Satellite Control Simulation for the Cosmic X-Ray Background NanoSat (CXBN)

Lance Simms

August 19, 2011

1 Introduction

The Cosmic X-Ray Background NanoSat (CXBN) control simulator is a Python based program that simulates the motion of the CXBN CubeSat as it orbits Earth. It models both the orbital propagation of the spacecraft around Earth using the Simple General Perturbations (SGP4) model as well as the rigid body dynamics of the spacecraft about its center of mass. The latter involves estimating the disturbing environmental torques, i.e. gravity gradient and atmospheric drag torques, and calculating the control torques necessary for CXBN to achieve its science goals.

The purpose of this document is to

- **1)** Present results from the simulations that show they are properly describing the dynamics of the spacecraft. To accomplish this, the simple case of a passive gravity gradient satellite without stabilization is presented. This case has been well studied, and for certain configurations, closed form solutions are known [1].

- **2)** Determine the proper forms for the control algorithms that will be used to orient the satellite properly for the CXBN science goals.

2 Methods

The CXBN simulator uses a finite-difference time-stepping approach to evolve the satellite trajectory and orientation with time. The initial time, t_o, is arbitrarily chosen as 2012/06/01 at 00:00:00 and the time-step, denoted by d_t, is chosen as 0.01 seconds. This time-step is sufficiently small to keep the numerical error tame for a large amount of orbits.

2.1 Orbital Propagation

At each time-step, the PyEphem software package (http://rhodesmill.org/pyephem/index.html) calculates the position of the satellite in coordinates of (right-ascension, declination, elevation). These are converted to Earth-Centered Inertial (ECI) coordinates to yield a position vector $\vec{r}_{eci} = (X_E, Y_E, Z_E)$ relative to the center of Earth. The velocity of the spacecraft is then evaluated by considering the positions at times $t - dt$ and $t + dt$, before and after the current time, respectively. With the position vector, $\vec{r}_{eci}$, and velocity vector, $\vec{v}_{eci}$, an Orbit Reference Frame (OBF) is determined as follows:

- The X_R axis is chosen so that $+X_R$ is aligned with the spacecraft velocity vector $(\vec{v}_{eci})$.

- The Z_R axis is chosen so that $-Z_R$ is aligned with the vector to the center of the earth $(-\vec{r}_{eci})$.

- The Y_R axis completes a right-handed triad $(Y_R = Z_R \times X_R)$.

2.2 Rigid Body Dynamics

The next thing that is handled is the orientation of the spacecraft. At time t_o, the body axes of the satellite (X_B, Y_B, Z_B) are chosen so that they are aligned with the Orbit Reference Frame axes (in certain cases, an initial Yaw, Pitch, and Roll at time t_o will be chosen). These frames are depicted in Figure 1.

The Body Frame and Orbit Reference Frame will in general differ at later times due to the fact that the satellite is moving and external torques are forcing its orientation to change. To handle the evolution of the spacecraft orientation, the Euler equations:

$$\dot{\omega}_1 = \frac{1}{I_1}\big((I_3 - I_2)\omega_2\omega_3 + T_1\big), \tag{1}$$

$$\dot{\omega}_2 = \frac{1}{I_2}\big((I_1 - I_3)\omega_1\omega_3 + T_2\big), \tag{2}$$

$$\dot{\omega}_3 = \frac{1}{I_3}\big((I_2 - I_1)\omega_1\omega_2 + T_3\big), \tag{3}$$

are solved, where ω is the instantaneous angular velocity of the spacecraft (as measured in body coordinates), $\dot{\omega}$ is the time derivative of the angular velocity, and $\vec{T}$ is the torque acting on the spacecraft. I is the moment of inertia tensor. One should note that it has been defined along the principal axes of the satellite so that it is diagonal.

