RSN Lab-4 Report

Navigation with IMU and Magnetometer

Data collection was done in the NUANCE car of Northeastern. The initial data is stationary collected with car pulled over near Ruggles circle. Then, we went around the circle a few times and took the car for a brief drive to collect both GPS and IMU data.

The units for data collected and represented are as follows:

Acceleration: m/s^2

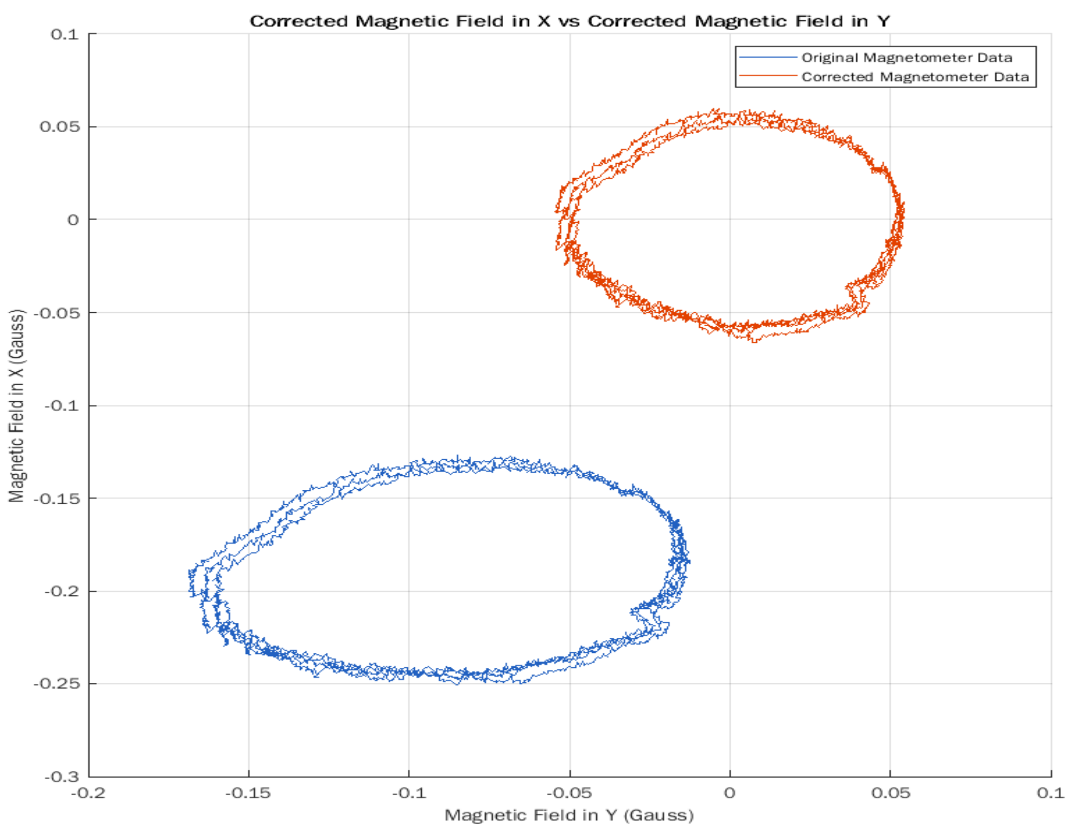
Magnetic Field: Gauss

Gyroscope: rad/s

Time (Seq): seconds

All the graphs use the same convention.

Magnetometer Calibration:



