Robin Yohannan
EECE 5554 Section 2
Lab 2 – GPS Device #06 & IMU #13
2/13/20

1. Objective

The objective of this lab was to characterize IMU noise and investigate navigation using IMU data by dead reckoning while combating sensor noise.

2. Methods

The inertial measurement unit (IMU) used in this lab was the VectorNav VN-100. The IMU has a 3-axis accelerometer, 3-axis gyroscope, and 3-axis magnetometer. For the first part of the experiment, the IMU was secured to a flat surface in a basement and data was collected while the IMU was stationary. In the second part of the experiment, the IMU was secured to the flat part of a car dashboard and a GPS was placed on the roof.

3. Results & Discussion

Part 1. Stationary IMU Data

The following plots present the average of 4 one-minute trials of stationary IMU collection.

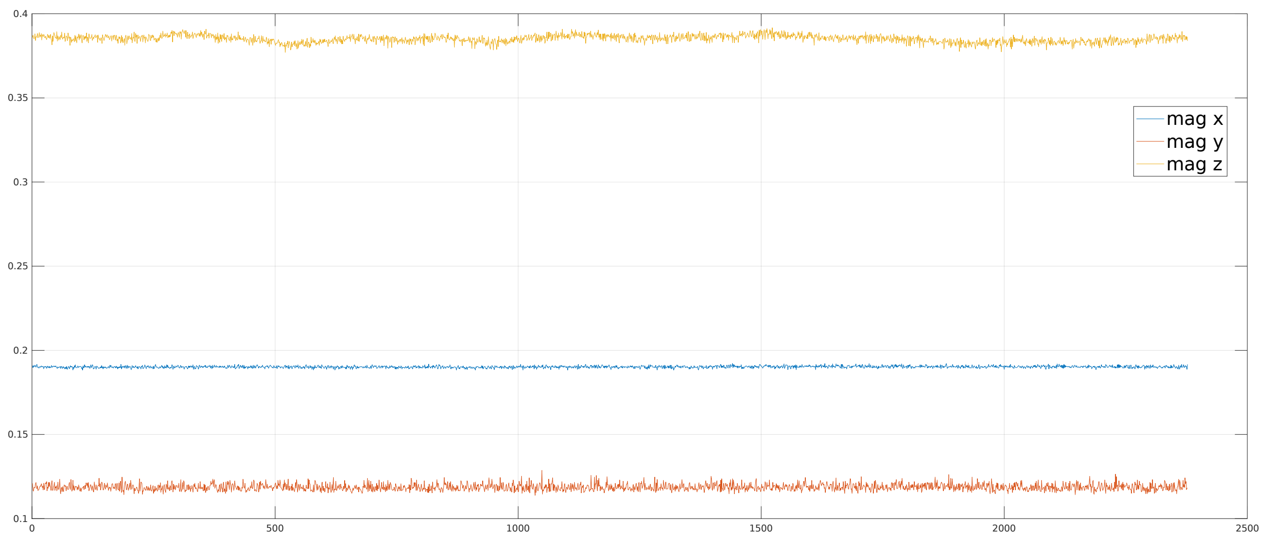

Figure 1: Stationary magnetometer data. The X-axis unit is the number of samples and the Y-axis unit is in Gauss.

