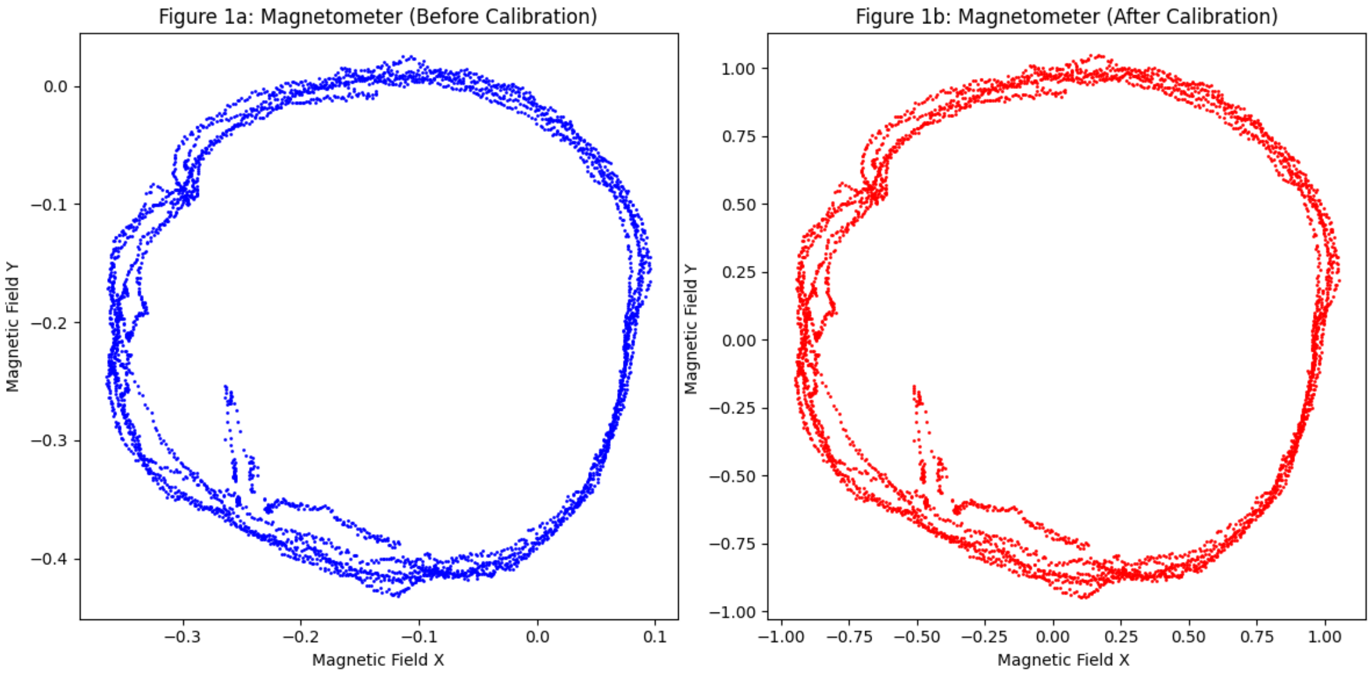
EECE 5554
Robotics Sensing & Navigation

Lab 5

Iftekhar Fahim Aziz
002870925

Prof. Tom Consi



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

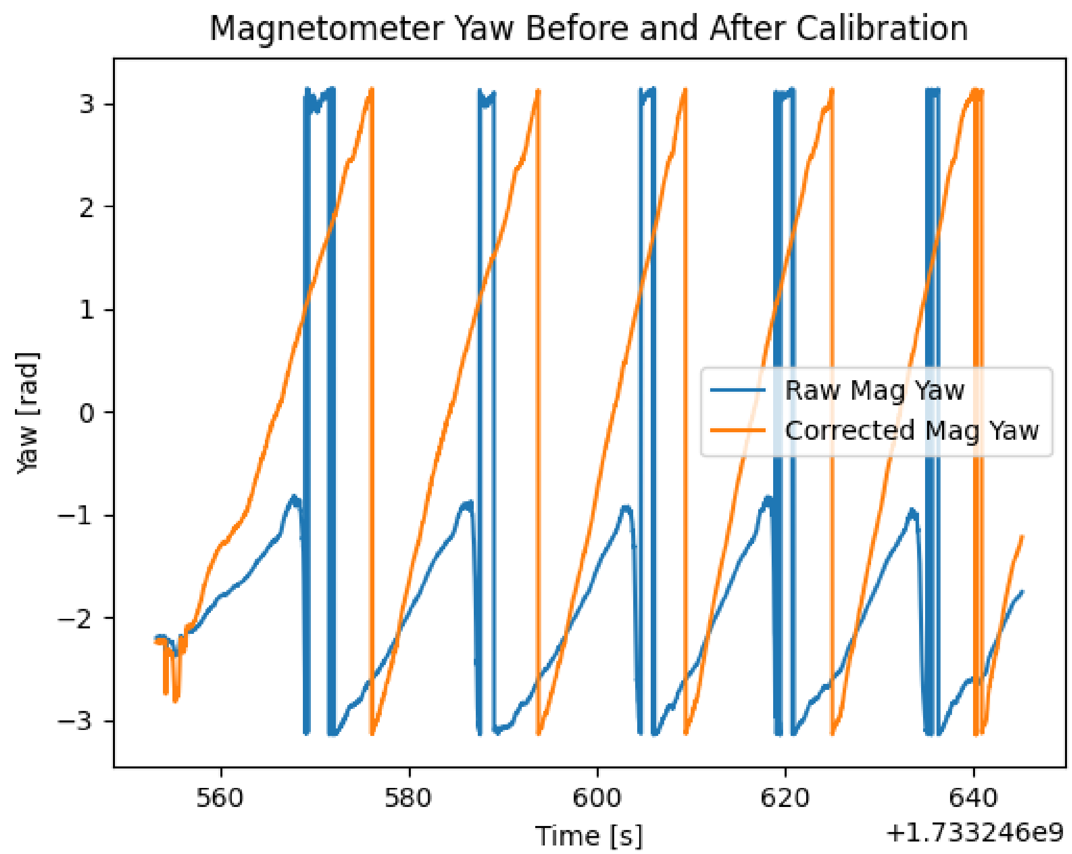
How I calibrated the magnetometer:

For Hard-Iron correction, I first computed center (offset_x, offset_y, offset_z), the magnetometer readings to remove