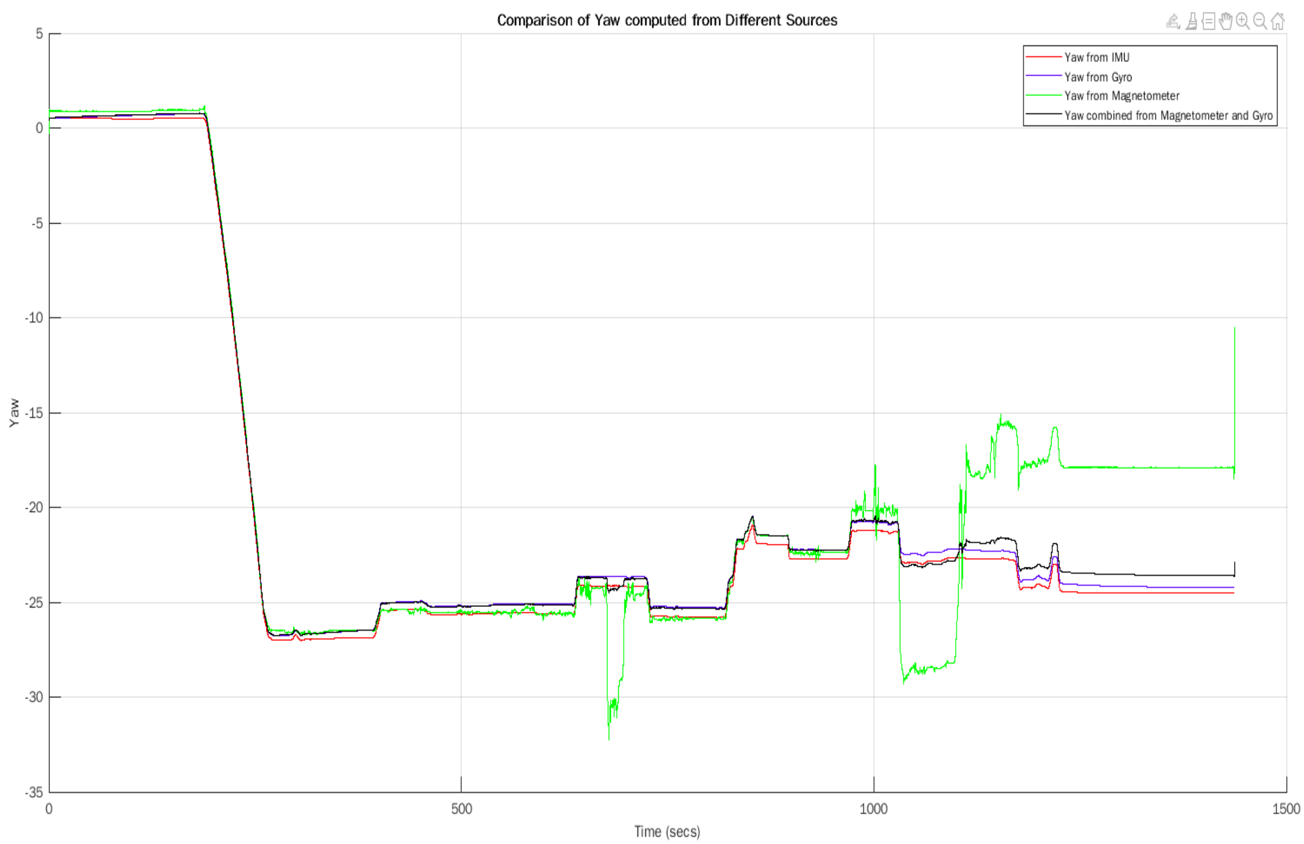
From the above data, it can be observed that the Hard Iron compensation is significantly more than that of the Soft Iron Corrections. The data collected during circling around Ruggles is used to calibrate the Magnetometer data and make these corrections. The Hard Iron offset takes the data to the origin point while the soft iron correction does the work of scaling the ellipse into a circle.

Bank angle and Elevation angle are calculated in order to make tilt corrections before processing it for hard iron or soft iron corrections.

The major axis of the ellipse is scaled by a factor of 0.69 for soft iron compensation.

The corrections calculated in this manner are then applied to the complete dataset.

Computing Yaw and Comparison:



The above graph shows a comparison between four values of Yaw:

1) Yaw data directly from IMU
2) Yaw calculated from Magnetometer
3) Yaw integrated from Gyro
4) Yaw calculated from combining measurements of yaw from magnetometer and yaw from gyro by using a complementary filter

The yaw from magnetometer is calculated after applying the necessary corrections. The yaw from uncorrected magnetometer data is drastically different and inaccurate when compared to the yaw calculated from the corrected magnetometer data.

