IMU to IMU Extrinsic Calibration
Using linear and non-linear regression to solve for rotation and translation between IMUs

Ben Boyack

Theory of Predictive Modeling

CS 580/PHSCS 513R

Why is Calibration Necessary?

Autonomous robotic systems use sensors such as camera, light detection and ranging (LiDAR), inertial measurement units (IMU), GPS, etc. to navigate the world around them. Many autonomous robots (like self-driving cars) require extreme precision in sensor measurements to ensure the safety of people.

To maintain high precision, each sensor has both intrinsic calibration (to account for manufacturing tolerances and other known biases) and extrinsic calibration (rotation and translation between sensors) performed. The extrinsic calibration between two IMU sensors is the focus of this project and is essential for transforming sensor data from local coordinate frames to the robot's coordinate frame.

Current Problems

Current methods for extrinsic IMU calibration, whether IMU to IMU or IMU to LiDAR, have high rotational accuracy but low translational accuracy (with ground truth found using CAD models). Many of these methods use different types of Kalman Filters to estimate the rotation and translation.

To better understand the nuances and issues involved with the lack in accuracy of extrinsic IMU calibrations, this project focused on the **extrinsic calibration** between **two different IMU sensors** using both non-linear regression (for rotation) and linear regression (for translation). The sensors under test are mounted on a golfcart as shown in the image to the right. Sensor 1 is by the driver's seat and sensor 2 is on the top.

IMU Sensor Measurements

Each IMU has a gyroscope (measures angular velocity. Gyro for short) and an accelerometer (measures linear acceleration). We will denote these as g and a respectively as seen below. The extrinsic rotation can be solved using just the gyro data. However, since the IMU sensor only measures relative rotation and acceleration, the translation is much harder to calculate.

$$g_i = \begin{bmatrix} g_{i_x} \\ g_{i_y} \\ g_{i_z} \end{bmatrix}, a_i = \begin{bmatrix} a_{i_x} \\ a_{i_y} \\ a_{i_z} \end{bmatrix}$$

Non-linear Regression for Rotation

When aligning the orientation between two IMUs, we can use the equation $g_2 = Rg_1$.
We can represent the rotation matrix using quaternions for robustness. A quaternion is defined as $q = q_w + q_x + q_y + q_z$. If it is a unit quaternion, we can represent the rotation matrix using quaternions as follows:

$$R(q) = \begin{bmatrix} 1 - 2s(q_y^2 + q_z^2) & 2s(q_x q_y - q_z q_w) & 2s(q_x q_z + q_y q_w) \\ 2s(q_x q_y + q_z q_w) & 1 - 2s(q_x^2 + q_z^2) & 2s(q_y q_z - q_x q_w) \\ 2s(q_x q_z - q_y q_w) & 2s(q_y q_z + q_x q_w) & 1 - 2s(q_x^2 + q_y^2) \end{bmatrix}$$

$R(q)g_1$ can be written as the A matrix for non-linear regression.

