EECE5554 Lab4 Report Yao Zhou

1. Part A

Based on the lab4 instruction, our team collected the dead-reckoning data using professor's car. There are two parts of data were collected separately, where the first part was collected at the circular route near Ruggles, the second part was semi-rectangular data near the Back Bay. For ROS driver, we have used previous ROS driver to publish two topics (IMU topic publish, GPS topic publish) simultaneously.

2. Part B

a) Heading estimation

i. Magnetometer Calibration

GPS has "hard iron" and "soft iron" effects, the deformation of hard iron will only shift the center of the circle away from the origin and will not distort the shape of the circle in any way. Soft iron twisting, on the other hand, distorts and distorts the existing magnetic field. Soft iron distortions are easy to identify when plotting the magnetic output, as they distort the circular output into an ellipse. So, before we start processing our GPS data, we need to remove the "hard iron" and "soft iron" effects. Figure 1 shows the magnetometer calibration before and after calibration. The blue line shows the data before calibration and the orange line shows the data after calibration. There is no noticeable distortion in the image, just a phase difference. So, we can easily see that most of the errors are caused by hard iron

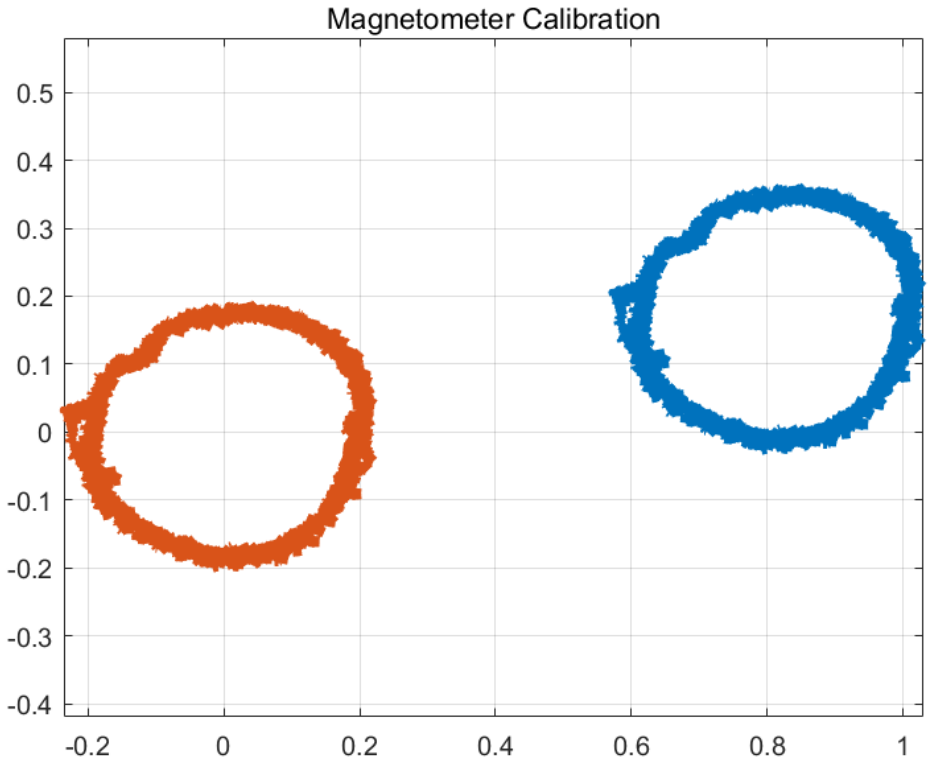


Figure 1 Magnetometer Calibration. Blue – uncalibrated, Orange -calibrated

ii. Yaw angle comparison from Magnetometer and gyro

After corrected the magnetic field, we can compare yaw angles from GPS and IMU, from the Figure 2, we can notice from the graph that there is only a very small difference between the two figures.