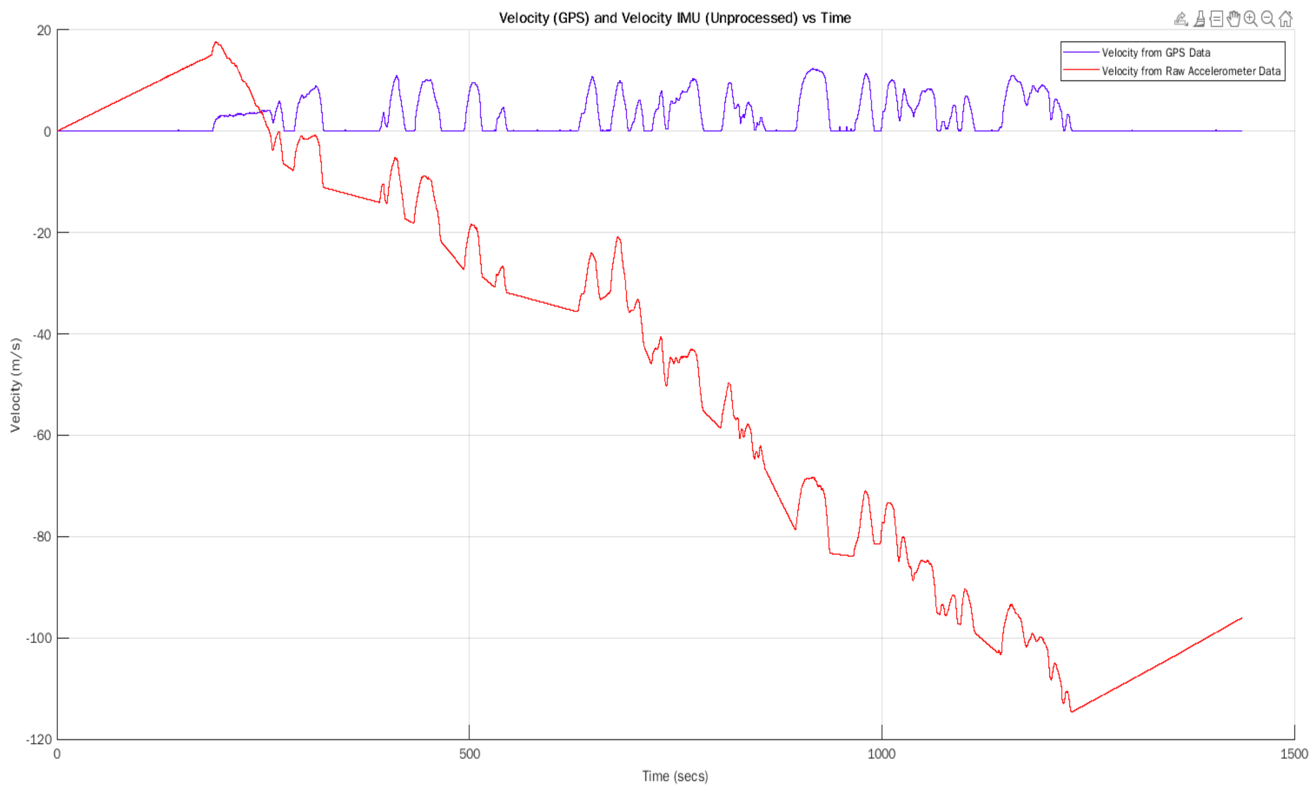
This can serve as a proof that the data is in fact, calibrated. A little deviation is observed which can be because of environmental factors as well as irregular driving or other components.

The yaw integrated from gyro nearly overlaps that of the one received from the IMU data.

The combined yaw from magnetometer and gyro also coincides with the original yaw data.

The noise that can be observed in the yaw combined using a complementary filter can be attributed to the residue noise of the yaw from magnetometer data which can be removed by processing it further.

Estimation of Forward Velocity and Comparison:



The data in red is the velocity calculated by integrating the accelerometer data without any processing while the data in blue is the velocity calculated from the GPS data. We can observe that there is an active negative bias in the unprocessed accelerometer data which continuously impacts the velocity and accumulates so much over time that the velocity reaches reasonably improbable values and that too in negative. This error can be removed in multiple ways.