$$A(q) = R(q)g_1 = \begin{bmatrix} 1 - 2(q_y^2 + q_z^2)g_{1_x} + 2(q_x q_y - q_z q_w)g_{1_y} + 2(q_x q_z + q_y q_w)g_{1_z} \\ 2(q_x q_y + q_z q_w)g_{1_x} + 1 - 2(q_x^2 + q_z^2)g_{1_y} + 2(q_y q_z - q_x q_w)g_{1_z} \\ 2(q_x q_z - q_y q_w)g_{1_x} + 2(q_y q_z + q_x q_w)g_{1_y} + 1 - 2(q_x^2 + q_y^2)g_{1_z} \end{bmatrix}$$
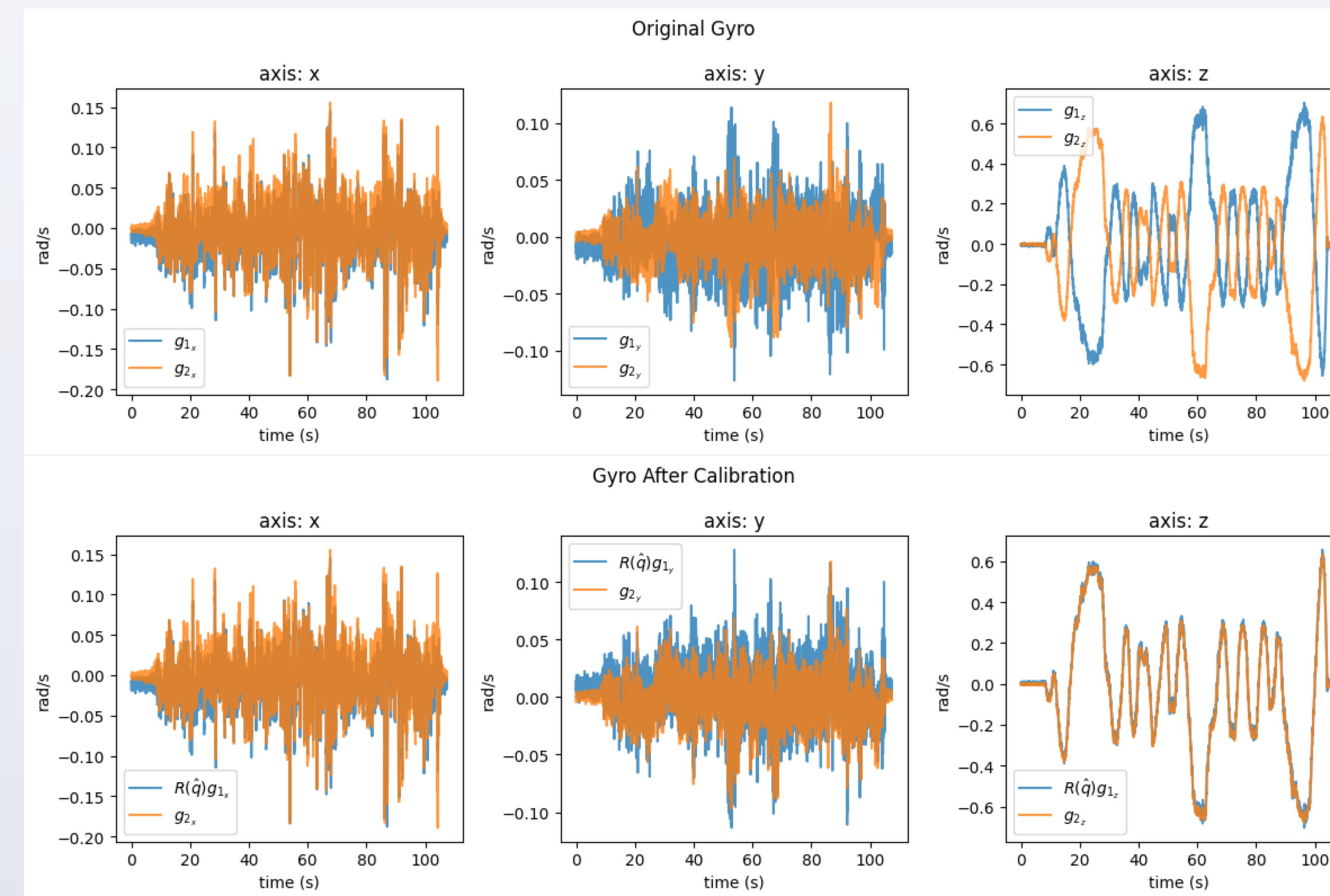
Now we can formulate this as a non-linear least-squares problem by finding the q that minimizes this cost function

$$0 = \underset{q}{argmin} \left\| A(q) - g_2 \right\|_2$$

Rotation Calibration Results

The rotational calibration was expected to be about a 180° rotation about the x-axis. As seen in the plot showing the original gyro data below, the data between IMU 1 and IMU 2 are very similar for the x axis but mirror images of each other in the y and z axis.