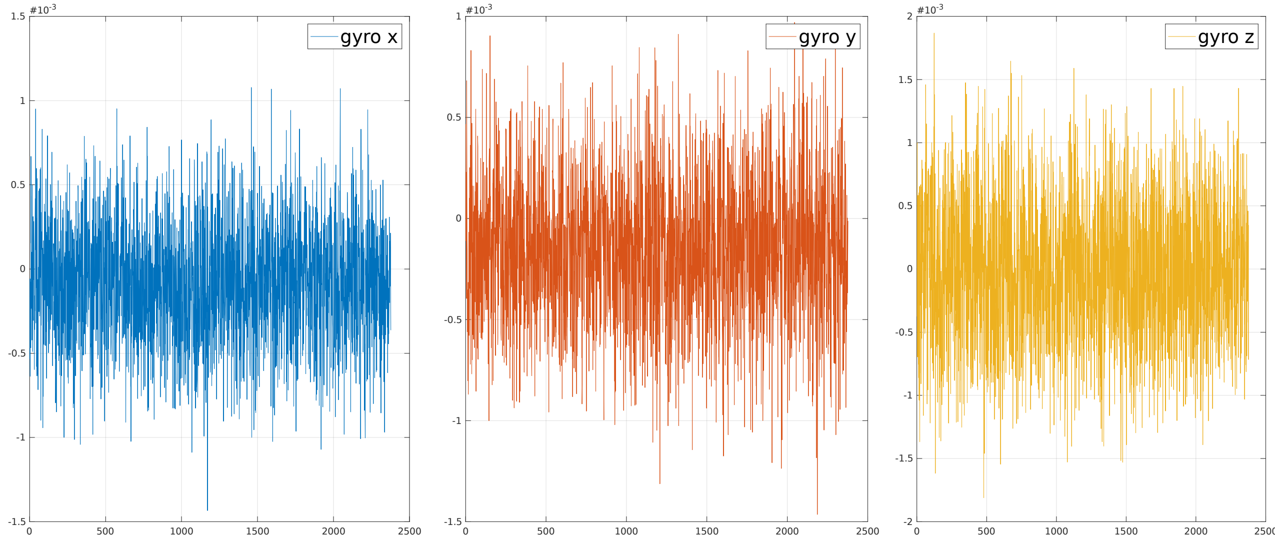

Figure 2: Stationary gyroscope data. The X-axis unit is the number of samples and the Y-axis unit is in radians/second.

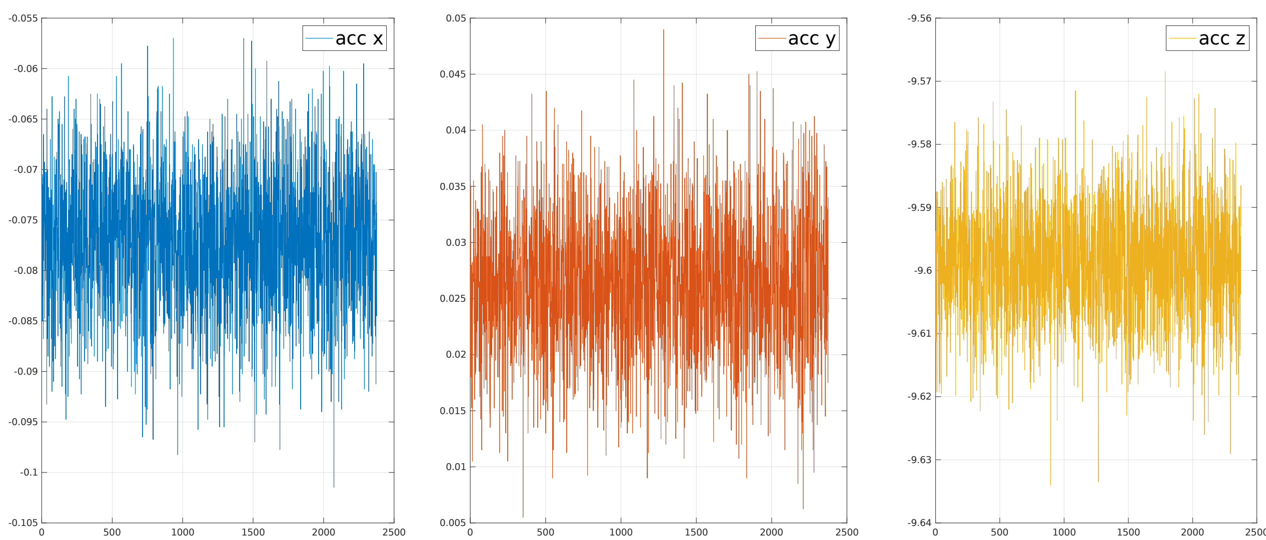

Figure 3: Stationary accelerometer data. The X-axis unit is the number of samples and the Y-axis unit is in m/s^2.

