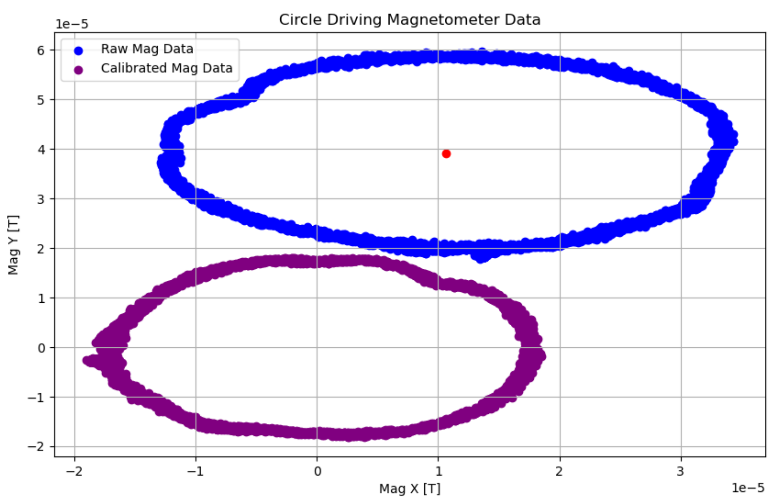
Lab 4 Report

Jon Bear

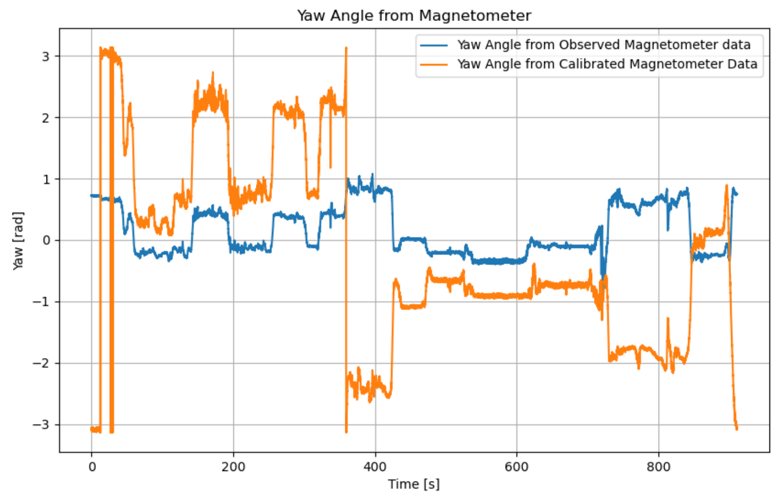
October 2024

1 Estimate the heading (yaw)

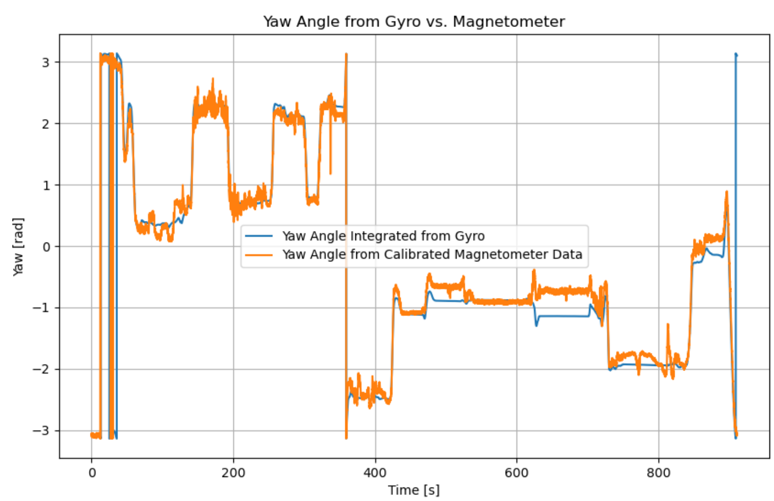
1.1 Magnetometer Calibration



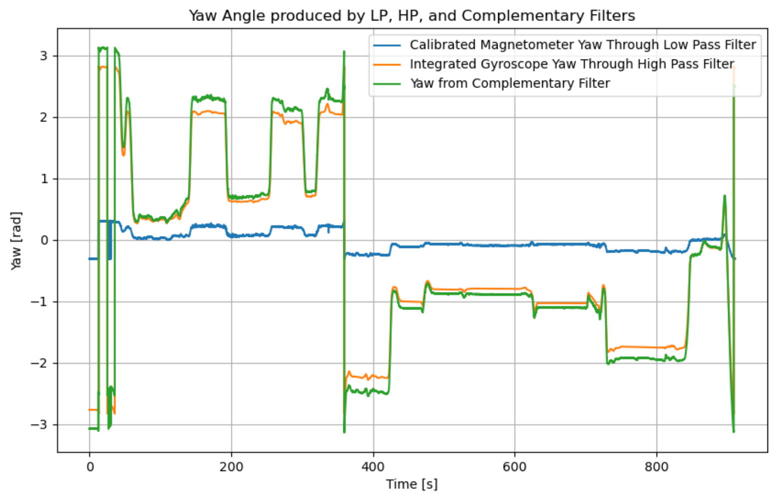
1.2 Magnetometer Data Before and After Correction



1.3 Magnetometer Yaw and Yaw Integrated from Gyro