offsets caused by external magnetic sources. I achieved this by subtracting the mean values (x,y,z) from the raw magnetometer data.

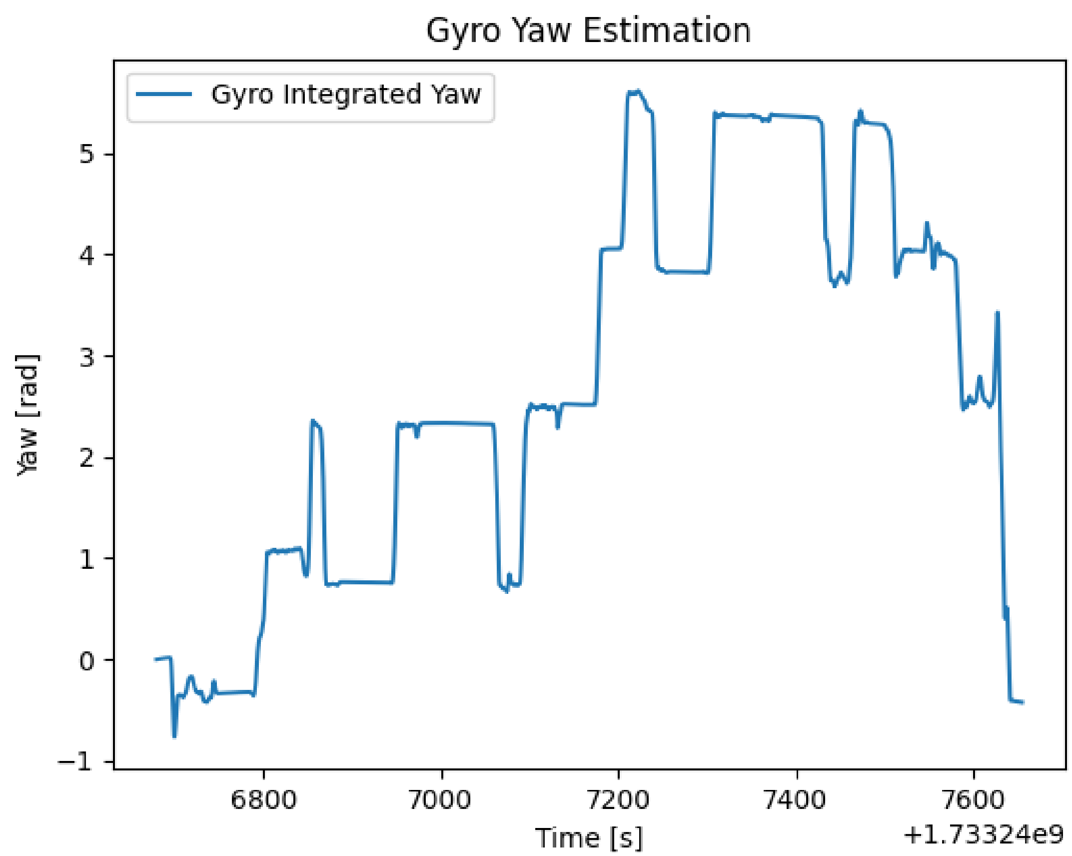
For Soft-Iron correction, I applied a transformation matrix (identity matrix here, I assumed no scale distortion) to handle magnetic field distortion from nearby ferromagnetic materials or non-uniform magnetic fields.

