MEM 355 Final Report

By Jake Green, Andrew DiDomenico, Nick Morgan, Jared Matlis

Extravehicular activities (EVA), the work completed by an astronaut outside a spacecraft, can be dangerous. In order to mitigate that danger, CosmosXYZ has proposed a controlled system for linear translation and posture control. To consider this system mathematically, the following equations were used:

$$M\ddot{q} + D\dot{q} + Kq = BT_c + Ff \ ; \ q = \begin{bmatrix} \theta \\ \psi \end{bmatrix}$$

The above equation describes the motion of an astronaut free from the effects of gravity, which is what is expected in deep space. The matrices in this equation are defined below with their corresponding units.

$$M = \begin{bmatrix} 60.1429 & 18.5714 \\ 18.5714 & 14.2857 \end{bmatrix} \text{ in } \frac{rad}{s^2}$$

$$D = \begin{bmatrix} 0 & 0 \\ 0 & 2 \end{bmatrix} \text{ in } \frac{rad}{s}$$

$$K = \begin{bmatrix} 0 & 0 \\ 0 & 1000 \end{bmatrix} \text{ in } rad$$

$$B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ in } Nm$$

$$F = \begin{bmatrix} -0.042857 \\ -0.071429 \end{bmatrix} \text{ in } N$$

For analyzing this system, the following assumptions were made:

- The backback is considered two rigid bodies connected by a hinge at the hip joint where k is the spring restraint for the hinge and c the viscous damping.
- T_c is the torque applied from our control in the form of a reaction wheel, with f as the force from the thruster
- The pitch of the torso will be measured in radians (θ) by a dedicated sensor

schematic

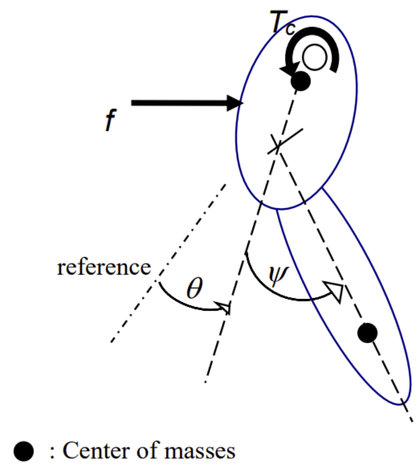

Figure 1: Schematic of Proposed System [1]

1. The states being used to represent this system as well as their first derivatives are defined below, first in an abbreviated form and second in its full form (after substituting out q and $\dot{q}$).

$$x = \begin{bmatrix} q \\ \dot{q} \end{bmatrix} \qquad \dot{x} = \begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix}$$

$$x = \begin{bmatrix} \theta \\ \psi \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \qquad \dot{x} = \begin{bmatrix} \dot{\theta} \\ \dot{\psi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix}$$

Since the state space form is $\dot{x} = Ax + Bu$, the equation of motion needs to be rearranged so as to isolate the $\ddot{q}$ term.

$$\ddot{q} = -M^{-1}Kq - M^{-1}D\dot{q} + M^{-1}BT_C + M^{-1}Ff$$

Utilizing the trivial equation $\dot{q} = \dot{q}$, the state space representation of this system can be determined.

$$\begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}D \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ M^{-1}B & M^{-1}F \end{bmatrix} \begin{bmatrix} T_C \\ f \end{bmatrix}$$

After substituting in each matrix, this can be expanded into the full state space representation.

$$\begin{bmatrix} \dot{\theta} \\ \dot{\psi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 36.111 & 0 & 0.0722 \\ 0 & -116.944 & 0 & -0.0234 \end{bmatrix} \begin{bmatrix} \theta \\ \psi \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.0278 & 0.00139 \\ -0.0361 & -0.00681 \end{bmatrix} \begin{bmatrix} T_C \\ f \end{bmatrix}$$

Finally, since the only measured quantity is θ, the measurement matrix will take the following form.

$$z = Mx$$

$$z = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \psi \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}$$

2.

a. The eigenvalues and eigenvectors of the A matrix were calculated using MATLAB.

```
[V,D] = eig(A)
```

Figure 2: MATLAB Code for calculating Eigenvalues and Eigenvectors of A Matrix

$$\lambda = \{\, 0,\ 0,\ -0.0117 \pm 10.8i\}$$

$$E(\lambda) = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}; \begin{bmatrix} -1 \\ 0 \\ 0 \\ 0 \end{bmatrix}; \begin{bmatrix} -0.0005 + 0.0272i \\ -0.001 - 0.088i \\ -0.2938 - 0.0057i \\ 0.9541 \end{bmatrix}; \begin{bmatrix} -0.0005 - 0.0272i \\ -0.001 + 0.088i \\ -0.2938 + 0.0057i \\ 0.9541 \end{bmatrix}$$

This system is asymptotically stable due to the eigenvalues having either negative or zero real components. This means they fall on the left hand side or along the imaginary axis of the root locus graph.

b. The following are equations for the frequency (ω_n) and damping ratio (ζ) of the system.

$$\omega_n = \sqrt{\lambda_R^2 + \lambda_I^2} \qquad \zeta = \frac{\lambda_R}{\omega_n}$$

The MATLAB calculated eigenvalues can be broken up into real and imaginary parts, in the above equation denoted λ_R and λ_I, respectively. Utilizing this fact, we are able to calculate both the frequency and the damping ratio of the system.

$$\omega_n = \sqrt{-0.0117^2 + 10.8^2}$$
$$\omega_n = \sqrt{0.0013689 + 116.64}$$
$$\omega_n = \sqrt{116.6413689}$$
$$\omega_n = 10.80063373 \approx 10.8$$

$$\zeta = \frac{\lambda_R}{\omega_n} = \frac{-(-0.0117)}{10.8} = 0.00108333 \approx 0.00108$$

3. Before designing a controller and observer, the controllability and observability of the system needs to be checked. This is done by verifying that the controllability and observability matrices -- defined below -- have full rank.

$$C_M = \begin{bmatrix} B & AB & A^2B & A^3B \end{bmatrix}$$
$$O_M = \begin{bmatrix} M \\ MA \\ MA^2 \\ MA^3 \end{bmatrix}$$

The controllability matrix will change depending on the inputs.

For (a) both torque and force inputs, the matrix B becomes:

$$(a) \qquad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.0278 & 0.00139 \\ -0.0361 & -0.00681 \end{bmatrix}$$

For (b) only the torque input, the matrix B becomes:

$$(b) \qquad B = \begin{bmatrix} 0 \\ 0 \\ 0.0278 \\ -0.0361 \end{bmatrix}$$

For (c) only the force input, the matrix B becomes:

$$(c) \qquad B = \begin{bmatrix} 0 \\ 0 \\ 0.00139 \\ -0.00681 \end{bmatrix}$$

By using MATLAB, the controllability matrices and observability matrix can be determined, and thereafter the ranks can be calculated.

```
C = vpa([B, A*B, A^2*B, A^3*B]);
C_Tc = vpa([B_Tc, A*B_Tc, A*A*B_Tc, A*A*A*B]);
C_f = vpa([B_f, A*B_f, A*A*B_f, A*A*A*B]);

rank(C)
rank(C_Tc)
rank(C_f)

M_s = [1, 0, 0, 0];
O = [M_s; M_s*A; M_s*A*A; M_s*A*A*A];
rank(O)
```

Figure 3: C and M matrix rank MATLAB Code

The output for each of these ranks is 4, and since this is full rank for 4x4 matrices, the system will be observable and controllable in scenarios (a), (b), and (c). The observability matrix, calculated with respect to θ, the measurement of the position sensor, also had a full rank of 4. Due to this, the system is observable from the position sensor.

4. For the design of the controller and observer, the following conditions were considered. First, the force (f) is to be ignored, so as to simplify the calculations. Second, the only actuator for this design should be the torquer. Third, the design must have a settling time of ten seconds, maximum overshoot of 21%, torque limit of fifteen Nm, and a set point of 30 °. The latter is the angle at which the torso should remain during operation.

a. The desired percent overshoot and settling time for the full-state feedback system are $\%OS = 21\%$ and $T_s = 10\ s$. With these two parameters, the damping ratio and natural frequency can be solved.

$$\zeta = \frac{\ln(\%OS)}{\sqrt{\pi^2 + \ln^2(\%OS)}}$$

$$\zeta = \frac{\ln(0.21)}{\sqrt{\pi^2 + \ln^2(0.21)}}$$

$$\zeta = 0.445$$

$$T_s = \frac{4}{\omega_n \zeta}$$
$$\omega_n = \frac{4}{\zeta T_s}$$
$$\omega_n = \frac{4}{0.445 * 10}$$

$$\omega_n = 0.899$$

Using the general form for a second-order system, the desired characteristic equation will have two poles that satisfy the equation:

$$s^2 + 0.8s + 0.808 = 0$$

However, the full desired characteristic equation of the system will be of fourth order, taking the form:

$$(s+a)(s+b)\left(s^2 + 0.8s + 0.808\right) = 0$$

where a and b will be an additional two poles that will be chosen arbitrarily. Next, define the control matrix K such that:

$$K = \begin{bmatrix} k_1 & k_2 & k_3 & k_4 \end{bmatrix}$$

The characteristic equation of the full-state feedback control system will then be:

$$\det(sI - (A - BK)) = 0$$

$$s^4 + (-0.0361k_4 + 0.0278k_3 + 0.234)s^3 + (0.0039k_3 - 0.0361k_2 + 0.0278k_1 + 117.0)s^2 \\ +(1.95k_3 + 0.0039k_1)s + 1.95k_1$$

To solve for the k values, this characteristic equation is set equal to the desired characteristic equation. By comparing like terms, a system of equations can be established.

$$\begin{bmatrix} -0.0361 & 0.0278 & 0 & 0 \\ 0 & 0.0039 & -0.0361 & 0.0278 \\ 0 & 1.95 & 0 & 0.0039 \\ 0 & 0 & 0 & 1.95 \end{bmatrix} \begin{bmatrix} k_4 \\ k_3 \\ k_2 \\ k_1 \end{bmatrix} = \begin{bmatrix} 0.8 + a + b - 0.234 \\ ab + 0.8(a+b) + 0.808 - 117.0 \\ 0.8ab + 0.808(a+b) \\ 0.808ab \end{bmatrix}$$

Values of a and b were chosen arbitrarily as $a = 10$ and $b = 5$. After substituting these values into the matrix above, the system was solved in MATLAB and the values of k were found to be:

$$\begin{bmatrix} k_4 \\ k_3 \\ k_2 \\ k_1 \end{bmatrix} = \begin{bmatrix} -410.6 \\ 26.7 \\ 1520.0 \\ 20.7 \end{bmatrix}$$