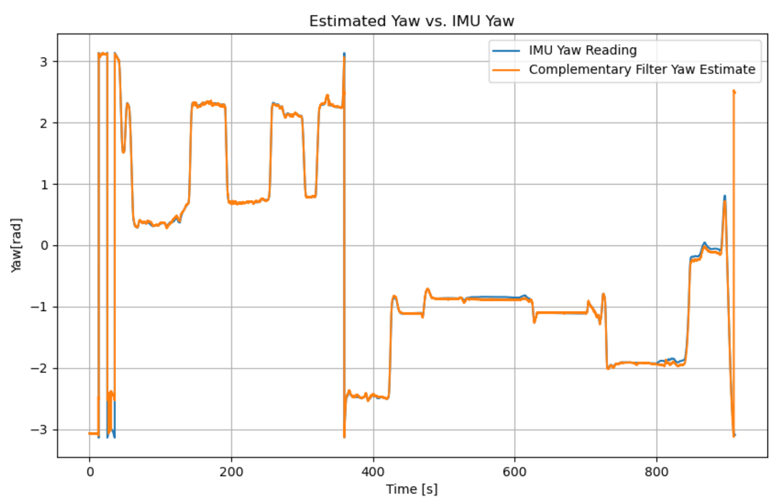
1.4 Filter Plots



$\alpha = 0.9$

1.5 Yaw from Complementary Filter and IMU

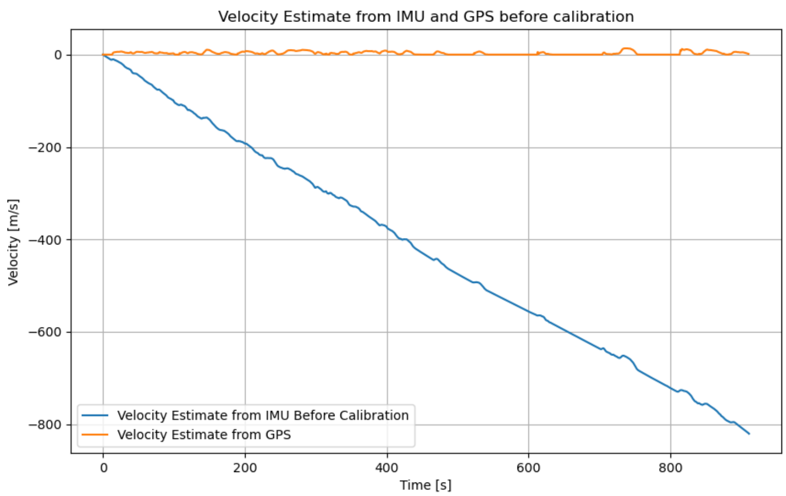
The following plot shows that the output estimate of the complementary filter follows the yaw observed by the IMU extremely well, indicating that the estimate can be appropriately used as the heading of the vehicle.



