Dynamic Analysis of 2.5 Axis Camera Gimbal

Intermediate Dynamics Final Project, Fall 2024

Due December 9th, 2024

Lola Anderson (lra57)

I. Description of Chosen System

In this report I will go into detail about the dynamics of a 2.5 axis camera gimbal. For background, when the system is powered on, it will be used for automatic overhead tracking of planes, however when the system is powered off and the motors act as axis constraints, allowing the camera and gimbal arms to rotate freely. The gimbal itself may not be entirely stationary and may be held by a human. In this case, the system should be fairly stable when the base is moved or rotated. This situation yields a number of interesting dynamic phenomena which are analysed and modelled in this report.

Figure 1: CAD Model of 2.5 Axis Camera Gimbal

As shown in figure 1, the camera gimbal has adjustable arms, allowing for different positioning of the center of gravity. The lengths of the arms are adjusted for dynamic simulations in order to target certain effects.

One of the interesting behaviors I will explore is called gimbal lock. This is when two of the Gimbal Axes Align, causing a restriction in the motion that can be achieved by the gimbals movement. Another behavior we will look into is how the gimbal reacts to translational acceleration of the base with several different initial conditions. The final phenomenon that I will explore is how this gimbal will behave when it is held at different angles and then released with gravity being the only force acting on the gimbal.

II. Setup of the Analytical Models

For my analytical model of the system, I have simplified the scenario by considering the masses of the motors and camera as point masses, rather than rigid bodies. I also chose to consider the gimbal arms as having negligible mass and moments of inertia.

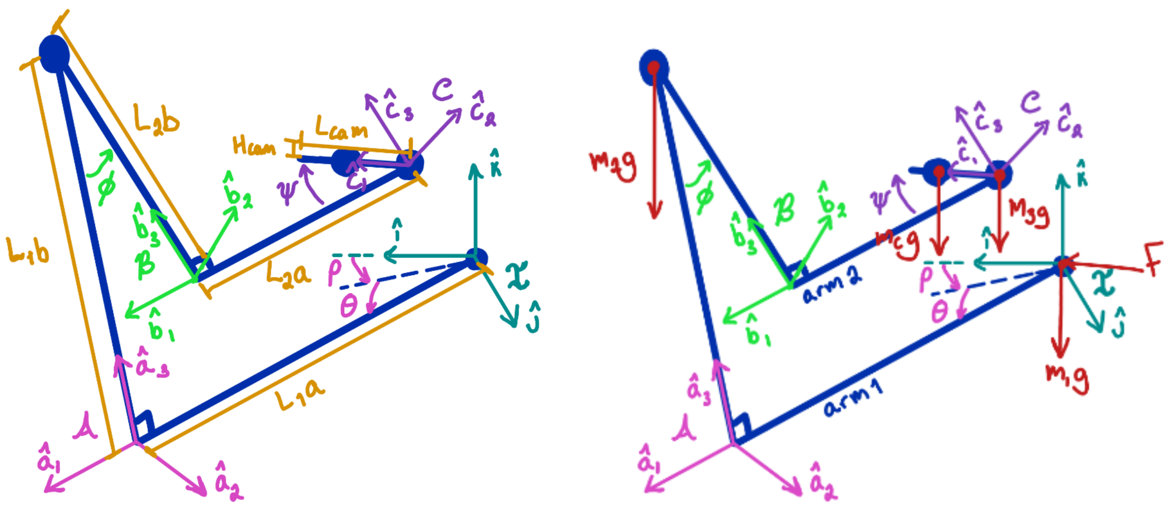
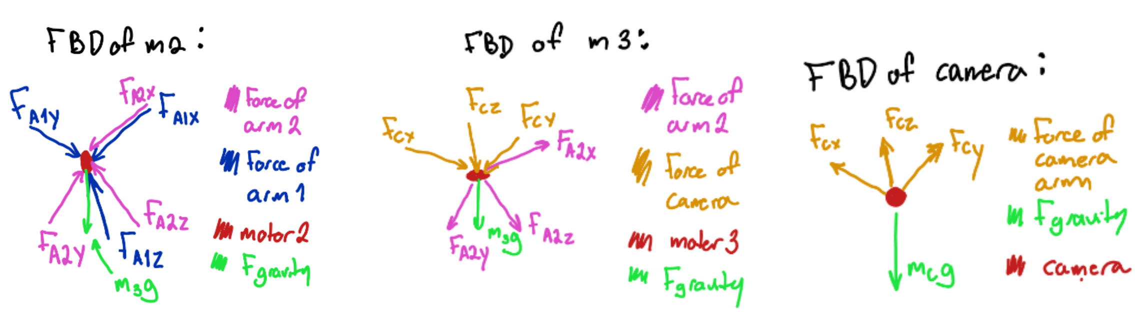