To verify that these values yield a response close to the desired characteristics, this closed loop system was plotted in MATLAB, where the new A matrix will equal $A - BK$.

```
A_k = A - B_Tc * K_vals;
full_state = ss(A_k, B_Tc, [1, 0, 0, 0]*(180/pi), 0);
figure(1), step(full_state);
```

Figure 4: MATLAB Code to Plot Full-State Feedback System

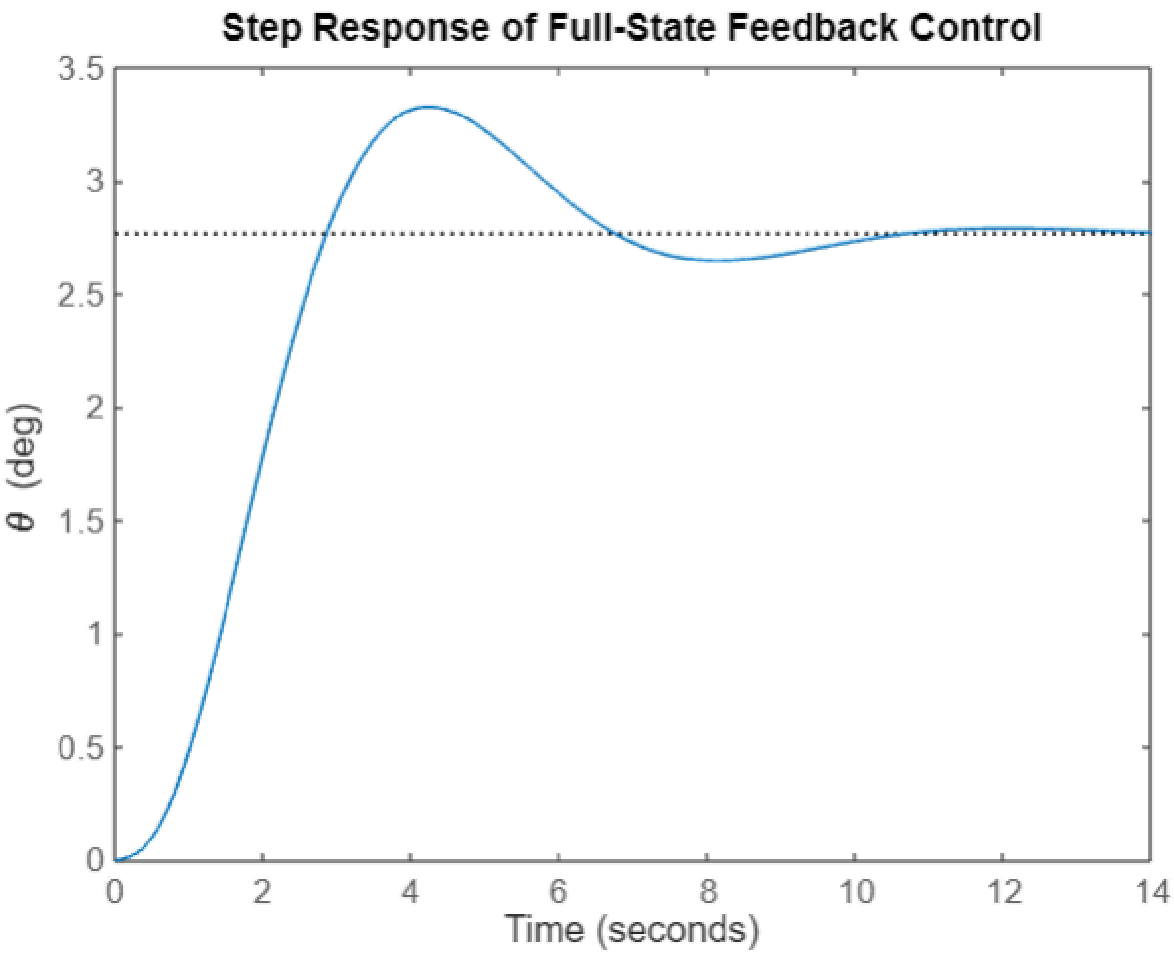

Figure 5: Plot of Full-State Feeback System

Analyzing the characteristics of this graph reveals that the settling time is around 9.6 seconds and the percent overshoot is around 20.3%, both of which are fairly close to the desired values.

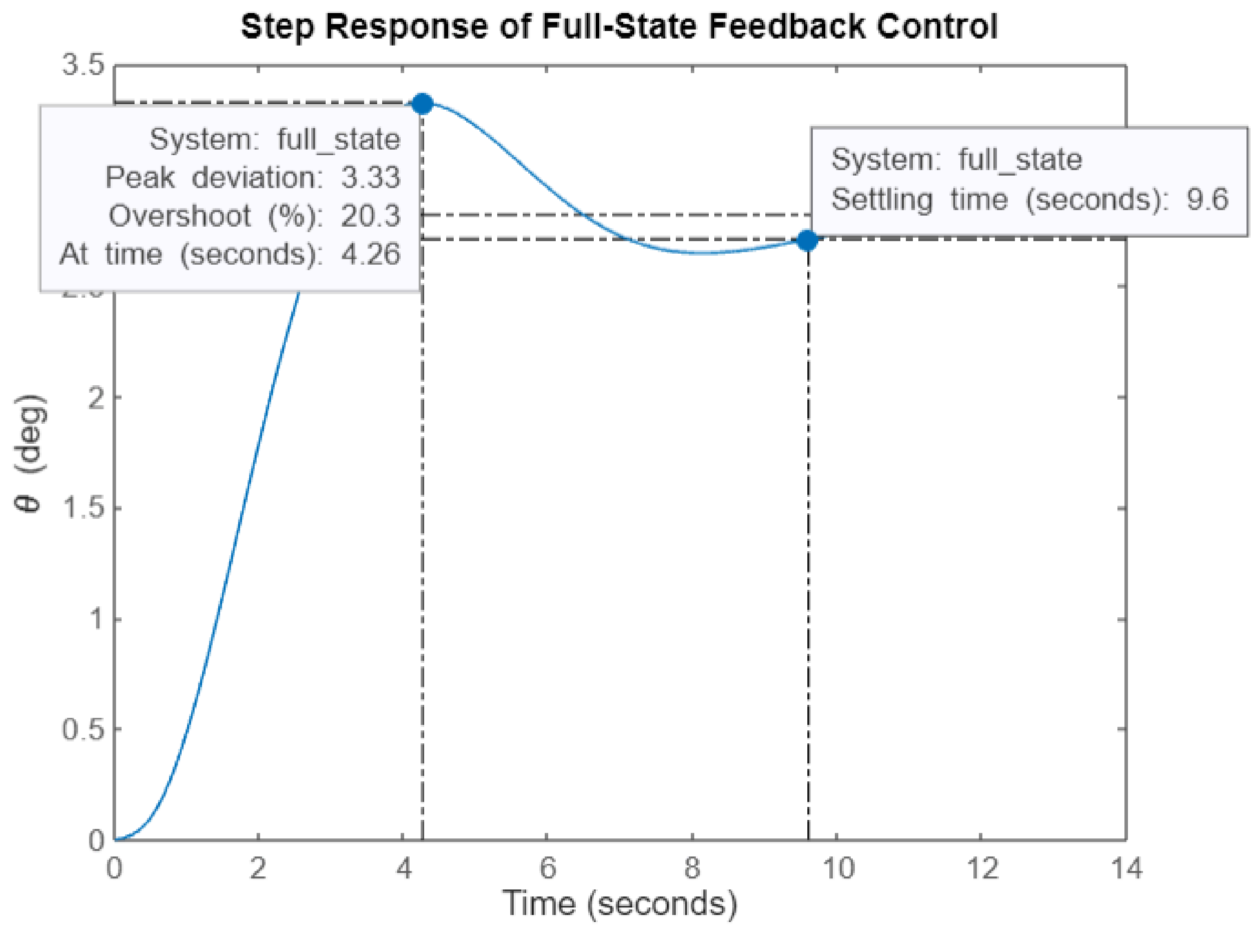

Figure 6: Plot of Full-State Feedback System with Settling Time and Overshoot

b. To design an observer, a very similar approach is taken, and, in fact, the desired characteristic equation will be the same. The observer matrix is defined as:

$$L = \begin{bmatrix} \ell_1 \\ \ell_2 \\ \ell_3 \\ \ell_4 \end{bmatrix}$$

The characteristic equation for the system with the observer will then be:

$$\det(sI - (A - LC)) = 0$$

$$s^4 + (30.0\ell_1 + 0.234)s^3 + (30.0\ell_3 + 7.02\ell_1 + 116.944)s^2 +$$

$$(2.166\ell_4 + 7.02\ell_3 + 1083.33\ell_2 + 3508.32\ell_1)s + (1083.33\ell_4 + 3508.32\ell_3 + 0.1985\ell_2)$$

Again, the like terms of this characteristic equation and the desired characteristic equation are compared to create a linear system which can be solved to find the ℓ values.

$$\begin{bmatrix} 0 & 0 & 0 & 30.0 \\ 0 & 30.0 & 0 & 7.02 \\ 2.166 & 7.02 & 1083.33 & 3508.32 \\ 1083.33 & 3508.32 & 0.1985 & 0 \end{bmatrix} \begin{bmatrix} \ell_4 \\ \ell_3 \\ \ell_2 \\ \ell_1 \end{bmatrix} = \begin{bmatrix} 0.8 + a + b - 0.234 \\ ab + 0.8(a+b) + 0.808 - 116.944 \\ 0.8ab + 0.808(a+b) \\ 0.808ab \end{bmatrix}$$

Values of a and b were again chosen arbitrarily as $a = 10$ and $b = 5$. After substituting these values in, the solution to the system is:

$$\begin{bmatrix} \ell_4 \\ \ell_3 \\ \ell_2 \\ \ell_1 \end{bmatrix} = \begin{bmatrix} 15.566 \\ -54.136 \\ 52.12 \\ 40.4 \end{bmatrix}$$

The observer error is defined as follows:

$$\tilde{x} = x - \hat{x}$$
$$\dot{\tilde{x}} = \dot{x} - \dot{\hat{x}}$$

Therefore, the error over all states will be given by:

$$\begin{cases} \dot{\tilde{x}} = (A - FM)\tilde{x} \\ y = C\tilde{x} \end{cases}$$

The error for each state must be found individually, and thus four different state spaces were made, each looking at a different state. A reasonable set of initial conditions were chosen as specified below, then, using MATLAB, the error for the observer design detailed above was plotted.