Once $\dot{\omega}$ has been determined, it is used to evolve the orientation of the satellite according to the prescription given by Whitemore[2]. This method utilizes quaternions to express the orientation of the satellite and it updates them in such a way that they always maintain a unit norm. With the updated quaternion obtained, the Direction Cosine Matrix between the Body Frame and the Orbit Reference Frame is calculated. From the DCM, the new Yaw, Pitch, and Roll angles can be obtained. From here, another time-step is taken and the procedure is repeated.
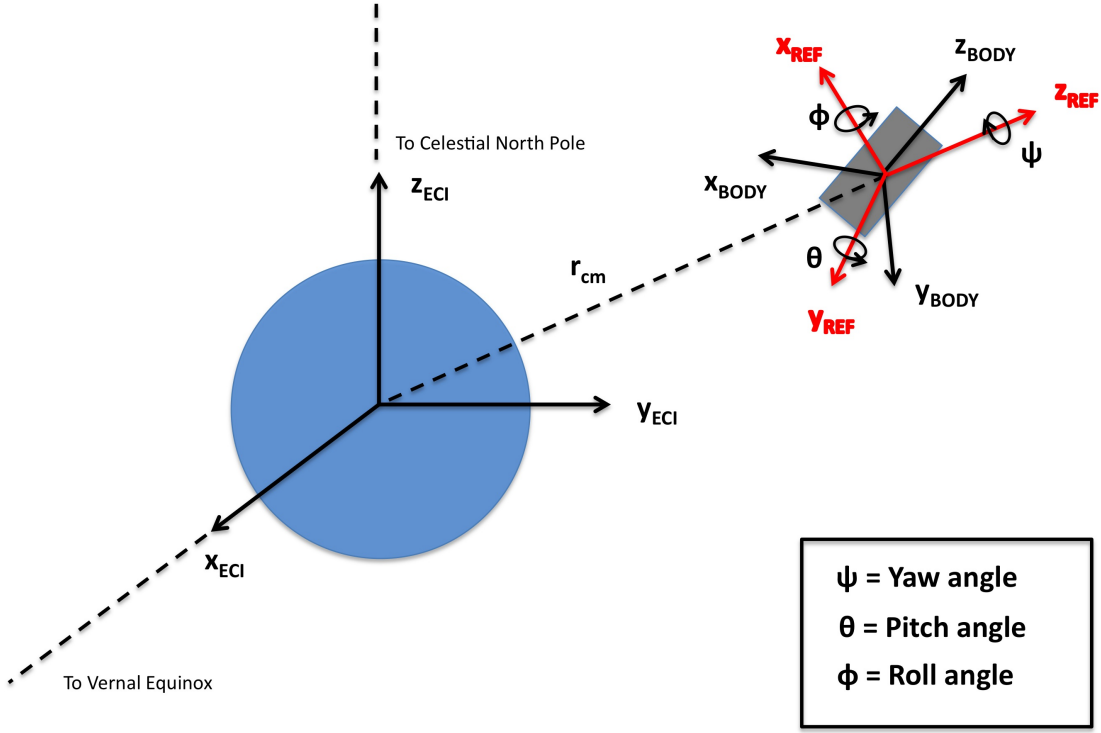
Figure 1: The coordinate reference frames used in the simulation. **E** correponds to the Earth Centered Inertial (ECI) reference frame, **B** correponds to the body reference frame, and **R** correponds to the satellite reference frame.

3   Initial Test with Passive Gravity Gradient Satellite

To validate the simulator is properly calculating gravity gradient torque, solving Euler's equations, updating quaternions, and expressing orientations with Direction Cosine Matrices, two simple scenarios for a passive gravity gradient (GG) stable satellite were carried out.

The GG satellite is one where the moment of inertia about the nadir-pointing axis, I_z, is made much smaller than I_x and I_y. The large moment arms for X and Y create a large torque when the Z axis points away from $-\vec{r}_{eci}$ and this torque tries to restore the satellite to a nadir pointing. If there is a source of damping, the satellite can be maintained in nadir. Without damping, the

satellite will oscillate, and in certain cases the oscillation is very well behaved.

