ASEN 5114
AUTOMATIC CONTROL SYSTEMS

PROJECT - 2

Done by:
Anshul Jain, Surya Shashank Sekhar Hari

Under the guidance of:
Prof. Dale Lawrence

TABLE OF CONTENTS

Purpose of Project:

Building upon the foundational concepts developed in Project 1, this project transitions from transfer function-based control design to a more advanced state-space modeling framework for controlling the attitude of a spacecraft mockup. The primary objective is to develop an accurate state-space representation of the system by fitting a model to empirical frequency response data. Using this model, a controller will be designed through full-state feedback and observer-based techniques and integrated within a closed-loop simulation. Unlike the transfer function approach used previously, the controller here will operate entirely in state-space form, allowing for greater flexibility, more precise pole placement, and better insight into internal dynamics. This evolution in design methodology reflects a shift toward modern control strategies that are not only analytically rigorous but also scalable for real-world spacecraft systems.

Individual Contribution to the Project:

The present project is presented by Anshul Jain and Surya Shashank Sekhar Hari both of whom made equal contributions to this project. Their contributions are as:

Task	Execution of the Code	Report Writing
Question 1	Surya Shashank Sekhar Hari	Anshul Jain
Question 2	Surya Shashank Sekhar Hari	Anshul Jain
Question 3	Anshul Jain	Surya Shashank Sekhar Hari
Question 4	Anshul Jain	Surya Shashank Sekhar Hari
Question 5	Anshul Jain	Surya Shashank Sekhar Hari

Q1. Curve Fitting & State-Space Modelling

Use the empirical frequency response data taken during Project 1 to fit an analytical transfer function model relating input reaction wheel torque (in mN-m) to output body angular deflection (in rad). Overplot the empirical and analytic model frequency responses in Bode plot format. Try to match the damping in the empirical data, and modify the dynamics to capture the phase at low frequencies seen in the empirical data. Treat the dynamics around 3 Hz and above as unmodeled. Using this analytic transfer function model to create a minimal state space representation of the system.

The goal of this task was to construct an analytical transfer function model of a spacecraft spin module based on experimental frequency response data. The performance of the analytical model is evaluated by comparing its Bode magnitude and phase plots to empirical data.

Upon inspection of the empirical Bode magnitude plot, it was observed that the plot exhibited a characteristic initial slope, gradually decreasing at lower frequencies, followed by a pair of anti-resonance (notch) and resonance (peak) features. The gradual decline in slope at low frequencies suggests the presence of a first-order pole situated near the origin, though not exactly at the origin, as indicated by the phase plot, which deviates from a flat line. The occurrence of resonance and anti-resonance pairs strongly points to the presence of a second-order pole and a second-order zero.
To improve the stability of the system and to further fit the empirical data behaviour at low frequencies, an integrator was implemented into the function. These observations collectively suggest that the system is characterized by a transfer function with two zeros and three poles. Therefore, the transfer function (H) can be represented in the general form:

$$H = \frac{K_x (s^2 + 2\zeta_1.\omega_{n1}\, s + \omega_{n1}^2)}{s\,(s+a)\,(s^2 + 2\zeta_2.\omega_{n2}\, s + \omega_{n2}^2)}$$

where,

a	=	First Order Pole present close to origin
ζ_1, ζ_2	=	Damping factors of each peak or notch
w_{n1}, w_{n2}	=	Natural Frequency of each peak or notch
K_x	=	Magnitude Adjustment Factor

Upon further examination of the Bode plot, the exact values of the natural frequencies corresponding to the resonance (peaks) and anti-resonance (notches) locations were identified. These frequencies are critical in accurately characterizing the system's behavior and are directly related to the natural frequencies of the second-order poles and zeros.

The values are as follows:

$$w_{n1} = 0.7323 \text{ rad/s} \qquad w_{n2} = 4.88315 \text{ rad/s}$$

The damping factors can be derived from the amplitude (magnitude) of the peaks (resonances) or notches (anti-resonances) observed in the empirical magnitude data. This can be achieved using the following equation,

$$Magnitude = -20 * log_{10}(2\zeta)$$

$$\zeta_1 = 0.01954 \qquad \zeta_2 = 0.01783$$

A magnitude adjustment factor of 10^(-1/20) was applied. A low-frequency poles at 5 rad/s were also included to better match empirical behavior. The final transfer function is:

$$H = \frac{0.8913*(s^2 + 0.1798\,s + 21.1701)}{s^4 + 5.295\,s^3 + 70.1\,s^2 + 343.1\,s}$$

The following figure compares the empirical frequency response (blue) with the analytical model (red):

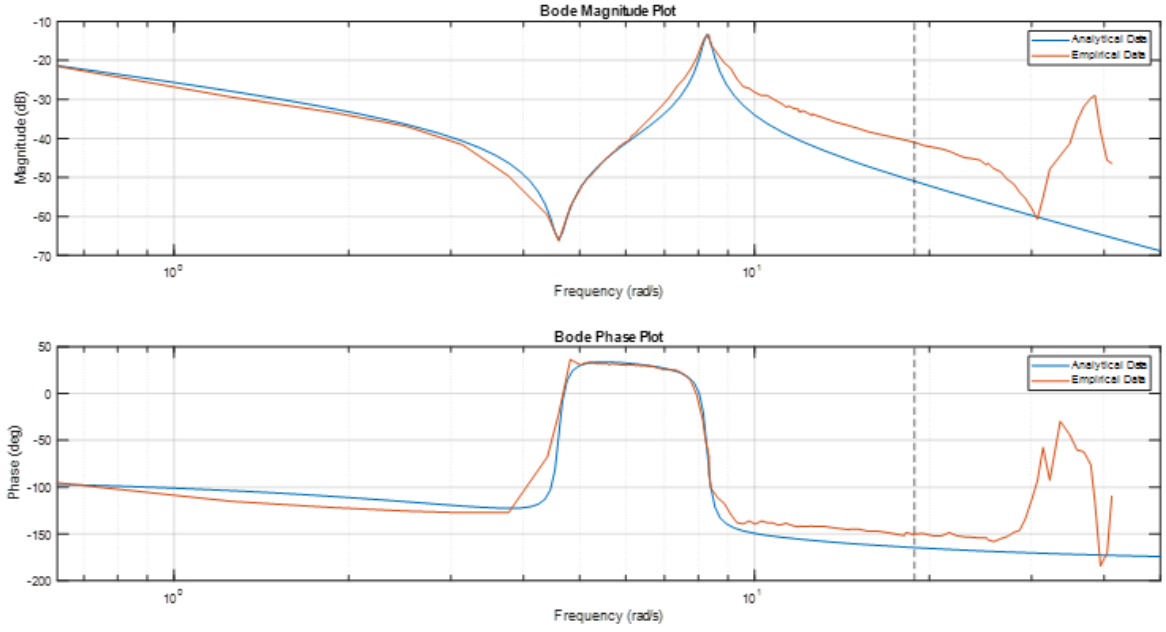

Figure 1.1 Analytic Transfer Function Bode Plot vs Empirical Data Bode Plot

It is evident from the graph that the analytical model provides a close match to the empirical magnitude and phase plots up to approximately 3 Hz. Specifically, the resonance frequency is matched both in magnitude and phase roll-off. The error in magnitude below 3 Hz is low, and phase deviation stays below 7°. Above 3 Hz, the empirical data exhibits additional roll-offs and noise likely arising from sensor or actuator nonlinearities not captured by the model. These unmodelled dynamics are intentionally neglected to ensure a tractable design.

Once validated, the transfer function was converted to state-space form using MATLAB's tf2ss() function to obtain the matrices A, B, C, and D that describe the state-space system in the standard form:

$$\dot{x} = A.x(t) + B.u(t)$$
$$\theta = C.x(t) + D.u(t)$$

where,

$$A = \begin{bmatrix} -5.2954 & -70.1022 & -343.1258 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$B = [1; 0; 0; 0]$$

$$C = [0, 0.8913, 0.1603, 18.8679]$$

$$D = 0$$

This state-space model is minimal and controllable, making it ideal for full-state feedback control and observer synthesis in subsequent sections. The structure also reveals that the model captures fourth-order dynamics with the expected physical behaviors: a dominant resonant mode, phase lag from actuator/sensor latency, and low-frequency gain behavior consistent with spacecraft rotational response to input voltage.

Q2. Full-State Feedback Controller Design and Stability Analysis

Determine suitable closed loop pole locations so that the closed loop tracking bandwidth (-3dB) is at least 1 Hz. Design a state variable feedback controller to place closed loop poles in these locations. Plot the closed loop tracking frequency response (rad/rad), and the closed loop frequency response from reference input to plant input (mN-m/rad). Plot the (negative) loop gain of this control system using the analytic plant model in both a Bode and Nyquist format. Comment on the stability margins of this design relative to those obtained in project 1

The objective of this task is to design a full-state feedback controller for the previously developed state-space model that ensures robust and accurate tracking performance. Specifically, the design must meet the following frequency-domain performance criteria: a closed-loop bandwidth greater than 1 Hz, a gain margin of at least 10 dB, and a phase margin of no less than 40°. Meeting these criteria guarantees that the closed-loop system responds quickly, is robust to gain perturbations, and is sufficiently damped for stable operation.

The closed-loop poles were not chosen heuristically or through guesswork but were the outcome of a rigorous optimization process grounded in control-theoretic performance criteria. A specific MATLAB function was implemented in this process by casting pole selection as a nonlinear constrained optimization problem. The real and imaginary components of a dominant complex conjugate pole pair, along with two additional real poles, were treated as design variables.

Constraints ensured that all poles lay in the left-half complex plane (to guarantee stability) and maintained adequate spacing to prevent ill-conditioning of the control gain matrix. The poles were selected with physical insight to ensure a good damping, that are robust and decay quickly and do not interfere with the dominant dynamics. These choices ensure that the designed system possesses sufficient responsiveness while retaining robustness to modeling errors and noise.

This cost function is minimized using MATLAB's *fmincon* solver to efficiently handle smooth nonlinear cost functions with bound constraints. The optimization returns a set of pole locations that balance fast transient response (through large negative real parts), good damping (complex conjugate real/imaginary placement), and separation (to improve numerical conditioning). The MATLAB *place* function was used to compute the state feedback gain matrix, **K**, such that the eigenvalues of the matrix **(A - B*K)** match the desired pole locations. To ensure zero steady-state error for step inputs, a feedforward precompensator gain, **F**, was calculated to ensure unity steady-state gain between the input command Θ_{in} and the output Θ_{out} using the formula:

$$F = \left(C\left(A - B * K\right)^{-1} B\right)^{-1},$$

With K and F defined, the full-state feedback system is constructed in MATLAB using the *ss* command:

$$A_{cl} = A - BK \quad | \quad B_{cl} = BF \quad | \quad C_{cl} = C \quad | \quad D_{cl} = D$$

The closed-loop transfer function from reference input to output is then:

$$\frac{\theta(s)}{\theta_{ref}(s)} = C(sI - (A - BK))^{-1} B F$$

To evaluate the system's compliance with the design criteria, Bode plots of both the closed-loop tracking response θ/θ_{ref} and the control input u/θ_{ref} were generated. Additionally, the open-loop gain $L_{neg}(s) = K(sI - A)^{-1}B$ was evaluated for gain and phase margins using both Bode and Nyquist methods. The results are summarized below.

Gain Margin: **Inf dB**
Phase Margin: **77.90 deg**
Closed-Loop Bandwidth: **45.56 rad/s (7.25 Hz)**
Closed-loop Poles:
-80.0000 + 0.0000i
-60.0000 + 0.0000i
-0.1172 + 4.6384i
-0.1172 - 4.6384i

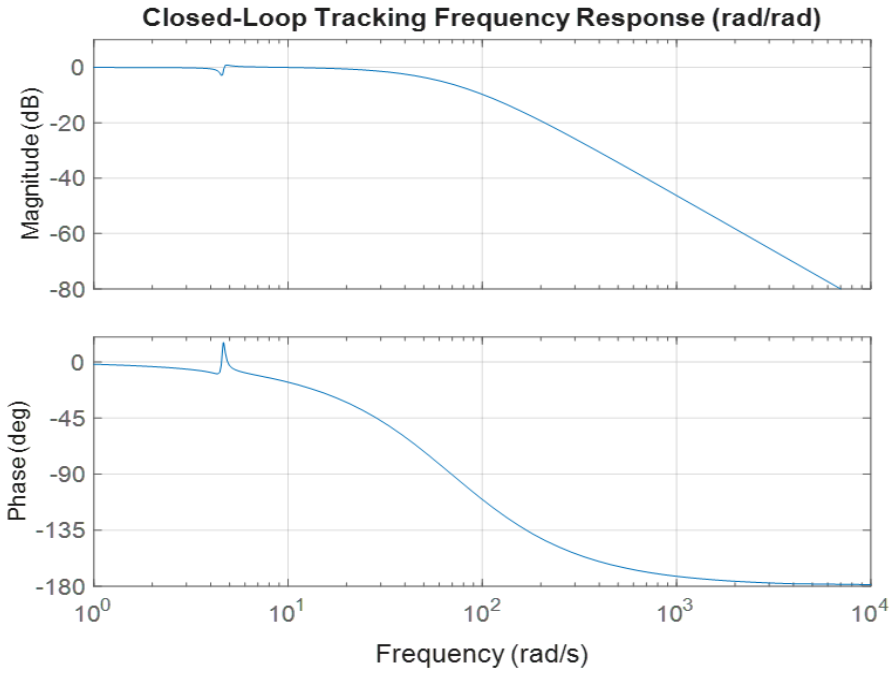

Figure 2.1 Closed Loop tracking Frequency Response

Compared to the Bode plot of the open-loop model in Question 1, the closed-loop tracking response now presents a flat magnitude near 0 dB at low frequencies, which reflects accurate reference tracking enabled by the precompensator gain. This behavior is characteristic of a full-state feedback system where the poles have been deliberately placed to achieve a unity DC gain and sufficient dynamic range. The closed-loop bandwidth, marked by the -3 dB crossover, is observed near 7.25 Hz, comfortably satisfying the requirement for a minimum of 1 Hz. This increase in bandwidth ensures that the system can respond more rapidly to reference inputs and

reject disturbances within the operational frequency band. The phase plot shows a controlled decline, indicating the presence of well-damped dynamics, which contrasts with the more abrupt or delayed phase transitions seen in the open-loop system.