$$\begin{bmatrix} \theta_0 \\ \psi_0 \\ \dot{\theta}_0 \\ \dot{\psi}_0 \end{bmatrix} = \begin{bmatrix} 0.1 \\ 0.4 \\ 0.2 \\ 0.2 \end{bmatrix}$$

```
%observer error state matrices
err_A = A-L_vals*M;
init_conds = [0.1; 0.4; 0.2; 0.2];
err_C_th = [1, 0, 0, 0];
err_C_ps = [0, 1, 0, 0];
err_C_thdt = [0, 0, 1, 0];
err_C_psdt = [0, 0, 0, 1];

%observer error state space models
obs_err_th = ss(err_A, [0; 0; 0; 0], err_C_th*(180/pi), 0);
obs_err_ps = ss(err_A, [0; 0; 0; 0], err_C_ps*(180/pi), 0);
obs_err_thdt = ss(err_A, [0; 0; 0; 0], err_C_thdt*(180/pi), 0);
obs_err_psdt = ss(err_A, [0; 0; 0; 0], err_C_psdt*(180/pi), 0);

%graph observer error for each state on the same graph
figure(2), initial(obs_err_th, obs_err_ps, obs_err_thdt, obs_err_psdt, init_conds);
```

Figure 7: Observer Error MATLAB code

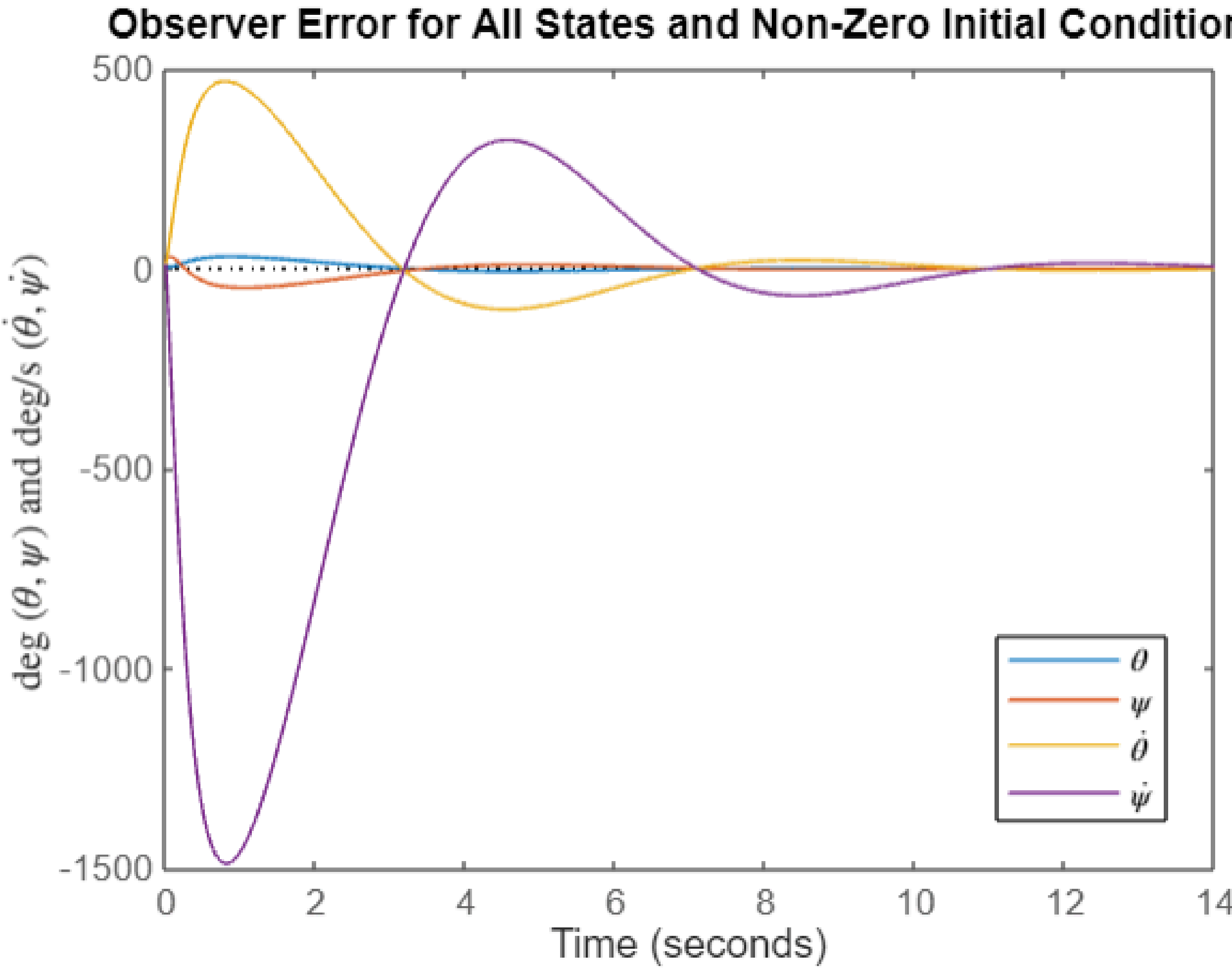

Figure 8: Observer Error for all States, $a = 10,\ b = 5$

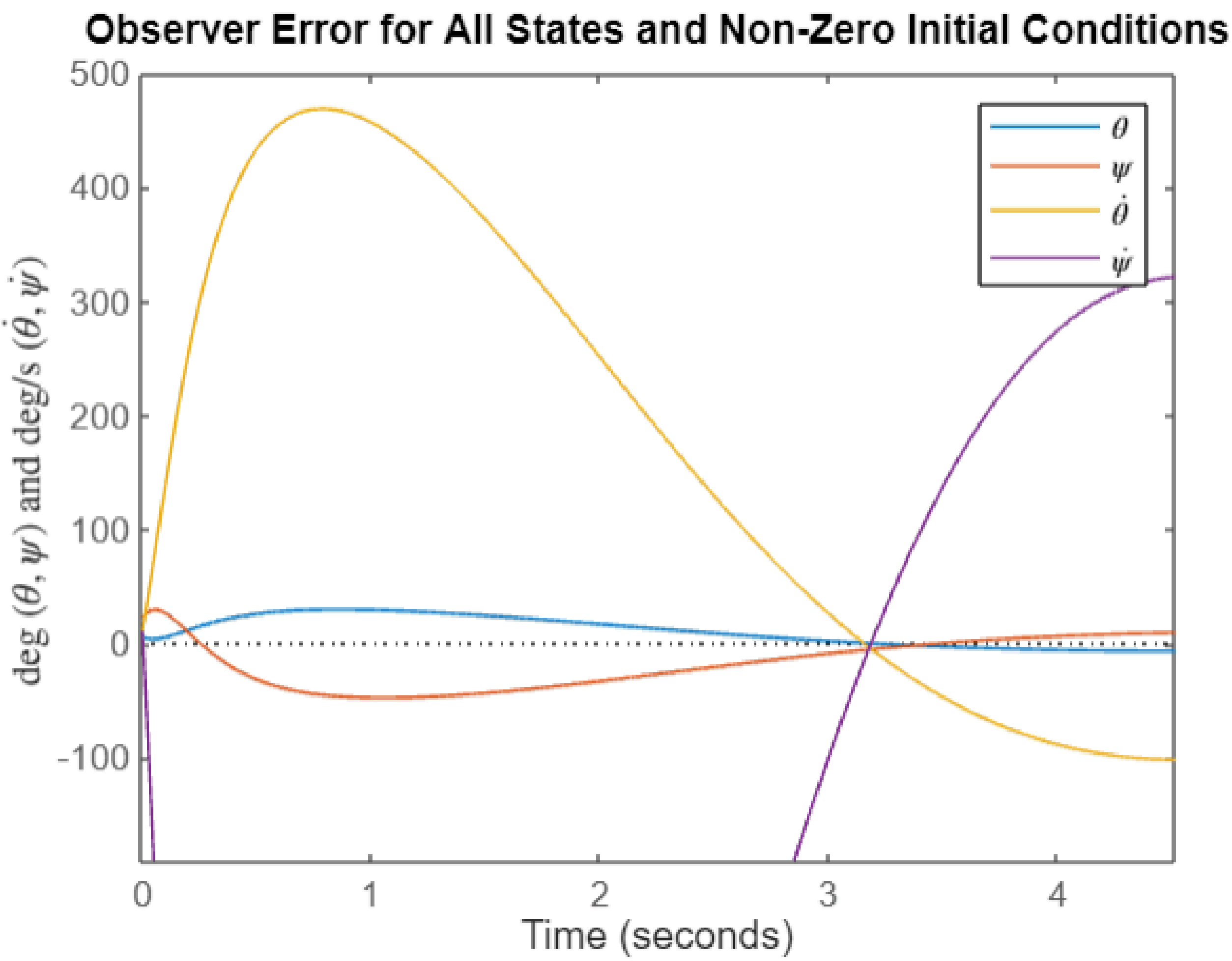

Figure 9: Close up Observer Error for all States to Distinguish the Peak of $\dot{\theta}$, $a = 10,\ b = 5$