Figure 2 and 3: A simplified gimbal with relevant lengths defined (left) and an FBD of the simplified model gimbal (right)

The diagrams above show the three non inertial reference frames aligned with the gimbal arms and the camera lens, which I will be using during this analysis. It also defines the inertial reference frame, centered at the origin. The model may translate in the î direction, and the angles ρ, ϕ, and Ψ are free to change. The force F in the FBD will sometimes be present and when it is, the model may translate in the î direction. When F is not prescribed, the system will be fixed at the origin.

Individual mass FBDs:



III. Equations of Motion

Definition of Rotation Matrices

First, I define my rotation matrices, this will help me find my velocity and position vectors easier later on. The frames I define DCMs for here are illustrated in figure 2.

The A reference frame is aligned with arm 1 of the gimbal and is constructed by rotating around the y axis in the I frame by an angle θ. This creates an intermediate frame. Then another direction cosine matrix is used to rotate around the z axis of the intermediate frame by an angle ρ. To create this rotation in a single matrix we can multiply the two matrices together.

$$ {}^{I}R^{\theta} = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix}, \quad {}^{\theta}R^{\rho} = \begin{bmatrix} \cos(\rho) & -\sin(\rho) & 0 \\ \sin(\rho) & \cos(\rho) & 0 \\ 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{I}R^{A} = {}^{I}R^{\theta} \cdot {}^{\theta}R^{\rho} $$

The B reference frame is aligned with the arm 2 of the gimbal and is constructed by rotating around the x axis in the A frame by an angle ϕ. We can multiple the rotation matrix between A and I with the DCM for a of ϕ about the x axis, in order to obtain the rotation matrix from B to I.

$$ {}^{\rho}R^{\phi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{bmatrix}, \quad {}^{I}R^{B} = {}^{I}R^{A} \cdot {}^{\rho}R^{\phi} $$

The C reference frame is aligned with the the camera lens and is constructed by rotating around the z axis in the B frame by an angle φ. We can multiply the rotation matrix between B and I with the DCM for a of φ about the x axis, in order to obtain the rotation matrix from the B frame to Inertial.

$$ {}^{\phi}R^{\psi} = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad {}^{I}R^{C} = {}^{I}R^{B} \cdot {}^{\phi}R^{\psi} $$

Euler-Lagrange Approach

Define the **position vectors** of the motors and camera in the inertial frame, then take the z-coordinate in order to find the potential energy equation.

$$\vec{p}_{m1} = \begin{bmatrix} x \\ 0 \\ -1 \end{bmatrix}, \quad z_1 = (\vec{p}_{m1})_z$$

$$\vec{p}_{m2} = \vec{p}_{m1} + {}^I R^A \cdot \begin{bmatrix} L_{1a} \\ 0 \\ L_{1b} \end{bmatrix}, \quad z_2 = (\vec{p}_{m2})_z$$

$$\vec{p}_{m3} = \vec{p}_{j2} + {}^I R^B \cdot \begin{bmatrix} -L_{2b} \\ 0 \\ -L_{2a} \end{bmatrix}, \quad z_3 = (\vec{p}_{m3})_z$$

$$\vec{p}_{\text{cam}} = \vec{p}_{m3} + {}^I R^C \cdot \begin{bmatrix} \frac{L_{\text{cam}}}{3} \\ 0 \\ H_{\text{cam}} \end{bmatrix}, \quad z_c = (\vec{p}_{\text{cam}})_z$$

Find the potential energy equation:

$$E_p = m_1 g z_1 + m_2 g z_2 + m_3 g z_3 + m_c g z_c$$

Find the expressions for **velocity vectors.** As well as the velocity magnitude. Use these equations to find the kinetic energy equation.