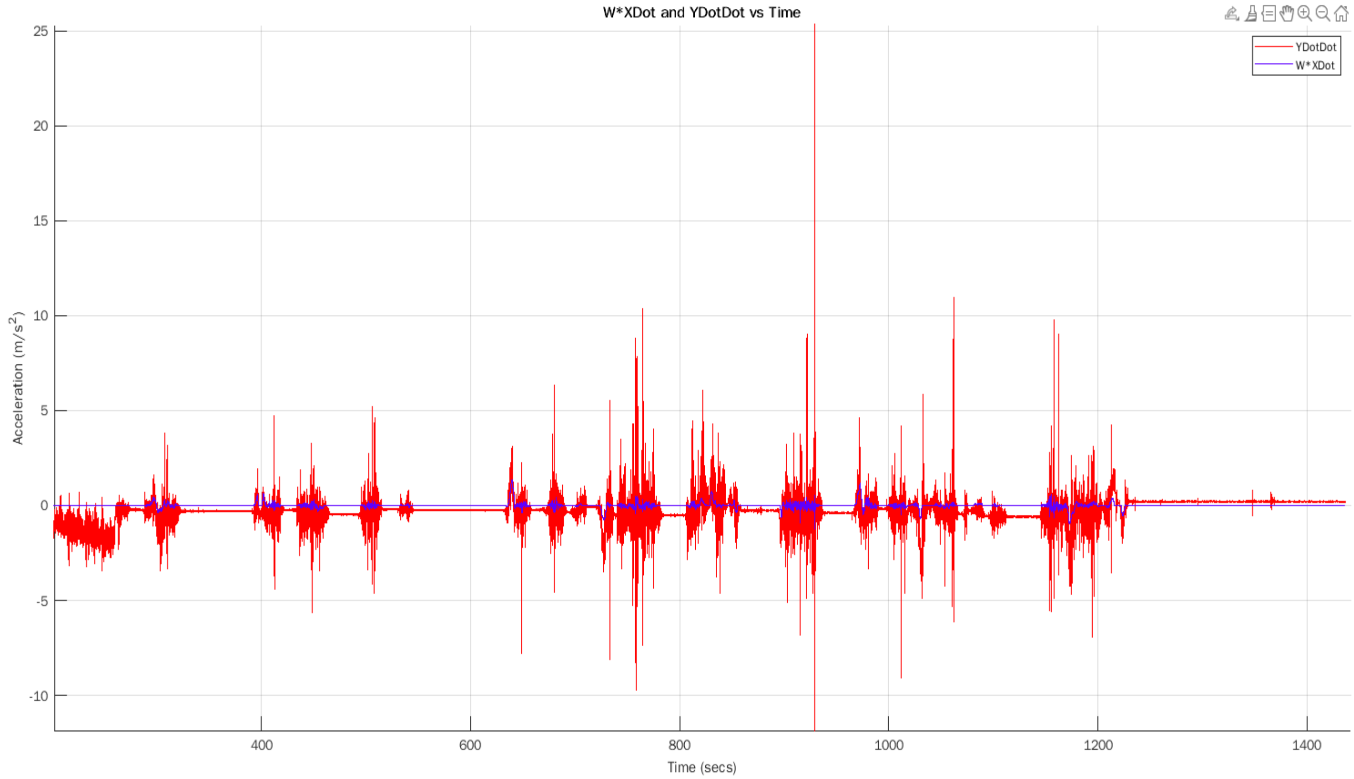
The way I have kept the errors in check is by adjusting the accelerometer data by taking into account the timestamps when the vehicle was at a stop which means that the acceleration and velocity components were zero during that particular time. Also, the car was never reversed so all negative velocities must be zeroed as well. I constantly check for the points and try to remove the bias every time the car is supposed to be stationary. This keeps the accelerometer values in check and gives a comparable trend of velocity which is consistent with the GPS data as well as shown below.

The data in blue is the same velocity data calculated from the GPS values and the data in red is the one that is calculated by integrating the adjusted and filtered acceleration data. The estimated data is not perfect yet and some noise can still be perceived but it can be considered satisfactory for the estimation of forward velocity. The values are consistent with the average velocity of the car and with the GPS data.

The remaining biases can be removed by better filtering the acceleration data and by scaling it.