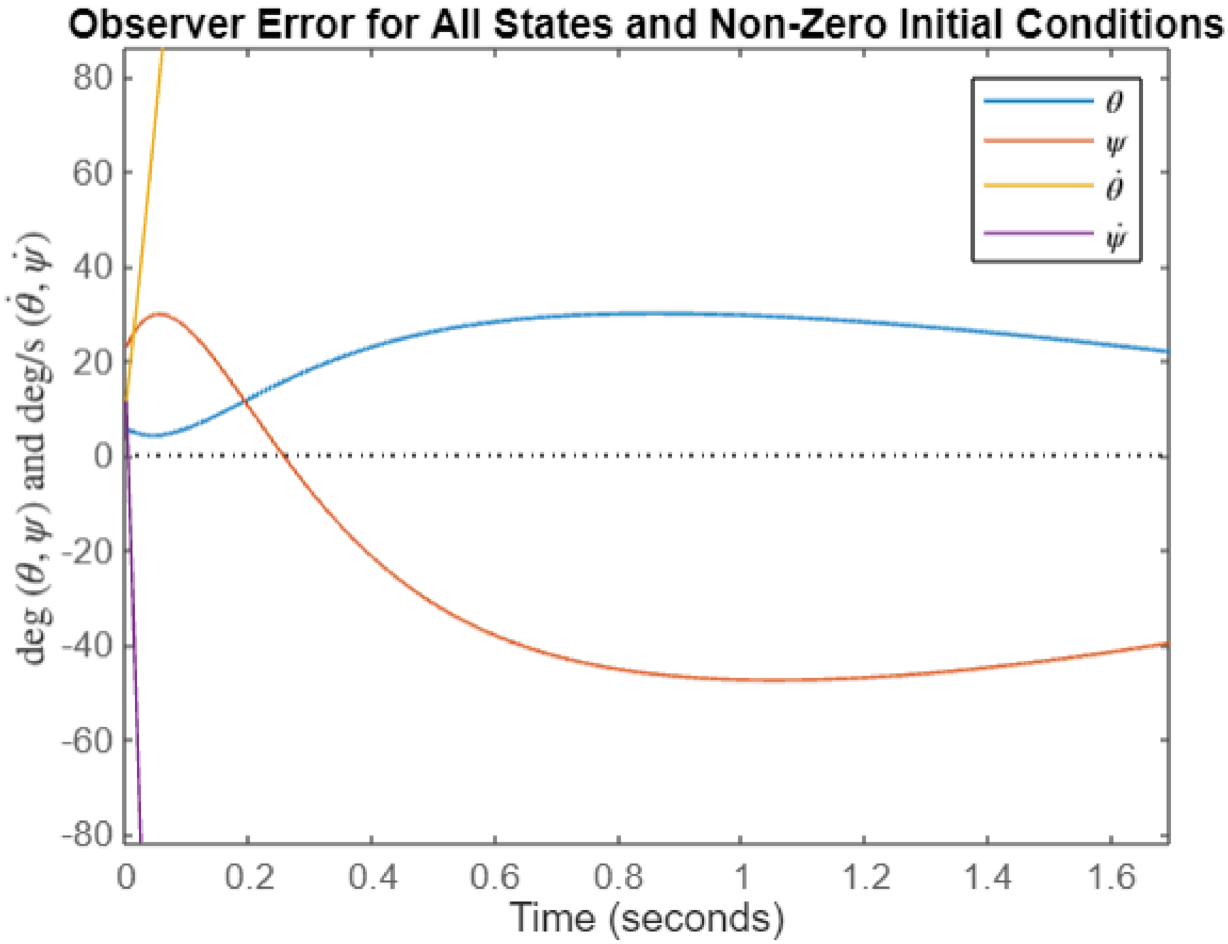

Figure 10: Close up Observer Error for all States to Distinguish the Peaks of θ and ψ, $a = 10,\ b = 5$

The observer error approaches zero at steady state for all states, though the amplitude of the first spike is quite large for $\dot{\psi}$ and $\dot{\theta}$.

c. Now that both the full-state feedback gain and observer gain matrices have been determined, the closed-loop system can finally be constructed. The state space representation of this system will take the following form:

$$\begin{bmatrix} \dot{x} \\ \dot{\tilde{x}} \end{bmatrix} = \begin{bmatrix} A - BK & BK \\ 0 & A - LM \end{bmatrix} \begin{bmatrix} x \\ \tilde{x} \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u_a$$

Substituting in each matrix yields the closed-loop system for this controller design.

$$\begin{bmatrix} \dot{\theta} \\ \dot{\psi} \\ \ddot{\theta} \\ \ddot{\psi} \\ \dot{\tilde{\theta}} \\ \dot{\tilde{\psi}} \\ \ddot{\tilde{\theta}} \\ \ddot{\tilde{\psi}} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ -0.576 & -6.145 & -0.742 & 11.488 & 0.576 & 42.256 & 0.742 & -11.416 \\ 0.748 & -62.072 & 0.963 & -15.058 & -0.748 & -54.872 & -0.963 & 14.824 \\ 0 & 0 & 0 & 0 & -15.566 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 48.969 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 57.778 & 36.111 & 0 & 0.072 \\ 0 & 0 & 0 & 0 & -188.240 & -116.944 & 0 & -0.234 \end{bmatrix} \begin{bmatrix} \theta \\ \psi \\ \dot{\theta} \\ \dot{\psi} \\ \tilde{\theta} \\ \tilde{\psi} \\ \dot{\tilde{\theta}} \\ \dot{\tilde{\psi}} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0.0278 \\ -0.0361 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} u_a$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \psi \\ \dot{\theta} \\ \dot{\psi} \\ \tilde{\theta} \\ \tilde{\psi} \\ \dot{\tilde{\theta}} \\ \dot{\tilde{\psi}} \end{bmatrix}$$

In order to achieve the desired set-point at steady state, the set-point control must be calculated, for which the following formula is used.

$$u = -\left(CA^{-1}B\right)^{-1}$$

Here the A, B, and C matrices are pulled from the closed-loop state space model. Substituting these matrices in, the set-point control is then calculated.

$$u = 20.7179$$

d. Before the system can be fully simulated, the input must be calculated from the set-point control.

$$T_C = 20.7179 * \frac{\pi}{6}$$
$$T_C = 10.8479\ Nm$$

Since this input is below the torque limit of $15\ Nm$, the response of the closed-loop system with zero initial conditions can be plotted in MATLAB.

```
final_sys = ss(A_cl, B_cl*input, C_cl, 0);
figure(6), step(final_sys);
```

Figure 11: MATLAB Code for Closed-Loop System

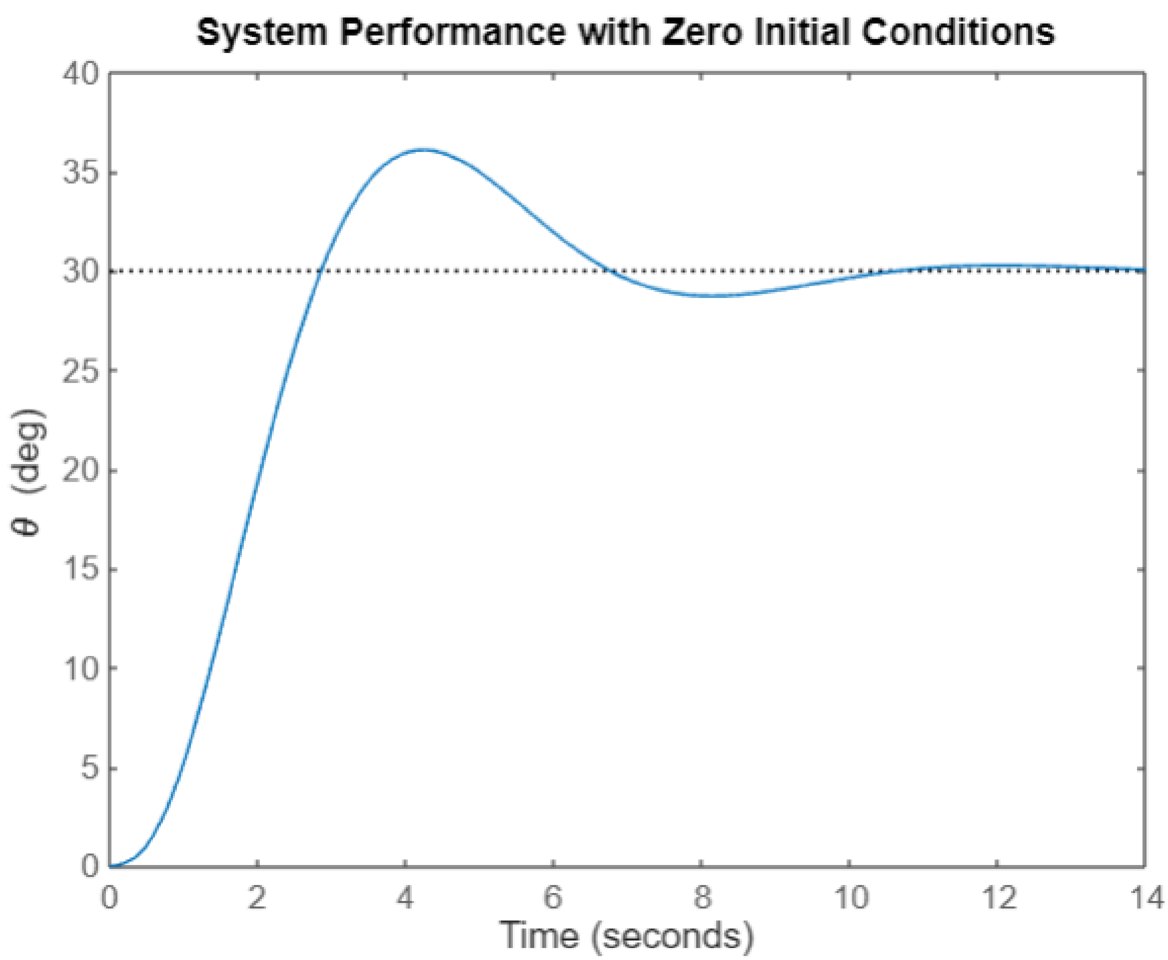

Figure 12: Performance of Closed-Loop System Step Response

By analyzing the graph, it can be determined that the percent overshoot is 20.3% and the settling time is 9.6 seconds. These values are fairly close to the specified overshoot and settling time, but the input still has room beneath the limit to tweak the performance.