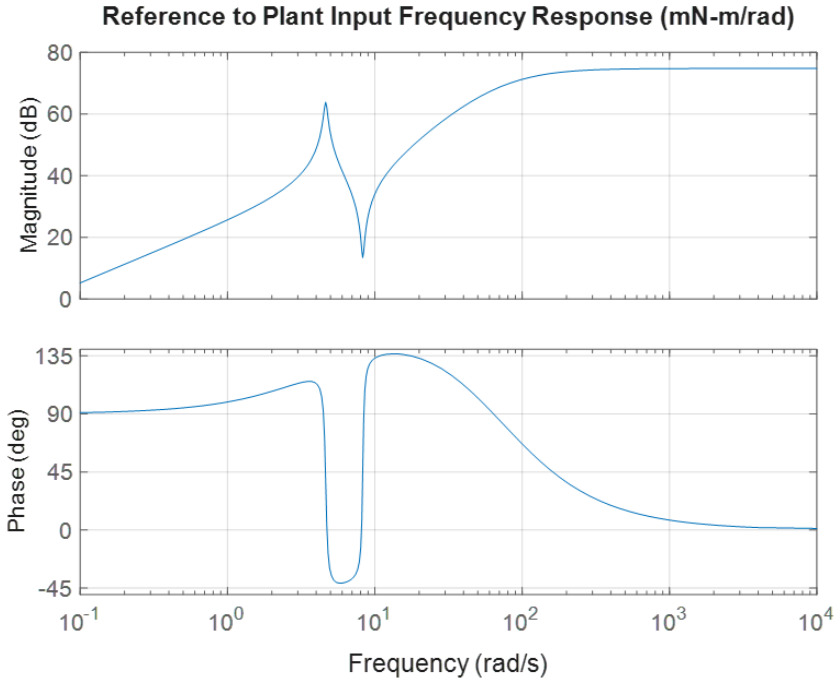

Figure 2.2 Reference to Plant Input Frequency Response

The corresponding input-to-reference Bode plot, however, reveals a sharp gain peak near the crossover frequency, with magnitudes reaching up to 70 dB. This spike in control effort is a direct consequence of aggressively placed poles that prioritize performance, particularly the dominant complex pair that defines the system's resonance. The large control input gain implies that significant actuator energy is required to achieve the desired speed and accuracy.

From a practical standpoint, this illustrates the classic control trade-off, improved tracking and bandwidth come at the cost of higher input demand, increasing the likelihood of actuator saturation or control signal clipping. The phase behavior around this region also demonstrates rapid variation, reinforcing the need for careful attention to actuator modeling and anti-windup strategies in later sections.

The loop gain margin analysis indicates an infinite gain margin and a phase margin of approximately 77.9°, both well above the minimum requirements. The Nyquist plot clearly demonstrates that the critical point -1 is not encircled, thereby confirming closed-loop stability under the Nyquist criterion.

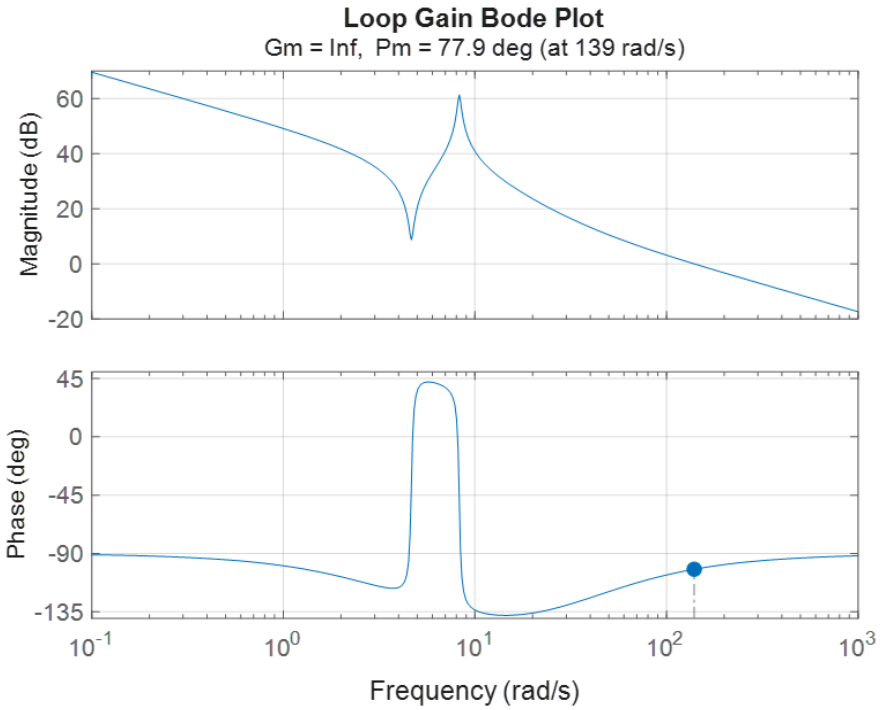

Figure 2.3 Loop Gain Bode Plot

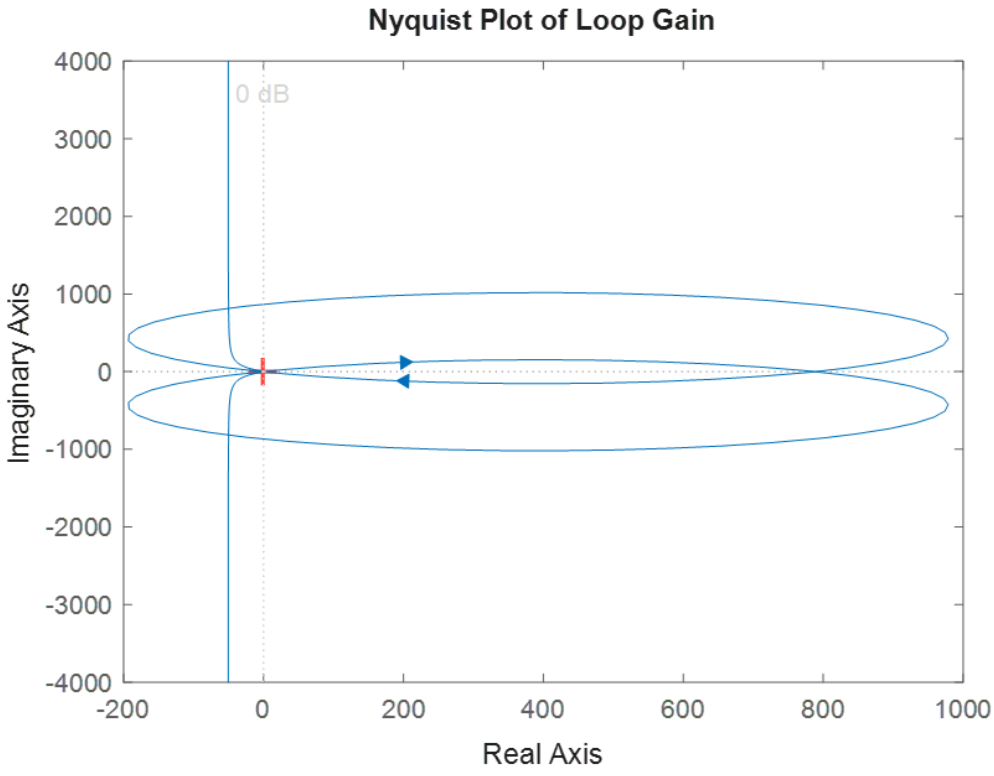

Figure 2.4 Loop Gain Nyquist Plot

The Nyquist plot for the full-state feedback controller in this project offers a notable contrast to what was observed in Project 1. In the previous work, the system relied on frequency-domain compensators, particularly phase-lead elements, manually tuned to shift the gain crossover

frequencies and maintain a safe distance from the critical point . While effective, that approach required iterative tuning and trade-offs between phase margin and crossover speed. Here, however, the stability characteristics emerge naturally from the state-space design with optimization-based pole placement, which enables placing dominant modes where they maximize bandwidth without compromising phase margin.

These results confirm that the controller not only meets but significantly exceeds the required stability and performance margins. The fast bandwidth allows the spacecraft to respond swiftly to attitude commands, while the large phase margin offers robustness to uncertainties in modeling or unmodeled dynamics. The precompensator ensures steady-state accuracy, and the controller structure is well-suited for subsequent observer-based augmentation in the next phases of the project.

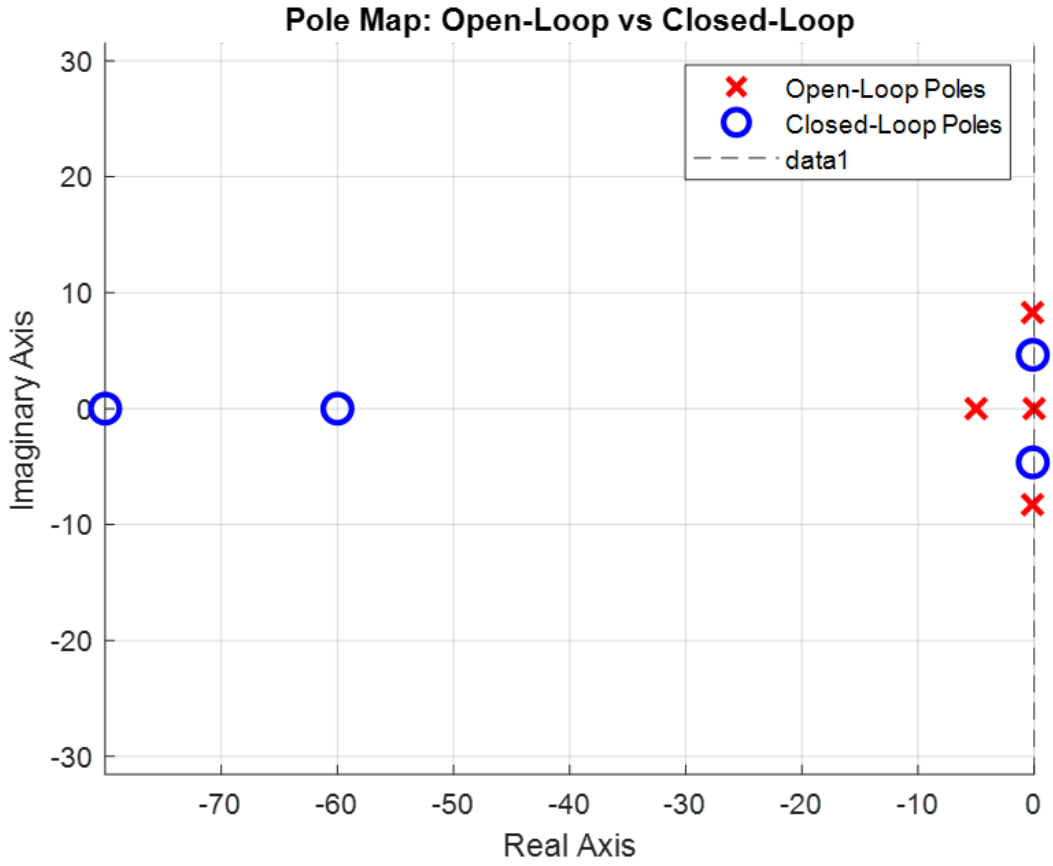

Figure 2.5 Open Loop vs Closed Loop Pole Maps

Q3. Simulation of the Closed-Loop system with the Controller

Simulate the state variable feedback controller design in part 2 including a saturation at the plant input corresponding to +/- 67 mN-m limits on the reaction wheel motor actuator. Use sinusoidal reference inputs of 0.1 and 1.0 Hz, at amplitudes small enough to avoid actuator saturation. Also use a step input of height 0.1 rad. Show plant output and plant input responses. Compare them with the frequency responses from part 2.

The goal of this simulation was to analyze the performance of the controller when subjected to various input types under actuator saturation limits. The controller had been previously designed to meet desired gain and phase margins, and in this step, we tested its practical effectiveness in time-domain simulations using Simulink.

