LQR Controller Design for Three-Axis Attitude Control System of LEO Satellites

Siddharth Dey (ME18B075)
AS5545 Project Report

Nov 2021

1 Introduction

The attitude and orbit control system (AOCS) (ref[[5]]) controls the attitude and position of a complete space vehicle or satellite. Based on this, the spacecraft orients its solar generators, thermal radiators, thrusters, and particularly its payload units, optical instruments, and antennas. We can use various sensors such as star trackers, sun sensors, infra-red Earth sensors, and magnetometers. Such sensors make use of specific properties such as the spectrum and intensity of the incoming radiation from a star or the geomagnetic field among various others to determine its current attitude. Once it is estimated, the satellite can try to reorient itself along the desired direction by using one of the actuators installed in the satellite. These actuators might include reaction wheels (RW), momentum wheels, thrusters, magnetic torquers, and so on. The reaction wheels make use of their angular momentum along the axis of rotation to generate a reaction torque on the platform on which it is mounted, thereby influencing the overall momentum of the satellite. A satellite might be subjected to multiple disturbance torques: (a) magnetic field in the vicinity of a planet (b) residual air-drag in low orbits around a planet (c) electromagnetic radiation from the Sun (d) gravitational field of a planet close to the satellite, all of which can cause the spacecraft to lose it stability. Hence, in this report, we shall formulate the mathematical model of the governing dynamics of a satellite and design an optimal closed-loop controller based on a Linear Quadratic Regulator (LQR) which can send the required signal to the actuators (reactions wheels in our case) to accurately stabilize its attitude, after getting the feedback of its current attitude from a Sun Sensor.

2 State-Space Plant Model of a Satellite

In this section, we shall derive the dynamics equation of the spacecraft (ref [1]) which we will use later as the plant model for the controller design. The total angular momentum of the spacecraft can be given as

$$\mathrm{H_b = I\omega_b + h_w} \tag{1}$$

where I is the inertia matrix of the satellite, ω_b is the body frame's angular velocity and h_w is the angular momentum of the reactions wheels, which is given by the following relationship:

$$\mathrm{h_w = LI_w\omega_w} \tag{2}$$

where $I_w = \mathrm{diag}(I_{w1}, I_{w2}, ..., I_{wn})$ is the RW diagonal inertia matrix and $\omega_w = [\omega_{w1}\omega_{w2}...\omega_{wn}]$ is the wheels angular speed.$L_{3\times n}$ is the RW distribution matrix. Here 'n' refers to the number of reaction wheels used which in our case is three deployed along the three body-axis, hence n=3. Using the Newton-Euler equations for angular momentum, we get:

$$\frac{\mathrm{dH_b}}{\mathrm{dt}} = -\omega_b(t) \times \mathrm{H_b(t) + T_d} \tag{3}$$

where T_d represents the disturbance torques. By taking the derivative of equation 1 and equating with equation 3, we get:

$$\frac{d\omega_b}{dt} = I^{-1}[-\omega_b(t) \times (I\omega_b + h_w) + T_d + T_c \tag{4}$$

where $T_c = \dot{h_w}$ is the command torque. The torque generated by the reaction wheel τ_r^b is given by:

$$\tau_r^b = \begin{bmatrix} \tau_{rx} \\ \tau_{ry} \\ \tau_{rz} \end{bmatrix} = (\frac{dL_r}{dt})^b + \omega_{bi}^b \times L_r = \begin{bmatrix} \dot{L}_{rx} + L_{rz}\omega_y - L_{ry}\omega_z \\ \dot{L}_{ry} + L_{rx}\omega_z - L_{rz}\omega_x \\ \dot{L}_{rz} + L_{ry}\omega_x - L_{ry}\omega_y \end{bmatrix} \tag{5}$$