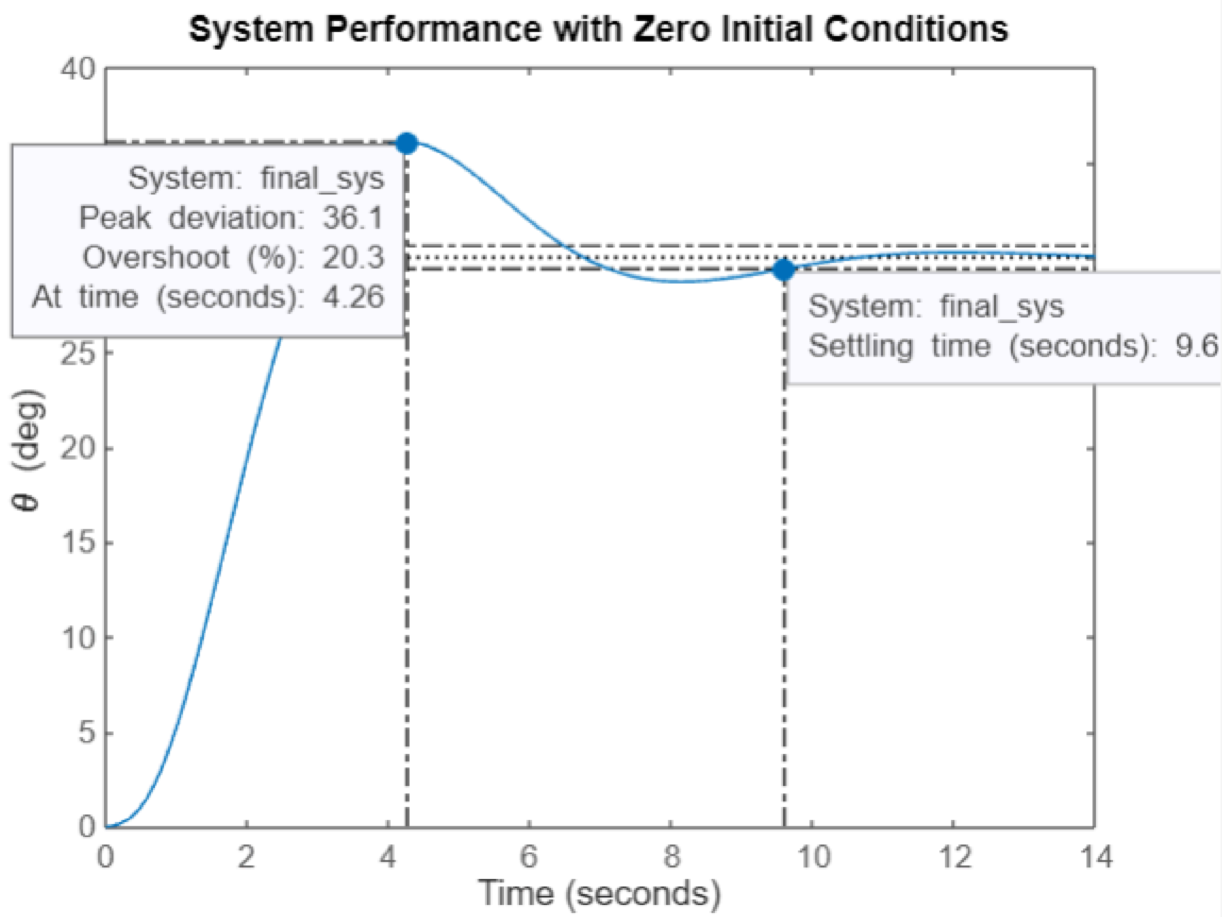

Figure 13: Plot of System Performance Without Initial Conditions

5. In order to demonstrate the function of our design, simulations were completed with zero initial conditions. By arbitrarily selecting two eigenvalues, denoted by a and b, the specifications for maximum torque, percent overshoot, settling time, and torso set point were verified. Through trial and error, a and b were set as the following:

$$a = 15;\ b = 4.5$$

These values correspond to a torque input of 14.6447 Nm, percent overshoot of 20.3%, settling time of 9.59 s, and a torso set point of 30. These values can be seen in the graph below.

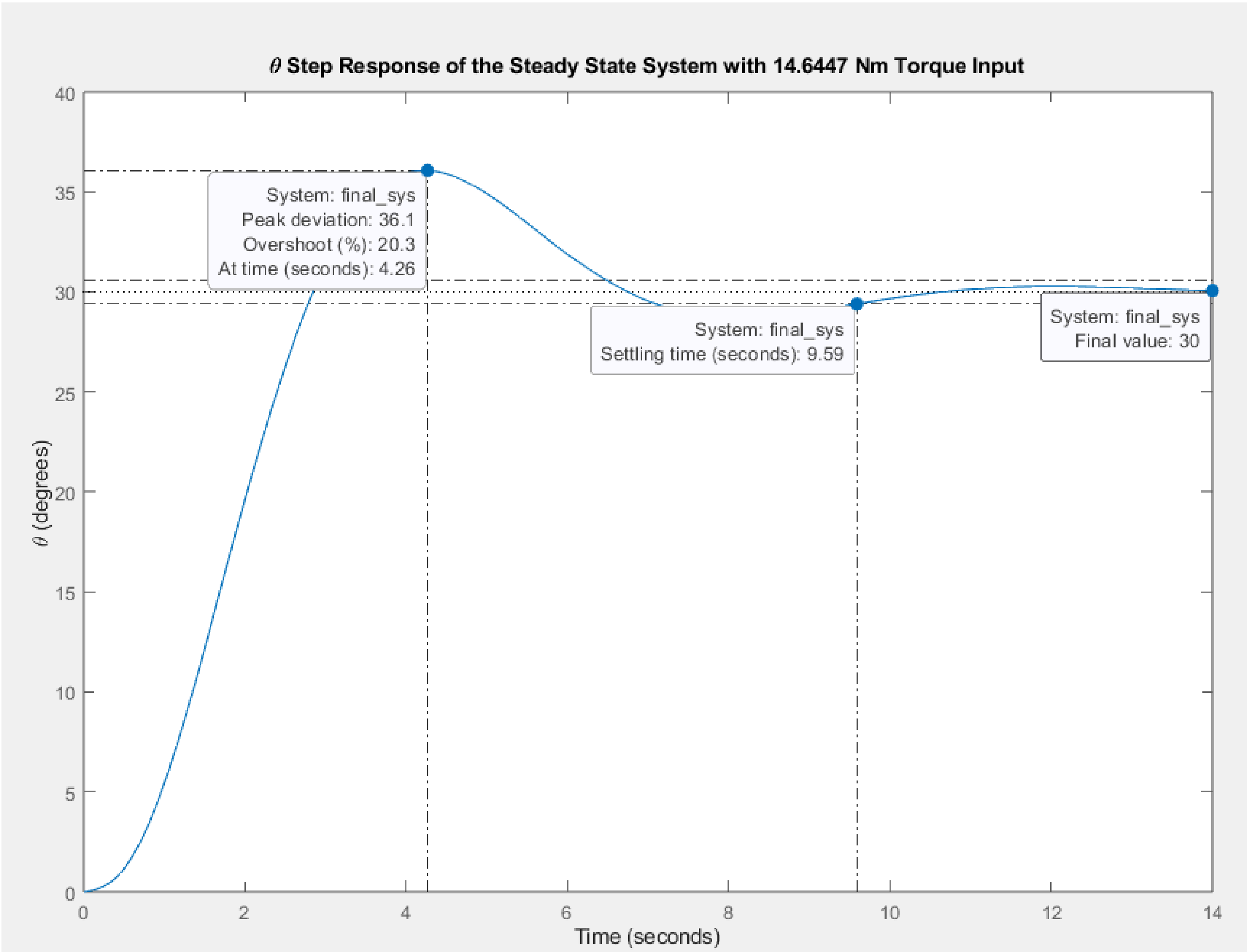

Figure 14: Response of θ with Respect to Time and Desired Values

As observed from the graph, these values are fairly close to the specified parameters. Moreover, though the settling time is very slightly slower than our initial design, the observer error shows a decent amount of improvement in all states.