3.1 Gravity Gradient Test 1

For the case where the Body and Orbit Reference Frames are initally aligned, the motion of the GG satellite is quite simple. The satellite is expected to rotate about the Y_B pitch axis with simple harmonic motion as the satellite orbits the Earth. The frequency of this motion is given by

$$\omega_{sim} = \omega_o \sqrt{3\frac{I_y - I_z}{I_x}}, \tag{4}$$

where ω_o is the orbital rate of the satellite in radians/second.

For this first test case, the values $I_x = I_y = 10$ kg·m^2 and $I_z = 0.1$ kg·m^2 were chosen. The satellite was placed in a circular, equatorial orbit at an elevation of ~ 800 km. At time t_o, the Body and Orbit Reference frames were aligned.

The orbital period was 14.719786 orbits/day, which corresponds to $\omega_o = 0.00107$ radians/sec. The expected frequency for the simple harmonic motion about the Y_B axis is thus $\omega_{sim} = 0.00184478$ radians/sec, which corresponds to a period of about 3400 seconds.

As shown in Figure 2, the frequency matches very closely to the latter value. It is about 3700 seconds, as opposed to 3400 seconds. Further, the values for the Yaw and Roll stay quite close to zero throughout the orbit. There is a small deviation of the Roll due to the fact that the approximation to the velocity vector is not quite perfect and the numerical error in solving the Euler equations is non-zero.

3.2 Gravity Gradient Test 2

For the second test case, the scenario in Example 5.3.1 of Sidi [1] has been followed. For this case, $I_x = 80$ kg·m^2, $I_y = 82$ kg·m^2, and $I_z = 4$ kg·m^2. were chosen. All other aspects of the configuration were the same is in Case 1, except that the satellite was given an inital Yaw of 5 degrees. Using Equation 4, the expected period is nearly the same as in Case 1.
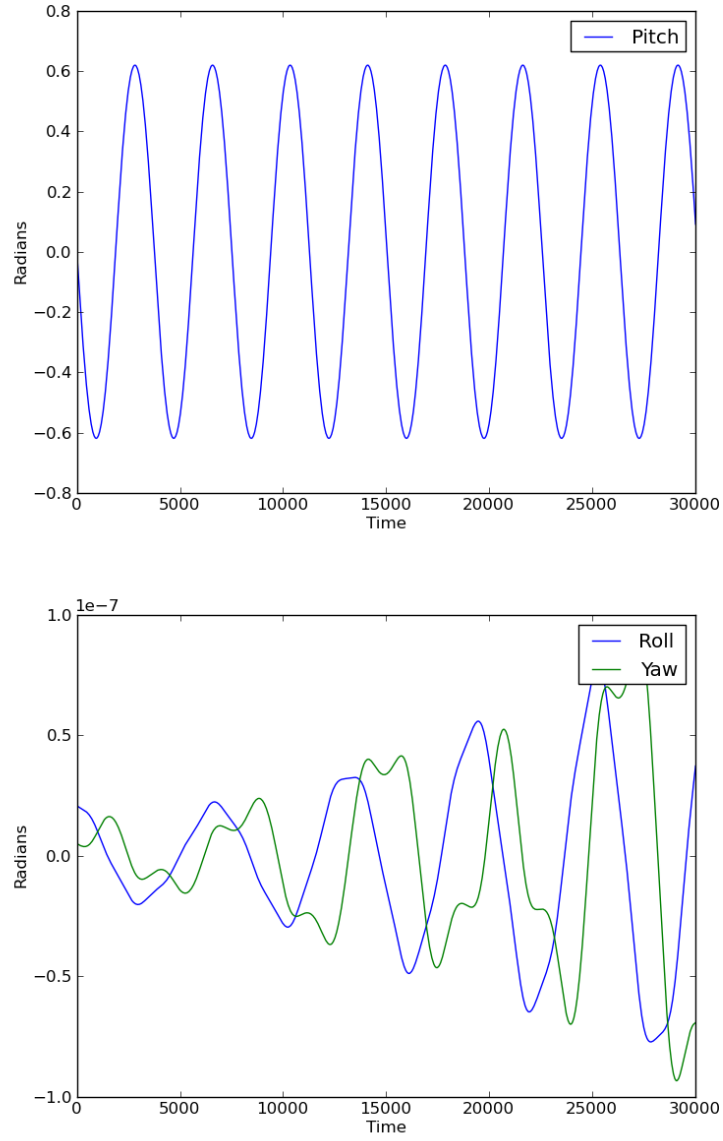
4