Given an initial guess of (0,0,0,1) which corresponds to a quaternion with no rotation, the non-linear regression result was (0.999784, 0.019481, 0.000000, 0.007224). A "1" in the first index corresponds to 180° rotation about the x axis which is as expected. The second row of plots below shows the calibration results. We do not have a CAD model for this robot and thus have no ground truth to compare against.

Linear Regression for Translation

It can be challenging to determine the translation between them IMUs due to their relative measurements. After finding the calibrated rotation matrix and using it to align their corresponding coordinate frames, we can treat the two IMUs as point masses on a rigid object and use the following equations from https://www.mechref.org/dyn/rigid_body_kinematics to solve for the translation.

$$\vec{r}_Q = \vec{r}_P + \vec{r}_{PQ} \tag{1}$$
$$\vec{v}_Q = \vec{v}_P + \vec{\omega} \times \vec{r}_{PQ} \tag{2}$$
$$\vec{a}_Q = \vec{a}_P + \vec{\alpha} \times \vec{r}_{PQ} + \vec{\omega} \times (\vec{\omega} \times \vec{r}_{PQ}) \tag{3}$$

Applying these equations to the two IMUs, let P be the point at IMU 1 and Q be the point at IMU 2. Since we are finding the translation from IMU 1 to IMU 2 and IMU 1 is at the origin, we can rewrite equation (1) above as

$$\vec{r}_2 = 0 + \vec{r}_{1,2} \rightarrow T = \vec{r}_{1,2}$$

where the relative distance between IMU 1 and IMU 2 is the external translation (T). Thus we can rewrite equations (3) as

$$\vec{a}_2 = \vec{a}_1 + \vec{\omega_1} \times T + \vec{\omega_1} \times (\vec{\omega_1} \times T) \tag{4}$$

The cross product can be represented as a matrix (see https://en.wikipedia.org/wiki/Cross_product)

$$\vec{\omega_1} \times T = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_z & 0 \end{bmatrix} T$$

This allows us to rewrite cross products in equation (4) as matrices.

$$\vec{a}_2 = \vec{a}_1 + [\vec{\omega_1}]T + [\vec{\omega_1}]\left([\vec{\omega_1}]T\right) = \vec{a}_1 + \left([\vec{\omega_1}] + [\vec{\omega_1}][\vec{\omega_1}]\right)T$$

$$\vec{a}_2 - \vec{a}_1 = \vec{a}_{diff} = \left([\vec{\omega_1}] + [\vec{\omega_1}]\right)^2 T$$