After plotting a histogram of the data shown above, the noise characteristics were observed to be Gaussian and the characteristics of the noise are presented below.

Statistic	Mag X	Mag Y	Mag Z	Gyro X	Gyro Y	Gyro Z	Acc X	Acc Y	Acc Z
Mean	0.1902	0.1187	0.3851	-0.0926e-3	-0.1475e-3	-0.0211e-3	-0.0771	0.0264	-9.5988
Std. Dev.	0.0006	0.0020	0.0022	0.3470e-3	0.3580e-3	0.5631e-3	0.0066	0.0060	0.0087

Table 1: Mean and standard deviation values for the 3D IMU data collected.

Part 2. Magnetometer Calibration

After collecting IMU and GPS data while driving a car in a loop around campus, the magnetometer data in the X and Y axes were analyzed to offset the hard and soft iron effects that were present during the trial. At the start of the trial, the car was driven in a circle 3 times and the magnetometer data collected in that period of time was sliced out for further processing. The iron effects in the magnetometer data were removed by fitting an ellipse onto the data using a MATLAB function [1] and finding the center and the lengths of the major axes of the fitted ellipse. The hard iron effect was removed by subtracting the fitted ellipse center X and Y coordinate from the magnetometer X and Y-axis data to center the data at the origin. The soft iron effect was removed by calculating the ratio of the major and minor axis of the fitted ellipse and multiplying it to the Y-axis magnetometer data. Figure 4 presents the magnetometer data in the X and Y axes before and after removing hard and soft iron effects.

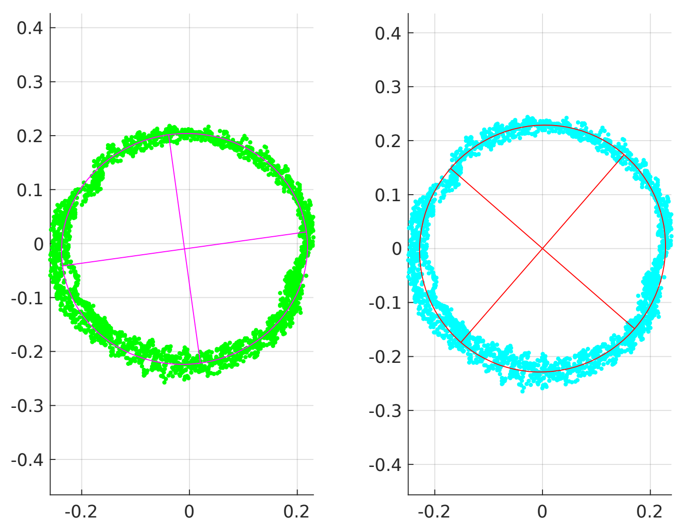

Figure 4. Magnetometer X and Y-axis data (original left, corrected right)

Part 3. Yaw Estimate

In order to estimate the yaw using the magnetometer and gyroscope data, the corrected magnetometer data was converted into heading by the following equation:

$$\psi_{mag} = -\tan^{-1}\left(\frac{y_{mag}}{x_{mag}}\right)$$

The yaw estimate from the gyroscope data was calculated by integrating the angular velocity measurements in the Z-axis. In order to reduce the impact of the error in the gyroscope measurements on the integrated yaw estimate, an offset was subtracted from the Z-axis angular velocity data. The offset was determined by taking stationary IMU data prior to starting the car and finding the mean of the measurements in the Z-axis, which in this experiment was found to be -0.1833 rad/s. The magnetometer data was also corrected using offsets also found using the stationary data, which in the X-axis was found to be -0.1832 G and in the Y-axis was found to be 0.036 G. Figure 5 shows both yaw estimates calculated from the magnetometer and gyroscope data.