Figure 2: Yaw, Pitch, and Roll for the simple test case of the passive gravity gradient satellite without damping. For this case, the initial Yaw, Pitch, and Roll were all zero. The Pitch angle undergoes simple harmonic motion with a period of about 3400 seconds, as expected. The Yaw and Roll angles are non-zero due to numerical error.
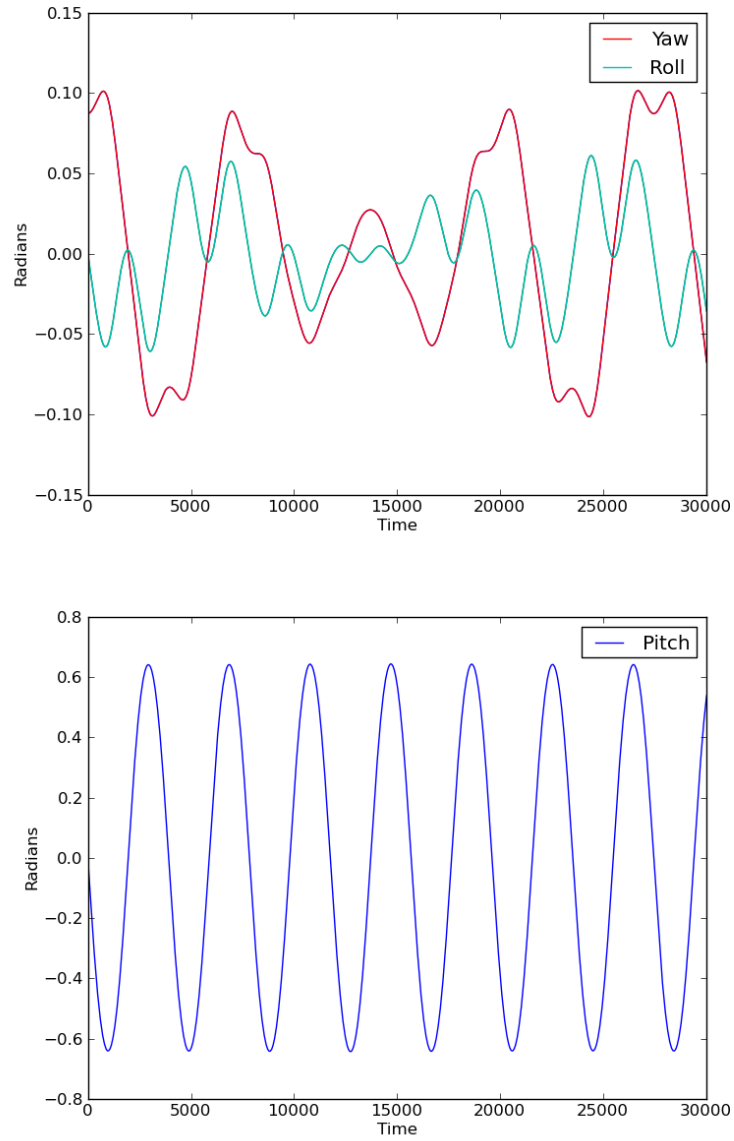
Figure 3: Yaw, Pitch, and Roll for the simple test case of the passive gravity gradient satellite without damping. For this case, the initial Yaw angle was 5 degress. The Roll and Pitch were both zero. The Pitch angle undergoes simple harmonic motion with a period of about 3400 seconds, as expected. The Yaw and Roll oscillate in a reasonable fashion.

4 Full CXBN Simulation

Along with the orbit propagation, the full CXBN simulation includes gravity gradient torque, aerodynamic torque, and a control torque from the magnetorquers. One important aspect of magnetic control is that magnetic sensing and controlling cannot take place at the same time. This is because any dipole moment generated by the magnetorquers will change the magnetic field sensed by the magnetometers. The magnetorquers must be inactive while the magnetometers are measuring the magnetic field of the Earth.