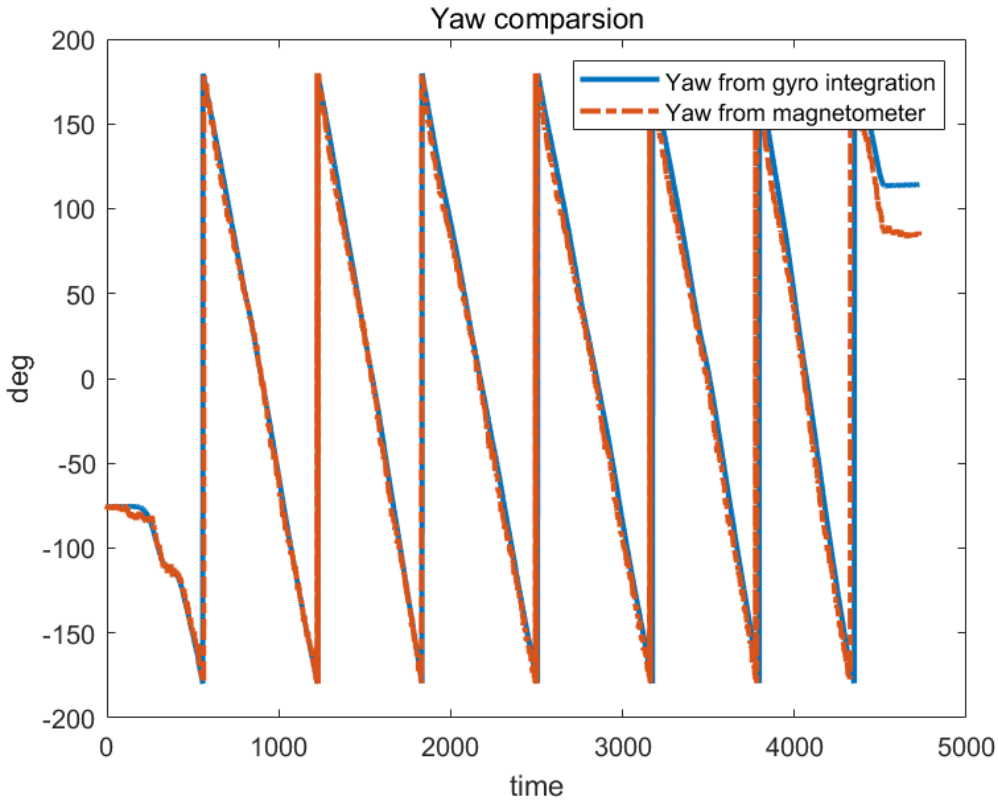


Figure 2 Yaw from Magnetometer & Gyro integration

But data here is still noisy. To reduce the impact of noise on the data, we filtered the magnetometer using lowpass filter and the gyro using high pass filter. The filter result shows below.

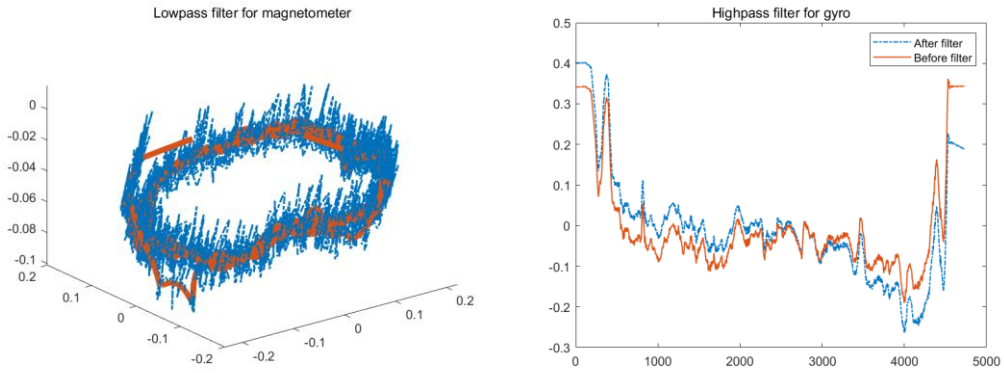


Figure 3 Low Pass Filter Blue-before, Orange-after Figure 4 High pass Filter

After filtering the data separately, I use a complementary filter on the two data to combine the magnetometer and yaw rate measurements described in class to obtain an improved yaw angle estimate. The figure below shows a comparison of yaw primary

transformed from the IMU and after fusion.