where $\omega_{bi}^b = [L_{rx}L_{ry}L_{rz}]^T = I_r\omega_r$ is the net moment vector of the reaction wheel and $L_r = [\omega_x\omega_y\omega_z]'$ is the body frame angular velocity vector. The standard 3-wheel configuration is applied here. As discussed in ref [3], the linearisation points for angular velocities ω_{ib}^b are selected as:

$$\omega_{ib}^b = [\dot{\Phi} \quad \dot{\Theta} \quad \dot{\Psi}]^T + \omega_0[-\Psi \quad -1 \quad \Phi]^T \tag{6}$$

where $\omega_{ib} = 2\dot{\varepsilon}$ is the angular velocity of satellite in body-axis and is given as:

$$\omega_{ib} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} 2\dot{\varepsilon_1} - 2\omega_0\varepsilon_3 \\ 2\varepsilon_2 - \varepsilon_0 \\ 2\dot{\varepsilon_3} + 2\omega_0\varepsilon_1 \end{bmatrix} \tag{7}$$

From equation (6) we can determine the following relationship between the quaternions $(\varepsilon_1, \varepsilon_2, \varepsilon_3)$ and the euler angles $(\varphi, \vartheta, \psi)$:

$$[\Phi \quad \Theta \quad \Psi]^T = [2\varepsilon_1 \quad 2\varepsilon_2 \quad 2\varepsilon_3]^T \tag{8}$$

Applying equations (6) and (7) to equation (3), we get the following equations in the component form:

$$I_x\ddot{\Phi} = \Phi[4\omega_0^2(I_z - I_y) - \omega_0\dot{\Theta}(I_z - I_y)] + \dot{\Theta}\dot{\Psi}(I_z - I_y) + \dot{\Psi}\omega_0(I_z - I_y + I_z) + \dot{L}_{rx} \tag{9}$$

$$I_y\ddot{\Theta} = 3\omega_0^2(I_x - I_z)\Theta + \Phi[\Psi\omega_0^2(I_x - I_z) + \dot{\Phi}\omega_0(I_z - I_x)] + \dot{\Psi}\Psi\omega_0(I_x - I_z) + \dot{\Psi}\dot{\Phi}(I_z - I_x) + \dot{L}_{ry} \tag{10}$$

$$I_z\ddot{\Psi} = \Psi[\omega_0^2(I_x - I_y) + \dot{\Theta}\omega_0(I_y - I_x)] + \dot{\Phi}[\omega_0(I_y - I_x - I_z) + \dot{\Theta}(I_x - I_y)] + \dot{L}_{rz} \tag{11}$$

We will use equations (9) to (11) to formulate our linear state space model of the form:

$$\dot{x}(t) = Ax(t) + Bu(t); y = Cx(t) + Du(t) \tag{12}$$

with the state variables 'x' and control input 'u' defined as:

$$x = [\Phi \quad \Theta \quad \Psi \quad \dot{\Phi} \quad \dot{\Theta} \quad \dot{\Psi}]^T; u = [\dot{L}_{rx} \quad \dot{L}_{ry} \quad \dot{L}_{rz}] \tag{13}$$

The matrix A,B,C,D are given as follows:

$$A = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 4\omega_0^2(I_z - I_y)/I_x & 0 & 0 & 0 & 0 & \omega_0(I_z - I_y + I_z)/I_x \\ 0 & 3\omega_0^2(I_x - I_z)/I_y & 0 & 0 & 0 & I_x \\ 0 & 0 & \omega_0^2(I_x - I_y)/I_z & \omega_0(I_y - I_x + I_z)/I_z & 0 & 0 \end{pmatrix}$$

$$B = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1/I_x & 0 & 0 \\ 0 & 1/I_y & 0 \\ 0 & 0 & 1/I_z \end{pmatrix}; C = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}; D = 0$$

We shall use this LTI system equation for devising the controller in a later section

