Fall 2023 Magneto-Attitude Propulsion Progress Report

The main accomplishment of the fall semester is completion of a six degree of freedom spacecraft model in Simulink. A key breakthrough is identification and correction of all the bugs in the code. Using the "Orbit Propagator" and "Attitude Dynamics" blocks instead of the "Spacecraft Dynamics" block from the Aerospace Blockset fixes one particularly perplexing bug where control torques cause angular acceleration of the spacecraft in the wrong direction. The model now efficiently produces expected results and is ready to aid the next phase of the project—formulating comprehensive design requirements for the Magneto-Attitude Propulsion spacecraft.

The Simulink model uses three coordinate systems: ECI, spacecraft body-fixed, and spacecraft center of mass-fixed coordinates. N refers to the ECI coordinate system, which is formed by basis vectors $\vec{n}_1, \vec{n}_2$, and $\vec{n}_3$. B refers to the spacecraft body-fixed coordinate system, which is formed by basis vectors $b_1, \vec{b}_2$, and $\vec{b}_3$. O refers to the spacecraft center of mass-fixed coordinate system, which is formed by basis vectors $\vec{o}_1, \vec{o}_2$, and $\vec{o}_3$. Defining $\vec{r}$ as the spacecraft's position relative to Earth's center, the O coordinate system is defined as follows: $\vec{o}_3 = -\frac{\vec{r}}{\|\vec{r}\|}$, $\vec{o}_2 = -\frac{\vec{r}\times\vec{v}}{\|\vec{r}\times\vec{v}\|}$, $\vec{o}_1 = \vec{o}_2 \times \vec{o}_3$.

The Simulink model uses the "Orbit Propagator" and "Attitude Dynamics" blocks from Simulink's Aerospace Blockset to propagate the spacecraft's orbit and attitude. It's important to note two things about these blocks: quaternions are scalar first, and the ICRF coordinate system is the same as our N coordinate system. Scalar last quaternions are used throughout the rest of the model, so it's important to be aware of any scalar first quaternions. There are two of these in the "Attitude Dynamics" block, one being the initial quaternion and the other being the output quaternion.

The spacecraft's position, velocity, attitude quaternion, and angular velocity are measured by the "Orbit Propagator" and "Attitude Dynamics" blocks and fed into the "Determine Control Inputs" block. This block uses the state information to produce information desired by the controller. Specifically, this block converts measured BqN to BqO and measured $B\omega^{B/N}$ to $B\omega^{B/O}$, where BqO is the quaternion rotation from O to B and $B\omega^{B/O}$ is the angular velocity of B with respect to O in B coordinates. The model uses a PD control law specified by equation 7.12 in *Fundamentals of Spacecraft Attitude Determination and Control* (Markley and Crassidis). That control law is $L = -k_p sign(\delta q_4)\delta q_{1:3} - k_d\omega$. The model doesn't directly calculate the error quaternion δq because it's set up to drive the B coordinates to the O coordinates, which it achieves by driving BqO to the identity quaternion. The implemented version of the control law above is $L = -k_p sign(q_4) q_{1:3} - k_d\omega$. k_p and k_d are mostly arbitrary, chosen to create a nice response for testing the model. Additionally, there's a dipole model of Earth's magnetic field within the "PD Controller" block. The orientation of Earth's magnetic field and the spacecraft's magnetic moment determine the available magnetic torque direction. The applied torque to the spacecraft is the projection of the control torque onto a unit vector pointing along the available magnetic torque direction.

A good next step is to demonstrate control of the spin rate of the spacecraft in the presence of a disturbance torque caused by the gravity gradient effect. The controller should be able to maintain a commanded spin rate of around 30 times the spacecraft's mean motion. Next, the model should transition from a spinning rigid body to something more realistic—two masses connected by tethers. The tethers should be modeled as springs with an appropriate stiffness. Applying Newton's second law to each mass creates an effectively rigid tensegrity structure when spinning fast enough. This model also captures the gravity gradient disturbance torque. A key question to answer with this model is: What spin rate is necessary to maintain the minimum required tension in the tethers? From there, the model should incorporate another control loop that causes the masses to trace the appropriate elliptical trajectory that produces the desired propulsive effect.