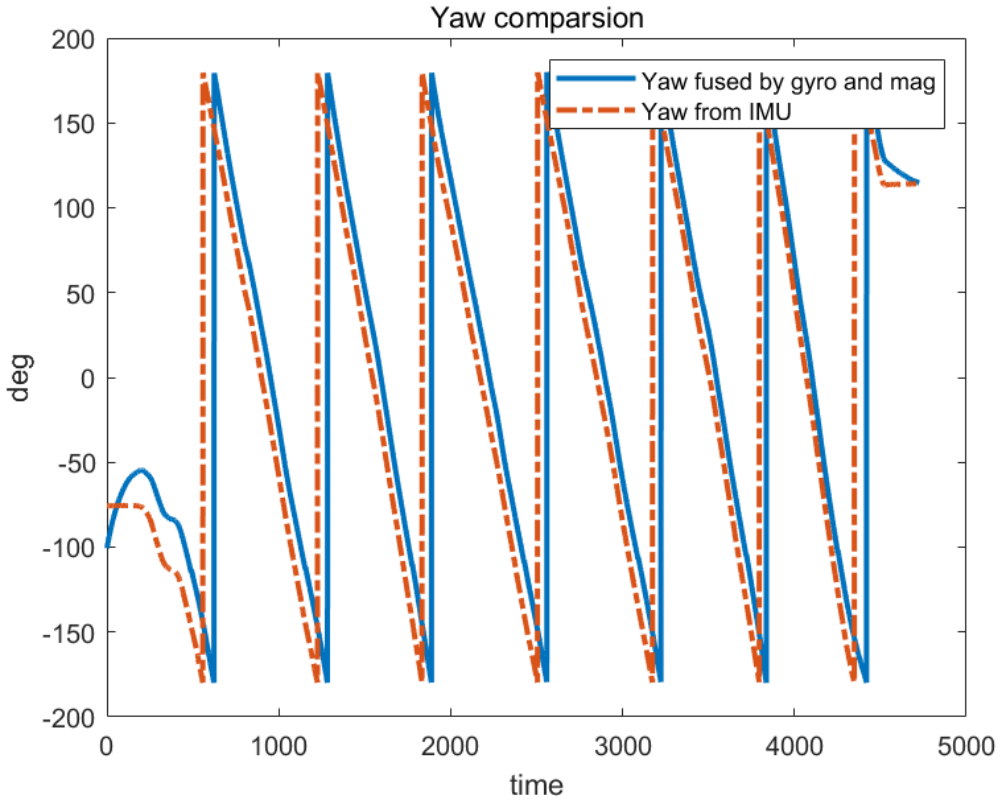
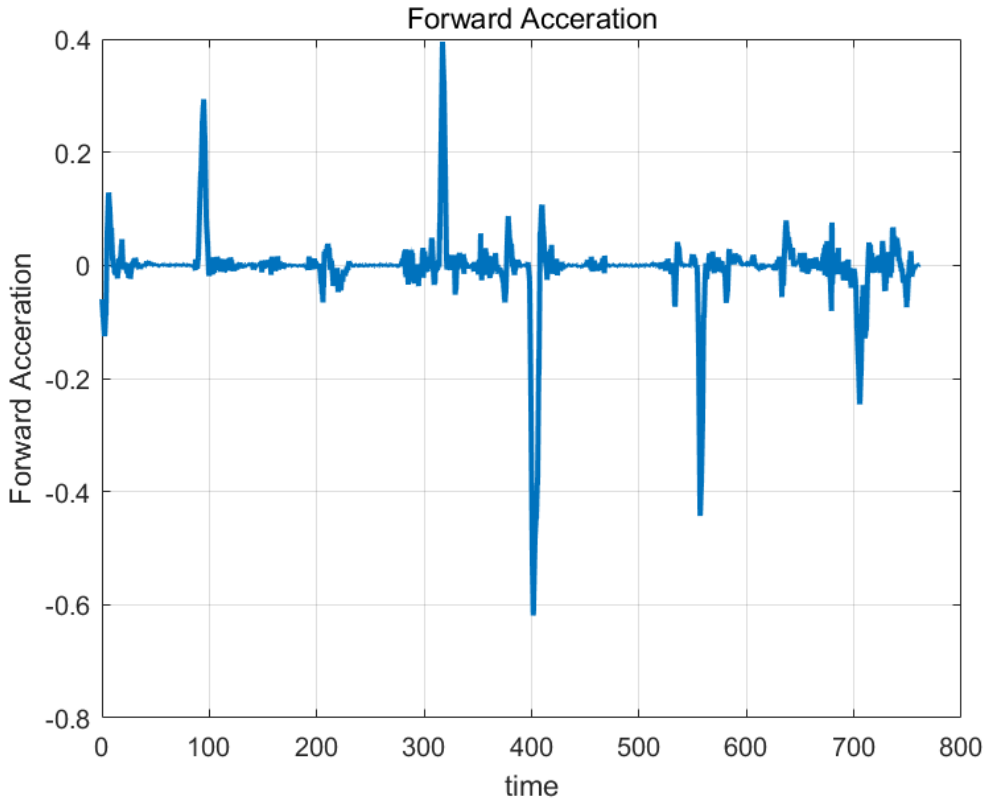