3 Linear Quadratic Regulator(LQR)optimum controller

The LQR controller is useful when it comes to controlling Multiple Input Multiple Output (MIMO) systems where pole placement cannot completely determine the controller gains or parameters. And in some cases, the designer might not know the desired pole location. In LQR control (derived in ref[2]), we assume the control vector to be linear with the state vector with the state vector as shown below:

$$u(t) = -Gx(t) \tag{14}$$

where G is a suitable gain matrix. Hence the closed loop dynamic behaviour becomes:

$$\dot{x} = Ax + Bu = Ax - BGx = A_c x \tag{15}$$

where,

$$A_c = A - BG \tag{16}$$

However, instead of seeking a gain matrix to obtain desired pole locations, we try to obtain a gain to minimize a specified performance criteria J (also known as the cost function) expressed in terms of an quadratic equation of the state 'x' and the control input 'u' as follows:

$$J_\infty = \int t^\infty (x'Qx + u'Ru)dt \tag{17}$$

where 'Q' and 'R' are symmetric matrices. x'Qx (' implies transpose operation) represents a penalty on the deviation of the state 'x' from the origin or zero vector and the term u'Ru which denotes the cost of control. For the steady state solution, we get:

$$J_\infty = x'\bar{M}x \tag{18}$$

where $\bar{M}$ satisfies the algebraic Riccati equation:

$$0 = \bar{M}A + A'\bar{M} - \bar{M}BR^{-1}B'\bar{M} + Q \tag{19}$$

and the optimum gain matrix becomesL

$$G = R^{-1}B'\bar{M} \tag{20}$$

For most design applications, the following facts about the solution of equation (19) will suffice:

- The system is asymptotically stable
- The system defined by (A,B) is controllable and system defined by (A,C) is observable

4 Attitude Determination for Feedback Control

LQR control needs full state feedback, i.e, all the state variable needs to be known at all timestamps. This requires us to have sensors to measure them. We usually do not have sensors to directly measure the velocity or angular velocity directly. We either estimate angular position and take its derivative to obtain angular velocity or use a accelerometer to obtain angular acceleration and them integrate it. The latter is usually not preferred as integration causes errors to accumulate over time. So, in this study, we shall use the six-eye CSS (Coarse Sun Sensor) configuration as discussed in ref [6] to determine the attitude and then take its derivative.

- A CSS cell/eye with a normal vector $\vec{n}_1$ converts the radiation intensity received into electric current $'I_1'$ according to:

$$I_1 = \begin{cases} I_{max}, & \text{if } \vec{n}_1 \cdot \vec{s} > 0 \\ 0, & \text{if } \vec{n}_1 \cdot \vec{s} \leq 0 \end{cases}$$

  where $\vec{s}$ is the solar vector. We consider another cell/eye with the normal vector $\vec{n}_{-1} = -\vec{n}_1$ whose current configuration 'I_{-1}' is as follows:

$$I_{-1} = \begin{cases} -I_{max}, & \text{if } -\vec{n}_1 \cdot \vec{s} > 0 \\ 0, & \text{if } -\vec{n}_{11} \cdot \vec{s} \leq 0 \end{cases}$$

- The six CSS eye configuration is then used with $\pm\vec{n}_1$, $\pm\vec{n}_2$, and $\pm\vec{n}_3$, which produce:

$$\begin{bmatrix} I_1 - I_{-1} \\ I_2 - I_{-2} \\ I_3 - I_{-3} \end{bmatrix} = I_{max} \begin{bmatrix} \vec{n}_1 \cdot \vec{s} \\ \vec{n}_2 \cdot \vec{s} \\ \vec{n}_1 \cdot \vec{s} \end{bmatrix} = I_{max} \begin{bmatrix} n_1' \\ n_2' \\ n_3' \end{bmatrix} s \tag{21}$$

$$s = \frac{1}{I_{max}} \begin{pmatrix} n_1' \\ n_2' \\ n_3' \end{pmatrix}^{-1} \begin{pmatrix} I_1 - I_{-1} \\ I_2 - I_{-2} \\ I_3 - I_{-3} \end{pmatrix} \tag{22}$$

  Once we have obtained the solar vector $\vec{s}$ in the body frame, we can estimate the attitude of the spacecraft based on star catalogues.