$$\vec{v}^I_{m1} = \begin{bmatrix} \dot{x} \\ 0 \\ 0 \end{bmatrix}^I, \quad |\vec{v}_{m1}| = \sqrt{\vec{v}^I_{m1} \cdot \vec{v}^I_{m1}}$$

$$\vec{v}^I_{m2} = \vec{v}^I_{m1} + {}^I R^A \cdot \begin{bmatrix} 0 \\ L_{1a} \cdot \dot{\rho} \\ 0 \end{bmatrix}^A, \quad |\vec{v}_{m2}| = \sqrt{\vec{v}^I_{m2} \cdot \vec{v}^I_{m2}}$$

$$\vec{v}^I_{m3} = \vec{v}^I_{m2} + {}^I R^B \cdot \begin{bmatrix} 0 \\ L_{2b} \cdot \dot{\phi} \\ 0 \end{bmatrix}^B, \quad |\vec{v}_{m3}| = \sqrt{\vec{v}^I_{m3} \cdot \vec{v}^I_{m3}}$$

$$\vec{v}^I_{\text{cam}} = \vec{v}^I_{m3} + {}^I R^C \cdot \begin{bmatrix} 0 \\ \frac{L_{\text{cam}}}{3} \cdot \dot{\psi} \\ 0 \end{bmatrix}^C, \quad |\vec{v}_{mc}| = \sqrt{\vec{v}^I_{mc} \cdot \vec{v}^I_{mc}}$$

Find the expression for the kinetic energy:

$$E_k = \frac{1}{2}m_1|\vec{v}^I_{m1}|^2 + \frac{1}{2}m_2|\vec{v}^I_{m2}|^2 + \frac{1}{2}m_3|\vec{v}^I_{m3}|^2 + \frac{1}{2}m_c|\vec{v}^I_{\text{cam}}|^2$$

Find the expression for the Lagrangian operator:

$$L = E_k - E_p$$

Define the generalized coordinates, velocities, and accelerations as:

$$\vec{q} = \begin{bmatrix} \rho \\ \phi \\ \psi \\ x \end{bmatrix}, \quad \dot{\vec{q}} = \begin{bmatrix} \dot{\rho} \\ \dot{\phi} \\ \dot{\psi} \\ \dot{x} \end{bmatrix}, \quad \ddot{\vec{q}} = \begin{bmatrix} \ddot{\rho} \\ \ddot{\phi} \\ \ddot{\psi} \\ \ddot{x} \end{bmatrix}$$

Jacobian operator:

$$J(f, \vec{q}) = \frac{\partial f}{\partial \vec{q}}$$

The equations of motion are given by:

$$J(J(L, \dot{\vec{q}}), \dot{\vec{q}}) \cdot \ddot{\vec{q}} + J(J(L, \dot{\vec{q}}), \vec{q}) \cdot \dot{\vec{q}} - J(L, \vec{q})^\top = \begin{bmatrix} 0 \\ 0 \\ 0 \\ F \end{bmatrix}$$

Where F is a prescribed external force applied to the base of the gimbal at the location of motor 1 in the î direction.

When plugged into matlab using the jacobian function, these functions will yield the equations of motion for this 2.5 axis gimbal

DAE Method Approach

For the DAE Method, I will be using the same rotation frames and matrixes as in the Lagrangian Method. The vector for my DAE unknowns is shown here and consists of 21 total unknowns. This means I will be deriving 21 equations to define my equations of motion

$$\text{DAE Unknowns} = \left[\ddot{x}_2, \ddot{y}_2, \ddot{z}_2, \ddot{\rho}, \ddot{x}_3, \ddot{y}_3, \ddot{z}_3, \ddot{\phi}, \ddot{x}_c, \ddot{y}_c, \ddot{z}_c, \ddot{\psi}, F_{\mathrm{a1x}}, F_{\mathrm{a1y}}, F_{\mathrm{a1z}}, F_{\mathrm{a2x}}, F_{\mathrm{a2y}}, F_{\mathrm{a2z}}, F_{\mathrm{cx}}, F_{\mathrm{cy}}, F_{\mathrm{cz}}\right]'$$

Before deriving the equations of motion, I will define velocities and accelerations for the three free masses using kinematics. These equations will be used later in the analysis.