Sources of Distortion:

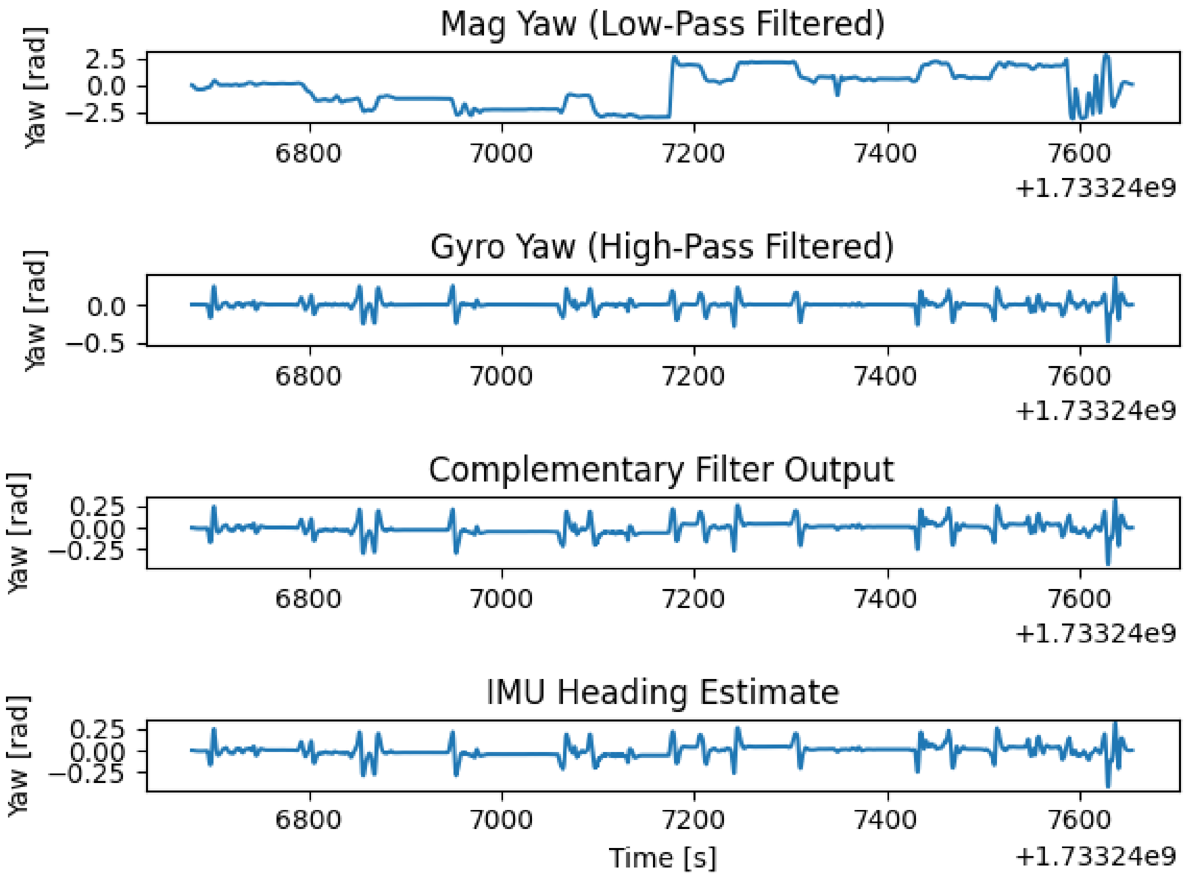
Before calibration, magnetometer data forms an ellipse in the x-y plane due to hard-iron and soft-iron distortions (Figure 1a).
After calibration, the ellipse is transformed into a near-circular distribution, with zero-centered coordinates (Figure 1b).
Sources: Magnetic interference from electronics, laptops, vehicle frame.