- To obtain the derivative terms, we can use a numerical approximation of the derivative, i.e, the difference in parameter values divided by the time stamp. However, the data might contain noise which will cause its derivative to spike randomly. Therefore, before taking the derivative, it is common to use a Low Pass Filter (LPF) which only allows frequencies upto a certain bandwidth to pass through and allows us to get rid of the noise which usually have high frequencies as shown below:

$$H(s) = \begin{cases} 1, & \text{when frequency is low} \\ 0, & \text{when frequency is high} \end{cases}$$

5 LQR Controller Design for Attitude Control

Now that we have derived the LQR optimal gain and the sensors for feedback, we shall next analyse the effect of considering different 'Q' and 'R' matrices of the LQR cost function on the desired response in the time domain.
For this analysis, we have the following requirements:

- The settling time to be less than 10 seconds
- Zero steady state error
- Power consumption to be less than 0.3 Watts

We have the cost function as:

$$J(x,u) = \int (x'Qx + u'Ru)dt \tag{23}$$

where $u(t) = -Kx(t)$ ('K' is the optimal gain matrix). The matrices Q and R need to be positive definite and symmetric. For this study, we assume them to be a diagonal matrix of the form:

$$Q = \mathrm{diag}[Q_1, Q_2, ..., Q_{nS}] \tag{24}$$

$$R = \mathrm{diag}[R_1, R_2, ..., R_{na}] \tag{25}$$

where n_s is the number of states and n_a is the number of actuators. We solve for the LQR problem using the MATLAB function 'lqr' with State Transition matrix and the cost matrix as input:

$$[K, P, E] = \mathrm{lqr}(A, B, Q, R) \tag{26}$$

which returns the optimal gain matrix 'K' and the eigenvalues of the closed loop system 'E', which can used to observe the pole locations of the plant. The block diagram of the plant can be seen in Fig 1

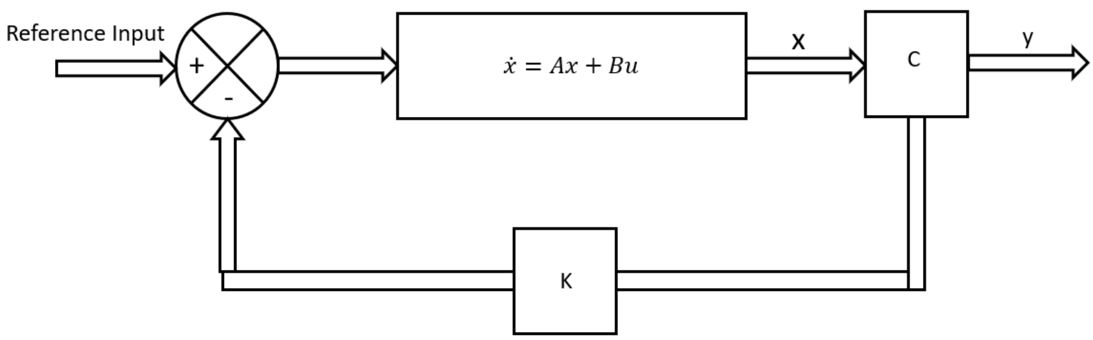


Figure 1: Closed Loop State Diagram

We shall assume the following values for the constants in the ODEs as mentioned in ref [1] for our simulations.

Satellite Parameters	Values	Units
Weight	120	Kg
Inertia Matrix	Ix = 9.8194, Iy = 9.7030, Iz = 9.7309	Kgm^2
Orbit	686 (LEO orbit)	Km
Orbit angular velocity	0.0010764	Rad/sec
Initial roll angle	3	Degrees
Initial pitch angle	1	Degrees
Initial yaw angle	1	Degrees
Initial angular velocities	[0, 0, 0]	Rad/sec

5.1 Results

The state space model made in Simulink can be seen in Fig 2. Here, reference input is the disturbance torque which is assumed to be a Dirac-Delta function. As mentioned in [4], we can get a Dirac-Delta function by taking the derivative of the step input.