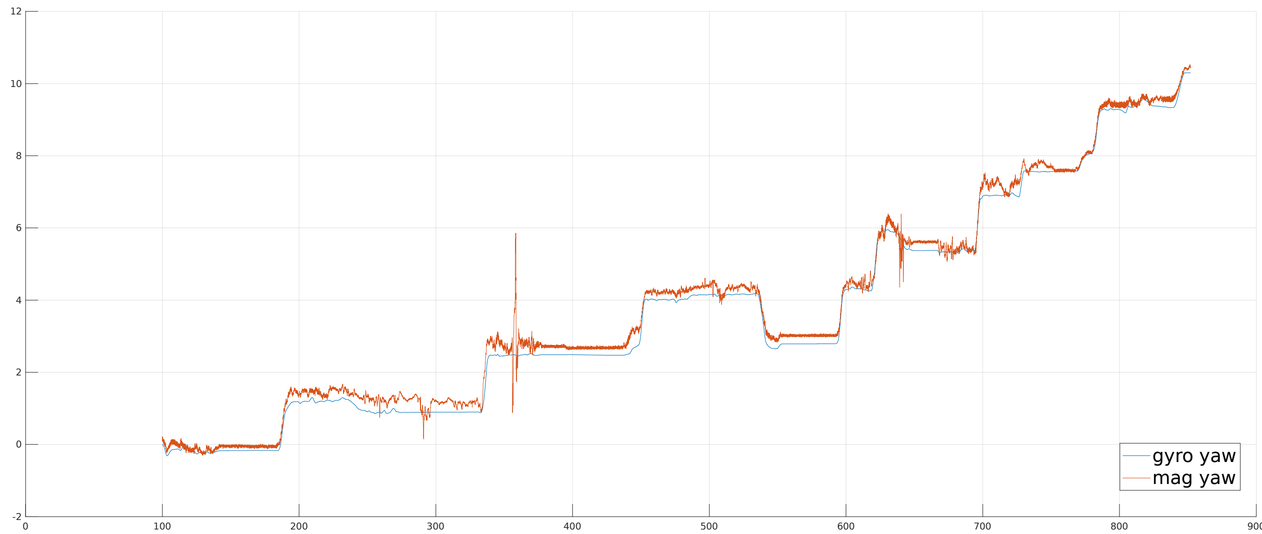

Figure 5: Yaw calculated from magnetometer and gyroscope data

In order for the magnitude of the estimates to match closely, the yaw integrated from the gyroscope data was divided by 40, which was chosen after inspecting the two graphs before scaling. Comparing the two estimates, the yaw calculated from the magnetometer data has more noise that the yaw calculated from the gyroscope data. In order to use both estimates to get a good yaw estimate in the short and long term, a discrete complementary filter was used to find a normalized weighted average of the two estimates, which has the form:

$$\psi_{est} = (1-\alpha)\,\psi_{gyro} + \alpha\,\psi_{mag}$$

Before using the complementary filter, the magnetometer data was passed through a low pass filter (see Figure 6) with a cutoff frequency of 0.5 Hz and a narrow transition region before calculating the yaw estimate.