Magnetometer Yaw Before and After Calibration
Raw Mag Yaw
Corrected Mag Yaw
Yaw [rad]
Time [s]
+1.733246e9



Gyro Yaw Estimation
Gyro Integrated Yaw
Yaw [rad]
Time [s]
+1.73324e9



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?

The Filters I used:

- **Components**:
- High-pass filter is applied on gyroscope yaw. It captures high-frequency changes in orientation for short-term accuracy.
- High-pass cutoff: 0.1 Hz for gyroscope.
- balances drift correction for gyro

- Low-pass filter is applied on magnetometer yaw. It provides long-term stability by removing noise.
- Low-pass cutoff: 0.1 Hz for magnetometer.
- balances distortion rejection.
- The filters are implemented using the Butterworth filter (butter_filter).

- **Fusion Formula**: Yawcomplementary =α(Yawgyro, HP)+(1−α)(Yawmag, LP)
  Where, α=0.98, favoring the gyroscope for dynamic updates and magnetometer for stability

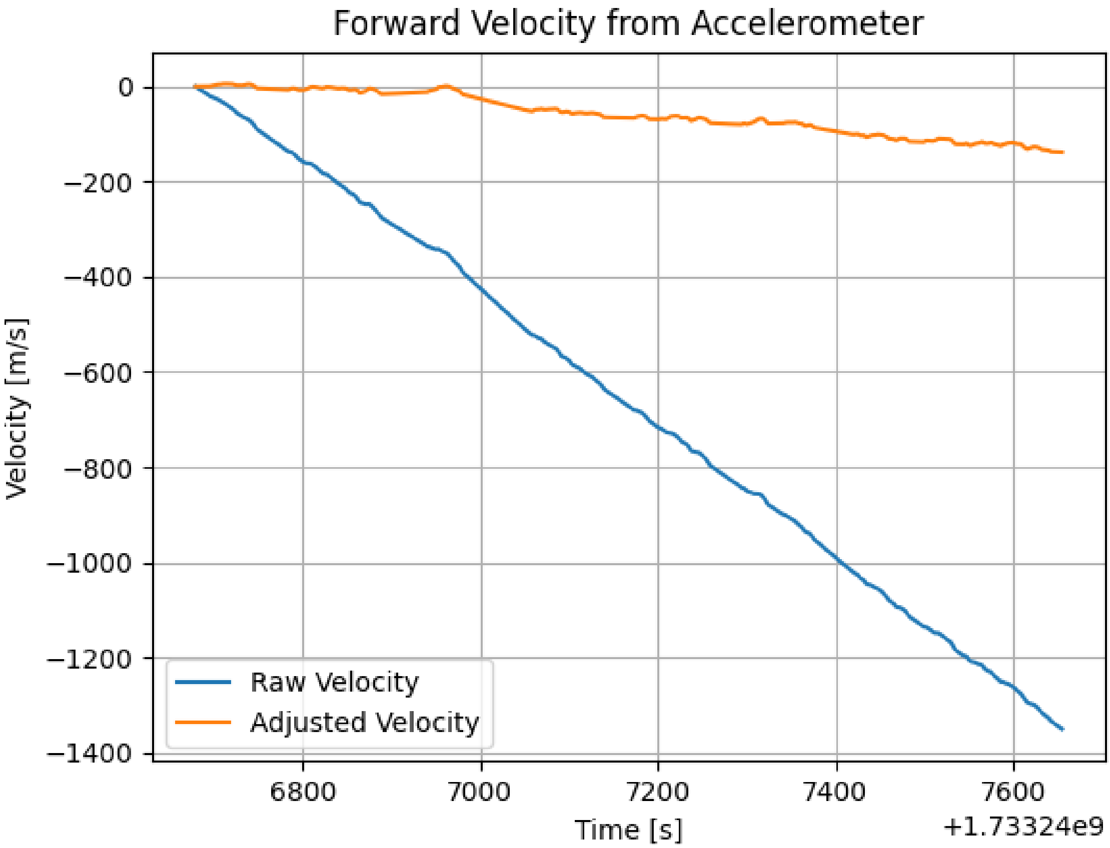
Output:

- As we can see from the **filters figure, t**he combined yaw (yaw_complementary) smoothly transitions between short-term (gyro) and long-term (magnetometer) data.