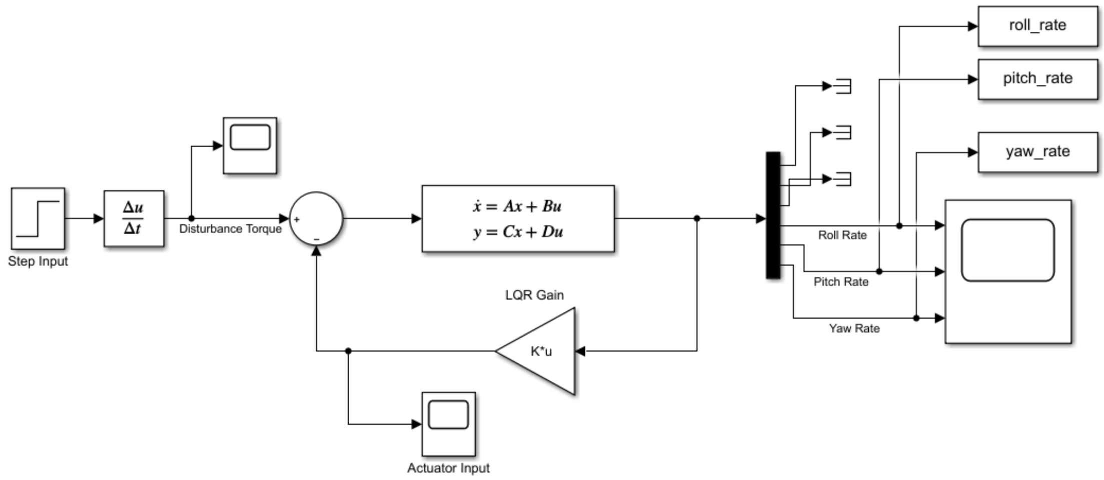


Figure 2: Simulink Block Diagram

q =	1	2	3	4	5	6	7	8
r = 0.1	0.752	0.786	0.799	0.806	0.808	0.809	0.809	0.807
r = 5	0.487	0.531	0.558	0.579	0.594	0.608	0.619	0.628
r = 10	0.446	0.486	0.512	0.531	0.546	0.558	0.569	0.579

Table 1: Rise time (in sec) of roll rate

q =	1	2	3	4	5	6	7	8
r = 0.1	0.827	0.846	0.856	0.862	0.866	0.869	0.872	0.874
r = 5	0.611	0.669	0.697	0.715	0.728	0.738	0.747	0.753
r = 10	0.495	0.611	0.648	0.669	0.685	0.697	0.707	0.716

Table 2: Rise time (in sec) of pitch rate

q =	1	2	3	4	5	6	7	8
r = 0.1	1.343	1.3602	1.335	1.294	1.251	1.229	1.239	1.28
r = 5	3.4e-5	2.7e-5	0.852	0.927	0.981	1.023	1.056	1.085
r = 10	4.2e-5	3.4e-5	2.9e-5	2.7e-5	2.5e-5	0.852	0.892	0.927

Table 3: Rise time (in sec) of yaw rate

q =	1	2	3	4	5	6	7	8
r = 0.1	3.245	3.164	3.109	3.057	3.008	2.965	2.926	2.891
r = 5	9.118	8.252	7.764	7.45	7.178	6.947	6.798	6.667
r = 10	10.089	9.118	8.620	8.252	7.950	7.764	7.601	7.450