To account for this, the simulator accounts for the measuring and torquing periods using a user controlled duty cycle. The duty cycle spans a period of $T_{SenseTorque}$, where $f_{SenseTorque} = 1/T_{SenseTorque}$. And during each period, the sensing occurs for a fraction F_{Sense}, after which the torquing is applied for a fraction F_{Torque}, where $F_{Sense} + F_{Torque} = 1$ and both fractions are positive.

The baseline parameters for the CXBN simulator are the following:

- $f_{SenseTorque} = 1\text{Hz}$

- $T_{SenseTorque} = 1$ second

- $F_{Sense} = 0.5$

- $F_{Torque} = 0.5$

4.1 Detumbling

After ejection from the pea-pod, the CubeSat will have some unknown angular velocity vector. This angular velocity, estimated to be approximately 20 degrees/second, will have an unknown orientation with respect to the spacecraft body. Thus, the satellite will almost certainly undergo torque-free precession, otherwise known as "tumbling".

The control loop used to de-tumble with the magnetometers is based on the well-known equation

$$\mathbf{m}^c = -k^c\dot{\mathbf{B}} - \mathbf{m}_{const}, \tag{5}$$

where

- $\mathbf{m}^c$ is the control dipole moment generated by the magnetorquers

- k^c is a control gain to be determined from simulation or empirical data

- $\dot{\mathbf{B}}$ is the rate of change of the magnetic field as sensed by the magnetometers on the cubesat

- $\mathbf{m}_{const}=[0,0,m_{const}]$ is a constant component of the dipole moment aligned along the z-axis.

This equation is easily understood by making the analogy with a compass needle that has some damping. The second term represents the constant dipole moment that the needle would generate, which causes the needle to align with the magnetic field. The first term damps the motion whenever the needle is swinging fast past the alignment vector.

A test of the control loop has been performed with a time step of 0.001 seconds and initial angular velocity of $\omega = [18.5°/s, -25.0°/s, 12.2°/s]$. The results are shown in figure 4. One can see from the figure that the initial precession is damped out after about 1500 seconds, which corresponds to roughly 1/2 of an orbit. The satellite is not completely stable due to the aerodynamic torque, but it does gradually approach the stable pointing antiparallel to the magnetic field.

4.2 Detumbling with Different Sense/Torque Frequencies, $f_{SenseTorque}$

As one might guess, the value of $f_{SenseTorque}$ must be chosen carefully. If it is too high, there will not be ample time for averaging of the signal from the magnetometers to yield a clean measurement. The change in magnetic field, $\Delta \vec{B}$ may also be too small if not enough time is waited in between averagings. If $\Delta B_{x,y,z} < 3\sigma_{magnetometer}$, where $\Delta B_{x,y,z}$ represents the change in each component of the field and $\sigma_{magnetometer}$ is the root mean square of the noise inherent in reading the magnetometer, then the torque generated will in a completely random direction.

On the other hand, if $f_{SenseTorque}$ is too low, then the magnetic field may change drastically between readings and create torques that are far too high. The value of $f_{SenseTorque}$= 1Hz is sufficiently high to ensure this is not the case. In 1 second, the spacecraft travels about 7.8 km. A degree of latitude corresponds to about 110 km, so that distance is roughly 0.072°. Over this angular scale, the magnetic field components change by less than 3%, even at points on the earth where the magnetic contours are closest together.

