northing coming from imu since data was quite large in size compare to gps so had to scale down, where the base path is converged in both cases yet its noticeable to have slide irregularities in IMU where path get deform after certain period and change the direction of path



8) considering the figure above it is conclude that for starters nearly 180 meter to northing the IMU and GPS perform as expected and their values were stable, but after that error started to grow and change the path trajectory where earlier it as lesser and increase over period of time, where it is not really possible to estimate the result based on how long imu will be able to navigate without a position fix where there are various parameters which can affect such as initial accuracy of position, and environment at which the operation being conducted, here the accuracy of system in experiment get decrease over time for this external sensor or Kalman filter can be used to decrease the error. For approximately 1 minute and 21 seconds the sensor supplied correct results after that values started to differ over span of experiment. Dead reckoning matched for starting of experiment but accuracy of imu tend to reduce due to constant errors getting integrated with time which allows the system to change the expected trajectory due to the behavior seen where at beginning the system was working accurately but over period of time accuracy dropped based on this we can expect the above case is somewhat true, where proper calibration and fixed mount can be used to decrease the error at some extend.

9) To calculate Xc

It can easily be done by taking this equation.

$$\ddot{y}_{obs} = \ddot{Y} + \omega\dot{X} + \dot{\omega}x_c$$

Since I know my Vehicle will not skidding in so Y (dot dot) will be going to zero; Now taking yobs (dot dot) = omega*(X dot) + (omega dot) *xc

Where, xc= [yobs (dot dot)-omega*(x dot)]/omega dot

Where Xc=0.587 approximate to 0.6 depend on filter cutoff frequency if applied on imu velocity