Table 4: Settling time (in sec) of roll rate

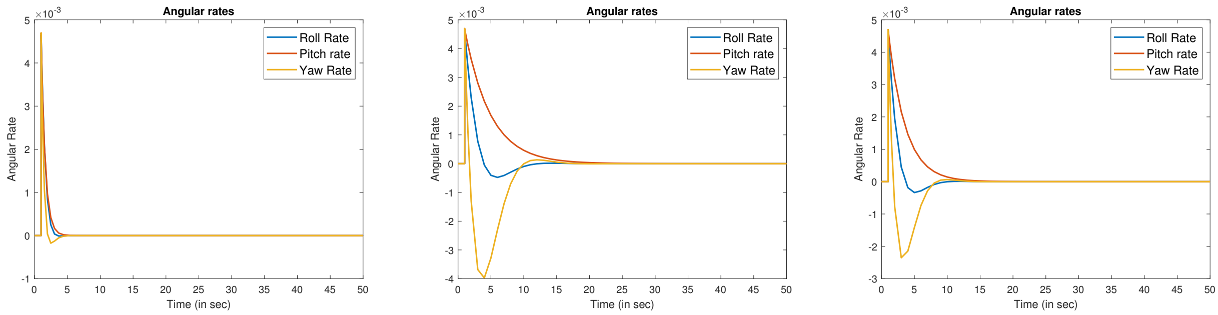
q =	1	2	3	4	5	6	7	8
r = 0.1	5.518	4.781	4.334	3.984	3.812	3.681	3.565	3.465
r = 5	13.771	11.706	10.652	9.928	9.462	9.022	8.741	8.491
r = 10	16.225	13.771	12.528	11.706	11.081	10.652	10.265	9.9278

Table 5: Settling time (in sec) of pitch rate

q =	1	2	3	4	5	6	7	8
r = 0.1	5.173	4.592	4.189	3.920	3.744	3.592	3.456	3.333
r = 5	11.407	7.966	7.641	7.353	7.075	6.910	6.806	6.711
r = 10	13.781	11.407	8.345	7.966	7.796	7.641	7.495	7.353

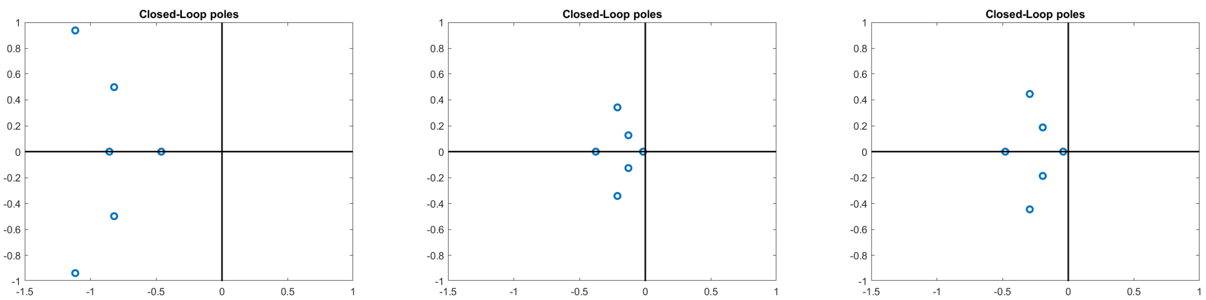
Table 6: Settling time (in sec) of yaw rate

We assume the cost matrix to be of the form $Q = q * I_{6\times6}$ and $R = r * I_{3\times3}$. We determine the lqr gain for 'q' ranging from 1 to 8 and 'r' ranging from 0.5 to 10 and compare the rise time and settling time for all the cases as can be seen in Tables [1] to [6]. After simulating for all the cases, we find that the settling time for roll, pitch and yaw rates is minimum for the case of q = 8 and r = 0.1. The rise time for roll and pitch rates is minimum for q = 1 and r = 10 and the yaw rate is least for q = 5 and r = 10. The time response for these three cases has been shown in Fig 3



(a) (b) (c)

Figure 3: (a) q = 8 and r = 0.1; (b) q = 1 and r = 10; (c) q = 5 and r = 10



(a) (b) (c)