Kinematics Equations of m_2:

$$r_{\vec{m_2/m_1}} = {}^{I}R^{A}\left[L_{1a}, 0, L_{1b}\right]^{A\,T}$$

$$\vec{v_{m_2}}^{I} = \dot{\rho} * \mathrm{diff}(r_{\vec{m_2/m_1}}^{I}, \rho) + \dot{\theta} * \mathrm{diff}(r_{\vec{m_2/m_1}}^{I}, \theta)$$

$$\begin{aligned} \vec{a_{m_2}}^{I} &= \ddot{\rho} * \mathrm{diff}(v_{\vec{m_2/m_1}}^{I}, \dot{\rho}) + \dot{\rho} * \mathrm{diff}(v_{\vec{m_2/m_1}}^{I}, \rho) \\ &+ \ddot{\theta} * \mathrm{diff}(v_{\vec{m_2/m_1}}^{I}, \dot{\theta}) + \dot{\theta} * \mathrm{diff}(v_{\vec{m_2/m_1}}^{I}, \theta) \end{aligned}$$

Kinematics Equations of m_3:

$$r_{\vec{m_3/m_1}} = {}^{I}R^{A}\left[L_{1a}, 0, L_{1b}\right]^{A\,T} + {}^{I}R^{B}\left[-L_{2a}, 0, -L_{2b}\right]^{B\,T}$$

$$\vec{v_{m_3}}^{I} = \dot{\phi} * \mathrm{diff}(r_{\vec{m_3/m_1}}^{I}, \phi) + \dot{\rho} * \mathrm{diff}(r_{\vec{m_3/m_1}}^{I}, \rho) + \dot{\theta} * \mathrm{diff}(r_{\vec{m_3/m_1}}^{I}, \theta)$$

$$\begin{aligned} \vec{a_{m_3}}^{I} &= \ddot{\phi} * \mathrm{diff}(v_{\vec{m_3/m_1}}^{I}, \dot{\phi}) + \dot{\phi} * \mathrm{diff}(v_{\vec{m_3/m_1}}^{I}, \phi) \\ &+ \ddot{\rho} * \mathrm{diff}(v_{\vec{m_3/m_1}}^{I}, \dot{\rho}) + \dot{\rho} * \mathrm{diff}(v_{\vec{m_3/m_1}}^{I}, \rho) \\ &+ \ddot{\theta} * \mathrm{diff}(v_{\vec{m_3/m_1}}^{I}, \dot{\theta}) + \dot{\theta} * \mathrm{diff}(v_{\vec{m_3/m_1}}^{I}, \theta) \end{aligned}$$

Kinematics Equations of m_c:

$$r_{\vec{m_c/m_1}} = {}^{I}R^{A}\left[L_{1a}, 0, L_{1b}\right]^{A\,T} + {}^{I}R^{B}\left[-L_{2a}, 0, -L_{2b}\right]^{B\,T} + {}^{I}R^{C}\left[L_{cam}/3, 0, H_{cam}/2\right]^{C\,T}$$

$$\vec{v_{m_c}}^{I} = \dot{\psi} * \mathrm{diff}(r_{\vec{m_c/m1}}^{I}, \psi) + \dot{\phi} * \mathrm{diff}(r_{\vec{m_c/m1}}^{I}, \phi) + \dot{\rho} * \mathrm{diff}(r_{\vec{m_c/m1}}^{I}, \rho) + \dot{\theta} * \mathrm{diff}(r_{\vec{m_c/m1}}^{I}, \theta)$$

$$\begin{aligned} \vec{a_{m_c}}^{I} &= \ddot{\psi} * \mathrm{diff}(v_{\vec{m_c/m_1}}^{I}, \dot{\psi}) + \dot{\psi} * \mathrm{diff}(v_{\vec{m_c/m_1}}^{I}, \psi) \\ &+ \ddot{\phi} * \mathrm{diff}(v_{\vec{m_c/m_1}}^{I}, \dot{\phi}) + \dot{\phi} * \mathrm{diff}(v_{\vec{m_c/m_1}}^{I}, \phi) \\ &+ \ddot{\rho} * \mathrm{diff}(v_{\vec{m_c/m_1}}^{I}, \dot{\rho}) + \dot{\rho} * \mathrm{diff}(v_{\vec{m_c/m_1}}^{I}, \rho) \\ &+ \ddot{\theta} * \mathrm{diff}(v_{\vec{m_c/m_1}}^{I}, \dot{\theta}) + \dot{\theta} * \mathrm{diff}(v_{\vec{m_c/m_1}}^{I}, \theta) \end{aligned}$$

There are three free masses that these equations describe, with six degrees of freedom each. This gives us 3 linear momentum balances per mass, totalling to 9 equations. We also will have three angular momentum balances per mass which totals to another 9 equations. Then I will use the idea that the inward accelerations of each mass is equal to the centripetal force which I will derive using accelerations.