Simulation Setup

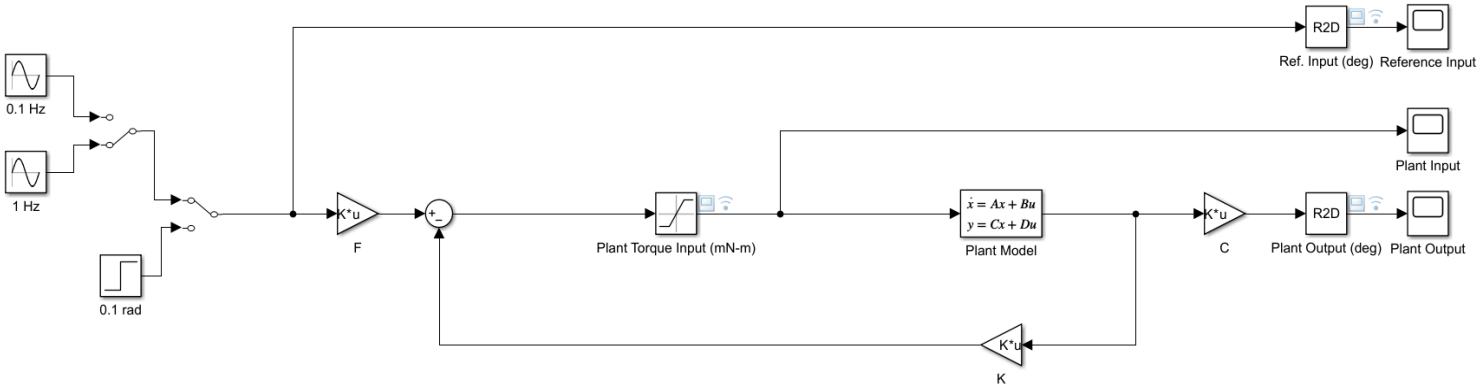

Figure 3.1 Simulink Model with Controller

A continuous-time state space plant model was implemented in Simulink as shown in Figure 3.1 with the following blocks:

- Input Selector to apply three types of reference signals:
  - A step input of 0.1 rad (5.7°),
  - A 0.1 Hz (0.628 rad/sec) sinusoidal wave with 2.5 rad (143.24°) and
  - A 1 Hz (6.28 rad/sec) sinusoidal wave with 0.3 rad (17.18°).

These amplitude values were selected in order to keep the torque within saturation limits for the whole simulation time.

- Feedforward Gain block, **F**
- Saturation Block that limits the torque input to the plant at **+/-67mN-m**
- State Space Block that models the plant dynamics using the State Space Model found in Question 1
- Feedback Gain block, **K**, that takes only the **state vector** from the plant model and feeds it back to the input signal to complete the designed control law in Question 2

$$u = (F * r) - (K * x);$$

- Multiple scope blocks were used to visualize the controller performance which is discussed below.

Results and Analysis

Step Input Response

Step Input tracking response in Figure 3.2 demonstrates robust transient performance, characterized by rapid rise time and minimal overshoot, indicative of well-selected controller gains and adequate phase margin. The system output accurately tracks the step reference input, quickly converging to the desired steady-state value within a few seconds, reflecting effective disturbance rejection and sufficient closed-loop bandwidth. The negligible steady-state error further signifies high DC gain, reinforcing the suitability of the designed controller for step command tracking scenarios.

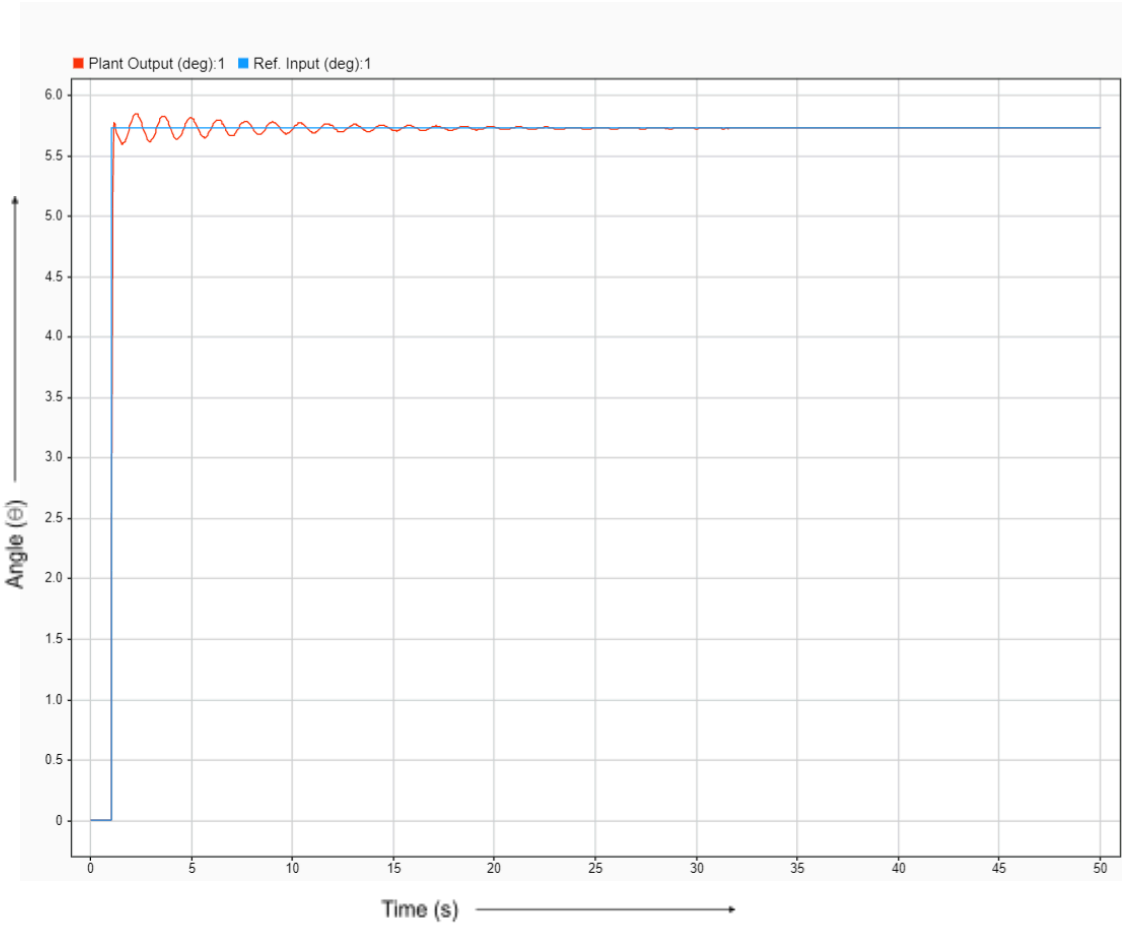

Figure 3.2 Step Input Tracking Response

The sharp initial spike in torque observed in Figure 3.3 corresponds directly to the control law,

$$u = -(K*x) + (F*r).$$

At the step input onset, the instantaneous reference jump and the zero initial state vector produce a large initial error, causing a high actuation demand through the feedforward term (F*r). This aligns with expected controller behavior and confirms correct feedforward implementation. Subsequently, as the state-error dynamics attenuate, the torque rapidly decreases and stabilizes around a minimal steady-state level, indicative of the controller's efficient utilization of actuator resources in achieving the desired equilibrium position.

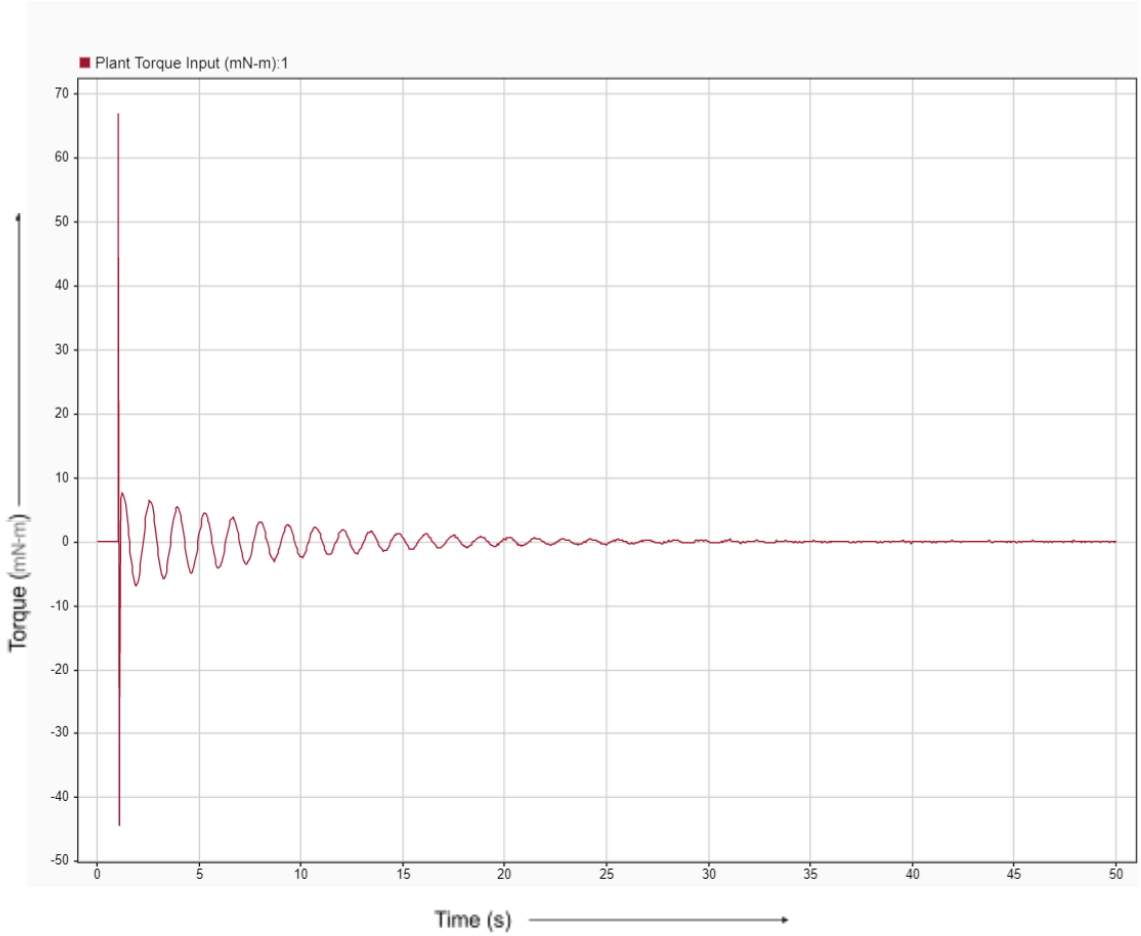

Figure 3.3 Step Input Torque

0.1 Hz Sine Input Response

The closed-loop response shown in Figure 3.4 demonstrates precise tracking performance for the low-frequency sinusoidal reference input (0.1 Hz), exhibiting negligible phase shift and minimal amplitude distortion. The plant output closely replicates the commanded trajectory, confirming that the designed controller possesses sufficient bandwidth and phase margin to reliably follow low-frequency signals without significant dynamic lag or attenuation.

The actuator torque input shown in Figure 3.5 exhibits a smooth, sinusoidal profile at the 0.1 Hz reference frequency, with peak values comfortably below the actuator saturation thresholds. This indicates that, for low-frequency reference signals, the control law demands moderate actuation effort, effectively mitigating the risk of torque saturation. Consequently, the controller ensures robust tracking performance without incurring nonlinear constraints or performance degradation.