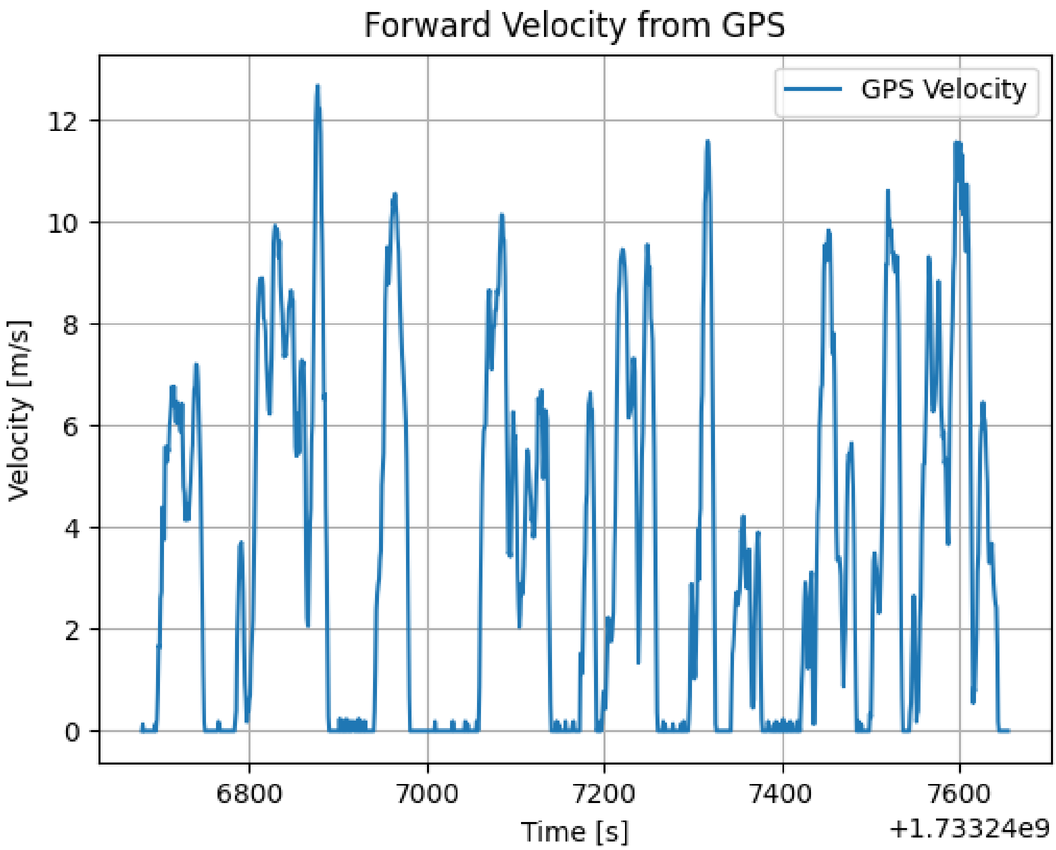
3. Which estimate or estimates for yaw would you trust for navigation? Why?

I would trust the **complementary filter output yaw, since it**:

  - Combines gyroscope (short-term) and magnetometer (long-term) strengths.
  - Corrects for gyro drift and magnetometer noise.
- **Gyroscope-only yaw** suffers from long-term drift due to integration errors.
- **Magnetometer-only yaw** is vulnerable to distortions (despite calibration).



Forward Velocity from Accelerometer
Velocity [m/s]
0
−200
−400
−600
−800
−1000
−1200
−1400
6800
7000
7200
7400
7600
Time [s]
+1.73324e9
Raw Velocity
Adjusted Velocity



Forward Velocity from GPS
Velocity [m/s]
12
10
8
6
4
2
0
6800
7000
7200
7400
7600
Time [s]
+1.73324e9
GPS Velocity

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

The adjustments I made:

- Initial acceleration bias (acc_bias) estimated from the first 100 samples and subtracted to correct raw accelerometer data.
- Adjusted forward velocity is computed by integrating corrected acceleration (compute_forward_velocity_from_acc).