Figures 4 and 5 show the difference between frequencies of 1Hz and 10Hz, respectively. One can see that the 10Hz loop results in more instabilities than the 1Hz loop because $\Delta \vec{B}$ is not large enough to generate the torque necessary to keep the spacecraft stably pointed along the magnetic field vector $\vec{B}$.
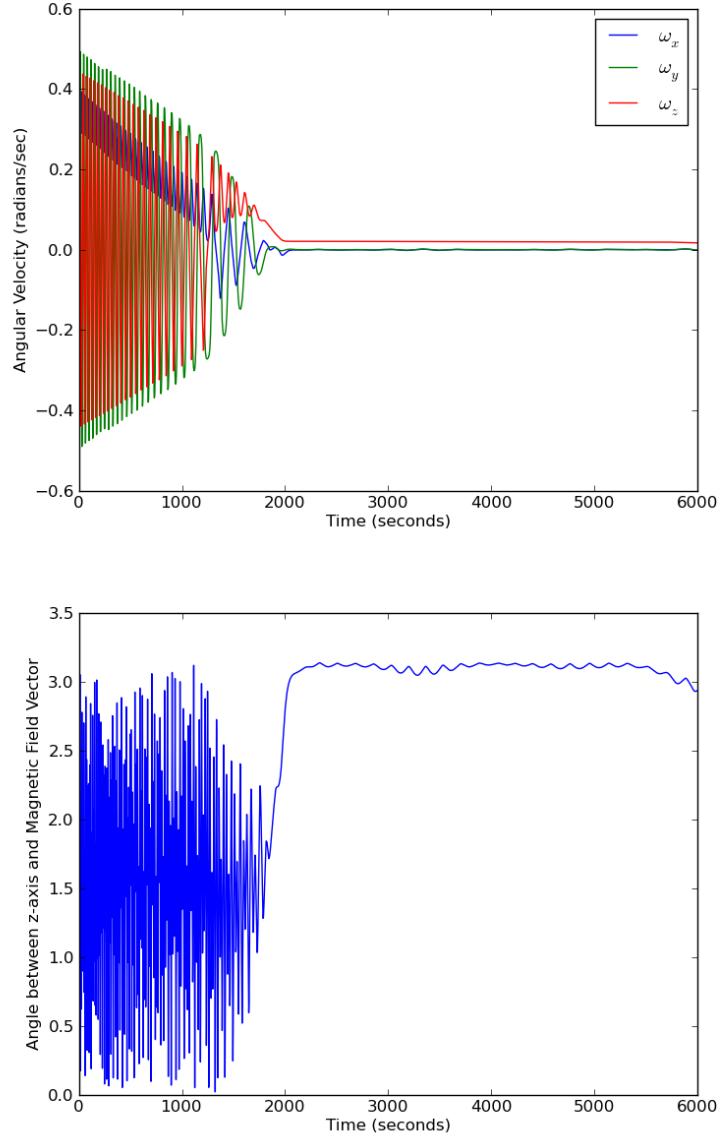
Figure 4: The upper plot shows the angular velocity (measured in the body frame) of the satellite during the de-tumbling phase. The initial angular velocity vector was $\omega = [18.5°/s, -25.0°/s, 12.2°/s]$ and the frequency for applying the torques is $f_{SenseTorque} = 1$Hz. The lower plot shows the angle between the magnetic field and the z-axis of the satellite. Note that orientations with +z and -z along the magnetic field are both stable solutions.
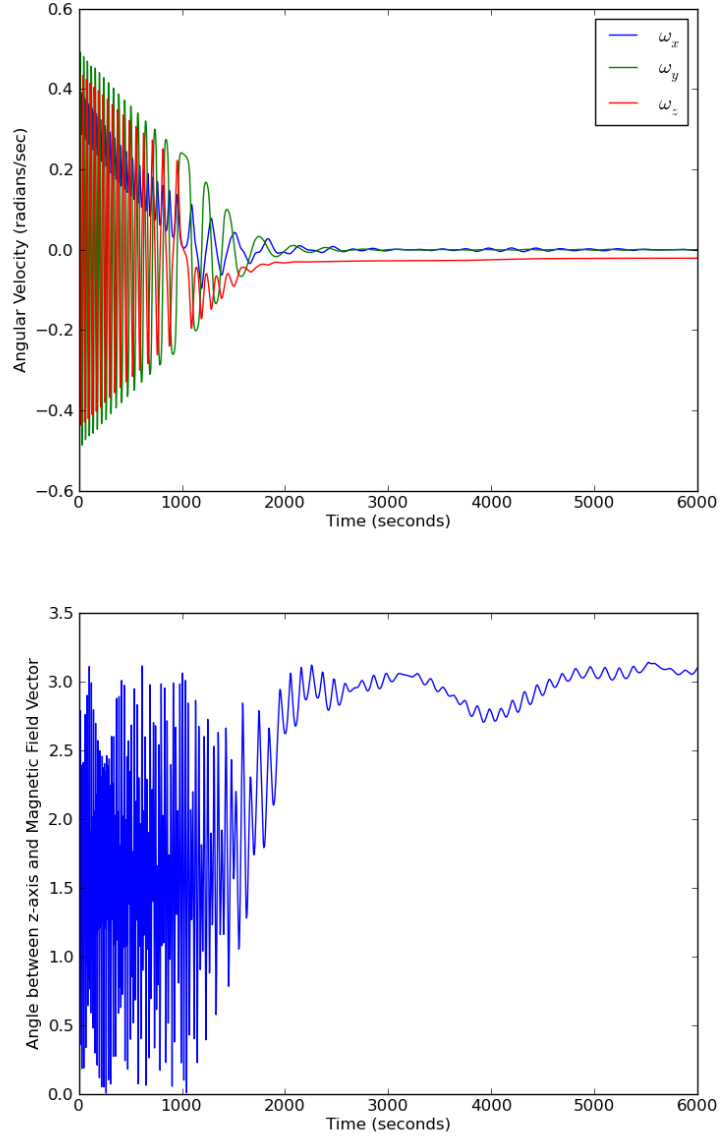
9