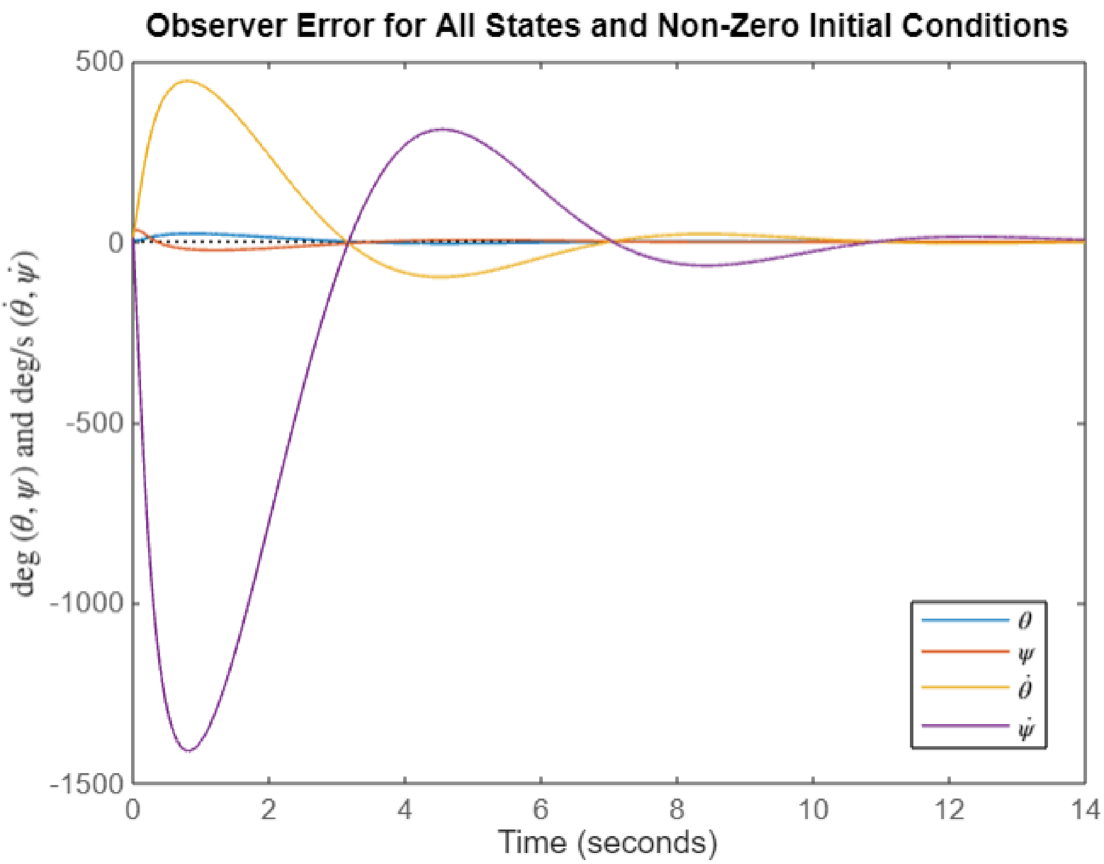

Figure 15: Observer Error for all States, $a = 15,\ b = 4.5$

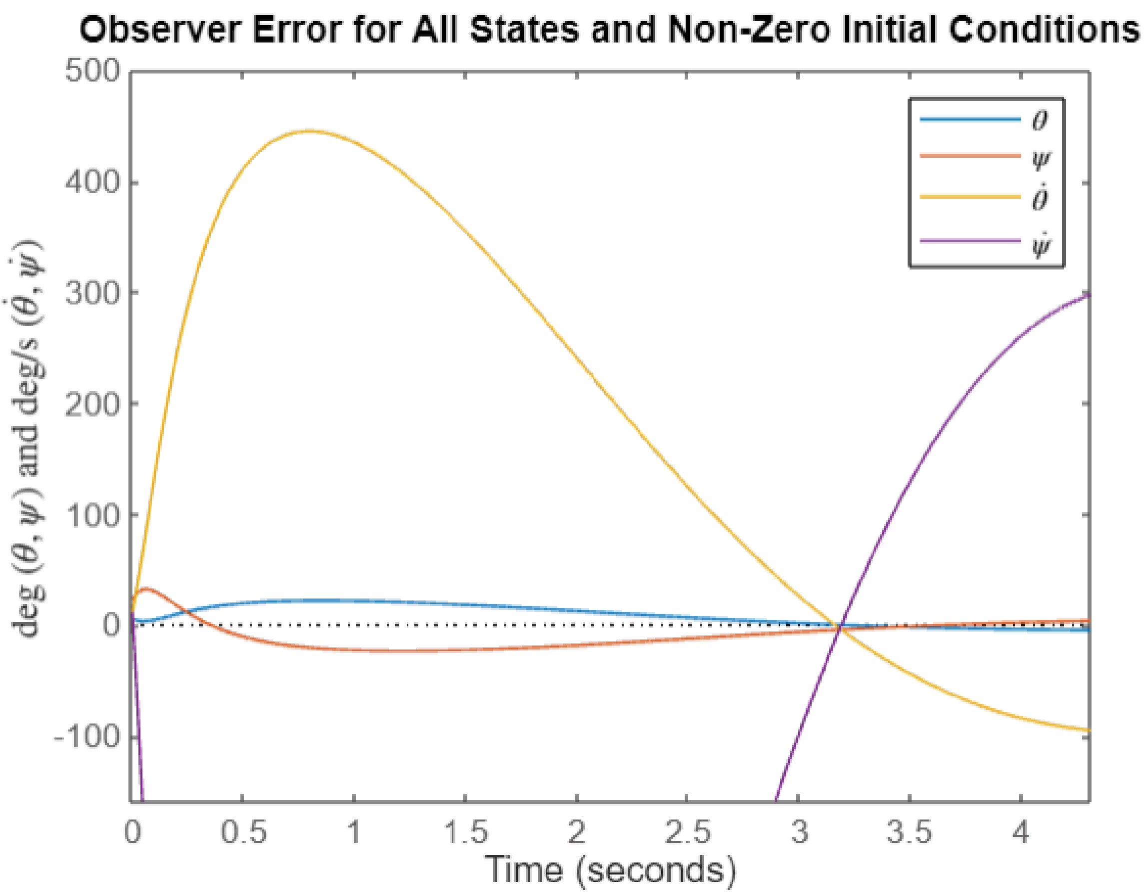

Figure 16: Close up Observer Error for all States to Distinguise the Peak of $\dot{\theta}$, $a = 15,\ b = 4.5$

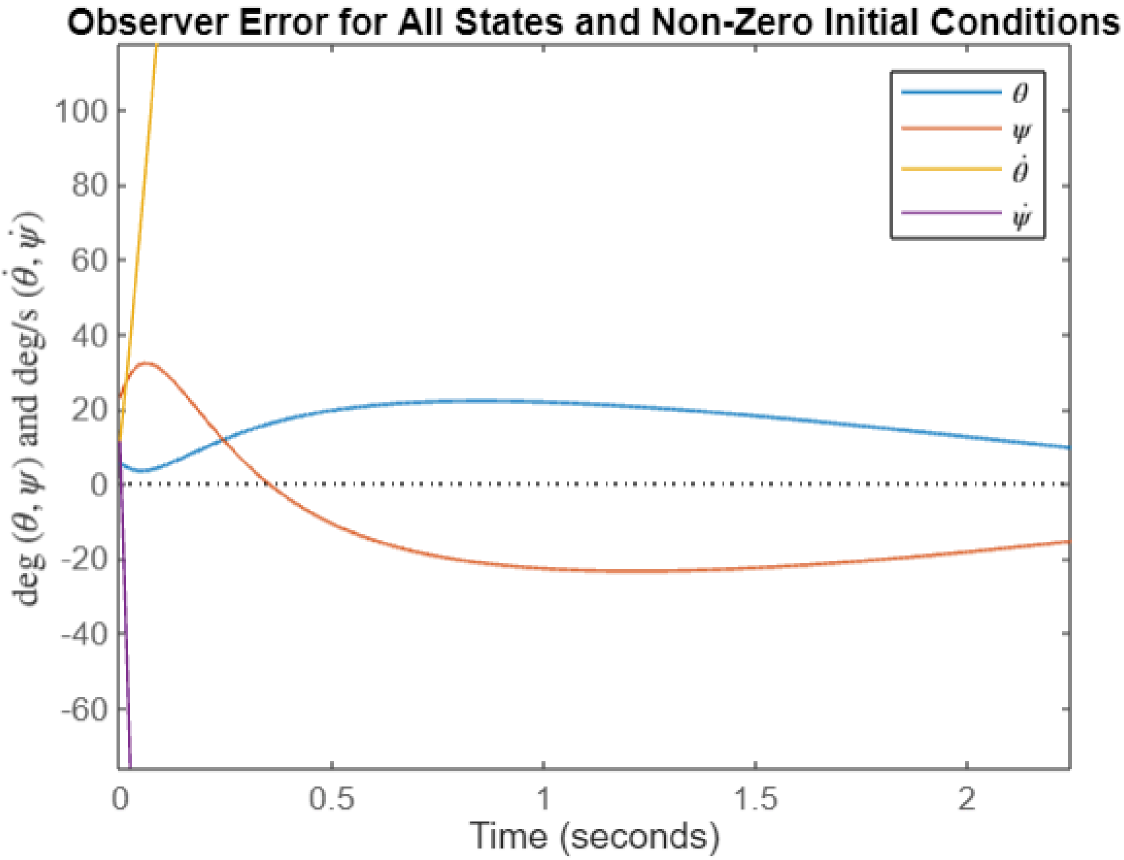

Figure 17: Close up Observer Error for all States to Distinguish the Peaks of θ and ψ, $a = 15,\ b = 4.5$

6. After simulating the state response of θ in order to verify the desired values, the other states can be similarly graphed in MATLAB. Below are all of the state responses of the system.

