**Lab 4 Navigation with IMU and Magnetometer**

The document aims to analyse the data collected from an IMU and a GPS sensor using self-written drivers in Python which was ran using ROS with analysis done using MATLAB.

**Magnetometer Calibration:**

Chart, line chart

Description automatically generatedGraphical user interface, chart

Description automatically generated

Figure 1 Figure 2

Figure 1. shows Magnetic field in X vs time and Figure 2. Shows the Magnetic field in Y vs time. These two graphs are from the data collected when driving around the Ruggles in circles. It can be noticed that while travelling in a circle the X and Y components of the magnetometer form a sine wave.

* To rectify the Hard Iron Errors, the x and y co-ordinates of the displaced and skewed ellipse's center can be used to subtract from the x and y values of the magnetic field respectively.
* After correcting the Hard Iron Errors, the ellipse’s center is shifted to the origin.
* To correct the Soft Iron errors of the Magnetometer, the ellipse needs to be rotated about the Z-axis. The rotation matrix required for this task is displayed below.

Rx 0.9845 -0.1756 0 Mx

Ry 0.1756 0.9845 0 My

Rz 0 0 1 Mz

* To transform the Rotated ellipse into a Circle, we need to adjust its size. This can be achieved by multiplying the coordinate of the data with sigma.

Sigma = short axis / long axis

Sigma = 0.8088

* The values mentioned earlier are utilized to fine-tune the data that was collected by the IMU during the drive around Boston for further analysis.

Chart, diagram

Description automatically generated

Figure 3

Figure 3. shows the Magnetometer Data before and after Magnetometer calibration.

Q1. The calibration of the magnetometer involved utilizing a rotation matrix demonstrated earlier. The distortions that affect the magnetometer are categorized into hard and soft iron errors. The presence of permanent magnets in close proximity to the IMU introduces hard iron errors. These errors may be caused by magnets in car instruments or speakers. Soft iron errors, on the other hand, occur temporarily due to magnetic fields induced by metal objects around the IMU.

**Sensor Fusion:**

* To determine the magnetic yaw, we employ both the raw magnetometer data and accelerometer data. Initially, we compute the pitch and roll of the accelerometer sensor through the following equations:

Pitch = atan2(accx, sqrt(accy2 + accz2))

Roll = atan2(accy, sqrt(accx2 +accz2))

* Afterward the magnetometer yaw is calculated using the formula provided,

Yaw = atan2((-ymag\*cos(Roll)+zmag\*sin(Roll)),(xmag\*cos(Pitch) + ymag\*sin(Pitch)\*sin(Roll)+ zmag\*sin(Pitch)\*cos(Roll)))

Chart

Description automatically generated

Figure 4

Figure 4 shows the data comparison between the raw magnetometer yaw vs the corrected yaw.

Chart, line chart

Description automatically generated

Figure 5

Figure 5 shows the comparison between the yaw angle plotted from the Magnetometer vs integrating the Gyro.

Chart, line chart

Description automatically generated

Figure 6

|  |  |  |
| --- | --- | --- |
| **Arguments Passed for the Filters Used** | | |
| **Filter Used** | **Alpha Values** | **Sample Rate in Hz** |
| Low Pass Filter | 3 | 40 Hz |
| High Pass Filter | 0.0003 | 40 Hz |
| Complementary Filter | 0.79 | 40 Hz |

Figure 6 shows the comparison between the complementary filter data vs the yaw data from the IMU

Chart, line chart

Description automatically generated

Figure 7

Figure 7 shows the plots results between Low-Pass Filter, High-Pass Filter and Complementary Filter

Chart, line chart

Description automatically generated

Figure 8

Figure 8 shows the Yaw data comparison between the 4 methods employed in it as per the legends given.

Q2. The complimentary filter combines the estimates from the gyroscope and magnetometer, considering the gyroscope’s high frequency and the magnetometer’s low frequency accuracy. This filter uses the low-pass filter applied on the magnetometer data and the high pass filter applied to the gyro data. The cutoff frequency is mentioned in the table above, it was chosen based on the sensor noise characteristics and the requirements devised from the analysis of the plot of the data made. The analysis from the data says that the magnetometer readings were good for static conditions whereas the gyroscope was good for the tilt in changing conditions.

Q3. The yaw estimate from the complementary filters seems to be the one closest to the one necessary. This might be the best reliable thing used for the navigation analysis as this takes into account, both the high frequency accuracy of the gyroscope and the low frequency accuracy of the magnetometer with the cut off values adjusted according to the requirement.

**Estimate the Forward Velocity:**

**Chart

Description automatically generated** **Chart

Description automatically generated**

Figure 9 Figure 10

Figure 9 and Figure 10 is showing the plot comparison between the GPS Velocity and the IMU velocity vs time after and before the adjustments have been made

Q4. The adjustments were made based on the observed discrepancies between the accelerometer-based velocity and the GPS-based velocity. Begining with the data, I computed the rate of change in acceleration by taking the time derivative of the velocity data from the IMU. Then, I employed loops to classify various segments within the data. Subsequently, I applied specific thresholds to reduce the stationary segments to a value close to zero. Following that, I utilized the most appropriate fit plot for each segment to reduce the portions displaying movement in the velocity plot. Finally, the GPS plot was graphed, as depicted in the Figure 10 above.

Q5. As for the discrepancies present, the IMU data shows some inconsistencies in the calculated velocity, which might be attributed to the accumulation of errors during integration. This could also be due to the noise, drift and biases in the accelerometer data. The accelerometer is sensitive to vibrations, while the GPS signal can be affected by factors like multipath errors, satellite geometry, and atmospheric conditions.

Q6. In an ideal situation the wX’ should closely match with 𝑦̈𝑜𝑏 as the lateral acceleration should be mainly due to the rotation of the vehicle but in this case, they do not closely match as expected due to the inaccuracies in the measurements i.e., the incorrect placement of the IMU sensor or caused due to the drift. The differences might also be due to noises in the sensor noise data. The data plots do not match all the way, but does match for certain parts of the data.

**Dead Reckoning :**

Chart, line chart

Description automatically generated

Figure 11

Figure 11 shows the comparison between the trajectory of the data from IMU & trajectory of the data from GPS

Q7. From the plotting on wX with comparison to Yobs, the Yobs almost overlaps thoughout the data, but the values don’t match, which makes me assume that the mounting of the device was in a wrong direction. This can be corrected in the future by checking the correct way of mounting the IMU.

Q8. To caluculate xc I used the equation given below,

Xc = (Acc in X – drift acc – Acc calibrated accel – (yaw from gyro\* linear acc of IMU in X)) ./ w\_dot + w.\*w

To better caluclulate the value of Xc, I took certain values of the total data, which helped me fit it to the best way possible. From the data, I can assume that the IMU sensor was mounted in the wrong direction , giving us negative results. With the total data analysis till now, I would say that the sensor might have been mounted towards a different direction. Mean of Xc was caluclated to be -3.6498 m

Q9. The ability to navigate without a position fix would depend on the quality of the inertial data and the duration of the loss of GPS signal. In this case, the IMU and GPS estimates of position seem to match closely for a substantial part of the route, but there are deviations in futher sections. This indicates that the actual performance of dead reckoning does not exactly match the stated performance. This might be due the the wrong position mounting of the IMU sensor , the noise or loss of data may also be the reason for this stated performance.