Figure 5: The same plots as in Figure 4 except that $f_{SenseTorque}$= 10Hz. One can see that the higher frequency results in less stability due to the fact that the magnetic field does not change appreciably over the 0.1 second interval. This will be especially problematic in the presence of noise.

10

5 Coarse Sun Acquisition with Solar Panels

Once detumbling has taken place, the next step is to orient the spacecraft so that its z-axis aligns with the sun pointing vector. In this position, the solar panels will be fully illuminated and provide maximum power to the spacecraft.

While the medium and fine sun sensors will eventually provide precise information on the alignment, at first the spacecraft will rely on the currents sensed in each solar panel. A misalignment will cause the spacecraft body to cast a shadow on some of the solar panels, and a subsequent reduction in current for those panels. For a given axis (x or y), the difference in currents between the two solar panels along that axis can be used as a proxy for the torque about that axis.

The acquisition algorithm goes as follows. First, the difference between the solar panel currents along the x and y axes are measured.

$$\Delta I_x = I_{+x} - I_{-x} \tag{6}$$
$$\Delta I_y = I_{+y} - I_{-y}. \tag{7}$$

Then, the desired angular angular velocities about these axes are calculated using a Proportional and Derivative term.

$$\omega_{x\ desired} = k_t \Delta I_x + k_\omega \omega_{x\ measured} \tag{8}$$
$$\omega_{y\ desired} = k_t \Delta I_y + k_\omega \omega_{y\ measured} \tag{9}$$
$$\omega_{z\ desired} = 0, \tag{10}$$

where $\omega_{desired}$ are the desired angular velocity components, $\omega_{measured}$ are the angular velocity components measured by the gyros, and k_t and k_ω are the P and D terms, respectively. A proper choice of these is dependent on the moment of inertia tensor of the satellite and the values of current expected in the solar panels.

Once these desired components have been calculated, the magnetic moment components are calculated as follows:

$$\mathbf{m}^c = -\frac{1}{|B|}\ \vec{\omega} \times \mathbf{B} \tag{11}$$

This last equation is a common method to calculate the magnetic moment components based upon a desired angular velocity.

