

Problem 1. Attitude Disturbance Analysis

Do calculations for the disturbances present for equatorial circular orbits at 200km and 350km. Consider a cube with two different halves. One half has twice the density (and therefore twice the mass) as the other. We will only be concerned with rotations about the axis out of the page.

I define \hat{x} along the velocity/sun vector, and \hat{y} in the opposite direction of the Earth. \hat{z} completes the right hand rule. See the attached code for derivations of the following values.

Part (a)

1. Drag Find the disturbance torque on the cube about the mass center (use $CD = 2.2$)

$$\underline{T} = \sum \underline{r}_i \times \underline{F}_i$$

$$\underline{F}_i = \frac{1}{2} \rho V^2 C_D \left(\hat{n}_i \hat{V} \right) A_i \left(-\hat{V} \right)$$

$$T = 5.45e - 06 \hat{z} \text{ Nm at altitude 200km. } T = 5.18e - 07 \hat{z} \text{ Nm at altitude 350km.}$$

Part (b)

2. Solar Pressure Find the disturbance torque on the cube about the mass center. Assume that the entire surface is covered in solar panels ($f_s \approx 0.21$, $f_d \approx 0.1$)

$$\underline{T} = \sum \underline{r}_i \times \underline{F}_i$$

$$\underline{F}_i = a_i \hat{s} + b_i \hat{n}_i$$

$$a_i = -P A_i \left(1 - f_{s,i} \right) \cos \theta_i$$

$$b_i = -2P A_i \left(f_{s,i} \cos \theta_i + \frac{1}{3} f_{d,i} \right) \cos \theta_i$$

$$T = 5.34e - 10 \hat{z} \text{ Nm at altitude 200km. } T = 5.34e - 10 \hat{z} \text{ Nm at altitude 350km.}$$

Part (c)

3. Gravity Gradient Find the disturbance torque on the cube about the mass center

$$\underline{T} = \sum \underline{r}_i \times \underline{F}_i$$

$$\underline{F}_i = \frac{\mu m_i}{r_i^2} (-\hat{r})$$

$$T = -0.031 \hat{z} \text{ Nm at altitude 200km. } T = -0.031 \hat{z} \text{ Nm at altitude 350km.}$$

Part (d)

4. Magnetic Dipole Find the disturbance torque on the cube about the mass center for two configurations

$$\underline{T} = \sum \underline{M}_i \times \underline{B}$$

$$\underline{M}_i = nIA\hat{c}$$

Magnetic Dipole Torque $T = 5.36e - 6\hat{x} - 9.29e - 4\hat{y}$ Nm at altitude 200km. $T = 5.36e - 6\hat{x} - 9.29e - 4\hat{y}$ Nm at altitude 350km.

Problem 2. Reaction Wheel and Thruster Analysis

Assume a maximum constant on orbit disturbance torque of 1×10^{-5} Nm on a single axis. Size a reaction wheel to keep a craft pointed despite this disturbance. Assume a saturation speed of 6000rpm for the wheel. It should be capable of eliminating the maximum on orbit disturbance without desaturating for two weeks.

Given,

$$H_{storage} = \int \frac{dH}{dt} = \int T_{disturbance,max} dt = T_{disturbance,max} t$$

$$\int \frac{dH}{dt} = \int I_{wheel} \alpha dt = I_{wheel} \omega_{saturated}$$

$$I_{wheel} \omega_{saturated} = \int I_{wheel} \alpha dt = \int \frac{dH}{dt}$$

$$= \int T_{thrusters} dt = T_{thrusters} t$$

as such,

$$T_{disturbance,max} t = I_{wheel} \omega_{saturated}$$

or

$$I_{wheel} = \frac{T_{disturbance,max} t}{\omega_{saturated}}$$

selecting $T = 1 \times 10^{-5}$ Nm, $t = 2 * 7 * 24 * 60 * 60 = 1209600$ s, and $\omega = 6000$ rpm, gives:

$$I_{wheel} = \frac{T_{disturbance,max} t}{\omega_{saturated}}$$

$$= \frac{1E - 5 \text{ Nm} * 1209600 \text{ s}}{6000 \text{ rpm} \cdot \frac{0.1047 \text{ rad/s}}{1 \text{ rpm}}}$$

$$= 0.01925 \text{ kg m}^2$$

To size the reaction wheel, we have:

$$\begin{aligned} I_{wheel} &= \frac{mr^2}{2} \\ &= \frac{\rho\pi r^4 h}{2} \end{aligned}$$

where the reaction wheel is modelled as a simple disc-shape. Choosing $h = r$ and $\rho = 8000 \text{ kg m}^{-3}$ (316 stainless steel) leads to

$$\begin{aligned} r &= \left(\frac{2I_{wheel}}{\pi\rho} \right)^{1/5} \\ &= \left(\frac{2 \cdot 0.01925}{\pi 8000} \right)^{1/5} \\ &= 0.069 \text{ m} \end{aligned}$$