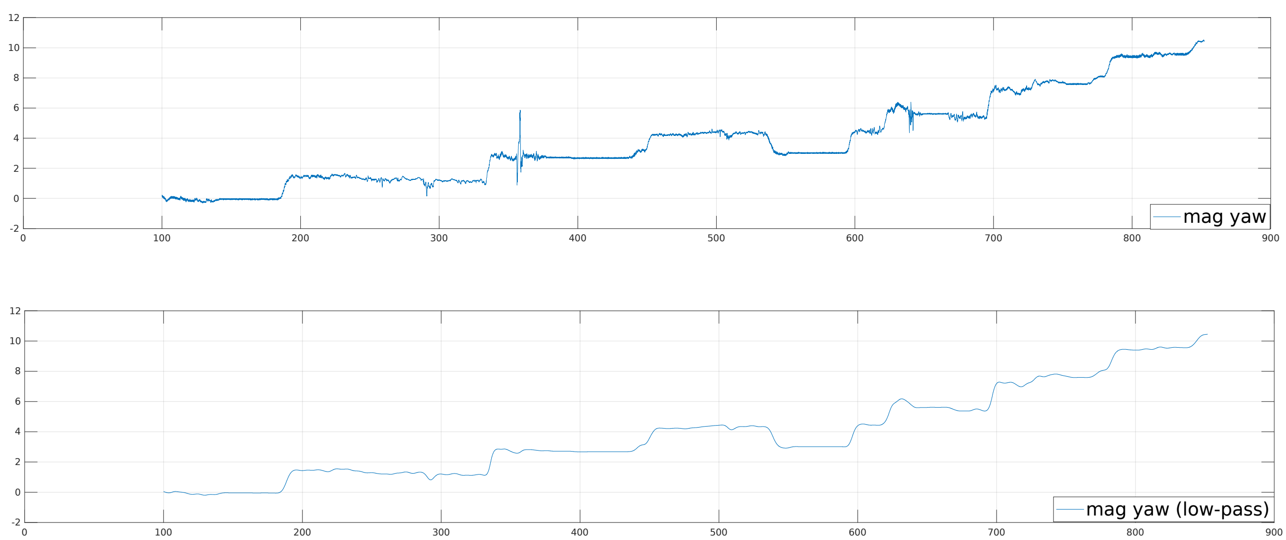

Figure 6: Yaw calculated before and after running magnetometer data through a low-pass filter

In the following figure, the resultant yaw estimate after using the complementary filter is plotted along with the yaw measurements collected directly from the IMU. The alpha value of the filter was eventually chosen to be 0.1 through a trial-and-error process in which the combined yaw estimate was compared to the IMU yaw.

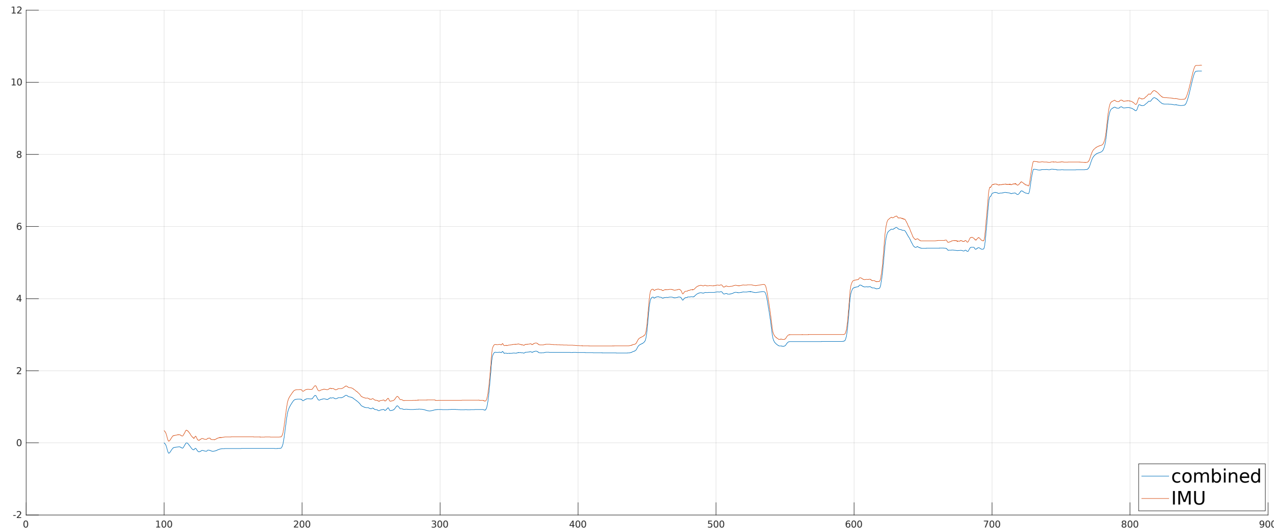

Figure 7: Estimated yaw from gyroscope and magnetometer data along with yaw from IMU

The yaw estimate combined from the gyroscope and magnetometer data tracks well with the yaw collected from the IMU, but has hard and soft iron effects removed, so the overall mean of the signal is shifted lower than the IMU yaw.