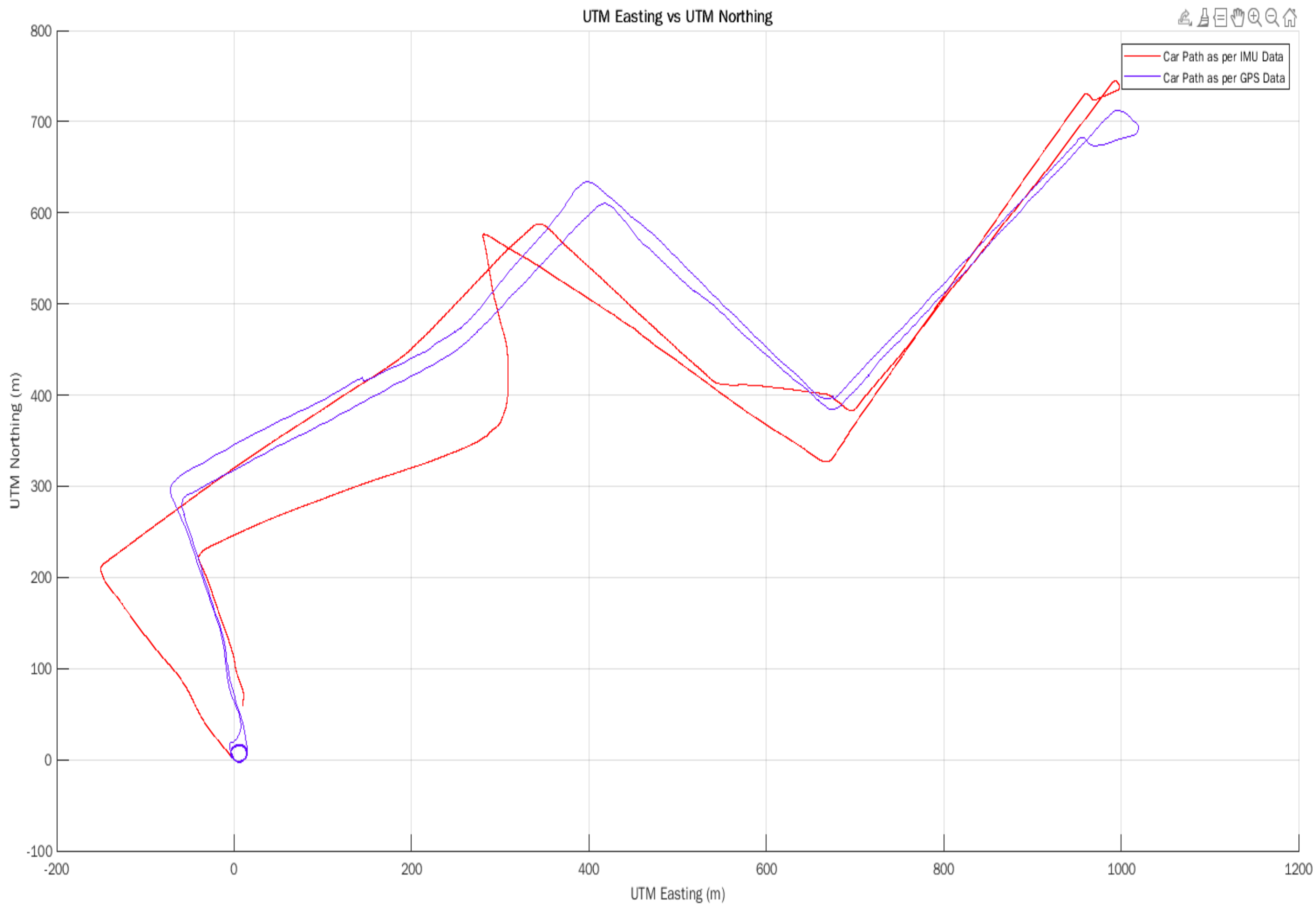
Dead Reckoning:

Trends in W*XDot and YDotDot:



It can be observed from the graph that W*XDot and YDotDot follow a similar pattern but the data in YDotDot is extremely noisy as compared to its counterpart. The major reason for this could be the assumption that Xc is zero. The actual value of Xc is very well not equal to zero and this causes the distortion in the data. While this is one of tha factors for the noise in YDotDot, the other factors that can contribute to this kind of distortion are vibrations felt by the sensors, disturbances caused by the movement of the car or human errors. Inspite of that, we can see that both the values average out near zero which is a validation of the given data.

Comparison of the path of the car using GPS and IMU data:



The above graph represents the path travelled by the car as perceived by the GPS data (in blue) and as sensed by the IMU data (in red). The starting point is the same for both the trajectories but the path traced using the IMU data was offset by an angle of 140 degrees in orientation and was comparatively gigantic when compared to the GPS data.

An angle correction of 140 degrees was applied to orient it in a way similar to that of the GPS data and a scaling factor of 0.043 was used to get this graph.

We can see that both the paths start at the same point but are quick in diverging. The first turn in nearly similar but as the error accumulates, the noise can be clearly observed. Still, it is noteworthy to observe that the path calculated from the IMU data closely follows the path traced by GPS which validates our computations and it was a high point when I observed that after closely following the path of GPS, IMU data traces the path back to its initial position (nearly) 😊.

The data from IMU that was used for creating this path was not the raw data but the processed data calculated for previous tasks after removing the noisy components. The yaw which was used was the final yaw combined from magnetometer and gyro data using a complementary filter and the velocity used was the one estimated in the task of estimation of forward velocity. This is also a way we can validate the previously calculated data.

Estimation of Xc:

```
Xc_estimation = -1.1613
```

We assumed Xc to be zero for the first subtask of our third task but it does not need to be zero necessarily. In fact, we positioned our inertial sensor between the front seats and back seats as that area was flat as compared to the other positions which is at a considerable distance from the dashboard as well.

This estimation of Xc is the mean from a particular timestamp when the car was supposed to be at a stop and it came out to be 1.16 meters (approximately) which is consistent with the placement of the inertial sensor as compared to the frame of the car.

The inertial sensor is displaced from the CM by (-1.16, 0, 0) where the negative sign is the indication of direction.