Linear Momentum Balances

Linear Momentum Balance on m_2 (3 Equations):

$$\sum \vec{F_{m_2}}^{I} = \begin{bmatrix} 0 \\ 0 \\ -m_2 g \end{bmatrix}^I + {}^I R^A \begin{bmatrix} F_{A1x} \\ F_{A1y} \\ F_{A1z} \end{bmatrix}^A + {}^I R^B \begin{bmatrix} F_{A2x} \\ F_{A2y} \\ F_{A2z} \end{bmatrix}^B$$

$$\sum \vec{F} = m_2 \vec{a_2}$$

$$\begin{bmatrix} 0 \\ 0 \\ -m_2 g \end{bmatrix}^I + {}^I R^A \begin{bmatrix} F_{A1x} \\ F_{A1y} \\ F_{A1z} \end{bmatrix}^A + {}^I R^B \begin{bmatrix} F_{A2x} \\ F_{A2y} \\ F_{A2z} \end{bmatrix}^B = m_2 \begin{bmatrix} \ddot{x}_2 \\ \ddot{y}_2 \\ \ddot{z}_2 \end{bmatrix}^I \quad \text{(Eqs 1-3)}$$

Linear Momentum Balance on m_3 (3 Equations):

$$\sum \vec{F_{m_3}}^{I} = \begin{bmatrix} 0 \\ 0 \\ -m_3 g \end{bmatrix}^I + {}^I R^B \begin{bmatrix} F_{A2x} \\ F_{A2y} \\ F_{A2z} \end{bmatrix}^A + {}^I R^C \begin{bmatrix} F_{Cx} \\ F_{Cy} \\ F_{Cz} \end{bmatrix}^C$$

$$\sum \vec{F} = m_3 \vec{a_3}$$

$$\begin{bmatrix} 0 \\ 0 \\ -m_3 g \end{bmatrix}^I + {}^I R^B \begin{bmatrix} F_{A2x} \\ F_{A2y} \\ F_{A2z} \end{bmatrix}^B + {}^I R^C \begin{bmatrix} F_{Cx} \\ F_{Cy} \\ F_{Cz} \end{bmatrix}^C = m_e \begin{bmatrix} \ddot{x}_3 \\ \ddot{y}_3 \\ \ddot{z}_3 \end{bmatrix}^I \quad \text{(Eqs 4-6)}$$

Linear Momentum Balance on m_c (3 Equations):

$$\sum \vec{F_{m_c}}^{I} = \begin{bmatrix} 0 \\ 0 \\ -m_c g \end{bmatrix}^I + {}^I R^B \begin{bmatrix} F_{A2x} \\ F_{A2y} \\ F_{A2z} \end{bmatrix}^A + {}^I R^C \begin{bmatrix} F_{Cx} \\ F_{Cy} \\ F_{Cz} \end{bmatrix}^C$$

$$\sum \vec{F} = m_c \vec{a_c}$$

$$\begin{bmatrix} 0 \\ 0 \\ -m_c g \end{bmatrix}^I + {}^I R^C \begin{bmatrix} F_{Cx} \\ F_{Cy} \\ F_{Cz} \end{bmatrix}^C = m_c \begin{bmatrix} \ddot{x}_c \\ \ddot{y}_c \\ \ddot{z}_c \end{bmatrix}^I \quad \text{(Eqs 7-9)}$$

Angular Momentum Balances

Angular Momentum Balance on m_2 about m_1 (3 Equations):

$$\sum \vec{M}_2 = \frac{d\vec{h_{m_2/m_1}}}{dt}$$

$$\vec{r_{m_2/m_1}}^I \times \sum \vec{F_{m_2}}^I = \frac{d}{dt}(\vec{r_{m_2/m_1}}^I \times m_2 \vec{v_{m_2/m_1}}^I) \qquad \text{(Eqs 10-12)}$$