As such, our reaction wheels are made of 316 stainless steel, and have a radius and a height of 0.069 meters. To calculate the burn time to desaturate the thrusters, we have

$$\begin{aligned} I_{wheel}\omega_{saturated} &= T_{thrusters}t \\ \text{or} \\ t &= \frac{I_{wheel}\omega_{saturated}}{T_{thrusters}} \end{aligned}$$

selecting $T = 1 \text{ Nm}$, $\omega = 6000 \text{ rpm}$, and $I = 0.01925 \text{ kg m}^2$, gives:

$$\begin{aligned} t &= \frac{I_{wheel}\omega_{saturated}}{T_{thrusters}} \\ &= \frac{0.01925 \text{ kg m}^2 \cdot 6000 \text{ rpm} \cdot \frac{0.1047 \text{ rad/s}}{1 \text{ rpm}}}{1 \text{ Nm}} \\ &= 12.25 \text{ s} \end{aligned}$$

Problem 3. Hubble Slew Problem

Either use published inertia values for Hubble or the estimates you made in the dynamics assignment. Assume torquers are aligned with the principle axis.

$$I_{xx}\dot{\Omega}_x + H_{spin}\Omega_y = T_x$$

$$I_{yy}\dot{\Omega}_y + H_{spin}\Omega_x = T_y$$

$$H_{spin} = I_{wheel}\omega_{wheel}$$

Part (a)

1. Two axis torques Find the required variable torques (x and y) to constantly accelerate the angular precession by 0.1 deg/s^2 on the x axis while trying to keep the angular procession on the y-axis at a constant 0.1 deg/s .

The HST inertia matrix ¹, was at one point measured as:

$$I = \begin{bmatrix} 36046 & -706 & 1491 \\ -706 & 86868 & 449 \\ 1491 & 449 & 93848 \end{bmatrix} \text{ kg} \cdot \text{m}^2.$$

Plugging in the given values of $I_{wheel} = 1 \text{ kg m}^2$, $\omega = 1500 \text{ rpm}$, $I_{xx} = 36046 \text{ kg m}^2$, $I_{yy} = 86868 \text{ kg m}^2$, $\dot{\Omega}_x = 0.01 \text{ deg s}^{-2}$, $\dot{\Omega}_y = 0 \text{ deg s}^{-2}$, $\Omega_y = 0.1 \text{ deg s}^{-1}$ gives

$$H_{spin} = 1 \cdot 1500 \cdot 0.1047 = 157.05$$

$$T_x = 36046 \cdot 0.0001745 + 157.05 \cdot 0.0017453 = 6.56$$

$$T_y = 86868 \cdot 0 + 157.05 \cdot \Omega_x$$

$$= 157.05 \cdot \Omega_x$$

Part (b)

2. Two axis torques Find the required variable torque (x) to constantly accelerate the angular precession by 0.1 deg/s^2 on the x axis while applying no torque to the y axis.

Plugging in the given values of $I_{wheel} = 1 \text{ kg m}^2$, $\omega = 1500 \text{ rpm}$, $I_{xx} = 36046 \text{ kg m}^2$, $I_{yy} = 86868 \text{ kg m}^2$, $\dot{\Omega}_x = 0.01 \text{ deg s}^{-2}$, $\Omega_y = 0.1 \text{ deg s}^{-1}$ gives