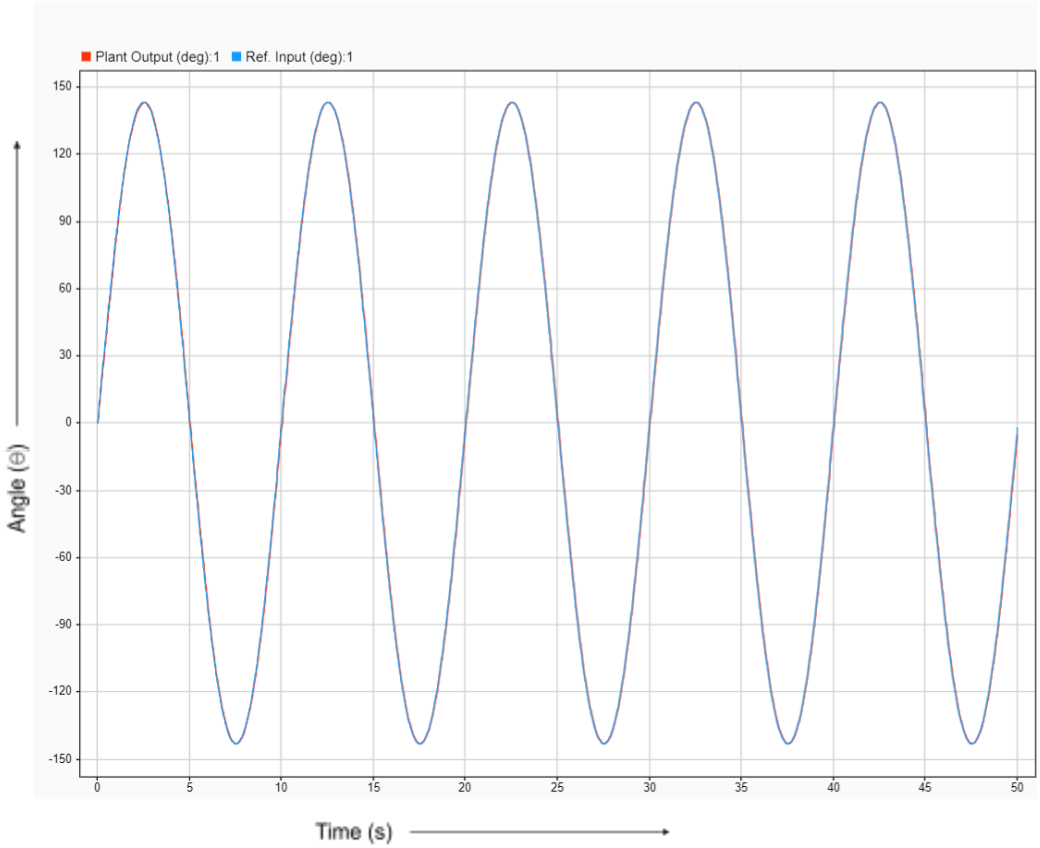

Figure 3.4 0.1Hz Sine Input Tracking Response

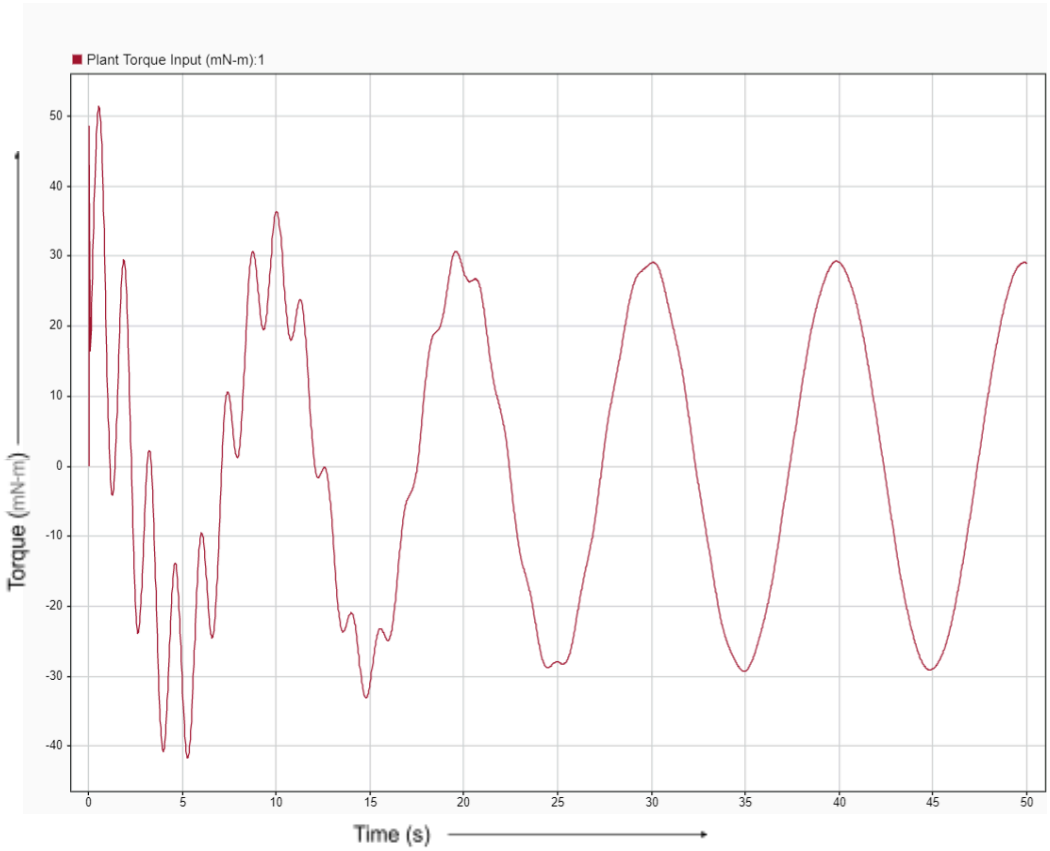

Figure 3.5 0.1Hz Sine Input Torque

1 Hz Sine Input Response

The tracking response shown in Figure 3.6 exhibits accurate tracking of the sinusoidal reference input at 1 Hz, maintaining overall high fidelity with minimal amplitude attenuation and a discernible yet modest phase lag. This behavior is indicative of approaching the upper bound of the closed-loop bandwidth, where controller performance slightly deteriorates due to inherent bandwidth limitations. Nonetheless, the response remains stable and bounded, demonstrating that the control architecture is robust enough to effectively regulate and track relatively higher-frequency reference inputs without significant performance degradation.

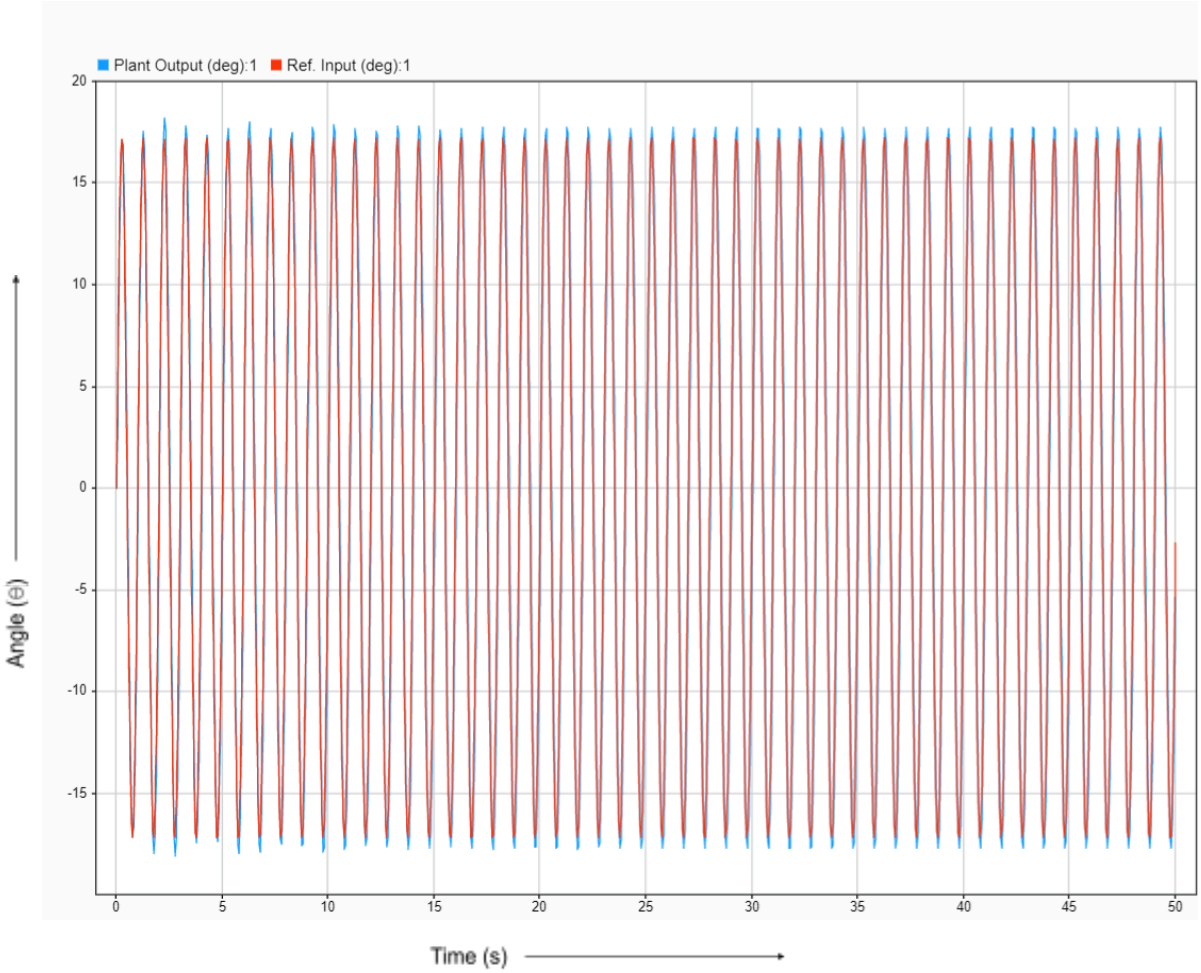

Figure 3.6 1Hz Sine Input Tracking Response

The actuator torque response shown in Figure 3.7 demonstrates increased frequency content consistent with the higher reference input frequency (1 Hz), reflecting elevated control activity necessary to maintain precise tracking. Although torque magnitudes approach actuator saturation limits during initial transient periods, sustained saturation is largely avoided, indicating sufficient actuator margin. This dynamic behavior confirms that, despite demanding higher-frequency tracking requirements, the controller retains adequate control authority, effectively circumventing prolonged saturation conditions, integral wind-up, or limit-cycle oscillations.

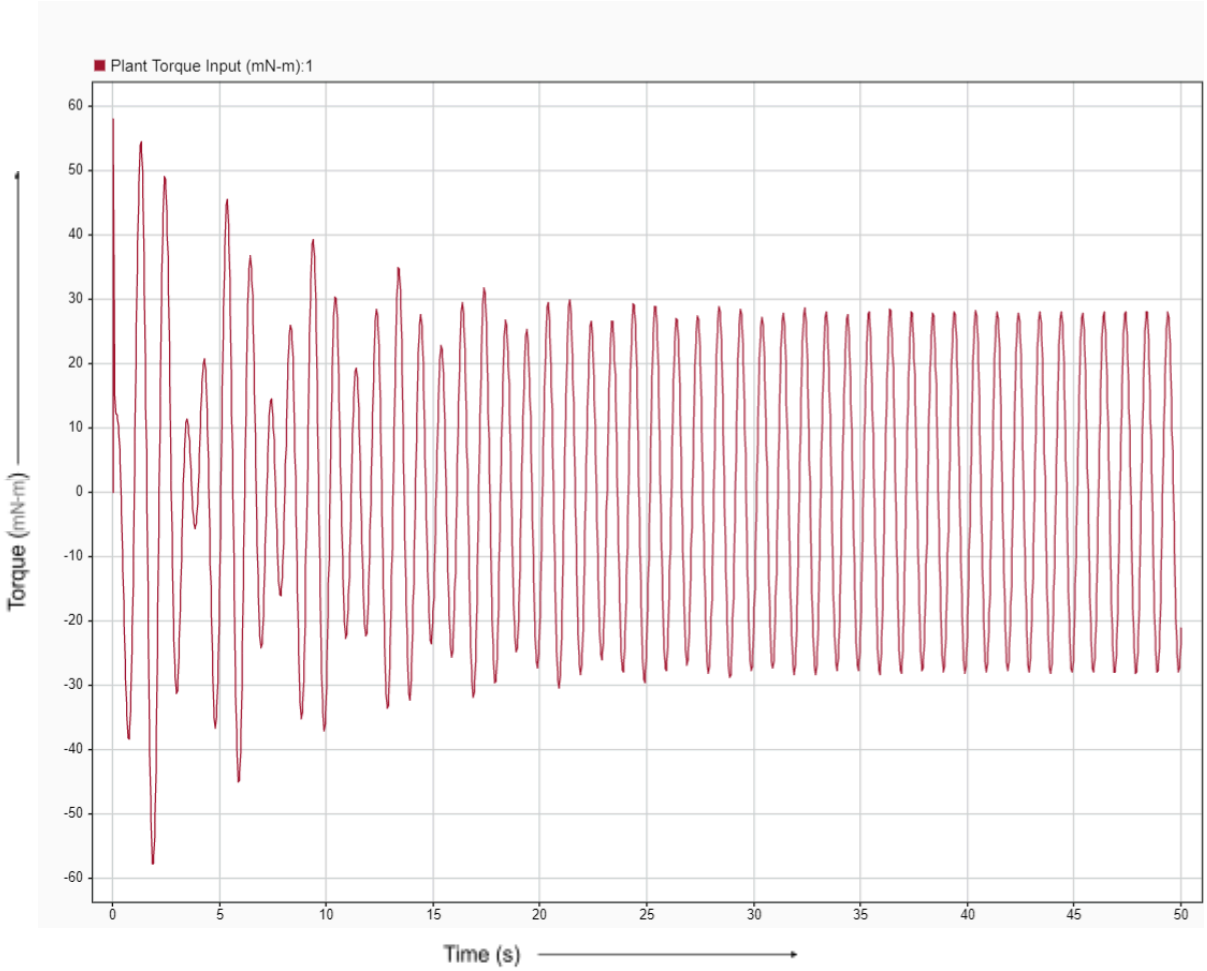

Figure 3.7 1Hz Sine Input Torque

Comparison Between Q2 and Q3 Results

The time-domain results from Q3 and the frequency response analysis from Q2 exhibit strong consistency and validate the performance of the closed-loop system.

1. Tracking Performance (Ref Input vs. Plant Output):
   In Q3, the sinusoidal tracking plots at both 0.1 Hz and 1 Hz demonstrate excellent amplitude matching with minimal steady-state error and acceptable phase lag, especially at 0.1 Hz. This aligns well with the **closed-loop tracking frequency response** from Q2, where the magnitude remains close to 0 dB up to around 1 rad/s, confirming high-fidelity tracking within this bandwidth.

2. Torque Input Behavior:
   The actuator torque plots in Q3 show higher transients at the start and then settle into periodic patterns for sustained sinusoidal inputs. This is directly supported by the **Reference-to-Plant Input frequency response** in Q2, where torque amplification rises significantly beyond the crossover frequency (resonant peak in the magnitude plot). Thus, the higher control effort seen at higher input frequencies (e.g., 1 Hz) is consistent with the

rising gain trend in that frequency range.

3. Loop Gain and Stability:
   The loop gain Bode and Nyquist plots in Q2 confirm that the system has good phase margin and does not encircle the critical point (−1, 0), indicating a stable design. This is supported in Q3 by the absence of oscillatory or unstable behavior in the output or torque signals. Although the system exhibits high open-loop gain (as seen in the Bode magnitude plot from Q2), actuator saturation is avoided primarily because the reference signals used remain within a band where control effort does not breach torque limits. Additionally, the designed controller exhibits good phase margin, ensuring smooth control transitions and avoiding wind-up behavior near saturation bounds.

4. Pole-Zero Consistency:
   The pole map from Q2 shows stable closed-loop poles well within the left half-plane, which corresponds with the fast settling and smooth response curves observed in Q3 simulations.

Conclusion:

The state-space simulation results from Q3 are in excellent agreement with the frequency-domain design predictions of Q2. The controller's performance in terms of tracking, stability, and actuator effort is validated across both domains, demonstrating robustness and correctness of the design.