Angular Momentum Balance on m_3 about m_1 (3 Equations):

$$\sum \vec{M}_3 = \frac{d\vec{h_{m_3/m_1}}}{dt}$$

$$\vec{r_{m_3/m_1}}^I \times \sum \vec{F_{m_3}}^I = \frac{d}{dt}(\vec{r_{m_3/m_1}}^I \times m_3 \vec{v_{m_3/m_1}}^I) \qquad \text{(Eqs 13-15)}$$

Angular Momentum Balance on m_c about m_1 (3 Equations):

$$\sum \vec{M}_c = \frac{d\vec{h_{m_c/m_1}}}{dt}$$

$$\vec{r_{m_c/m_1}}^I \times \sum \vec{F_{m_c}}^I = \frac{d}{dt}(\vec{r_{m_c/m_1}}^I \times m_c \vec{v_{m_c/m_1}}^I) \qquad \text{(Eqs 16-18)}$$

Constraint Equations

Centripetal Force on m_2:

$$F_{cent_2} = (m_2 * {}^I R^{AT} * \vec{a_{m_2/m_1}}^I) \cdot [1,0,0]'$$

$$F_{cent_2} = ({}^I R^{AT} * \sum \vec{F_{m_2}}^I) \cdot \left[1,0,0\right]'$$

$$({}^I R^{AT} * \sum \vec{F_{m_2}}^I) \cdot \begin{bmatrix}1\\0\\0\end{bmatrix} = (m_2 * {}^I R^{AT} * \vec{a_{m_2/m_1}}^I) \cdot \begin{bmatrix}1\\0\\0\end{bmatrix} \qquad \text{(Eq 19)}$$

Centripetal Force on m_3:

$$F_{cent_3} = (m_3 * {}^I R^{BT} * \vec{a_{m_3/m_1}}^I) \cdot [0,0,1]'$$

$$F_{cent_3} = ({}^I R^{BT} * \sum \vec{F_{m_3}}^I) \cdot \left[0,0,1\right]'$$

$$({}^I R^{BT} * \sum \vec{F_{m_3}}^I) \cdot \begin{bmatrix}0\\0\\1\end{bmatrix} = (m_3 * {}^I R^{BT} * \vec{a_{m_3/m_1}}^I) \cdot \begin{bmatrix}0\\0\\1\end{bmatrix} \qquad \text{(Eq 20)}$$

Centripetal Force on m_c:

$$F_{cent_c} = (m_c * {}^I R^{CT} * \vec{a_{m_c/m_1}}^I) \cdot [1,0,0]'$$

$$F_{cent_c} = ({}^I R^{CT} * \sum \vec{F_{m_C}}^I) \cdot \left[1,0,0\right]'$$

$$({}^I R^{CT} * \sum \vec{F_{m_C}}^I) \cdot \begin{bmatrix}1\\0\\0\end{bmatrix} = (m_c * {}^I R^{CT} * \vec{a_{m_c/m_1}}^I) \cdot \begin{bmatrix}1\\0\\0\end{bmatrix} \quad \text{(Eq 21)}$$

With all of these equations defined, I can construct my A matrix and my F matrix, then use matlab to invert my A matrix and solve these equations of motion using ode45.