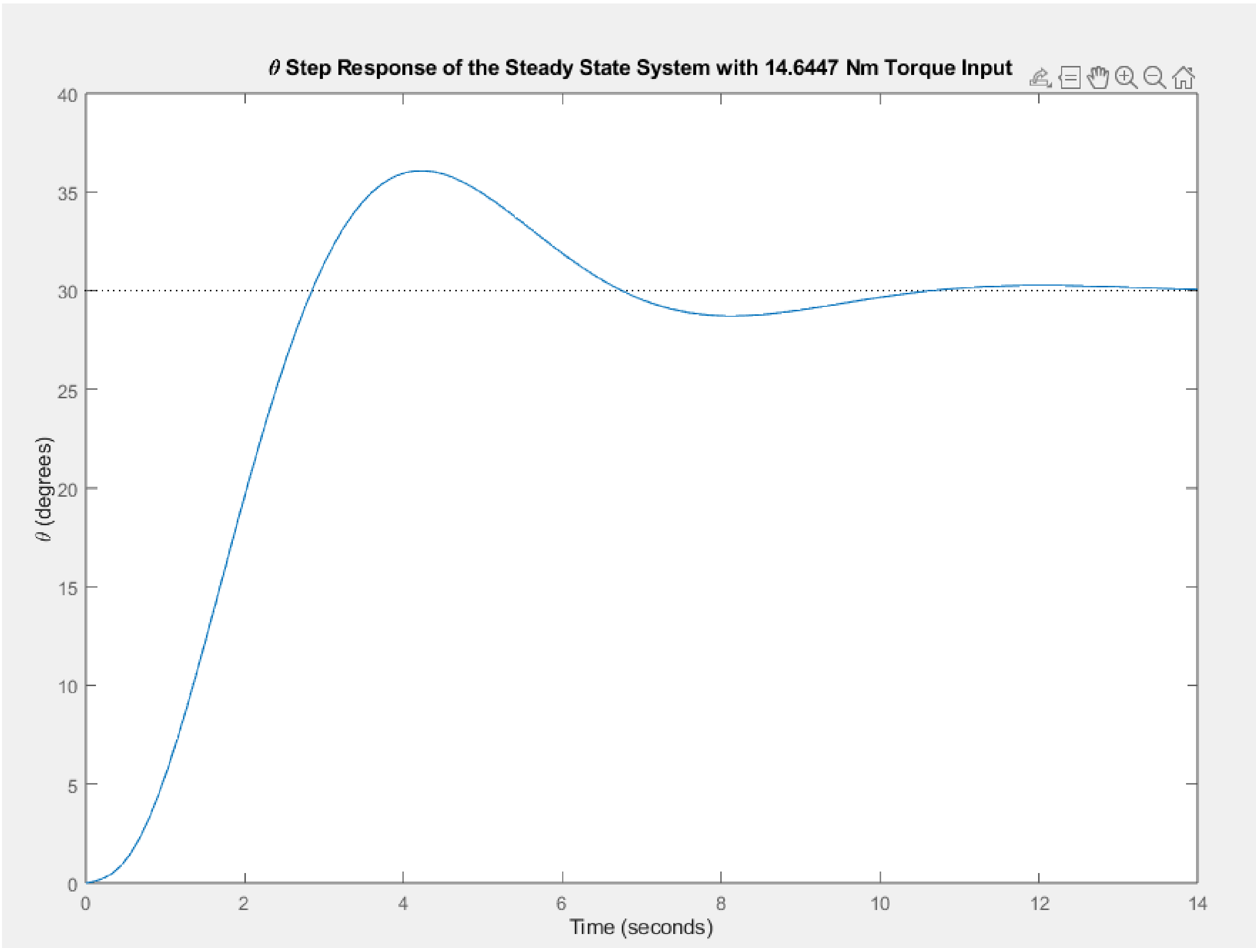

Figure 18: Response of State θ with Respect to Time

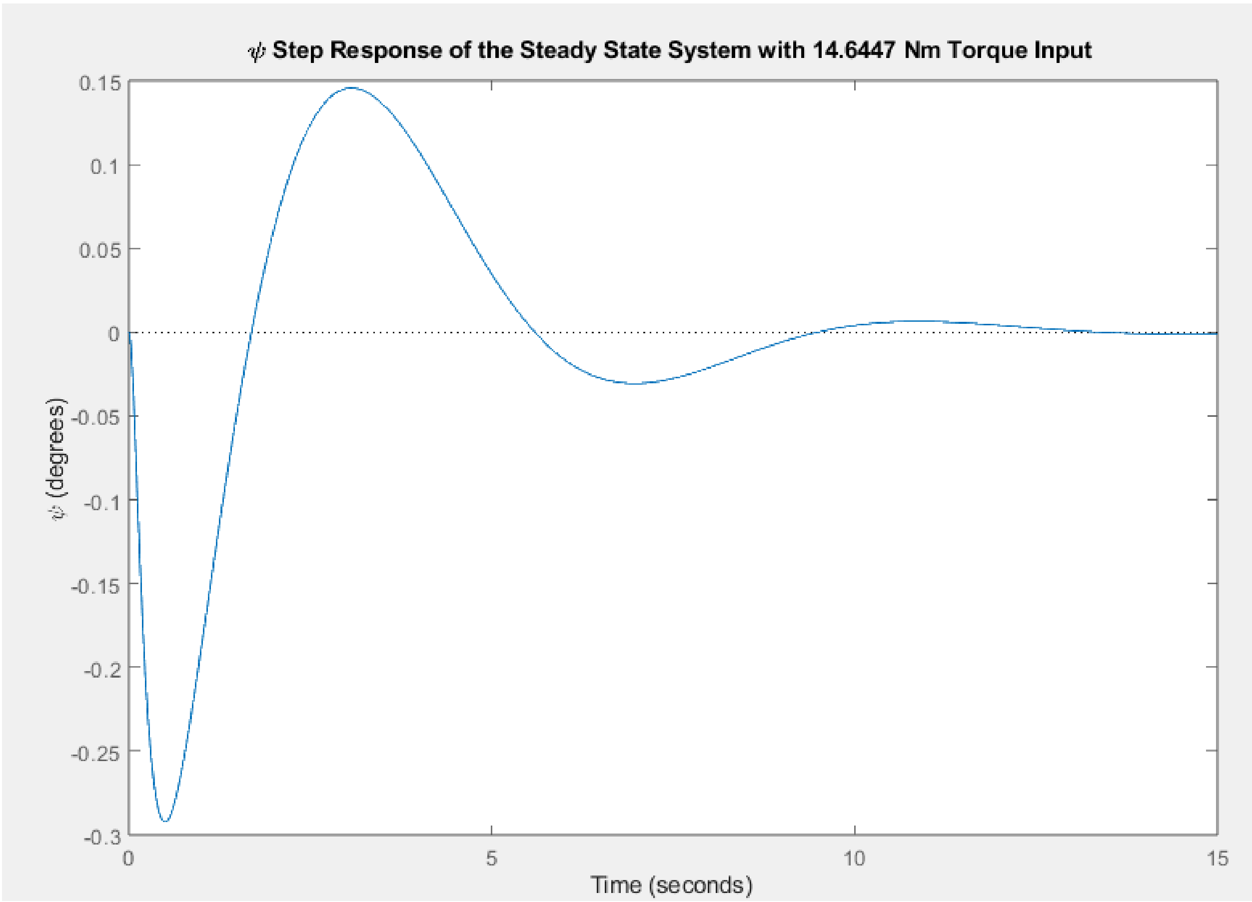

Figure 19: Response of State ψ with Respect to Time

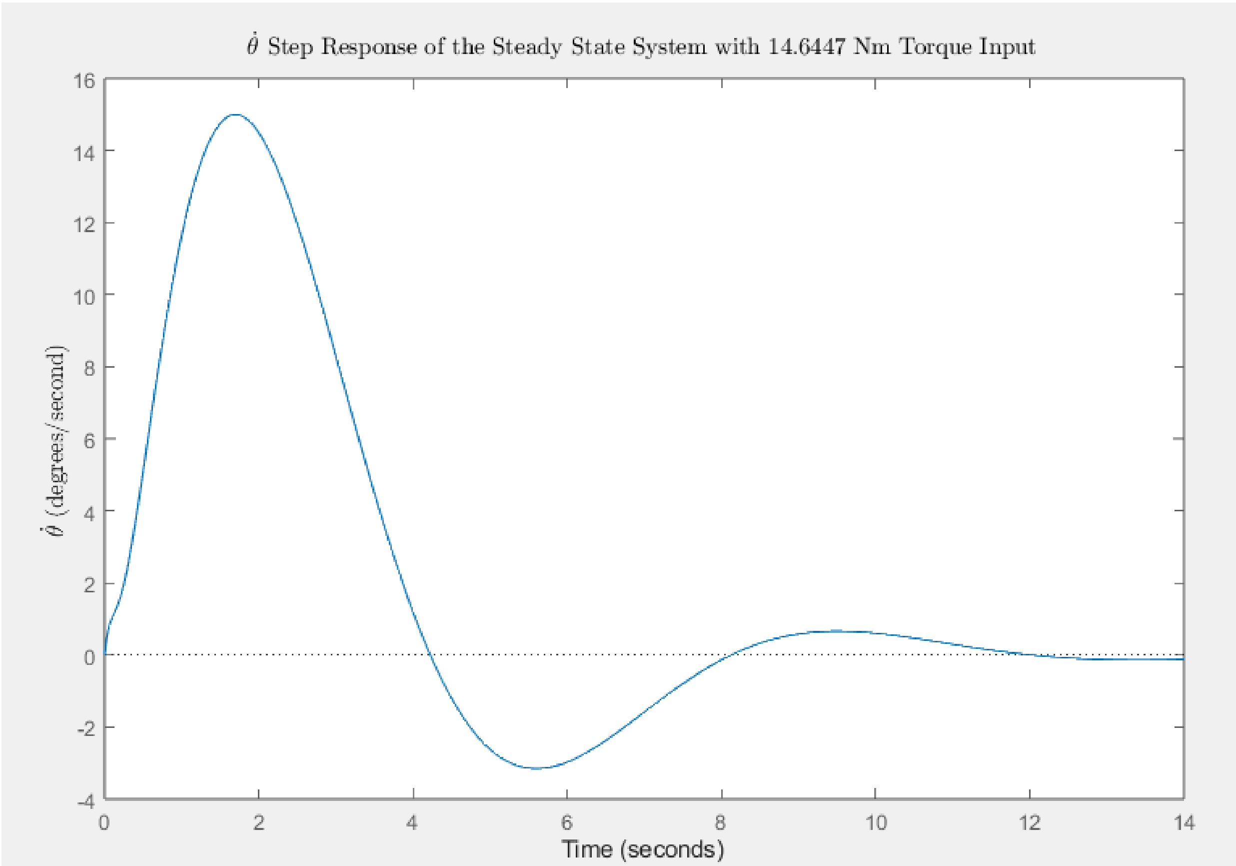

Figure 20: Response of State $\dot{\theta}$ with Respect to Time

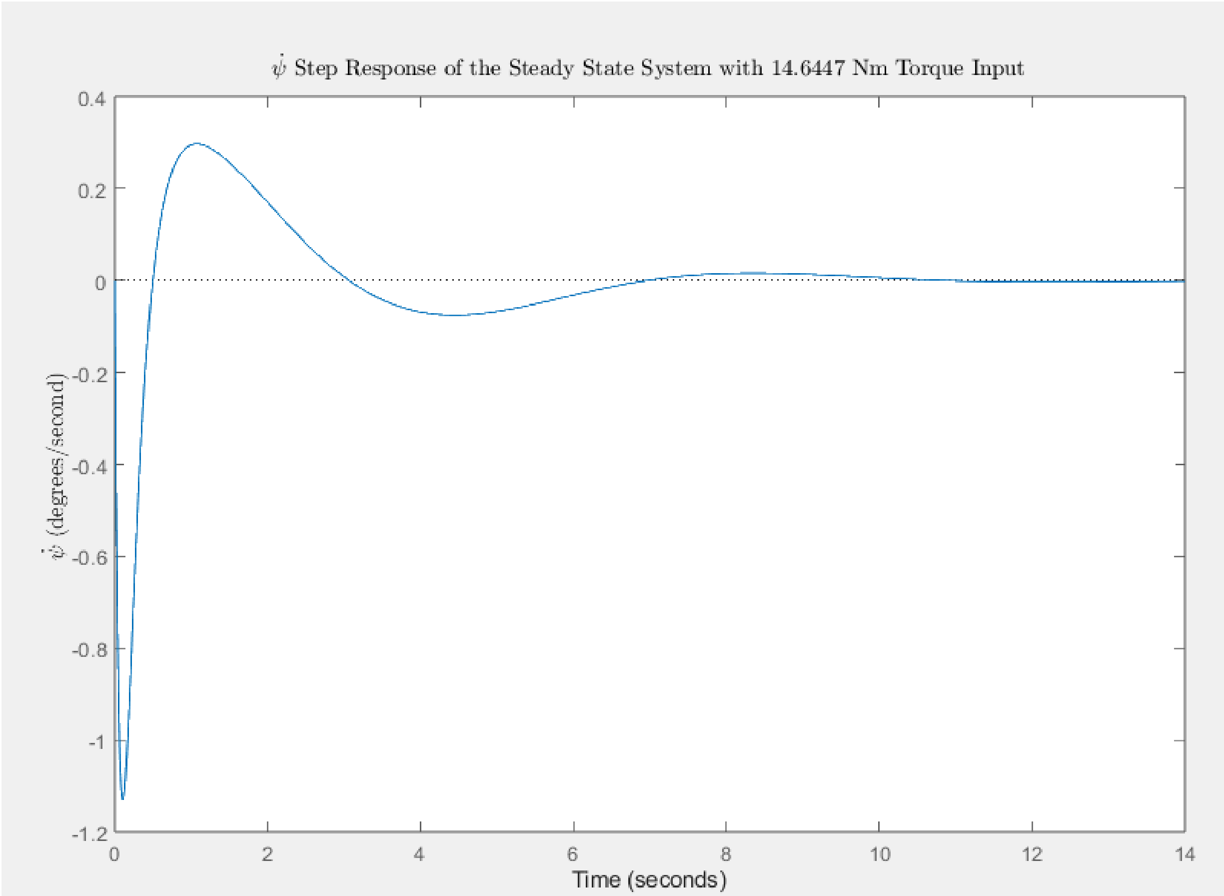

Figure 21: Response of State $\dot{\psi}$ with Respect to Time

The changes in θ and ψ on the graph correspond to the change in angles shown in Figure 1. Through our control of the system, they achieve steady state at 30° and 0°, respectively. The 30° angle θ is depicting the top body on our schematic to be at an angle of 30° from the reference point. The 0° angle ψ depicts a steady state where the top and bottom body on the schematic are parallel to each other, this maintains posture while allowing for linear translation.