2 Estimate the forward velocity

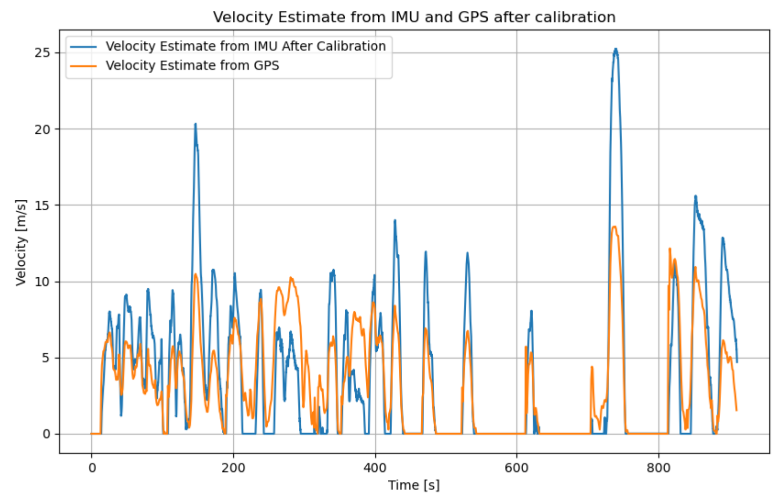
2.1 Velocity Estimates Before Adjustment

The following plot shows that purely integrating the forward acceleration observed by the IMU to determine the forward velocity of the vehicle does not make sense. The integrated forward velocity does have similar behavior relative to itself as the velocity calculated from the gps relative to itself. However, the integrated forward velocity decreases consistently, while the gps velocity never goes below 0. The gps velocity makes sense here, as the vehicle never moved backward, so the velocity should never be below 0. The integrated velocity does go below 0 so it does not make sense.



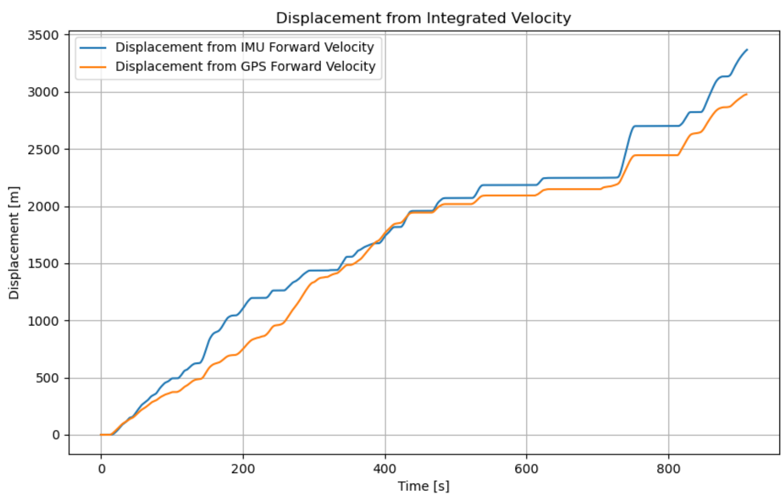
2.2 Velocity Estimates After Adjustment

In order to correct the acceleration data to make the forward velocity integration more reasonable, I first recognized that the vehicle both started and ended at complete rest. Since the acceleration was sampled at consistent intervals, it follows that the mean of the acceleration should be 0, so I centered the acceleration by subtracting the mean. Second, I recognized that this principle applies to two other situations: when the vehicle is at rest, and between rests. Using the gps velocity as ground truth, I can see exactly when in the recording the vehicle was at rest and when it was moving. For both situations, I applied the same centering technique for that time until the end of the data, as the bias experienced during that event would still affect the data as time goes on. In addition, The acceleration of a vehicle at rest is 0, excluding the first and last sample in the time frame of that rest, so I zeroed the acceleration when the vehicle is at rest. Next, I used the truth that the vehicle only moved forward to set any resulting negative values of the resulting integration to 0. Lastly, the magnitude of the integrated forward velocity did not match up well with the magnitude of the gps velocity, so I scaled the acceleration by a factor of 1.96 to achieve more appropriate behavior.