Figure 6 shows results from simulations where these values were chosen at $k_t = 0.001$ and $k_\omega = -0.10$. One can see that the angle between the z-axis and the sun pointing vector quickly goes to zero. The angular velocity about the z-axis does not go to zero. This must be dealt with at a later stage when the sun sensors take over: a task that remains to be completed by the CXBN team.
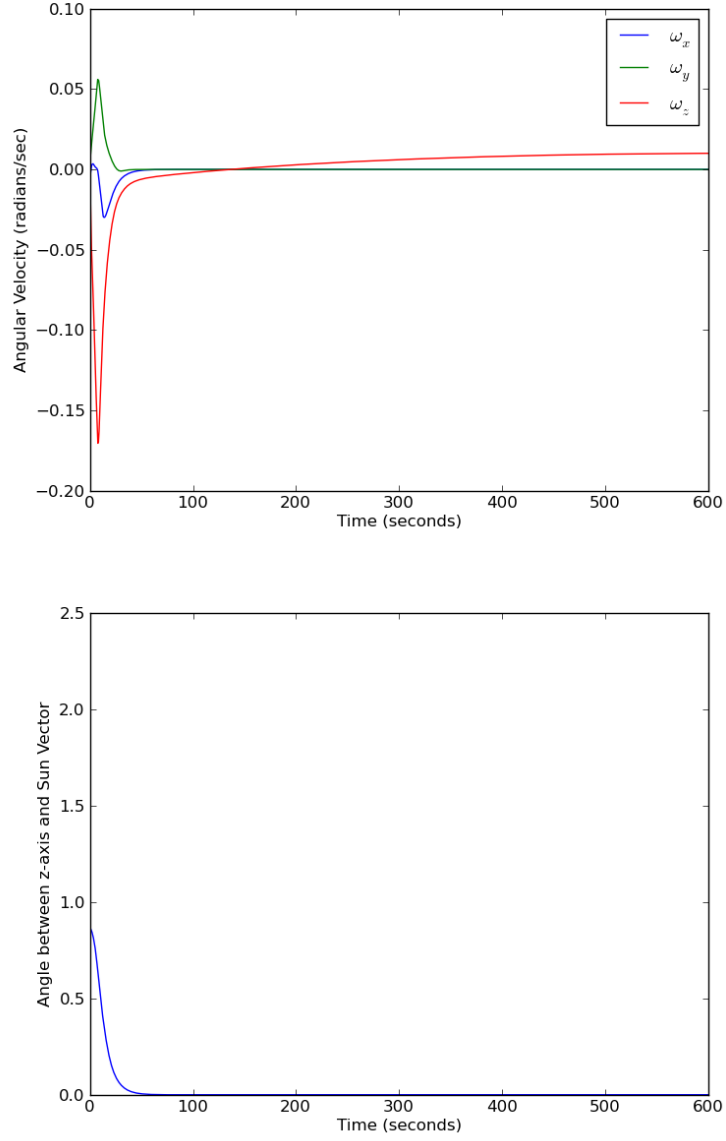
Figure 6: The upper plot shows the angular velocity (measured in the body frame) of the satellite during sun acquisition with the solar panel currents. The initial angular velocity vector was $\omega = [0°/s, 0°/s, 0°/s]$, based on the assumption that the satellite had already detumbled. The lower plot shows the angle between the sun pointing vector and the z-axis of the satellite.

12

6   Conclusion and Next Steps

The previous results show that the CXBN simulator is properly handling the dynamics of the satellite as it orbits Earth. There is a good match between the numerical and analytical solutions of the passive GG satellite; the small discrepancy is likely a result of the PyEphem treatment of the orbit or numerical error. The aerodynamic torque also produces the expected behavior. With respect to the control torques, in detumbling, the magnetic torque damps the motion of the satellite and aligns the z-axis with the magnetic field of the Earth. And in sun acquistion, it similary aligns with the sun pointing vector.

The next steps should be to

1. Create a function to perform the spin-up phase.

2. Create a function to model the medium and fine sun sensors.

3. Create a function to model the Canopus pipper.

References

[1] Marcel J. Sidi. *Spacecraft Dynamics and Control: a Practical Engineering Approach.* Cambridge University Press, Cambridge, UK, 1997.

[2] Stephen A. Whitmore. Closed-form integrator for the quaternion (euler angle) kinematics equations, 1998.