Figure 5 Yaw after fusion & from IMU

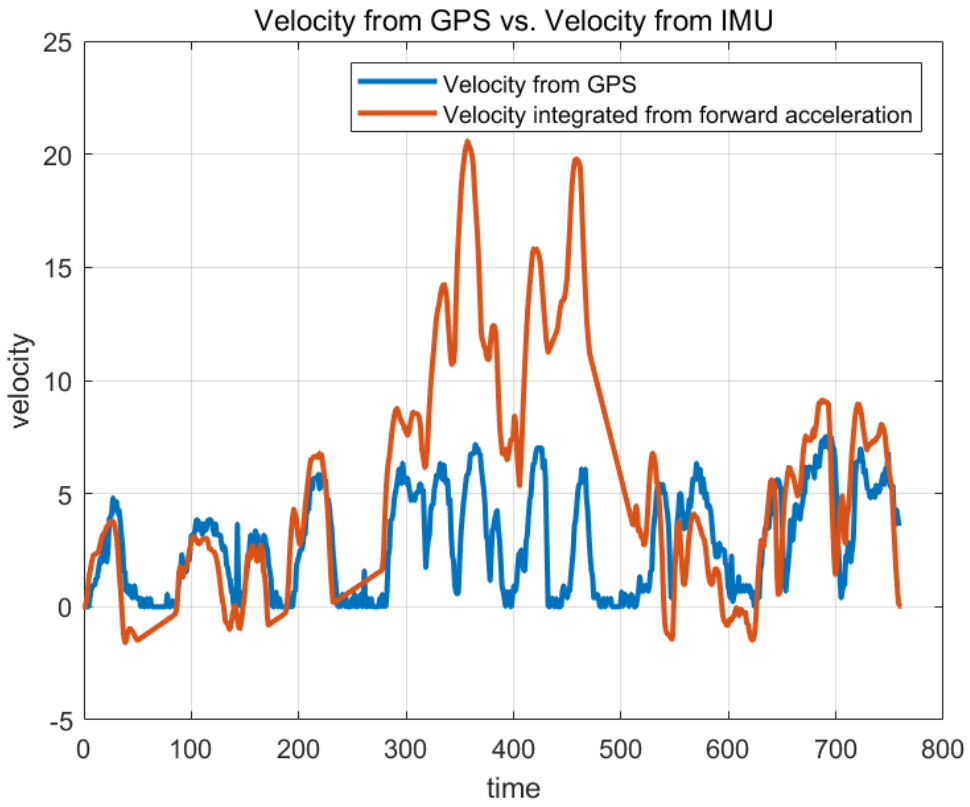
As mentioned earlier, the orientation calculated by the IMU hardware uses a Kalman filter, and the addition of an accelerometer adds some extra precision by tracking the direction of gravity relative to the moving system. As can be seen above, the fused gyroscope and magnetometer yaw estimates are very close to those estimated by the IMU's Kalman filter.

b) Estimate the forward velocity

The IMU is placed in the professor's vehicle with the X direction facing forward. Therefore, we can use the Pythagorean theorem to get the speed of the GPS, and integrating the accelerometer data in the X direction can estimate the speed of the IMU. As you can see from the forward acceleration plot, there is a positive bias in most of the data.



This can be mitigated in a number of ways. The first step is to remove the initial bias by subtracting the average of the first few minutes when the vehicle was stationary. So I designed a function to determine if the car stopped. In theory, more high frequency noise is only produced when the car is moving, so I set the acceleration to zero for the moment when the high frequency noise is low. Also, throughout the plot, there are other areas where the vehicle stops, but with an offset from zero. To mitigate the accumulation of integration errors, which can be visualized by the lack of noise when the car is clearly not moving, these data points can be directly set to zero. The result of this function is shown below.