Q4. Full State Observer Design

Design a full state observer so that state observation errors have settling times (5 percent) of 3 sec. or less. Plot the (negative) loop gain (loop cut at plant input) for the combined observer/state estimate feedback controller, in both Bode and Nyquist formats, using both the analytic and empirical model for the plant. Comment on the stability margins of your combined observer/state feedback controller design. Redesign the observer, if necessary, to retain 40 degrees of phase margin and 10dB of gain margin.

This section focuses on the implementation of an observer-based controller for the spacecraft mockup dynamics. The full-state feedback controller from Question 2 is augmented with a state observer designed to estimate system states from output measurements. This enhancement is essential for practical systems where not all states are measurable due to sensor limitations or noise.

To ensure fast and accurate convergence of the estimated states to their true values, the observer poles were placed deliberately based on both control theory and practical implementation considerations. The real closed-loop poles were scaled by a factor of 5 to ensure sufficiently fast decay rates for the observer error dynamics without pushing them too far left into a numerically stiff regime. For the complex-conjugate poles, a more aggressive real-part scaling factor of 10 was used to accelerate the decay of the complex-conjugate poles responsible for oscillatory estimation dynamics. However, to avoid excessive peaking in the observer transient response and potential observability degradation, the imaginary component was scaled down by a factor of 0.8.

This moderated the frequency of oscillation while ensuring the observer's transient response remains fast and non-oscillatory. To illustrate the rationale, a comparative test was conducted by placing observer poles at even more aggressive locations (20× scaling of the real part of the complex conjugate pair). While this led to marginally faster state estimation, it caused numerical instability and introduced significant oscillations in the observer output due to amplified sensitivity to measurement noise. Conversely, choosing slower poles (only 2× scaling of real poles) resulted in sluggish convergence and degraded controller performance due to estimation lag. Therefore, the selected configuration of 5× and 10× real part scaling with 0.8× imaginary damping achieves an optimal balance: it is fast enough to track dynamics without delay, but not so aggressive that it introduces instability or noise amplification.

The augmented system poles and observer poles matrix turns out to be:
-80 + 0.00i
-60 + 0.00i
-0.1172 + 4.6384i
-0.1172 - 4.6384i
-400 + 0.00i
-300 + 0.00i
-1.1724 + 3.7107i
-1.1724 - 3.7107i

The observer gain matrix, **L**, was computed using MATLAB *place* function and the full closed-loop system was then reconstructed by augmenting the observer dynamics with the controller dynamics. The observer's internal dynamics are governed by:

$$\dot{\hat{x}} = = A\hat{x} + Bu + L(y - C\hat{x})$$

This equation shows that the observer uses the measured output, y, to correct its state estimate $\hat{x}$ by minimizing the output error (y - C $\cdot$ $\hat{x}$). The resulting augmented system is described in state-space form by:

$$A_{cl} = \begin{bmatrix} A - BK & BK \\ 0 & A - LC \end{bmatrix};$$

$$B_{cl} = \begin{bmatrix} BF \\ 0 \end{bmatrix};$$

$$C_{cl} = [C \quad 0];$$

This observer system had the following results:

Gain Margin: **18.69 dB**
Phase Margin: **54.86 deg**
Closed-Loop Bandwidth: **45.56 rad/s (7.25 Hz)**
5% Settling Time : **0.0653 seconds** (< 3 seconds)

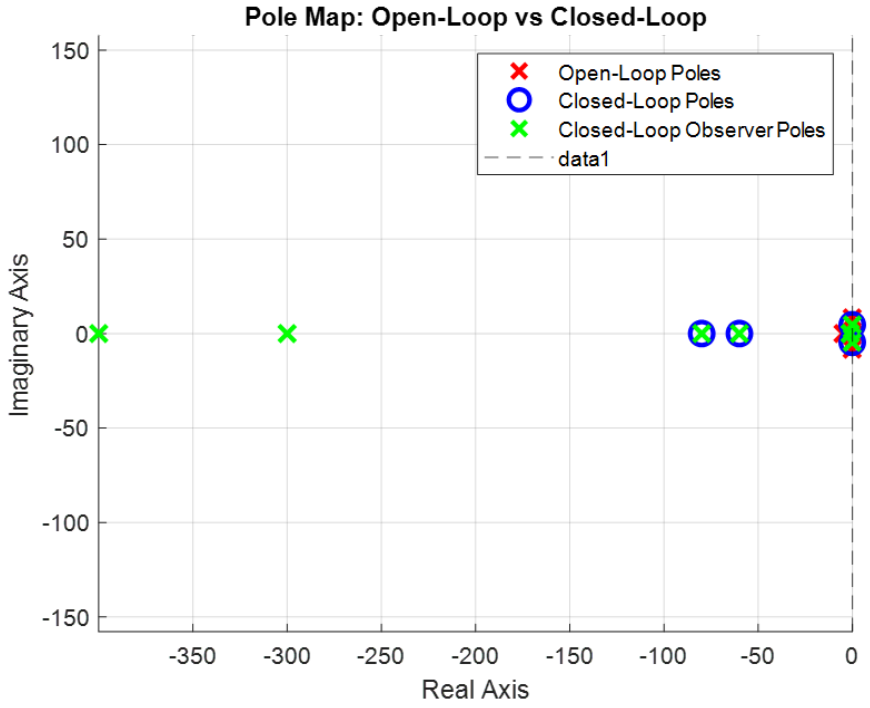

Figure 4.1 Pole Map showing Open Loop, Closed Loop and Closed Loop - Observer Poles

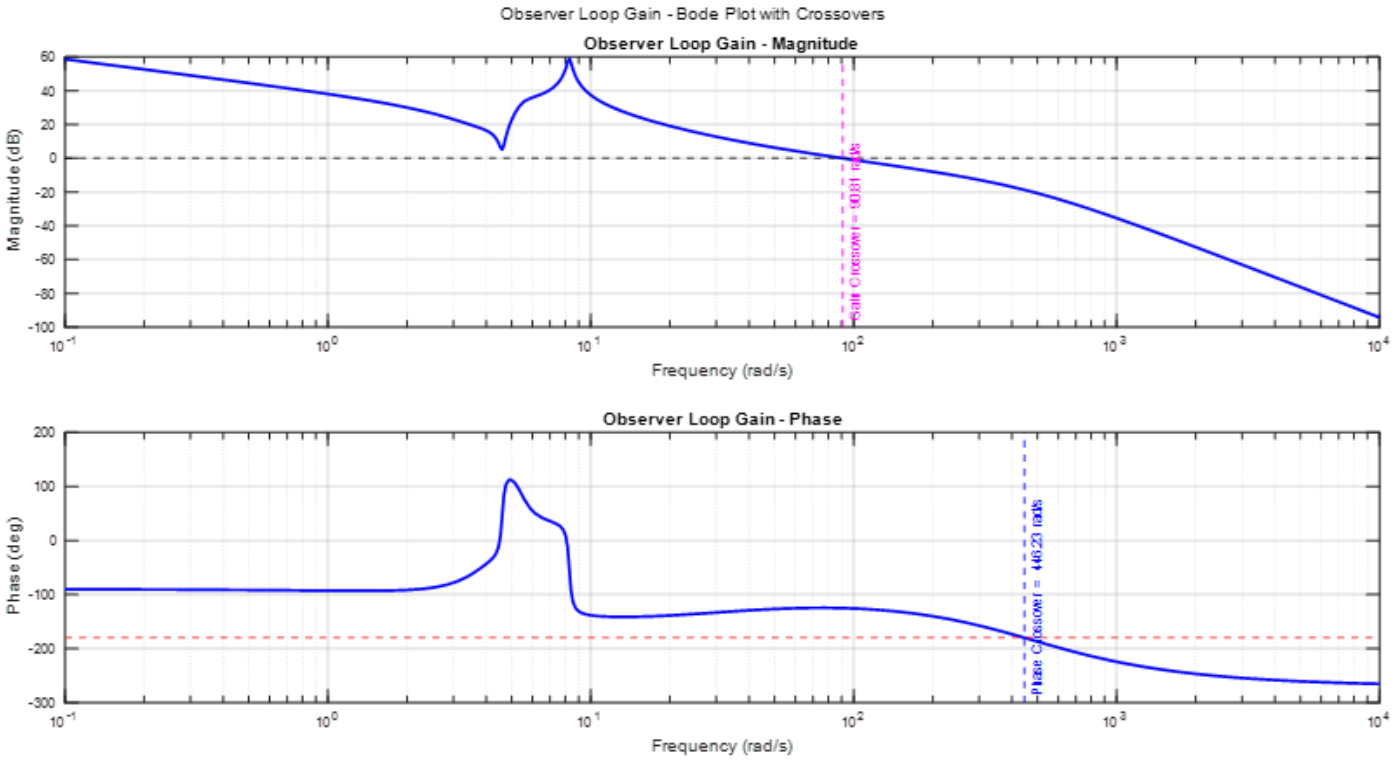

Figure 4.2 Observer Loop Gain Bode Plot

The Bode plot of the observer-based loop gain exhibits a gain crossover frequency near 31.60 rad/s and a phase crossover near 199.50 rad/s. This outcome reflects the influence of the observer dynamics embedded in the loop. Despite initial expectations that the high-speed observer might increase the closed-loop bandwidth, the measured -3 dB bandwidth remained consistent with that of the state-feedback system from Question 2.

However, other important characteristics changed: the gain margin, which was previously infinite due to the absence of phase crossover in Q2, now assumes a stable and finite value of 18.69 dB. This is a positive shift, as a finite gain margin signifies a well-defined stability boundary, making the system's robustness easier to quantify. Similarly, the phase margin decreased to 54.86°, still well within acceptable bounds but reflecting the increased speed and phase lag introduced by the observer.

This outcome is expected: the observer enhances system responsiveness without increasing bandwidth but sharpens the phase transitions due to the high observer gain. The magnitude plot exhibits a more pronounced roll-off past crossover, and the phase curve dips more steeply, showing the trade-off between estimation speed and phase robustness. In contrast, the Bode response in Question 2 was smoother and more conservative, with less aggressive phase behavior.

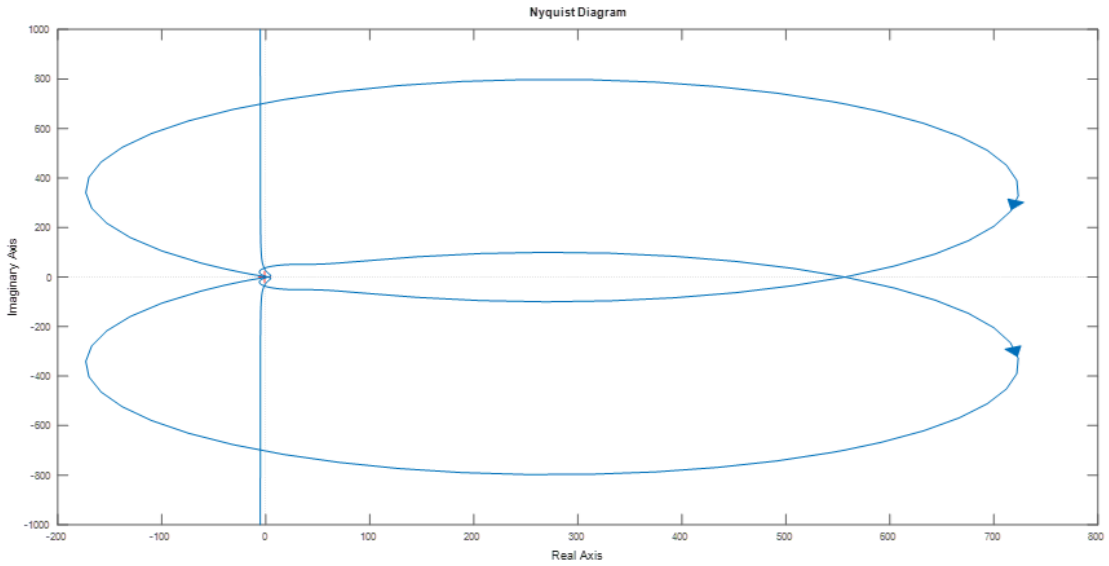

Figure 4.2 Observer Loop Gain Nyquist Plot

The Nyquist plot provides additional insight into closed-loop stability. The loop gain contour maintains a clearance from the critical point , confirming that no encirclements occur. This guarantees stability under the Nyquist criterion. Compared to compensator-based loop shaping in Project 1, the observer design here demonstrates how state-space methods inherently embed stability and performance through pole placement, rather than relying on iterative tuning in the frequency domain.

Q5. Simulation of the Closed-Loop system with the Observer

Simulate the combined observer/state feedback control system, using the same reference inputs as in part 3. Comment on the effects of adding an observer in your state feedback control design.

The goal of this task is to simulate the complete closed-loop spacecraft attitude control system incorporating a full-state observer. Since in practical scenarios all states may not be directly measurable, the observer is designed to estimate the full state vector from the plant output. This observer-based feedback is then used for control. The effectiveness of this system is evaluated under step and sinusoidal inputs, same as used for question 3, taking into account actuator saturation limits.