It's important that the spacecraft is always spinning around the principal axis with highest moment of inertia, and it turns out that this requirement isn't always automatically satisfied. The tether material is quite heavy, and if the masses come too close together, the spin axis can become the intermediate axis. The spacecraft has too much angular momentum to make super-spin feasible, so the solution is to maintain some minimum distance between the two masses. This should not cause any trouble, but it's an important design requirement to be aware of.

Another interesting idea that warrants investigation is orientation of the spacecraft relative to its orbit. The spacecraft is always spinning about the truss between the two masses. The original idea is to have the spin axis point orbit normal, which makes the ellipse traced by the spinning masses in the plane of the orbit. Another idea is to have the spin axis oriented along $0.7071\vec{o}_1 + 0.7071\vec{o}_3$, which is in the plane of the orbit. Some complexity would be removed because the masses would be able to spin in a circle rather than an ellipse. However, the angular momentum must always point along the $0.7071\vec{o}_1 + 0.7071\vec{o}_3$ axis, which requires tilting it 2π radians every period. This must be accomplished using magnetic torque.

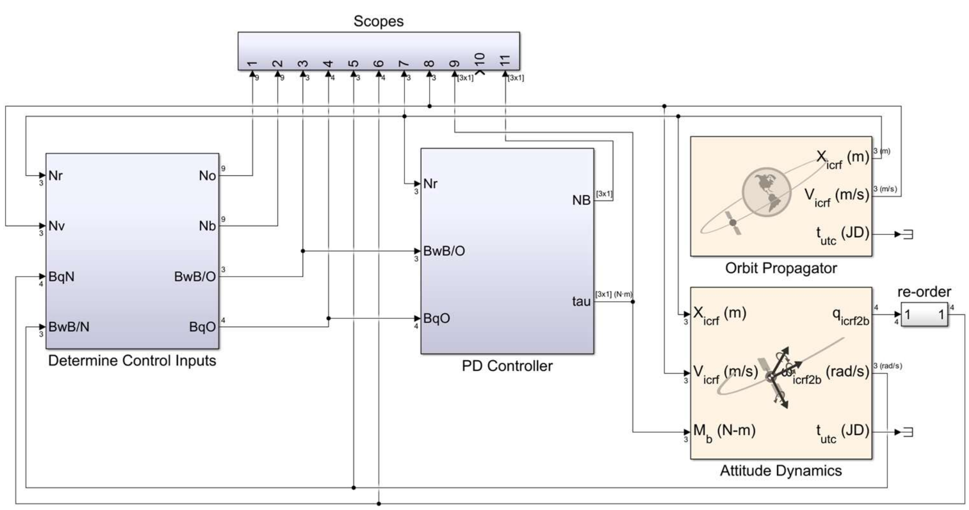

Figure 1: Spacecraft model Simulink blocks

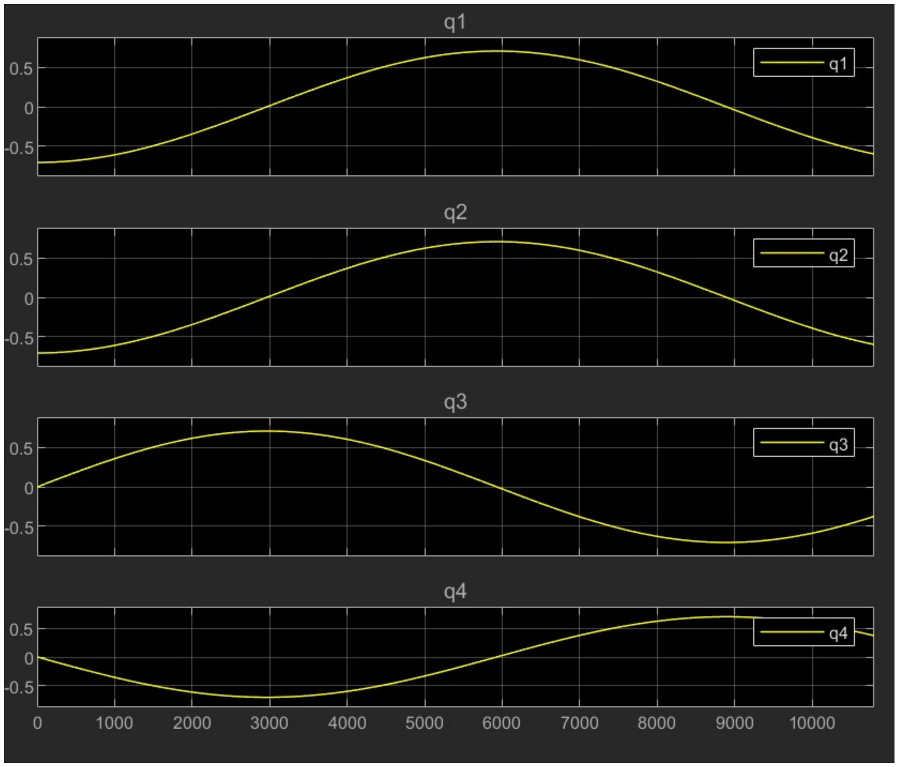

Figure 2: BqN, quaternion rotation from N (ECI) to B over a little less than 2 orbits

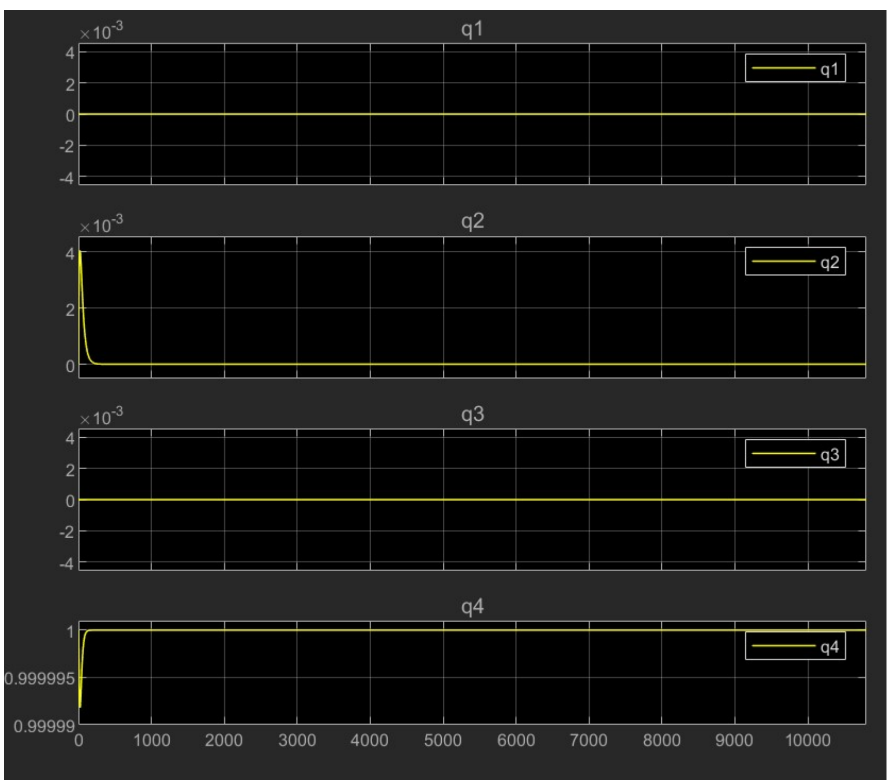

Figure 3: BqO, quaternion rotation from O to B over a little less than 2 orbits