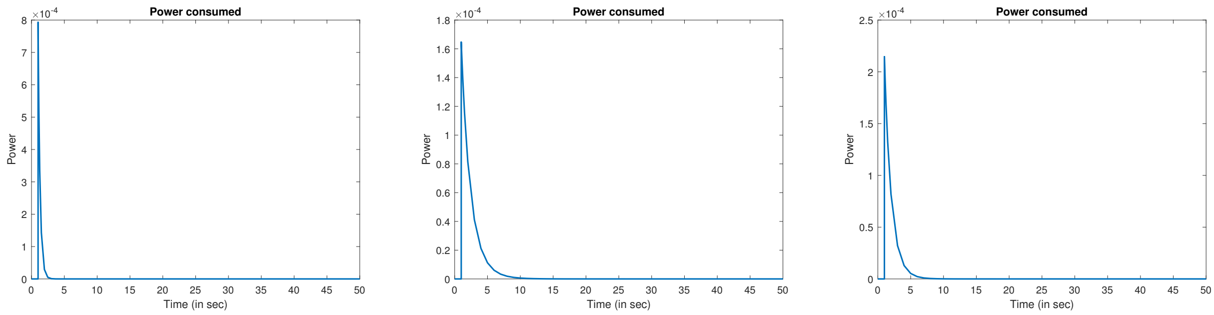
Figure 4: Location of poles on the complex plane for (a) q = 8 and r = 0.1; (b) q = 1 and r = 10; (c) q = 5 and r = 10

A system is said to be asymptotically stable if all the poles of the plant are on the left half of the complex plane. For a closed loop control with linear feedback, the poles are obtained by finding the eigenvalues of the closed-loop dynamics matrix A_c:

$$\text{poles} = \text{eig}(A_c) = \text{eig}(A - BK) \tag{27}$$

where 'eig()' denotes the eigenvalues of the input matrix. As shown in Fig 4, all the poles for the 3 cases of optimal settling time and rise time are located on the left half. Next, we want to calculate the power consumed in these 3 cases. The power at any instant 't' is given as the dot product of Torque applied and angular velocity as shown below:

$$\text{Power(inWatt)} = T.\Omega \tag{28}$$



(a) (b) (c)

Figure 5: Power consumed for applying external torque for (a) q = 8 and r = 0.1; (b) q = 1 and r = 10; (c) q = 5 and r = 10

As seen in Fig 5, the maximum power values for q = 8 with r = 0.1, q = 1 with r = 10 and q = 5 with r = 10 are 0.793 mW, 0.164 mW and 0.214 mW respectively.

6 Conclusion

In this report, we formulated the dynamics of a LEO satellite and formulated a full state feedback control based on LQR as it does not need prior knowledge of the poles of the plant and is applicable to all LTI systems subjected to certain conditions. We iterated through various cost matrix 'Q' and 'R' and investigated the time response of the angular velocities along subjected to an impulse disturbance torque. We also analysed the location of the poles for stability and power consumed by the controller.

For future work, we can examine how the controller handles disturbances such as aerodynamic torque, Solar Radiation Pressure Torque and Gravity-Gradient Torque. We can also the study other linear controllers such as PID, H-infinity controller and compare their performance. We also need to model the sensor noise and uncertainty in actuator and include them as transfer functions in the closed-loop model. Usually, when it is difficult to do so, we integrate robust controllers to handle deviations from the considered plant model.

References

[1] Charles U Eze, Christian C Mbaocha, and James O Onojo. Design of linear quadratic regulator for the three-axis attitude control system stabilization of microsatellites.

[2] Bernard Friedland. *Control system design: an introduction to state-space methods*. Courier Corporation, 2012.

[3] CD Hall. Spacecraft control and dynamics. *AOE4140 Lecture Notes, Aerospace and Ocean Engineering Virginia Tech*, 2002.

[4] Mathworks1. Attitude control system.

[5] OHB. Attitude control system.

[6] Prof. Luoyi Tao. Lecture notes.