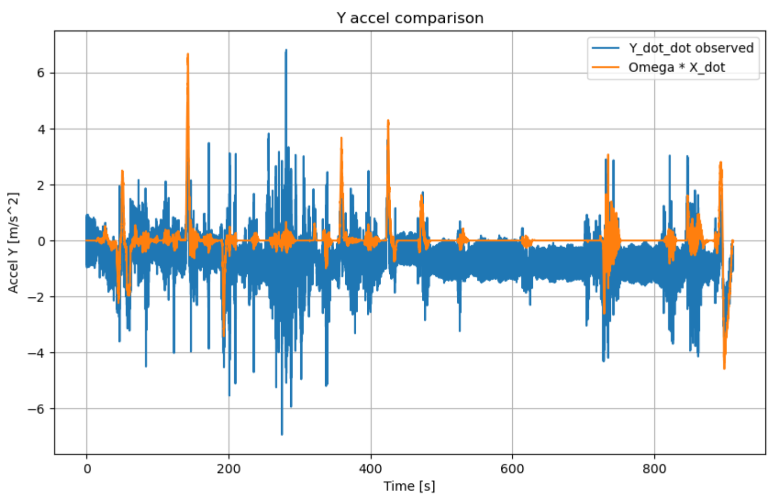
3 Dead Reckoning with IMU

3.1 Displacement from Calculated Velocities



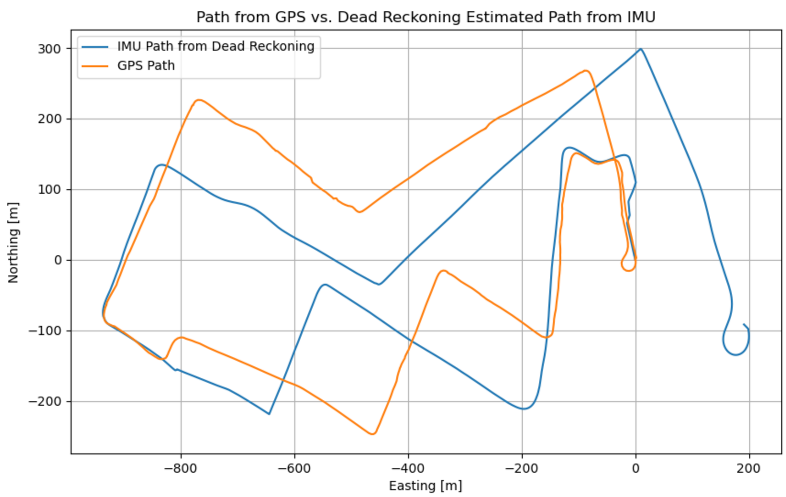
3.2 $\omega\dot{X}$ and $\ddot{y}_{obs}$

The following plot shows $\omega\dot{X}$ plotted against $\ddot{y}_{obs}$. Both datasets show an angular velocity (a.k.a. turn) at the same time, however the magnitudes of these events do not line up most of the time, and are inconsistently different, making it difficult to correct. This difference is likely due to 2 factors. First, The IMU was placed a distance x_c towards the front of the vehicle along the x-axis, causing acceleration observed in the y direction to be felt at a higher magnitude. Second, the IMU may not have been oriented to point perfectly straight ahead, and may have been slightly jostled at times during data collection, leading to both an accumulated bias both from the beginning and at the time of jostling. Our group tried very hard to correct for this bias, however it appears we were not perfect, leading to some bias.



3.3 Followed Path Estimate

Scaling factor: 1.96 (as noted earlier) Rotation angle: arctan(4) = 1.326 rad (observed from gyro data)



4 Analysis

4.1 How did you calibrate the magnetometer from the data you collected? What were the sources of distortion present, and how do you know?

In order to calibrate the magnetometer data, I had to control for two sources of distortion: hard-iron and soft-iron. By looking at our magnetometer data when driving in a circle, we know that the data should reflect an approximate circle centered at 0,0. We observe hard-iron distortion when the center of the ellipse of data is not centered at 0,0 and control for this distortion by subtracting the x and y mean of the magnetometer from the whole data set. We observe soft-iron distortion when the data is not shaped like a circle, and control this distortion by applying a combined rotation-scale matrix to the data so that it looks like a circle. As a result, we can take the x and y means, as well as the matrix, and apply them to the driving data to obtain the calibrated magnetometer data.

4.2 How did you use a complementary filter to develop a combined estimate of yaw? What components of the filter were present, and what cutoff frequency(ies) did you use?