IV. Animation Still Frames

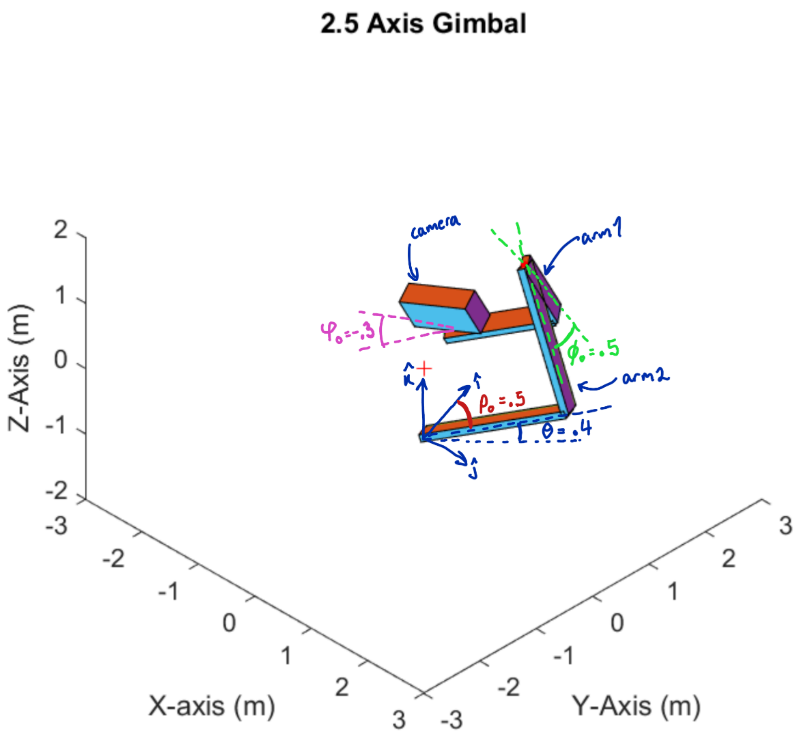


Figure 4.1: Initial Setup

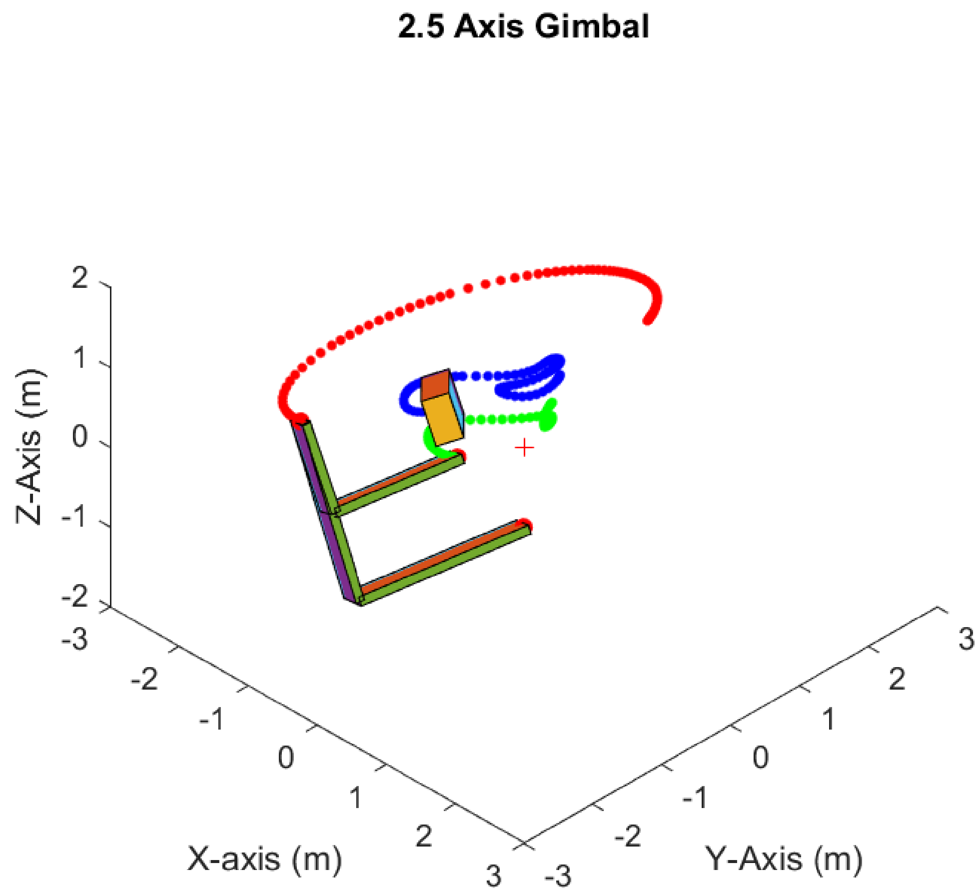
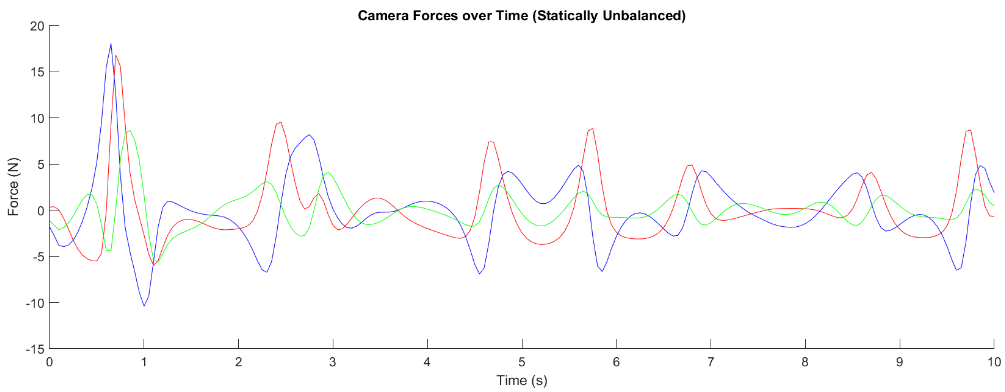