This follows the traditional linear regression form of $\left\| A\theta - b \right\|_2$ where $A(t) = [\vec{\omega_1}] + [\vec{\omega_1}]^2$, $\theta = T$, and $b = \vec{a}_{diff}$.
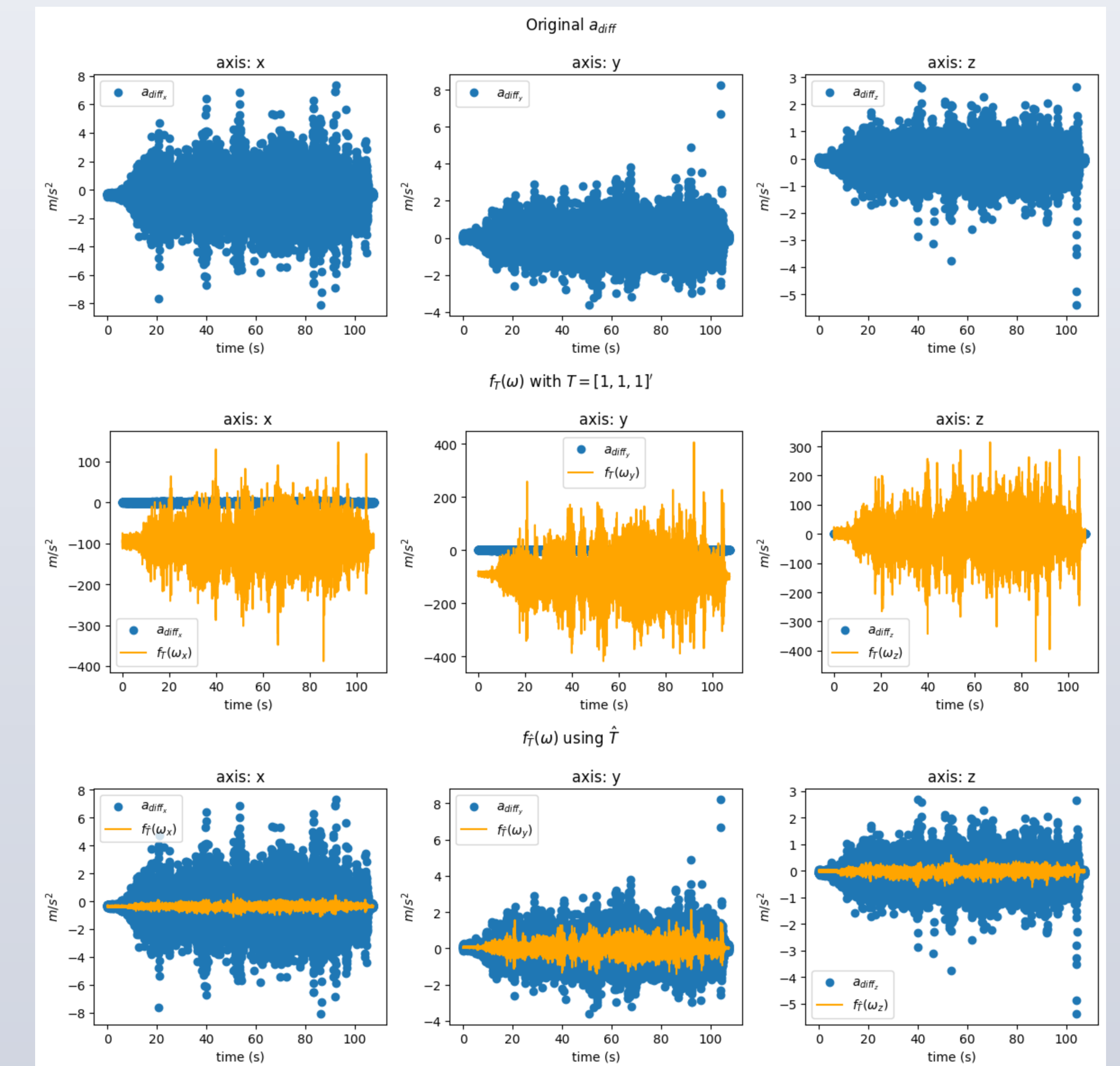
Translation Calibration Results

The expected translation via hand measurement as well as the calibrated translation can be seen in the table below. The calibrated translation is not as expected.

	Hand Measured	Calibrated
x	-0.8128	0.00378549
y	-0.0762	-0.00038839
z	1.5748	0.00480272

By applying the calibrated rotation found earlier, the difference in linear acceleration between the two IMUs has a mean of 0. Apart from noise, the deviations from the mean should be related to rigid body kinematics as described in the previous section. However, for a ground vehicle like the one I used, it needs to move forward to turn which makes it challenging to get the linear acceleration of one IMU to be larger than the other. To add to this problem, the x and y translations are less than a meter thus making the figurative lever arm rather short. The third figure below shows that the model did learn the parameters to fit the data, but the noise may be too large to get accurate results. Calculating the derivative of the angular velocity is very noisy and thus more advanced algorithms or prior filtering could be used to try to increase accuracy.

It appears to be common practice among other IMU calibration methods to move the IMUs in specific patterns to exploit certain measurements. However, this is not possible for IMUs fixed on a large ground vehicles like mine.

Conclusion

IMU to IMU calibration is more challenging than most people think. IMUs only make relative measurements which can easily be used to calculate the extrinsic rotation; however, the lack of absolute measurements significantly complicates the translational calibration, especially for ground vehicles that have limited motion. In practice hand measured translational measurements are often good enough for most scenarios.

Code Repository

https://github.com/bboyack/cs580-imu2imucalib