From these new velocity estimates, we can observe that the peaks and troughs are aligned in time, but the scale changes over time. This is mainly due to the accumulation of errors due to the integration of noisy acceleration signals. Also, the route traversed is not flat and may contribute some gravitational component to the forward acceleration in the pitch angle

c) Dead Reckoning with IMU

The following equations of motion can be simplified by assuming that the vehicle moves in a 2D plane. The center of mass of the vehicle is at (X, Y,0) and the rotation rate around CM is (0,0, ω). The position of the inertial sensor is defined as (x, y,0) and its position relative to the center of mass of the vehicle is (xc,0,0). I used two methods to obtain two Ydot, one directly from IMU and the other through ωX, the comparison graph is shown below.

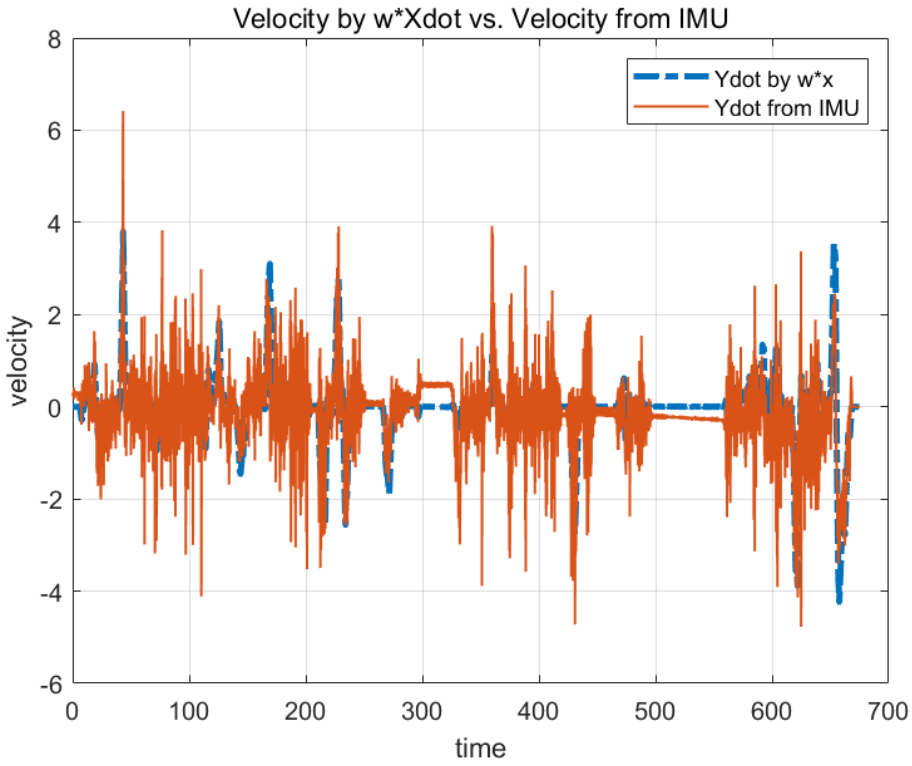


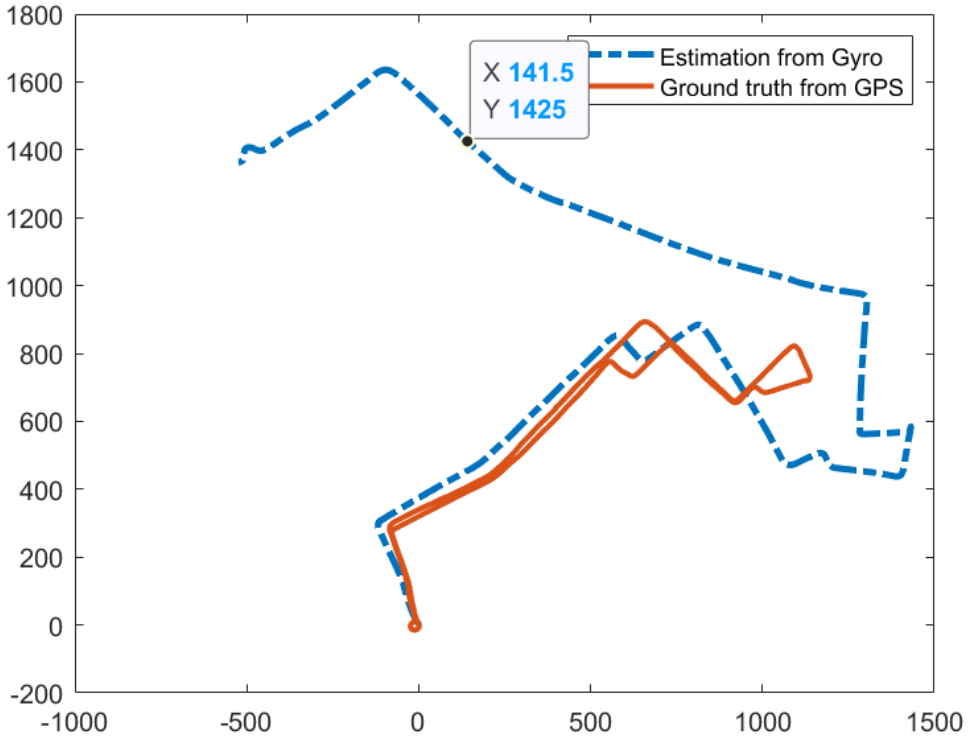
Figure Estimation of forward velocity

iii. Estimated trajectory Comparison