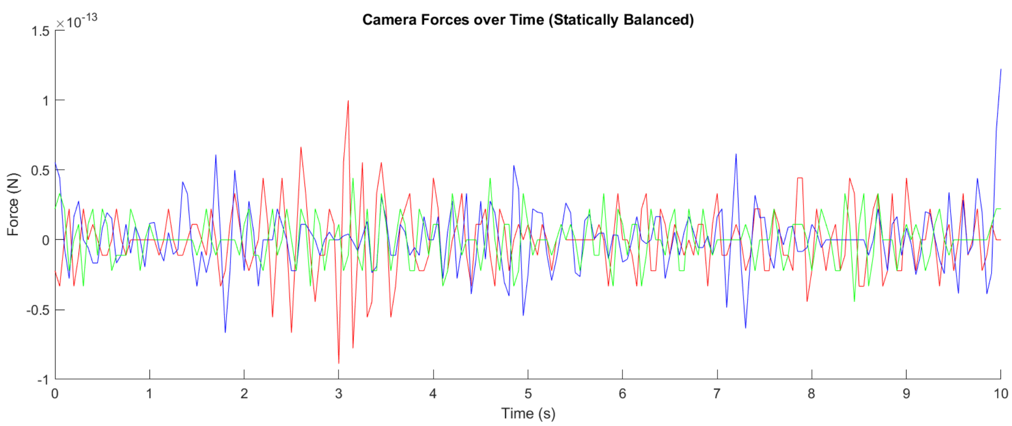
Figure 4.2: Animation at t=3.2s

V. Discussion

For my Animations. I have two separate Matlab Scripts. The first is the DAE Method. In this script I attempted to allow for prescribed changes in θ. This script takes over an hour to run, but appears to give a correct result when θ is held constant. The second script consists of the Lagrange Euler method. This approach runs significantly faster, most likely because the matrix being inverted is significantly smaller and is only inverted once.

Because I was not able to get the transformation and changing theta parts of either of my matlab scripts working, the phenomenon I explored was how the forces on the camera change when the cg of the camera is located at the intersection of all axes versus when it is not. Typically, a camera gimbal will place the cg of the camera at the point where all of the constrained axes intersect. This is considered a balanced gimbal. These are two graphs of force on the camera over time with the same initial conditions for a balanced versus an unbalanced system:





It is clear from the graphs that when the gimbal is balanced, the camera experiences far lower forces, there are higher vibrations however according to the output of my lagrange code. These vibrations however are most likely due to the decision to approximate the motors and camera as point masses.

Key Takeaways

For a more accurate code in the future, I would estimate the moments and products of inertia in order to obtain a more accurate simulation of the movement this system would experience. Overall however, when the system is unbalanced, the gimbal performs very close to how it would in real life when using point masses to estimate. From this project I learned that in engineering, it is best to break down problems into manageable sizes and then solve them piece by piece, checking your accuracy along the way. This would have made my debugging process easier, because with a complex system such as this, it can be hard to isolate when there is a problem in the equations of motion and what the cause is. Checking as I went would have made this easier for me. I also learned that making approximations and simplifications can lead to misleading results which don't always correlate with reality, such as the high frequency force oscillations I encountered when the gimbal was balanced. In order to determine if these oscillations in the force are real, I would first conduct another simulation that includes the moments of inertia of the camera, then I would set up a real life test and gather data to see if the oscillations still appear.