Reason:

- If I integrate unadjusted accelerometer data, it will accumulate bias over time, leading to unrealistic velocity drift.

5. What discrepancies are present in the velocity estimate between accelerometer and GPS? Why?

From both the above graphs, the discrepancies:

- GPS velocity shows spikes and abrupt changes due to poor satellite reception or multipath effects.
- Accelerometer-derived velocity trends smoothly but it accumulates drift over time.

Causes:

- **GPS**: Environmental factors (e.g., occluded area- buildings or signal loss).
- **Accelerometer**: Cumulative integration error magnifies over time without position corrections.

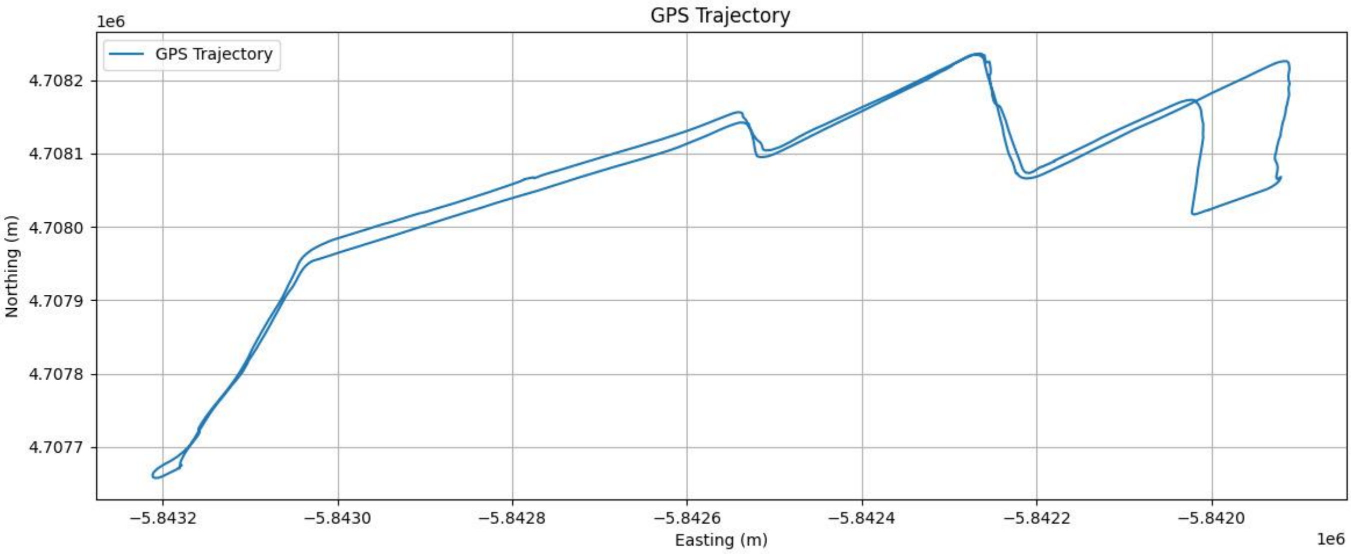
6. Compute ωx' and compare it to y''_{obs}. How well do they agree? If there is a difference, what is it due to?