Then I use the magnetometer's heading to rotate to a fixed (east, north) frame of reference. Denote this vector by (ve, vn). Integrate it to estimate the vehicle's trajectory (xe, xn). Below is the equation used.

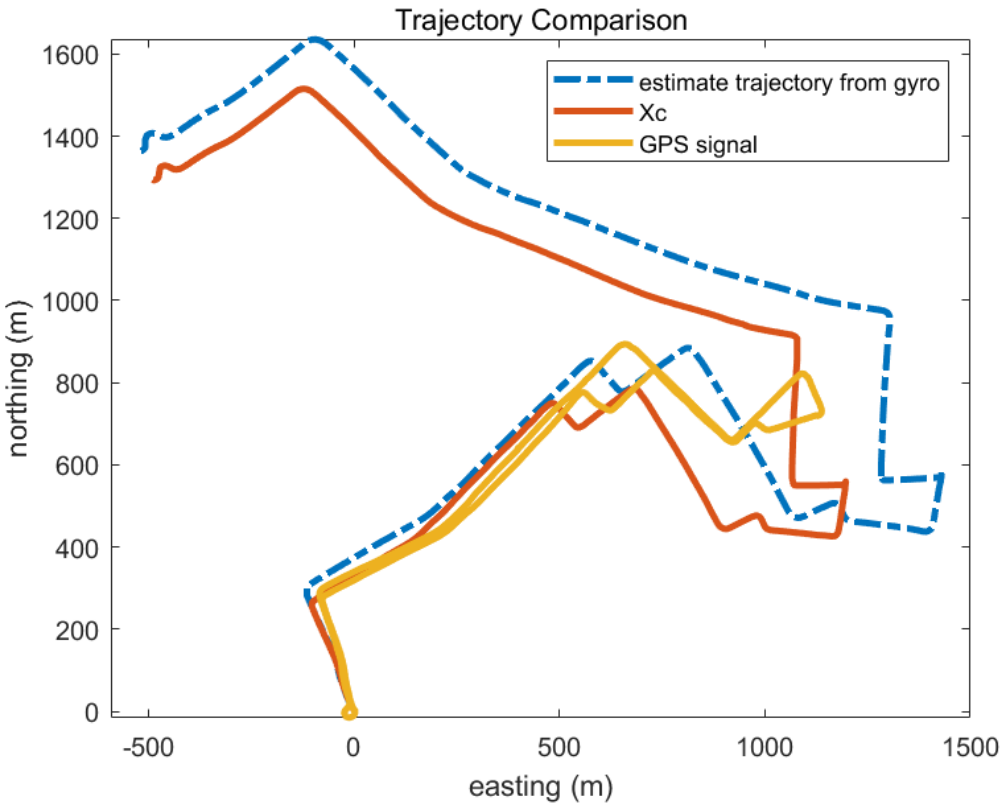
$$\ddot{x} = \dot{v} + \omega \times v = \ddot{X} + \dot{\omega} \times r + \omega \times \dot{X} + \omega \times (\omega \times r)$$

The comparison of estimated trajectory and ground truth from GPS shows below. The stop detection function is used here to reduce the noise.



iii. Xc Estimation

The inertial sensor is offset from CM by r = (xc,0,0), i.e. the vector is constant in the vehicle frame. I used the formula above to differentiate the numerical solution for Xc = 0.20376. The visualization results are shown below.



As expected, based on the error propagation in the velocity and forward position maps above, these maps suffer from significant scaling issues. It is easy to see that the latter system drifts significantly over time compared to GPS, which is constantly updating its global position. Drift can also be seen in the direction measured by the sensor