Midterm Exam

Problem 1 (100 Points)

Design and simulate a controller for the International Space Station (ISS) attitude maneuver described below. Assume the ISS inertial matrix $\boldsymbol{J}^b_{cm}$ to be

$$\boldsymbol{J}^b_{cm} = \begin{bmatrix} 24181836 & 3783405 & 3898808 \\ 3783405 & 37621803 & -1171849 \\ 3898808 & -1171849 & 51576634 \end{bmatrix} \text{ kg m}^2$$

For simplicity, you can assume the ISS is in perfectly circular equatorial orbit with 400km radius. You can also use an inclined orbit if you prefer. You can choose where the ISS is in its orbit at the initial time. The LVLH frame ℓ is defined with the x-axis pointing in the direction of motion, the body z-axis pointing to nadir, and the y-axis completing a right-hand triad.

The initial LVLH-to-body nominal quaternion is

$$\boldsymbol{q}^b_\ell(t_0) = \begin{bmatrix} 0.0280 \\ -0.0788 \\ 0.1141 \\ 0.9899 \end{bmatrix}$$

The maneuver lasts 7110 seconds, the final LVLH-to-body nominal quaternion is

$$\boldsymbol{q}^b_\ell(t_f) = \begin{bmatrix} -0.0607 \\ -0.0343 \\ -0.7045 \\ 0.7062 \end{bmatrix}$$

Part 1 (25 points)

Numerically calculate and plot the ISS nominal LVLH-to-body quaternion and angular velocity as a function of time. In defining this nominal trajectory, you can assume the initial and final angular velocities are zero.

Add the time varying inertial-to-LVLH attitude to the trajectory above and plot the ISS nominal inertial-to-body quaternion and angular velocity as a function of time.

Part 2 (35 points)

Design a PD controller to track this trajectory. Include gravity gradient perturbations in the true dynamics. Additionally, assume the true initial ISS angular velocity coincides with orbit rate. Feed the control torque directly to the dynamics and plot the following:

1. The attitude control error vs. time
2. The attitude rate error vs. time
3. The true inertial-to-body attitude vs. time
4. The true LVLH-to-body attitude vs. time

Part 3 (35 points)

Add a steering law to your controller and simulate four-double-gimbal CMGs mounted in parallel. Each CMG has angular momentum magnitude of $h_0 = 4881$ kg m^2/s and two gimbal angles $\alpha^{(i)}$ and $\beta^{(i)}$. The total angular momentum of the wheels coordinatized in body coordinates is

$$\boldsymbol{h}^b_{cmg} = h_0 \sum_{i=1}^{4} \begin{bmatrix} \sin\alpha^{(i)} \\ \cos\alpha^{(i)}\cos\beta^{(i)} \\ \cos\alpha^{(i)}\sin\beta^{(i)} \end{bmatrix}$$

The initial gimbal angles are

$$\alpha^{(1)} = \alpha^{(2)} = \alpha^{(3)} = \alpha^{(4)} = 0$$
$$\beta^{(1)} = \pi/4 \qquad \beta^{(2)} = -\pi/4$$
$$\beta^{(3)} = 3\pi/4 \qquad \beta^{(4)} = -3\pi/4$$

Produce the following plots:

1. The attitude control error vs. time
2. The attitude rate error vs. time
3. The gimbal rates of the CMGs vs. time
4. The magnitude of the sum of the angular momentum of the wheels vs. time
5. The true LVLH-to-body attitude vs. time

Do not worry if you can't complete the maneuver because the CMG saturates or you hit a singularity. In fact, for a long time, many NASA engineers believed it was impossible to perform such a large ISS maneuvers without the help of RCS.

Instructions

The top level of your code should contain these three separate components. The components can be function calls for a scripted language or subsystem blocks for a graphic language.

1. Flight software (FSW) component containing the nominal attitude calculation, the controller and the steering law. FSW receives as inputs the true attitude and rate of the ISS as well as the gimbal angles of the CMGs. FSW produces as outputs the gimbal rate commands to the CMGs (or the control torque when we are not using the CMGs).
2. Actuator component containing the CMGs. CMGs receive as inputs the gimbal rate commands and the ISS rate. CMGs outputs the torque produced on ISS as well as the gimbal angles.
3. Kinematics/Dynamics component including gravity gradient (K& D). K& D receives as inputs the CMG torque. It produces as outputs the ISS attitude and angular velocity.

You can include additional code to the top level to read/load/generate parameters/data or to generate plots. Additionally, you can make the true position/velocity/inertia of the ISS available (as an input or a global variable) to anyone that needs it. Similarly, you can feed the simulation time and the true inertial-to-LVLH attitude to anyone that needs it.

Submit all of your work in a single PDF file. Include all of your code in the PDF file.