In order to develop the complementary filter, I tried using two different methods. The first used a low pass butterworth filter on the magnetometer data and a high pass butterworth filter on the gyro data, and added both together to get an estimate. However, when I plotted the resulting yaw of the complementary filter, I noticed that it exhibited behavior wildly different than the yaw observed by the IMU, no matter the cutoff frequencies, and no alpha-scaling would solve this issue. The second method, which is the one reflected in the plots above, did not filter the data before applying the alpha-scaling. From this, I noticed the resulting yaw agreed almost perfectly with the yaw observed by the IMU with an alpha-scaling value of 0.9. This is likely because the IMU has a builtin filter, and attempting to filter on filtered data created incorrect behavior.

4.3 Which estimate or estimates for yaw would you trust for navigation? Why? (Your answer must not be the Yaw computed by the IMU)

Assuming the IMU yaw is the ground truth, the yaw estimate I trust the most is calculated from the GPS. I trust the GPS the most because the IMU is susceptible to biases that accumulate over time, whereas inconsistencies in the GPS are not accumulated. Additionally, the context for which we want the yaw is navigation. When navigating, the vehicle will never move in a direction other than forward relative to itself. This constraint allows the yaw to calculated fairly accurately from the GPS, since we know that the vehicle moved forward between two samples. If there existed a reason for the vehicle to move backward, or if the yaw needed to be determined on a higher precision scale, then the GPS would not be as effective as the IMU.

4.4 What adjustments did you make to the forward velocity estimate, and why?

I only made one adjustment to the forward velocity estimate, where I set all negative values of the velocity that resulted from the integration to 0. Considering the truth that the vehicle both started and ended at rest, there should not be an offset added post integration. Additionally, the forward velocity should be calculated via direct integration of the forward acceleration.

However, I did make multiple adjustments to the forward acceleration, as the data observed by the IMU did not agree with certain truths that should have been exhibited. I described the method and reasoning with which I calibrated the accelerometer data in section 2.2 of this report.

4.5 What discrepancies are present in the velocity estimate between accel and GPS. Why?

After applying the calibration methods previously described in Section 2.2 of this report, The velocity estimate from acceleration exhibited similarly shaped peaks and valleys as the estimate from the GPS, but with one major discrepancy: the magnitude. While the peaks and valleys are shaped nearly identically, the magnitudes of the peaks and valleys are wildly inconsistent. This is likely due to two possibilities. First, the accelerometer may have experienced some electromagnetic interference in the form of a low-frequency sinusoid. This would normally be corrected by applying a high-pass filter to the data, however upon implementation, this produced other errors that I was not able to account for in time. Second, and more likely (based on other observations), the IMU may not have been perfectly stable when mounted, and likely experienced some jostling during data collection that skewed the orientation of the accelerometer vector, leading to a different magnitude of the vector in the x direction.

4.6 Compute $\omega\dot{X}$ and compare it to $\ddot{y}_{obs}$. How well do they agree? If there is a difference, what is it due to?

The two plots agree in the sense that both indicate a turning event at the same time, however disagree on the magnitude of these events. The observed acceleration in the y-direction portrays an apparent bias that accumulates over time, such that there is an acceleration in the y-direction, even when the vehicle is at rest. The reasons for this bias are likely the same as the reasons stated in the previous question: either low-frequency sinusoidal EMF is affecting the accelerometer, or the IMU was not perfectly mounted and some jostling occurred.

4.7 Estimate the trajectory of the vehicle (xe,xn) from inertial data and compare with GPS by plotting them together. (adjust heading so that the first straight line from both are oriented in the same direction). Report any scaling factor used for comparing the tracks

Scale: 1.96
Rotation: 1.326 rad

4.8 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 2m) Did the stated performance for dead reckoning match actual measurements? Why or why not?

Given the specifications of the VN-100, it can be expected that the IMU will maintain an accurate position fix for about 60 seconds. It appears that the GPS and IMU estimates of position only matched within 2 meters for about 14 seconds. The stated performance for dead reckoning was higher than the actual measurements, as this performance assumes no initial velocity errors and dynamically known gyro biases, which could allow for random walk terms to contribute errors about 3 times higher.