Part 4. Forward Velocity Estimate

After calculating the yaw estimate from the gyroscope and magnetometer data, the linear acceleration in the X-axis, which was pointed forward in the vehicle, was integrated to find the forward velocity estimate. After calculating the velocity from the accelerometer data, the forward velocity using GPS coordinates was estimated by dividing the difference between each subsequent easting and northing by the time difference and calculating the L2 norm of the easting and northing derivatives. The following plot gives a side-by-side comparison of the estimated forward velocity integrated from the IMU accelerometer and the velocity derived from the GPS data:

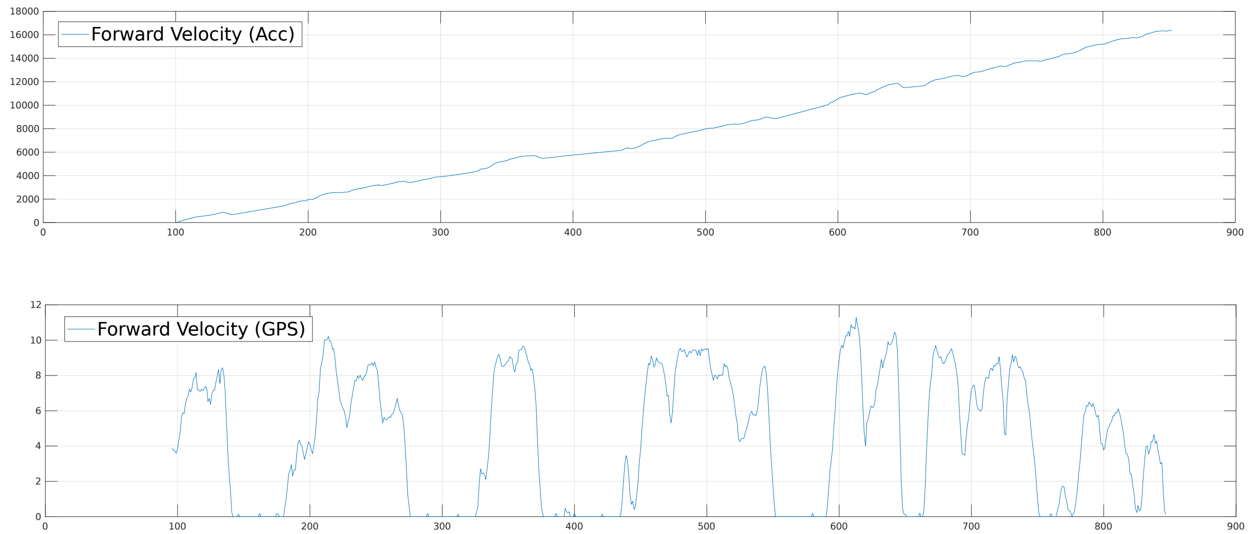

Figure 8: Forward velocity calculated from (unadjusted) accelerometer and GPS data

Comparing the forward velocities found using the two methods, the scale of the accelerometer-based velocity is unreasonable for a moving vehicle. Also the accelerometer-based velocity does not fall to zero when the car is stationary, while the GPS-based velocity does. This can be due to the errors in the accelerometer getting integrated and therefore compounding over the course of the trip.

A few adjustments were made to the accelerometer data to reduce the errors that would grow during integration. The accelerometer X-axis data was first subtracted by the mean acceleration in the first 10 seconds of the trial when the car was stationary. Then, a small bias was removed in the last third of the accelerometer data to reduce the accumulation of error over the course of the trial. Then, the data was passed through an algorithm that would set a window of accelerometer data to zero if all of the differences between pairs of measurements within the window were close to zero. The rationale for this was to make sure that when the vehicle was stationary, such as at a stoplight, the data reflected the situation by forcing the accelerometer data to be zero. After the accelerometer data was integrated, more adjustments were made to the forward velocity data, which consisted of subtracting the minimum velocity from all of the data points, passing the data through the algorithm used in the previous step again, and scaling the velocity by 1/7000 to end up with north and east position data that approximately matched the GPS data. The following figure compares the forward velocity integrated from the accelerometer data and calculated from the GPS data after the above adjustments were made:

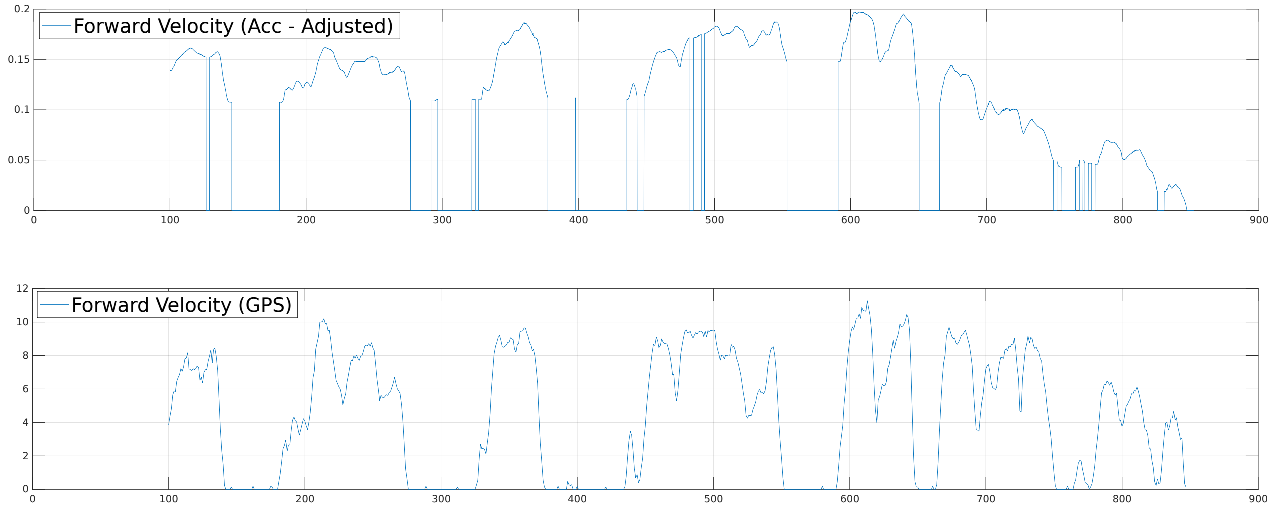

Figure 9: Adjusted forward velocity calculated from accelerometer and GPS data. The X-axis unit is in seconds and the Y-axis unit is in m/s^2

Part 5. IMU Dead Reckoning

In order to compare $\ddot{y}_{obs}$ with $\omega\dot{X}$, the yaw measurements were differentiated to obtain the rotation rate around the center of mass of the vehicle. The following plot shows a side-by-side comparison of the Y-axis accelerometer data with the forward velocity multiplied by the rotation rate:

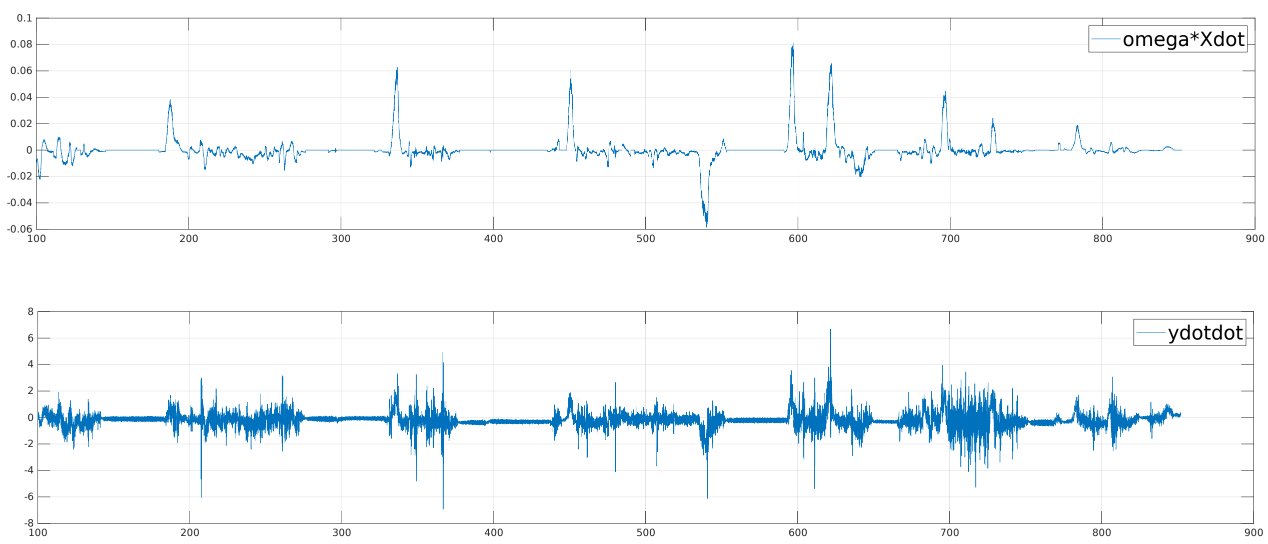

Figure 10: Comparison between $\ddot{y}_{obs}$ and $\omega\dot{X}$. The X-axis unit is seconds and the Y-axis unit is m/s^2

Comparing the Y-axis accelerometer data and the rotated forward velocity, the two plots appear similar but have different peaks and the Y-axis accelerometer has more noise in its measurements. The difference can be attributed to the x_c IMU displacement that is ignored in the rotated forward velocity value but impacts the Y-axis accelerometer measurements based on the equation:

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

The following plot shows the trajectory of the vehicle in the North-East coordinate frame after rotating the forward velocity into the North-East reference frame. The heading was adjusted by $\pi/14$ to make sure the initial track was inline with the GPS trajectory and the starting position was synchronized between the two trajectories. As was stated in the last part, the forward velocity using IMU data was scaled by a factor of 1/7000 to approximately match the size of the GPS trajectory.

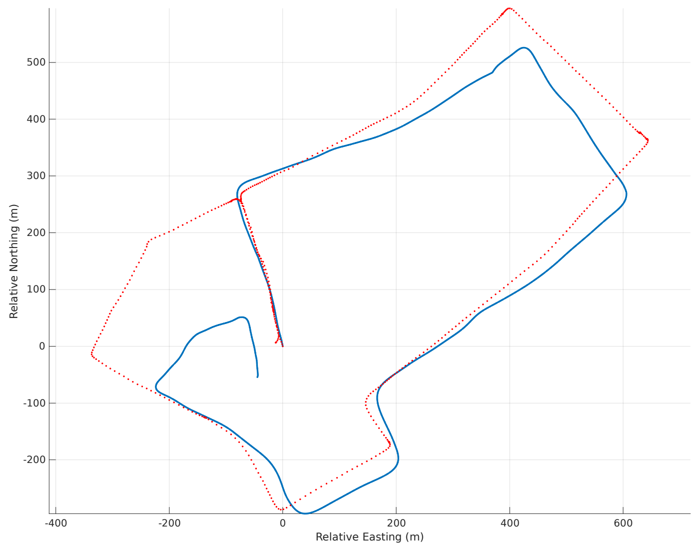

Figure 11: Vehicle trajectories calculated from GPS data (red) and IMU data (blue)

Part 6. Estimating x_c

In order to estimate the IMU offset x_c in the vehicle, the following equation:

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

was rearranged to the following:

$$x_c = \frac{\ddot{y}_{obs} - \omega\dot{X}}{\dot{\omega}}$$

assuming that $\ddot{Y}$ is equal to zero. Using that formula, the estimate for the IMU offset x_c was found to be -0.0025828 meters, which is a plausible result since the IMU was mounted on the dashboard of the vehicle, which is close to the center of mass of a front-drive vehicle.

References

[1] Ohad Gal (2020). fit_ellipse (https://www.mathworks.com/matlabcentral/fileexchange/3215-fit_ellipse), MATLAB Central File Exchange.