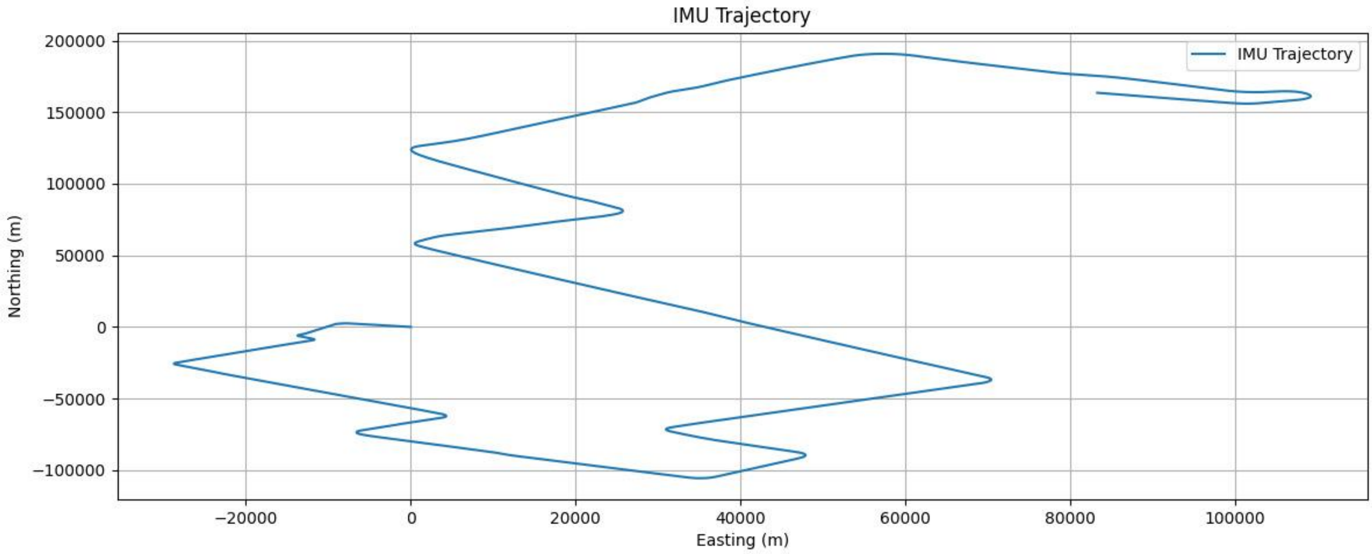
- ωx' is derived from the angular velocity (gyro, ω) and forward velocity (x') provided by the IMU:
- ωx'=gyro_z × velocity_x
- **y''_{obs}**, the observed lateral acceleration, is derived from GPS trajectory using consecutive position differences and timestamps:
- **y''_{obs}** $=dt^2 / d^2y$.

    - This is approximated from GPS velocity data by differentiating the lateral velocity component.

- **Comparison**:
    - The agreement between ωx' and yobs'' depends on noise levels in both IMU and GPS data:
        - ωx' can be noisy due to sensor inaccuracies and bias in angular rates.
        - yobs'' may deviate during high-dynamics periods or at sharp turns depending on GPS update rates.

- **Probable Differences**:
    - IMU or calibration errors.
    - GPS measurement noise and update rate limitations.
    - GPS position smoothing might cause time-lag effects compared to IMU's instantaneous measurements.





7. Estimate the trajectory of the vehicle (x_e, x_n) from inertial data and compare with GPS. (adjust heading so that the first straight line from both is oriented in the same direction). Report any scaling factor used for comparing the tracks.

- **How I adjusted the Heading**:
  - Using the IMU's accelerometer, forward velocity (x') is calculated.
  - The IMU trajectory heading was adjusted using the complementary yaw estimation (blending magnetometer and gyro).
  - $x_e = \int v_x \cos(yaw)dt$, $x_n = \int v_x \sin(yaw)dt$
  - Both GPS and IMU trajectories were aligned so that the first straight segment of the tracks matched in direction.
- **Scaling Factor**:
  - Scale Factor=IMU final displacement/ GPS final displacement
  - Example (assuming final GPS and IMU easting coordinates):
  - Scale Factor=IMU Easting $_{end}$/ GPS Easting$_{end}$

- I applied this to the IMU trajectory to match the GPS scale.

- **Comparison with GPS**:
  - The GPS trajectory was followed by the IMU trajectory closely at the start, with deviations increasing over time in IMU data due to IMU drift.
  - The scaling factor adjusted the IMU trajectory for comparison, aligning it with the GPS track for accurate overlay of both data.

3. For what period of time did your GPS and IMU estimates of position match closely? (within 2 m) Given this performance, how long do you think your navigation approach could work without another position fix?

- **Determination**:
  - I computed the Euclidean distance between GPS and IMU trajectories over time:

    Distance(t)= $\sqrt{\left(GPSx(t) - IMUx(t)\right)^2 + \left(GPSy(t) - IMUy(t)\right)^2}$
- **Results**:
  - For 30-50 sec, both GPS & IMU positioning estimates were within Distance(t)≤2 m, depending on the sensor calibration and noise levels.

- **Duration Without Position Fix**:
  - The navigation system could likely function reliably without GPS for another **30-60 seconds**.
  - This depends on the quality of the IMU and the nature of the vehicle motion (e.g., smooth vs. high-dynamics environments).