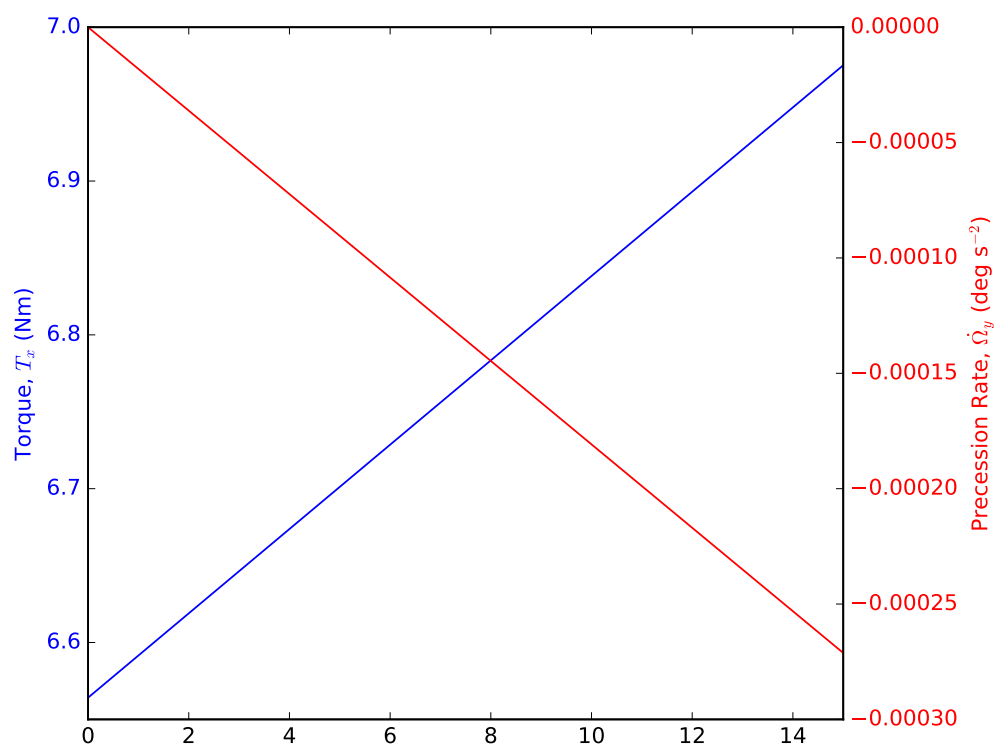
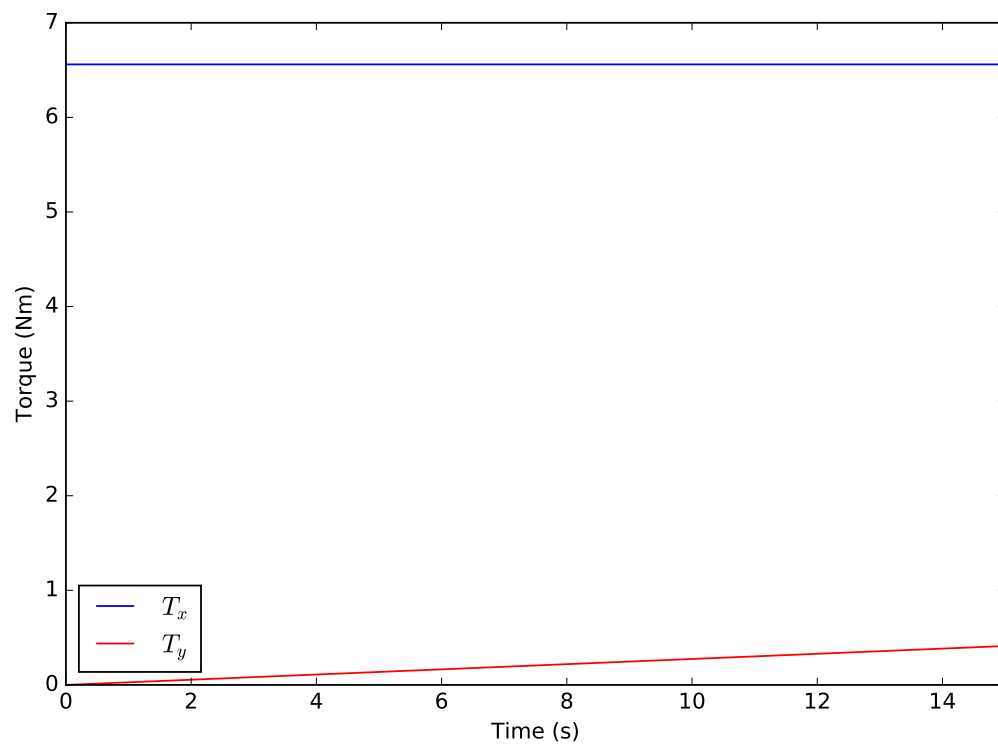
$$H_{spin} = 1 \cdot 1500 \cdot 0.1047 = 157.05$$

$$T_x = 36046 \cdot 0.0001745 + 157.05 \cdot \Omega_y$$

$$T_y = 86868 \cdot \dot{\Omega}_y + 157.05 \cdot \Omega_x$$

$$\dot{\Omega}_y = -\frac{157.05 \cdot \Omega_x}{86868}$$

¹Queen, S., "HRV GNC Peer Review, Flight Performance Analysis," Tech. rep., NASA Goddard Space Flight Center, 2004.