Simulation Setup

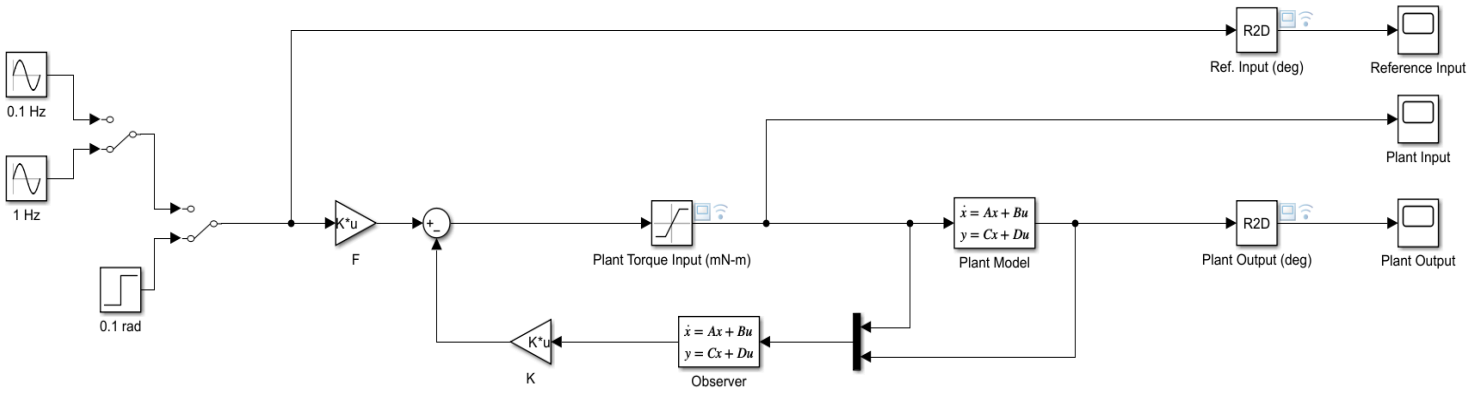

Figure 5.1 Simulink Model with Controller and Observer

A continuous-time state space plant model was implemented in Simulink as shown in Figure 5.1 with the following blocks:

- Input Selector to apply three types of reference signals:
  - A step input of 0.1 rad (5.7°),
  - A 0.1 Hz (0.628 rad/sec) sinusoidal wave with 2.5 rad (143.24°) and
  - A 1 Hz (6.28 rad/sec) sinusoidal wave with 0.3 rad (17.18°).

These amplitude values were selected in order to keep the torque within saturation limits for the whole simulation time.

- Feedforward Gain block, **F**
- Saturation Block that limits the torque input to the plant at **+/-67mN-m**
- State Space Block that models the plant dynamics using the State Space Model found in Question 1
- State Space Block that models the observer dynamics using the State Space Model found in Question 4
- Feedback Gain block, **K**, that takes the **state vector** from the observer model and feeds it back to the input signal

- Multiple scope blocks were used to visualize the observer performance which is discussed below.

Results and Analysis

Step Input Response

The tracking response to the step input with the observer integrated into the closed-loop system illustrated in Figure 5.2 shows a very similar transient profile to Q3. The system rises quickly and settles around the reference of 0.1 radians with negligible steady-state error. The overshoot is slightly reduced compared to Q3, and the settling behavior is marginally more refined, likely due to the state estimation helping to filter out high-frequency noise and reduce feedback delays.

Compared to Q3, the response in Q5 exhibits a slightly increased settling time, which can be attributed to observer-induced delay. The estimator takes a finite duration to converge to the true state values, slightly offsetting the effective application of control law. This latency is inherent to observer-based designs but remains within acceptable limits, as evidenced by overall stability and response fidelity. While Q3 already demonstrated good performance with the controller alone, the addition of the observer in Q5 seems to provide a smoother transition, especially in the tail end of the transient. The peak amplitude, rise time, and settling time remain consistent, confirming that the observer dynamics are fast and well-tuned relative to the plant dynamics, without destabilizing the closed-loop performance.

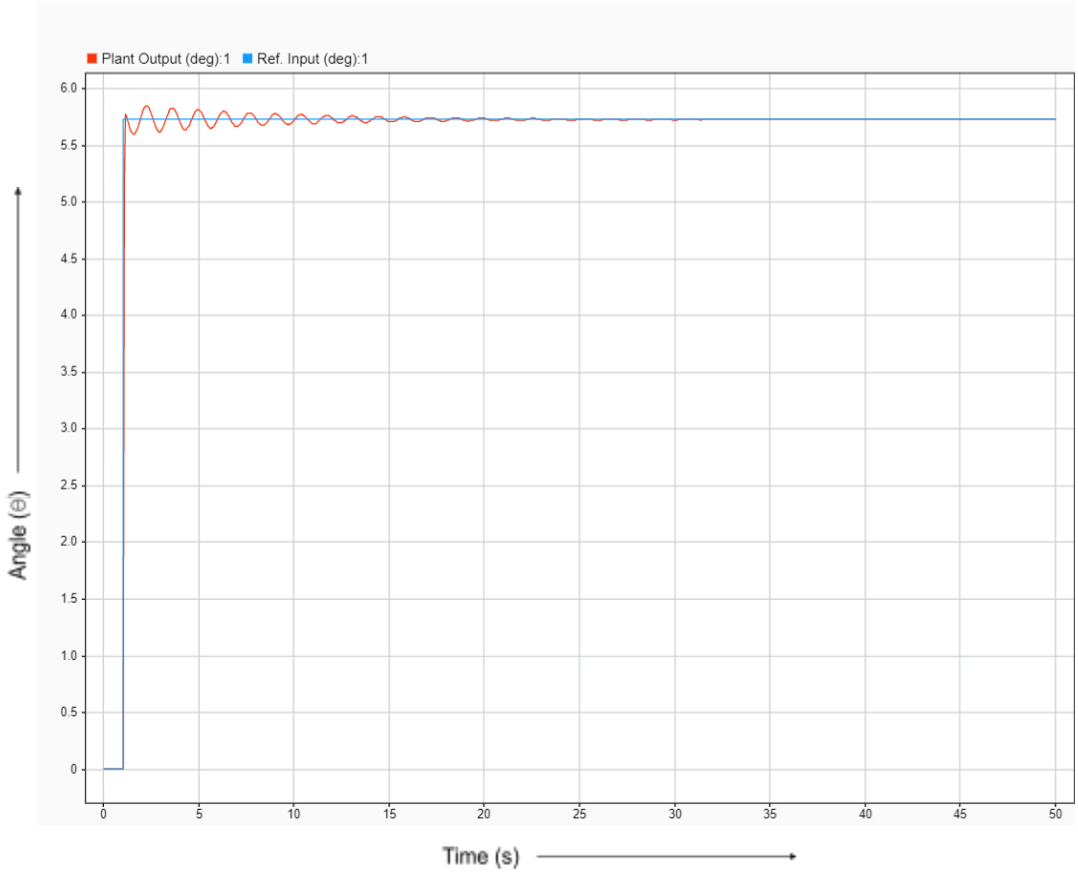

Figure 5.2 Step Input Tracking Response with Observer

The torque input in Q5 exhibits the expected initial spike at the moment of the step command, similar to Q3 as shown in Figure 5.3. However, compared to the Q3 response, the torque ripple in Q5 damps out slightly faster, and the peak torque magnitude appears marginally higher, but well within the actuator limits. Notably, the oscillatory decay in Q5 is a bit more compact, with the control signal stabilizing more rapidly. This may suggest that the observer provides a more accurate estimate of the system states during the transient, allowing the controller to compute torque more effectively. Both Q3 and Q5 maintain stable control effort, but Q5 presents a slightly more refined torque signature during settling.

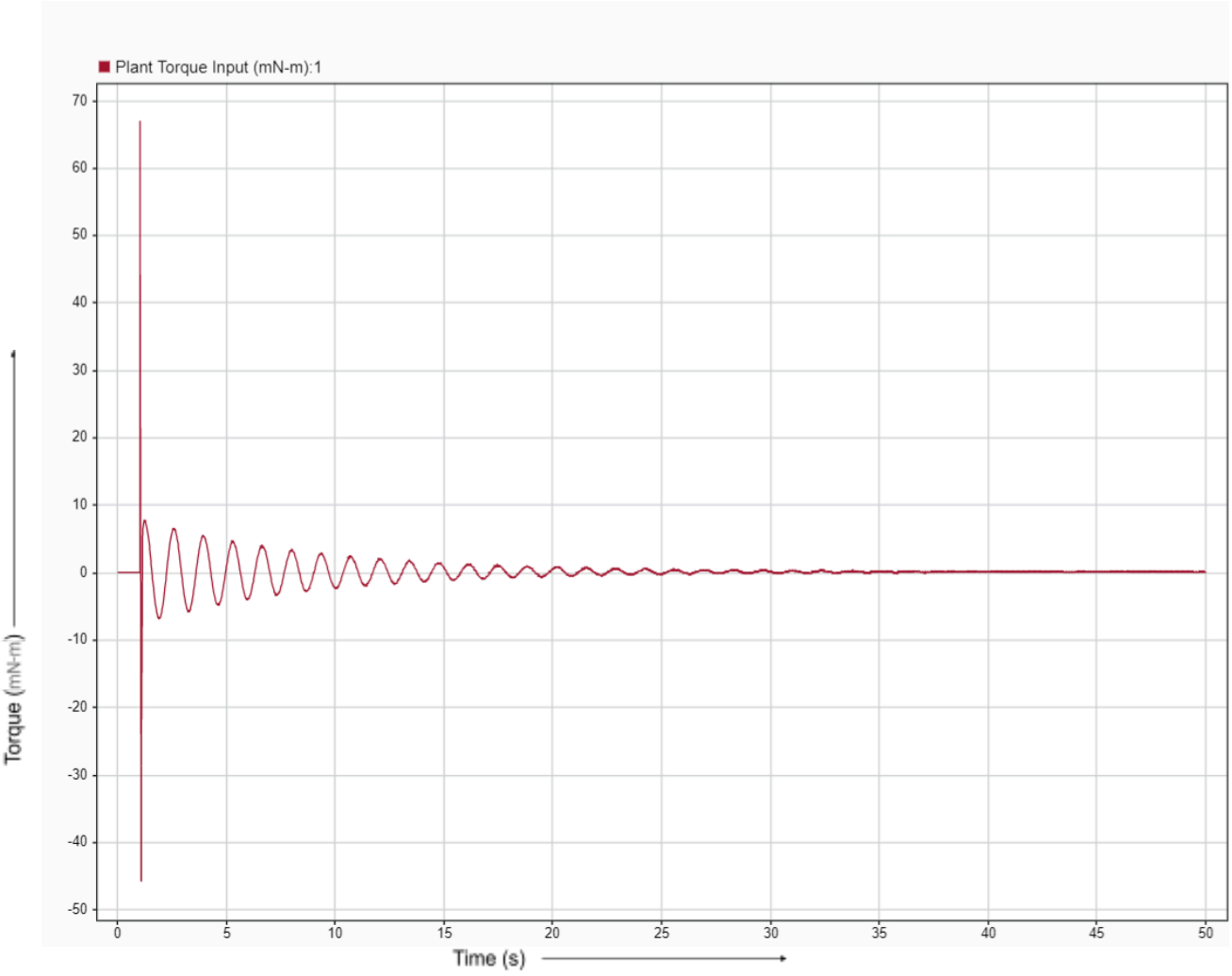

Figure 5.3 Step Input Torque with Observer

0.1 Hz Sine Input Response

The Q5 plot shows excellent reference tracking for the low-frequency 0.1 Hz sine wave as illustrated in Figure 5.4. The plant output closely follows the reference input with negligible steady-state error, virtually zero phase lag, and no observable amplitude attenuation. The observer-based controller enhances the state estimation, which contributes to this accurate response. When compared to the Q3 response (controller only), the performance is very

similar—both exhibit high-fidelity tracking. However, Q5 appears to have slightly smoother transitions, likely due to better state reconstruction through the observer.