Graphically, the change in θ and ψ over time, or $\dot{\theta}$ and $\dot{\psi}$, fall within expected bounds. On the θ graph at approximately two seconds, the slope is about 15. At two seconds on the $\dot{\theta}$ graph, the value is the slope of the θ graph. This can be seen on the rest of the θ and $\dot{\theta}$ graphs as well as the ψ and $\dot{\psi}$ graphs. Both $\dot{\theta}$ and $\dot{\psi}$ have a steady state value of 0. The goal of the controller is for θ and ψ to not change with time, and graphically this can be seen with the steady state of the $\dot{\theta}$ and $\dot{\psi}$ graphs approaching 0. Despite this proper steady state of $\dot{\theta}$ and $\dot{\psi}$, the start of $\dot{\psi}$ jumps down to approximately -1.1° before the control brings it back up to 0.2° and then down to 0°. This corresponds with the derivative of the ψ graph and shows how fast the angle ψ changes within the first two seconds of simulation.

Bibliography

[1] A. Yousuff, *MEM 355: Performance Enhancement of Dynamic Systems Final Project (Spring 2023-2024; 202335)).* [Accessed June 10, 2024]

Appendix

Full MATLAB Code:

```
% Problem 1
M_given = [60.1429, 18.5717;
     18.5714, 14.2857];
D_given = [0, 0;
     0, 2];
K_given = [0, 0;
     0, 1000];
B_given =[1; 0];
F_given = [-0.042857; -0.071429];

A = [zeros(2), eye(2);
     -inv(M_given)*K_given, -inv(M_given)*D_given]

B = [0, 0;
     0, 0;
     inv(M_given)*B_given, inv(M_given)*F_given]
```

```
%%
% Problem 2

[V,D] = eig(A)
```

```
%%
% Problem 3
B = [0, 0;
     0, 0;
     0.0278, 0.00139;
     -0.0361, -0.00681];
B_Tc = [0; 0; 0.0278; -0.0361]; %input matrix without force input
B_f = [0; 0; 1.39*10^(-3); -6.81*10^(-3)]; %input matrix without torque input

C = vpa([B, A*B, A^2*B, A^3*B])
C_Tc = vpa([B_Tc, A*B_Tc, A*A*B_Tc, A*A*A*B])
C_f = vpa([B_f, A*B_f, A*A*B_f, A*A*A*B])

rank(C) %part a
rank(C_Tc) %part b
rank(C_f) %part c

M_s = [1, 0, 0, 0];
O = [M_s; M_s*A; M_s*A*A; M_s*A*A*A]
rank(O) %part d
```

```
%%
%Problem 4 Part a
syms s; syms k1; syms k2; syms k3; syms k4;
syms a1; syms b1;
K = [k1, k2, k3, k4];
vpa(det(s*eye(4)-(A-B_Tc*K)), 3) %controller char eq

vpa(expand((s+a1)*(s+b1)*(s^2+0.8*s+0.808))) % desired char eq

%Two desired poles not determinable from specifications; change these
%values to design the controller
a = 15; b = 4.5;

%These two matrices are hardcoded in from setting desired and controller
%characteristic equations equal
K_c = [-0.0361, 0.0278, 0, 0;
       0, 0.0039, -0.0361, 0.0278;
       0, 1.95, 0, 0.0039;
       0, 0, 0, 1.95];
consts_k = [0.8+a+b-0.234;
            a*b+0.8*(a+b)+0.808-117;
            0.8*a*b+0.808*(a+b);
            0.808*a*b];
K_vals = fliplr(transpose(inv(K_c)*consts_k));
```

```
%create and graph the state space
A_k = A - B_Tc * K_vals;
full_state = ss(A_k, B_Tc, [1, 0, 0, 0]*(180/pi), 0);
figure(1), step(full_state);
title('Step Response of Full-State Feedback Control'), ylabel('\theta (deg)');
```

```
%%
% Problem 4 Part b
syms s; syms l1; syms l2; syms l3; syms l4;
M = [1, 0, 0, 0];
L = [l1; l2; l3; l4];
vpa(det(s*eye(4)-(A-L*M))) %observer char eq

%desired char eq will be the same as in Part a, matrices change due to the
%observer char eq
L_c = [0, 0, 0, 1;
       0, 1, 0, 0.234;
       0.0722, 0.234, 36.111, 116.944;
       36.111, 116.944, 0.006172, 0];
consts_L = [0.8+a+b-0.234;
            a*b+0.8*(a+b)+0.808-116.944;
            0.8*a*b+0.808*(a+b);
            0.808*a*b];
```

```
L_vals = flipud(inv(L_c)*consts_L);

%observer error state matrices
err_A = A-L_vals*M;
init_conds = [0.1; 0.4; 0.2; 0.2];
err_C_th = [1, 0, 0, 0];
err_C_ps = [0, 1, 0, 0];
err_C_thdt = [0, 0, 1, 0];
err_C_psdt = [0, 0, 0, 1];

%observer error state space models
obs_err_th = ss(err_A, [0; 0; 0; 0], err_C_th*(180/pi), 0);
obs_err_ps = ss(err_A, [0; 0; 0; 0], err_C_ps*(180/pi), 0);
obs_err_thdt = ss(err_A, [0; 0; 0; 0], err_C_thdt*(180/pi), 0);
obs_err_psdt = ss(err_A, [0; 0; 0; 0], err_C_psdt*(180/pi), 0);

%graph observer error for each state on the same graph
figure(2), initial(obs_err_th, obs_err_ps, obs_err_thdt, obs_err_psdt, init_conds);
title('Observer Error for All States and Non-Zero Initial Conditions'),
ylabel('deg ($\theta$, $\psi$) and deg/s ($\dot{\theta}$, $\dot{\psi}$)', 'interpreter','latex'),
legend('$\theta$', '$\psi$', '$\dot{\theta}$', '$\dot{\psi}$', interpreter='latex')
```

```
%%
% Problem 4 Part c and d

A_cl = [A-B_Tc*K_vals, B_Tc*K_vals;
            zeros(4), A-L_vals*M];
B_cl = [B_Tc; [0; 0; 0; 0]];
C_cl = [M, [0, 0, 0, 0]];

set_point = -inv(C_cl*inv(A_cl)*B_cl);
input = set_point * (pi/6)

final_sys = ss(A_cl, B_cl*input, C_cl*(180/pi), 0);
figure(3), step(final_sys);
title('System Performance with Zero Initial Conditions'), ylabel('\theta (deg)');
```

```
%%
% Problem 6
M = [0, 1, 0, 0]; %state: psi
C_cl = [M, [0, 0, 0, 0]];
```

```
final_sys_psi = ss(A_cl, B_cl*input, C_cl*(180/pi), 0);
figure(4), step(final_sys_psi);
title('\psi Step Response of the Steady State System with 14.6447 Nm Torque Input')
ylabel('\psi (degrees)');

M = [0, 0, 1, 0]; %state: theta dot
C_cl = [M, [0, 0, 0, 0]];

final_sys_thetadot = ss(A_cl, B_cl*input, C_cl*(180/pi), 0);
figure(5), step(final_sys_thetadot);
title('$\dot{\theta}$ Step Response of the Steady State System with 14.6447 Nm Torque Input', interpreter='latex')
ylabel('$\dot{\theta}$ (degrees/second)', interpreter='latex');

M = [0, 0, 0, 1]; %state: psi dot
C_cl = [M, [0, 0, 0, 0]];

final_sys_psidot = ss(A_cl, B_cl*input, C_cl*(180/pi), 0);
figure(6), step(final_sys_psidot);
title('$\dot{\psi}$ Step Response of the Steady State System with 14.6447 Nm Torque Input', interpreter='latex')
ylabel('$\dot{\psi}$ (degrees/second)', interpreter='latex');
```

Sample Outputs:

```
A =

         0         0    1.0000         0
         0         0         0    1.0000
         0   36.1120         0    0.0722
         0 -116.9456         0   -0.2339


B =

         0         0
         0         0
    0.0278    0.0014
   -0.0361   -0.0068


V =

   1.0000 + 0.0000i  -1.0000 + 0.0000i   0.0003 + 0.0272i   0.0003 - 0.0272i
   0.0000 + 0.0000i   0.0000 + 0.0000i  -0.0010 - 0.0880i  -0.0010 + 0.0880i
   0.0000 + 0.0000i   0.0000 + 0.0000i  -0.2938 - 0.0000i  -0.2938 + 0.0000i
   0.0000 + 0.0000i   0.0000 + 0.0000i   0.9514 + 0.0000i   0.9514 + 0.0000i
```

```
D =

   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i
   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i
   0.0000 + 0.0000i   0.0000 + 0.0000i  -0.1169 +10.8135i   0.0000 + 0.0000i
   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i  -0.1169 -10.8135i


C =

[      0,        0,    0.0278,    0.00139, -0.002607, -0.0004918, -1.303, -0.2458]
[      0,        0,   -0.0361,   -0.00681,  0.008443,   0.001593,   4.22,   0.796]
[ 0.0278,  0.00139, -0.002607, -0.0004918,    -1.303,    -0.2458, 0.6097,   0.115]
[-0.0361, -0.00681,  0.008443,   0.001593,      4.22,      0.796, -1.974, -0.3725]


C_Tc =

[      0,    0.0278, -0.002607, -1.303, -0.2458]
[      0,   -0.0361,  0.008443,   4.22,   0.796]
[ 0.0278, -0.002607,    -1.303, 0.6097,   0.115]
[-0.0361,  0.008443,      4.22, -1.974, -0.3725]
```

```
C_f =

[        0,    0.00139, -0.0004918, -1.303, -0.2458]
[        0,   -0.00681,   0.001593,   4.22,   0.796]
[ 0.00139, -0.0004918,    -0.2458, 0.6097,   0.115]
[-0.00681,   0.001593,      0.796, -1.974, -0.3725]


ans =

     4


ans =

     4


ans =

     4
```

```
O =

    1.0000         0         0         0
         0         0    1.0000         0
         0   36.1120         0    0.0722
         0   -8.4463         0   36.0951


ans =

     4


ans =

1.95*k1 + 0.00389*k1*s + 1.95*k3*s + 0.0278*k1*s^2 - 0.0361*k2*s^2 + 0.00389*k3*s^2 + 0.0278*k3*s^3 - 0.0361*k4*s^3 + 117.0*s^2 + 0.234*s^3 + s^4


ans =

0.808*a1*b1 + 0.808*a1*s + 0.808*b1*s + 0.8*a1*s^2 + a1*s^3 + 0.8*b1*s^2 + b1*s^3 + 0.808*s^2 + 0.8*s^3 + s^4 + 0.8*a1*b1*s + a1*b1*s^2
```

```
ans =

7.183e-16*l2 + 116.9*l3 + 36.11*l4 + 116.9*l1*s + 36.11*l2*s + 0.2339*l3*s + 0.07222*l4*s + 0.2339*l1*s^2 + l1*s^3 + l3*s^2 + 116.9*s^2 + 0.2339*s^3 + s^4


input =

   14.6447
```