.1 Python Code

```

1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 #####
5 # Problem 1
6 # Part 1
7 a = 10 * 1/100 # m
8 mu = 1/1000 * (100)**3 # kg/m^3
9 A = a**2 # m^2
10 CD = 2.2
11 theta = np.deg2rad(30) # rad
12
13 n1 = np.array([np.cos(theta), np.sin(theta), 0]) # red face
14 n2 = np.array([np.sin(theta), np.cos(theta), 0]) # top face
15 V = np.array([1, 0, 0])
16
17 low = {'v': 7789,
18        'rho': 8.28E-10,
19        'alt': '200km'}
20 high = {'v': 7702,
21         'rho': 8.05E-11,
22         'alt': '350km'}
23
24 m1 = mu * (a * a * (a/2))
25 m2 = (mu/2) * (a * a * (a/2))
26 r1 = np.array([0, a/4, 0])
27 r2 = np.array([0, -a/4, 0])
28 Rz = np.array([[np.cos(theta), -np.sin(theta), 0],
29                [np.sin(theta), np.cos(theta), 0],
30                [0, 0, 1]])
31 r1_rot = np.dot(r1, Rz)
32 r2_rot = np.dot(r2, Rz)
33
34 x = (m1 * r1_rot[0] + m2 * r2_rot[0])/(m1 + m2)
35 y = (m1 * r1_rot[1] + m2 * r2_rot[1])/(m1 + m2)
36 r = np.array([x, y, 0])
37
38 print('\nDrag Torque')
39 for alt in [low, high]:
40     T = np.array([0., 0., 0.])
41     for n in [n1, n2]:
42         F = 0.5 * alt['rho'] * alt['v']**2 * CD * np.dot(n, V) * A * (-V)
43         T += np.cross(r, F)
44     print(T, 'at altitude {}'.format(alt['alt']))
45
46 #####
47 # Part 2
48 P = 4.67E-6
49 s = np.array([1, 0, 0])
50 a_i = -P * A * (1 - .21) * np.cos(theta)
51 b_i = -2 * P * A * (.21 * np.cos(theta) + (1/3) * 0.1) * np.cos(theta)

```

```

52
53 print('\nSolar Pressure Torque')
54 for alt in [low, high]:
55     T = np.array([0., 0., 0.])
56     for n in [n1, n2]:
57         F = a_i * s + b_i * n
58         T += np.cross(r, F)
59     print(T, 'at altitude {}'.format(alt['alt']))
60
61 #####
62 # Part 3
63 mu = 0.3986E15
64 r_earth = np.array([0., 1., 0.])
65
66 print('\nGravity Gradient Torque')
67 for alt in [low, high]:
68     T = np.array([0., 0., 0.])
69     for m, r_rot in zip([m1, m2], [r1_rot, r2_rot]):
70         rr = r_rot + np.array([0, -6371E3, 0])
71         F = ((mu * m) / np.linalg.norm(rr)**2) * (-r_earth)
72         T += np.cross(r, F)
73     print(T, 'at altitude {}'.format(alt['alt']))
74
75 #####
76 # Part 4
77 n = 25
78 I = 0.1
79 A = 25 * np.pi * (1/100)**2
80 c = np.array([np.cos(theta), np.sin(theta), 0])
81 B = np.array([0., 0., 1.])
82
83 print('\nMagnetic Dipole Torque')
84 for alt in [low, high]:
85     T = np.array([0., 0., 0.])
86     for n in [n1, n2]:
87         M = n * I * A * c
88         T += np.cross(M, B)
89     print(T, 'at altitude {}'.format(alt['alt']))
90
91 #####
92 # Problem 3
93 # Part a
94 t = np.linspace(0, 15)
95 f, ax = plt.subplots()
96
97 ax.plot(t, 6.56 * np.ones_like(t), 'b-', label='$T_x$')
98 ax.plot(t, 157.05 * np.deg2rad(t * 1/100), 'r-', label='$T_y$')
99 plt.xlim(0, 15)
100 plt.legend(loc='best')
101 plt.xlabel('Time (s)')
102 plt.ylabel('Torque (Nm)')
103 plt.tight_layout()
104 plt.savefig('p3_a.pdf')
105 plt.close()

```

```
106
107 # Part b
108 t = np.linspace(0, 15)
109 f, ax = plt.subplots()
110
111 tx = 36046 * 0.0001745 + 157.05 * np.deg2rad(t * 1/100 + .1)
112 omega_dot_y = - np.rad2deg(np.deg2rad(157.05 * t * 1/100)/86868)
113 ax.plot(t, tx, 'b-', label='$T_x$')
114 ax.set_ylabel('Torque, $T_x$ (Nm)', color='b')
115 for tl in ax.get_yticklabels():
116     tl.set_color('b')
117
118 r_ax = ax.twinx()
119 r_ax.plot(t, omega_dot_y, 'r-', label='$\dot{\Omega}_y$')
120 r_ax.set_ylabel('Precession Rate, $\dot{\Omega}_y$ (deg s$^{-2}$)', color='r')
121 for tl in r_ax.get_yticklabels():
122     tl.set_color('r')
123
124 plt.xlim(0, 15)
125 plt.xlabel('Time (s)')
126 plt.tight_layout()
127 plt.savefig('p3_b.pdf')
128 plt.close()
```