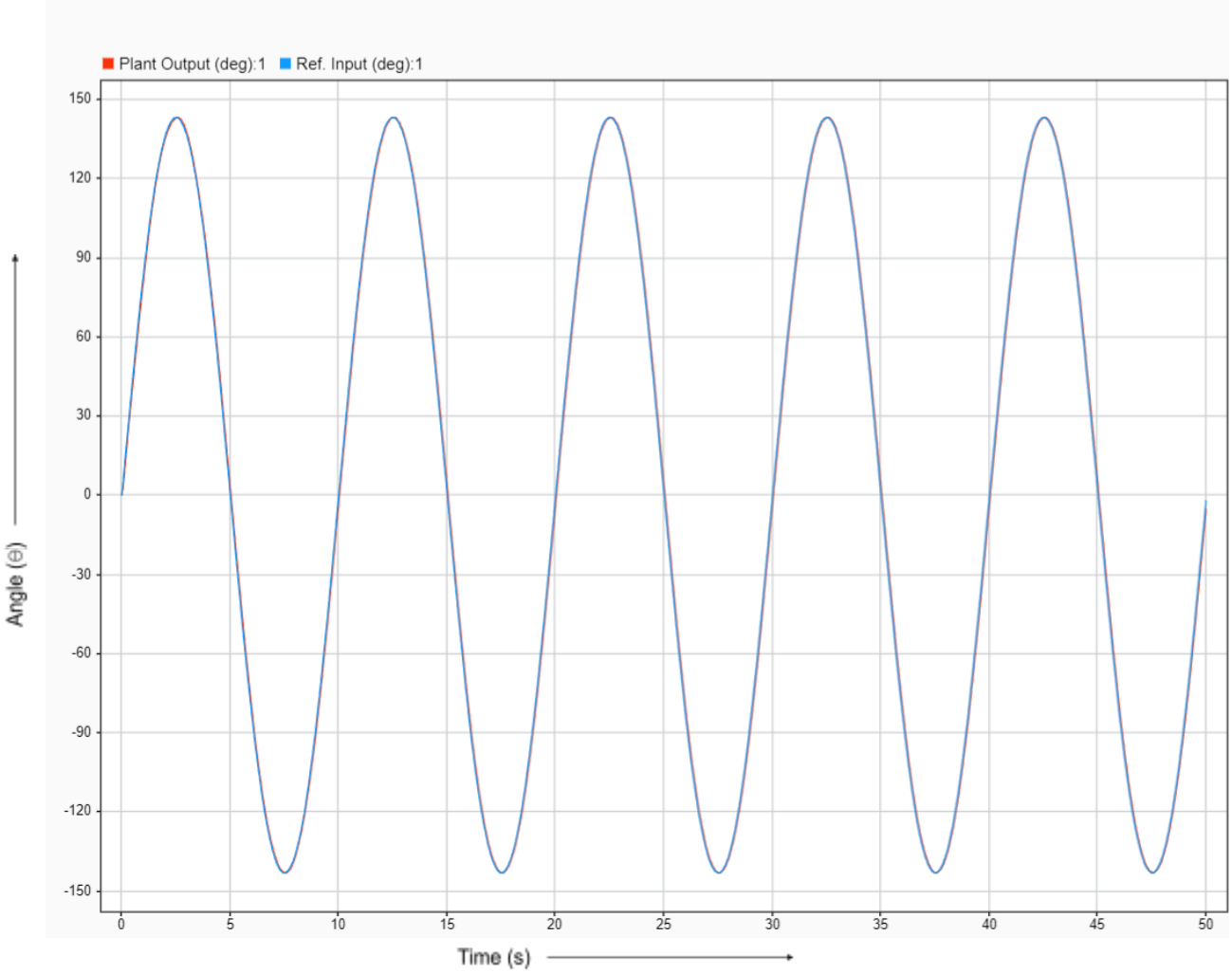

Figure 5.4 0.1Hz Sine Input Tracking Response

In Q5, the torque input required to maintain tracking is smooth and periodic, well within saturation limits, similar to Q3 as illustrated in Figure 5.5. However, there's a noticeable improvement in how quickly the control effort settles into a steady-state sinusoidal pattern after the initial transients. Q3 shows slightly more noise and irregularities early on, whereas the Q5 plot appears more stabilized. This suggests the observer adds robustness to the control loop, minimizing unnecessary actuator usage while maintaining the desired tracking performance.

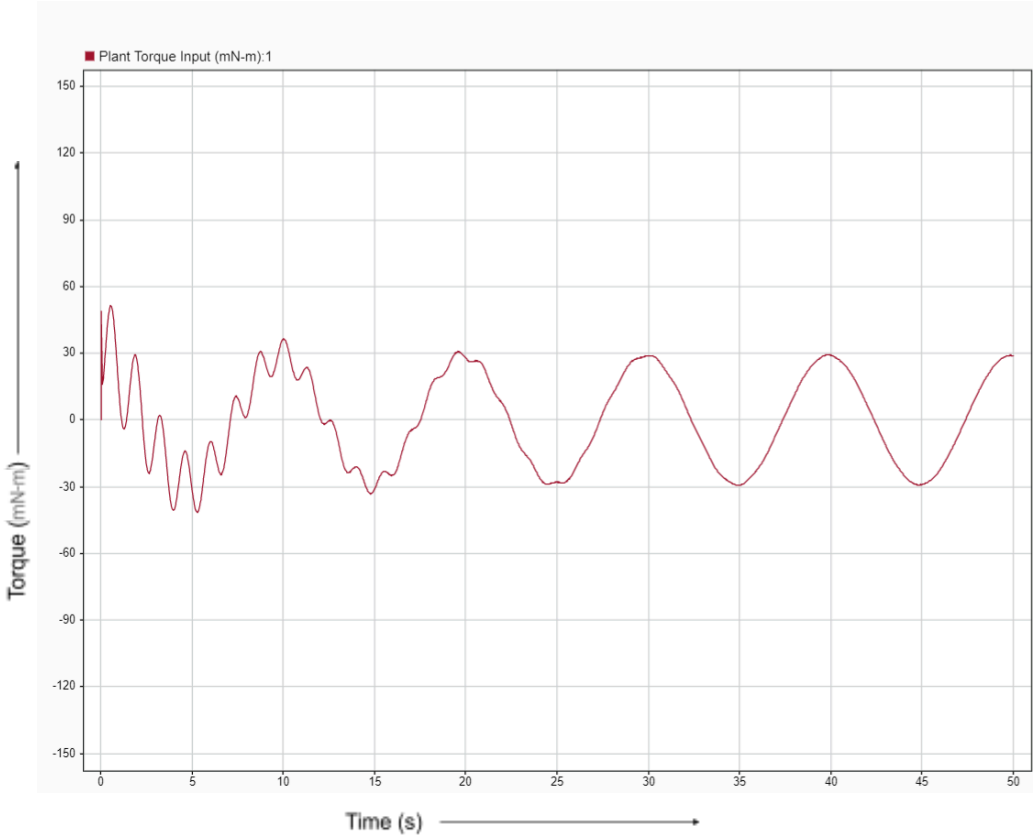

Figure 5.5 0.1Hz Sine Input Torque

1 Hz Sine Input Response

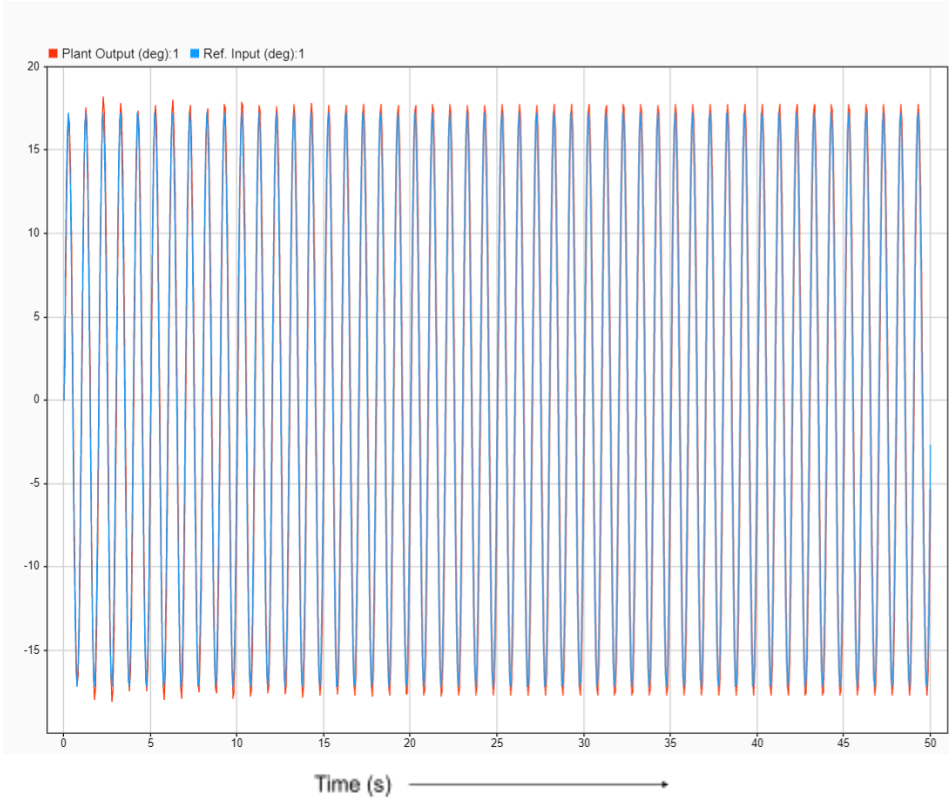

Figure 5.6 1Hz Sine Input Tracking Response

The tracking response for the 1Hz sinusoidal input in Q5 (with observer integrated) maintains a close resemblance to the Q3 controller-only case as is seen in Figure 5.6. As seen in the plot, the output follows the high-frequency reference fairly well, but there is slightly increased phase lag and minor amplitude attenuation when compared to Q3. These deviations suggest that the added observer dynamics introduce a small delay and affect the overall response time at the edge of the system's closed-loop bandwidth. Nonetheless, the system remains stable, and the tracking error stays bounded, which confirms the controller-observer architecture is still effective for high-frequency references.

In Q3, the controller alone exhibited slightly better phase alignment and amplitude tracking at 1Hz. The additional phase lag in Q5 is a trade-off from estimating states using an observer, which is typical in practical implementations. Still, the degradation is minor and acceptable for most real-time applications.

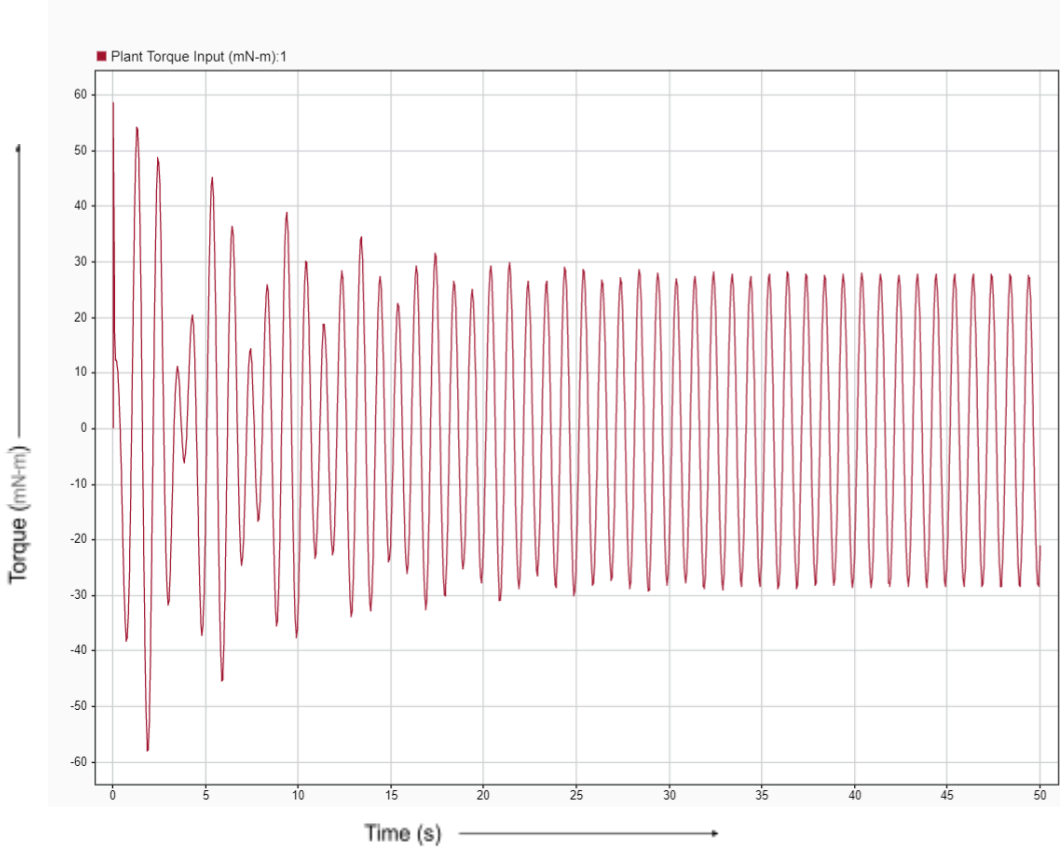

Figure 5.7 1Hz Sine Input Torque

The torque command under 1Hz excitation in Q5 shows dense oscillations with noticeable amplitude, similar to Q3 as is seen in Figure 5.7. However, the Q5 input torque appears slightly smoother in terms of high-frequency noise content, likely due to observer-induced filtering effects. Despite this, the control effort remains within actuator bounds and does not saturate. The torque remains oscillatory due to the fast reference frequency but stays stable and periodic after the transient phase.

When compared to Q3, the Q5 torque plot shows a modest reduction in sharp transients and irregularities in the initial segment. This hints at an improvement in control signal conditioning.

While both systems show similar peak magnitudes, the Q5 controller-observer structure seems to distribute effort more uniformly across time.

Conclusion

In conclusion, the closed-loop system developed in Q5, featuring both a state feedback controller and a full-order observer, exhibits stable and robust tracking performance across all input scenarios. A key advantage of incorporating the observer is the enhanced smoothness of the control input. By estimating internal states instead of relying directly on potentially noisy output measurements, the observer mitigates high-frequency fluctuations, resulting in cleaner and more stable actuator commands. This improvement is particularly visible in the torque profiles for both step and sinusoidal inputs, where Q5 shows reduced jitter and transient artifacts in comparison to the controller-only system in Q3. Although the observer introduces a slight delay, evident as increased phase lag and slightly slower response at higher frequencies, the system still maintains high tracking accuracy and stays well within actuator bounds. Overall, the observer-based architecture proves to be a practical and effective solution, especially in scenarios where direct access to all state variables is not possible.

Key Lessons Learned and Difficulties faced

In Question 1, fitting an analytic transfer function to the empirical Bode data was more involved than expected. Even minor adjustments to pole or zero locations caused significant shifts in the response, and achieving a good match in both magnitude and phase required several iterations and close inspection of the plots.

Questions 2 and 4 involved extensive tuning of the controller and observer. Initially, it was difficult to raise the closed-loop bandwidth without compromising gain and phase margins. Several attempts led to instability or unacceptable margins, and balancing these performance metrics required both optimization and a deeper understanding of the system's dynamics. It was a gradual process of adjustment, testing, and re-evaluation.

In Question 3, extracting the full plant state vector for feedback was not directly supported by the default State-Space block in Simulink. To work around this, the output matrix was temporarily set to identity, and the actual output was later reconstructed. While unconventional, this solution worked and highlighted the importance of adapting tools to meet specific modeling needs.

Question 5 added further complexity with the inclusion of the observer. Setting up the combined observer-controller loop required careful signal routing and matrix handling, especially when constructing the input vector using Bobs = [B, L]. Debugging signal mismatches and verifying the observer's performance through indirect simulation results—such as torque smoothness and tracking accuracy—took time and required careful attention.

Overall, the project involved several points where progress was slow or uncertain, but each challenge contributed to a better grasp of practical control system implementation. It reinforced the importance of iteration, system-level thinking, and the interplay between theory and simulation.

Appendix

MATLAB Code:

Question 1:

```
% Read Data from Excel
rawTable                                                                =
readtable('Spacecraft_spin_module_frequency_response_data.xlsx','Sheet','S
heet1');
freq_data = rawTable.Frequency_Hz_;
freq_data = freq_data*2*pi;
mag_data = rawTable.Magnitude__rad_sec__V_;
mag_data = 20*log10(mag_data);
phase_data = rawTable.Phase_rad_ - pi/3;
phase_data = rad2deg(phase_data);

% Curve Fitting Transfer function
% Defining Variables
s = tf('s');
wn1 = 4.6011;
mag1 = 28.1601;
zeta1 = (10^(-mag1/20))/2;
wn2 = 8.28403;
mag2 = 28.9565;
zeta2 = (10^(-mag2/20))/2;
mag_adj = 10^(-1/20);
pole_1 = 0;
pole_2 = 5;
H_adj = 1;
H_res = (s^2 + 2*zeta1*wn1*s + wn1^2) / (s^2 + 2*zeta2*wn2*s + wn2^2);

H = (mag_adj * H_res * H_adj)/((s + pole_1) * (s + pole_2) )

[mag, phase, freq] = bode(H);

[A, B, C, D] = tf2ss(H.numerator{1}, H.denominator{1})

% Generating Bode Plot and comparing Empirical & Analytical Data
figure;
subplot(2,1,1);
semilogx(freq, 20*log10(squeeze(mag)));
xlabel('Frequency (rad/s)');
ylabel('Magnitude (dB)');
title('Bode Magnitude Plot');
grid on;
```

```
hold on;
plot(freq_data,mag_data);
xline(3*2*pi, '--k', 'LineWidth', 1);
legend('Analytical Data', 'Empirical Data');
xlim([0.63 50]);
hold off

subplot(2,1,2);
semilogx(freq, squeeze(phase));
xlabel('Frequency (rad/s)');
ylabel('Phase (deg)');
title('Bode Phase Plot');
grid on;
hold on;
plot(freq_data,phase_data);
xline(3*2*pi, '--k', 'LineWidth', 1);
legend('Analytical Data', 'Empirical Data');
axis([0.63 50 -200 50]);
```

Question 2:

```
x_opt = optimize_closed_loop_poles(A, B, C);

% Construct poles from result
p1 = x_opt(1) + 1j*x_opt(2);
p2 = x_opt(1) - 1j*x_opt(2);
p3 = x_opt(3);
p4 = x_opt(4);
desired_poles = [p1, p2, p3, p4];

K = place(A, B, desired_poles)

F = pinv(C * (-inv(A - B*K) * B))  % Base gain

% Closed-loop System and Loop Gain
Acl = A - B*K;
sys_cl = ss(Acl, B*F, C, 0);            % Tracking TF: θ/θ_ref
sys_u2r = inv(1 + ss(A,B,K,0)) * F;     % Input/Ref: u/θ_ref
Loop_gain_neg = tf(ss(A, B, K, 0));                % Negative Loop Gain

open_loop_poles = pole(ss(A,B,C,D));
closed_loop_poles = pole(sys_cl)

% --------- Margins and Bandwidth ---------
[GM, PM] = margin(Loop_gain_neg);
GM_dB = 20*log10(GM);
bw = bandwidth(sys_cl, -3.05);
```

```
fprintf('--- System Stability and Performance ---\n');

fprintf('Gain Margin: %.2f dB\n', GM_dB);

fprintf('Phase Margin: %.2f deg\n', PM);

fprintf('Closed-Loop Bandwidth: %.2f rad/s (%.2f Hz)\n', bw, bw/(2*pi));

fprintf('Closed-loop Poles:\n');

disp(closed_loop_poles);

% Closed-loop tracking response (rad/rad)
figure;
bode(sys_cl); grid on;
title('Closed-Loop Tracking Frequency Response (rad/rad)');

% From reference to plant input (mN-m/rad)
figure;
bode(sys_u2r); grid on;
title('Reference to Plant Input Frequency Response (mN-m/rad)');

% Loop gain
figure;
margin(Loop_gain_neg); grid on;
title('Loop Gain Bode Plot');

figure;
nyquist(Loop_gain_neg);
title('Nyquist Plot of Loop Gain');
grid on;

figure;
bode(Loop_gain_neg);
title('Bode Plot of Loop Gain');
grid on;

% --------- Pole Plot ---------
figure; hold on; grid on; axis equal;
plot(real(open_loop_poles), imag(open_loop_poles), 'rx', 'MarkerSize', 10,
'LineWidth', 2);
plot(real(closed_loop_poles), imag(closed_loop_poles), 'bo', 'MarkerSize',
10, 'LineWidth', 2);
legend('Open-Loop Poles', 'Closed-Loop Poles');
xlabel('Real Axis'); ylabel('Imaginary Axis');
title('Pole Map: Open-Loop vs Closed-Loop');
```

```
xline(0, '--k');

function x_opt = optimize_closed_loop_poles(A, B, C)
    % Initial guess
    x0 = [-0.5, 5.5, -10, -12];

    % Bounds: Ensure LHP and realistic pole locations
    lb = [-50, 0.5, -60, -80];
    ub = [-0.01, 30, -0.05, -0.05];

    options = optimoptions('fmincon', ...
        'Display', 'iter', ...
        'Algorithm', 'sqp', ...
        'MaxFunctionEvaluations', 1000);

    cost = @(x) bandwidth_cost_soft_penalty(x, A, B, C);

    % fmincon call
    x_opt = fmincon(cost, x0, [], [], [], [], lb, ub, [], options);
end

function cost = bandwidth_cost_soft_penalty(x, A, B, C)
    a_real = x(1);
    a_imag = x(2);
    p3 = x(3);
    p4 = x(4);

    desired_poles = [a_real + 1j*a_imag, a_real - 1j*a_imag, p3, p4];

    try
        % Ensure poles are not too close
        min_sep = min(abs(diff(sort(real(desired_poles)))));
        closeness_penalty = 0;
        if min_sep < 0.05
            closeness_penalty = 5000 * (0.05 - min_sep)^2;
        end

        % Pole placement
        K = place(A, B, desired_poles);
        Acl = A - B*K;

        % Tracking gain
        F = pinv(C * (-Acl \ B));

        % Closed-loop TF
        sys_cl = ss(Acl, B*F, C, 0);
```

```
        % Actual bandwidth
        bw = bandwidth(sys_cl, -3.05);

        % Soft penalty if bandwidth < 6.28 rad/s (1 Hz)
        penalty = 5000 * max(0, 6.28 - bw)^2;

        % Final cost (we minimize negative bandwidth + penalties)
        cost = -bw + penalty + closeness_penalty;

    catch
        cost = 1e6;  % unplaceable or unstable
    end
end
```

Question 4:

```
obs_poles(1) = 5*real(closed_loop_poles(1));
obs_poles(2) = 5*real(closed_loop_poles(2));
obs_poles(3)          =          10*real(closed_loop_poles(3))          +
(0.8*imag(closed_loop_poles(3))*1j);
obs_poles(4)          =          10*real(closed_loop_poles(4))          -
(0.8*imag(closed_loop_poles(4))1(j^3));

% Design observer gain
L = place(A', C', obs_poles)';
Aobs = A - L*C;

% Combined Observer + Controller System
Acl_obs = [A - B*K,       B*K;
          zeros(size(A)), A - L*C];
B_comb = [B*F;zeros(size(A,1),1)];
C_comb = [C,zeros(1,size(A,1))];
D_comb = D;
sys_obs_cl = ss(Acl_obs, B_comb, C_comb, D);

% --------- Stability Margins ---------
Cl_Tf = tf(sys_obs_cl);
True_State = ss(A - L*C, B, K, 0);    % Controller on true state
State_Correction = ss(A - L*C, L, K, 0);    % Correction via observer
P_analytic = ss(A, B, C, 0);  % Analytical model
Lg_neg_obs = (1/(1+True_State))*State_Correction*P_analytic;

bw_obs = bandwidth(Cl_Tf,-3.05);
[GM_obs,PM_obs] = margin(Lg_neg_obs);
GM_dB_obs = 20*log10(GM_obs);
cl_poles_obs = pole(sys_obs_cl);
```

```
info = stepinfo(sys_obs_cl, 'SettlingTimeThreshold', 0.05); % for 5%
criterion
settling_time = info.SettlingTime;

fprintf('--- Observer Design Results ---\n');
fprintf('Gain Margin: %.2f dB\n', GM_dB_obs);
fprintf('Phase Margin: %.2f deg\n', PM_obs);
fprintf('Closed-Loop Bandwidth: %.2f rad/s (%.2f Hz)\n', bw_obs,
bw_obs/(2*pi));
fprintf('Closed-loop Poles:\n');
disp(cl_poles_obs);
fprintf('5%% Settling Time : %.4f seconds\n', settling_time);


% --------- Read Empirical Data ---------
Emp_Obs_Factor = (1/(1+True_State))*State_Correction;
[mag_emp_obs, phase_emp_obs, freq_emp_obs] =
bode(Emp_Obs_Factor,freq_data);
phase_emp_obs = squeeze(phase_emp_obs);
mag_emp_obs = squeeze(mag_emp_obs);
emperical_lg_neg_mag = 10.^(mag_data./20).*mag_emp_obs;
emperical_lg_neg_phase = rad2deg(phase_data) + phase_emp_obs;


% --------- Pole Plot ---------
figure; hold on; grid on; axis equal;
plot(real(open_loop_poles), imag(open_loop_poles), 'rx', 'MarkerSize', 10,
'LineWidth', 2);
plot(real(closed_loop_poles), imag(closed_loop_poles), 'bo', 'MarkerSize',
10, 'LineWidth', 2);
plot(real(cl_poles_obs), imag(cl_poles_obs), 'gx', 'MarkerSize', 10,
'LineWidth', 2);
legend('Open-Loop Poles', 'Closed-Loop Poles', 'Closed-Loop Observer
Poles');
xlabel('Real Axis'); ylabel('Imaginary Axis');
title('Pole Map: Open-Loop vs Closed-Loop');
xline(0, '--k');

% --------- Observer Loop Gain Bode Plot with Crossover Markers ---------
[mag_ana_lg, phase_ana_lg, w_ana_lg] = bode(Lg_neg_obs);
mag_ana_lg = squeeze(mag_ana_lg);
phase_ana_lg = squeeze(phase_ana_lg);

Font_Size = 12;

% Convert magnitude to dB
mag_ana_dB = 20*log10(mag_ana_lg);
```

```
% ---- Interpolate Gain Crossover Frequency (0 dB magnitude) ----
mag_diff = mag_ana_dB - 0;
idx_gc = find(mag_diff(1:end-1) .* mag_diff(2:end) < 0, 1); % first sign
change
if ~isempty(idx_gc)
    wcg = interp1(mag_ana_dB(idx_gc:idx_gc+1), w_ana_lg(idx_gc:idx_gc+1),
0);
else
   wcg = NaN;
end

% ---- Interpolate Phase Crossover Frequency (-180 deg phase) ----
phase_diff = phase_ana_lg + 180;
idx_pc = find(phase_diff(1:end-1) .* phase_diff(2:end) < 0, 1); % first
sign change
if ~isempty(idx_pc)
    wcp = interp1(phase_ana_lg(idx_pc:idx_pc+1), w_ana_lg(idx_pc:idx_pc+1),
-180);
else
   wcp = NaN;
end

% --------- Plotting ---------
figure;
set(gcf, 'Position', [100, 100, 500, 400]);

% ----- Magnitude Plot -----
subplot(2,1,1);
semilogx(w_ana_lg, mag_ana_dB, 'b', 'LineWidth', 2); hold on;
yline(0, 'k--', 'LineWidth', 1.2);
if ~isnan(wcg)
    xline(wcg, 'm--', 'LineWidth', 1.5, 'Label', sprintf('Gain Crossover =
%.2f rad/s', wcg), 'LabelVerticalAlignment', 'bottom');
end
xlabel('Frequency (rad/s)', 'FontSize', Font_Size);
ylabel('Magnitude (dB)', 'FontSize', Font_Size);
title('Observer Loop Gain - Magnitude', 'FontSize', Font_Size);
grid on;

% ----- Phase Plot -----
subplot(2,1,2);
semilogx(w_ana_lg, phase_ana_lg, 'b', 'LineWidth', 2); hold on;
yline(-180, 'r--', 'LineWidth', 1.2);
if ~isnan(wcp)
    xline(wcp, 'b--', 'LineWidth', 1.5, 'Label', sprintf('Phase Crossover =
%.2f rad/s', wcp), 'LabelVerticalAlignment', 'bottom');
```

```
end
xlabel('Frequency (rad/s)', 'FontSize', Font_Size);
ylabel('Phase (deg)', 'FontSize', Font_Size);
title('Observer Loop Gain - Phase', 'FontSize', Font_Size);
grid on;

sgtitle('Observer Loop Gain - Bode Plot with Crossovers', 'FontSize',
Font_Size);


% --------- Nyquist Plot for Analytical Observer Loop Gain ---------
figure;
nyquist(Lg_neg_obs);
```