Assignment For EE5101 - Linear Systems
Sem I AY2010/2011

Linear State Feedback Controller Design

Phang Swee King
A0033585A
Email: king@nus.edu.sg
NGS/ECE Dept.
Faculty of Engineering
National University of Singapore

October 17, 2010

Abstract

This project describes the problems, solutions and findings of a linear state-space system. In this report, a typical third-order single input single output plant is provided in the form of transfer functions. The state-space model and dynamic characteristics is to be investigated. In order to stabilize and gain full control of the plant, a state feedback controller using the Linear Quadratic Regulator (LQR) method is required. Then a practical situation in which the state variables are not fully measurable is considered. As a solution, state observer is used and it will be designed based on the pole placement method. Lastly, servo control is formulated and implemented into the control loop to realize reference tracking and disturbance rejection. All the above investigations will be carried out or simulated by MATLAB and Simulink.

Contents

Chapter 1

Introduction

Classical Control Theory was first introduced in 1930-1940, which had achieved a great success in the 20th century. However, it has some disadvantages, as it is based on input/output relationship called transfer function. In computing the transfer function of a system, all initial conditions are set to be zero. Also, it is difficult to design a controller by using classical methods like frequency response and root locus. Due to these inherent shortcomings, Classical Control Theory was gradually replaced by Modern Control Theory.

Contrary to Classical Control Theory, Modern Control Theory uses the time domain representation of the system known as the state-space representation. Thus it is able to provide a complete description of the system. At the late 20th century, many modern control ideas were developed upon the state-space framework, including feedback using pole placement, optimum control, decoupling control, servo control, state observer and many more. Furthermore, the state space representation can handle problems with multiple inputs and multiple outputs, and it is not restricted to linear systems or zero initial condition systems.

In our project, the performance of the plant is analyzed with non-zero initial state. Since the controller for the plant is of 3rd order, it is not easy to design the controller by using Classic Control Theory. Therefore, in this project, the controller is to designed based on the state-space representation in time domain.

Chapter 2

State-space Modelling and Analysis

A plant shown in Fig. 2.1 is to be investigated, where

$$G_1(s) = \frac{1}{s^2 + 2as + 5a}$$
$$G_2(s) = \frac{2}{s}$$

The state space model of the plant needs to be deduced and analyzed.

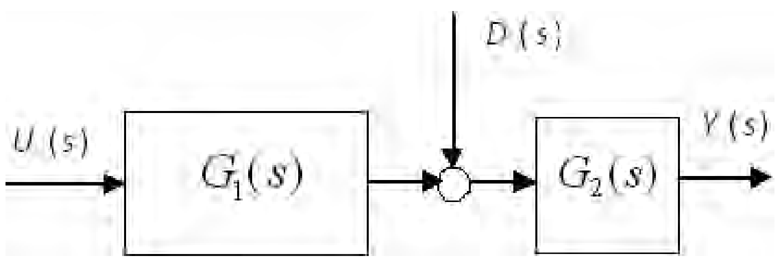

Figure 2.1: A 3rd-Order Plant

2.1 State-space Modeling

Based on the last 4 digits of my matriculation number, the parameter a is given by

$$a = 3.585 \tag{2.1}$$

Thus, in the simulation below, the transfer function for G_1 and G_2 will be assigned as

$$G_1(s) = \frac{1}{s^2 + 7.17s + 17.925} \tag{2.2}$$
$$G_2(s) = \frac{2}{s} \tag{2.3}$$

To determine the overall state-space model, we consider $G_1(s)$ (with order two) and $G_2(s)$ (with order one) together, the overall order of this plant will then be three. Thus, three state variables are needed to represent the full model, and they are not unique. We can arbitrary choose three state variables for our own convenience. Hence, we assign the output from G_2 as the first state variable, x_1, and output from G_1 as the second state variable, x_2. The last state variable, x_3, resides inside the G_1 block and we choose it to be the derivative of x_2 for easy calculation.

From the transfer function G_1, we have

$$(s^2 + 7.17s + 17.925)X_2(s) = U(s) \tag{2.4}$$

Taking La-place transform on both sides and consider $x_2'' = x_3'$ and $x_2' = x_3$, we have

$$\begin{aligned} x_2''(t) + 7.17x_2'(t) + 17.925x_2(t) &= u(t) \\ x_3'(t) &= -7.17x_3(t) - 17.925x_2(t) + u(t) \end{aligned} \tag{2.5}$$

For state x_1, we know

$$sX_1(s) = 2X_2(s) + 2D(s) \tag{2.6}$$

$$x_1'(t) = 2x_2(t) + 2d(t) \tag{2.7}$$

Thus, by letting $x = \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix}^T$, we can obtain the full state representation of the system

$$\begin{aligned} \dot{x} &= \begin{bmatrix} 0 & 2 & 0 \\ 0 & 0 & 1 \\ 0 & -17.925 & -7.17 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u + \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix} d \,, \\ y &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} x \end{aligned} \tag{2.8}$$

where $A = \begin{bmatrix} 0 & 2 & 0 \\ 0 & 0 & 1 \\ 0 & -17.925 & -7.17 \end{bmatrix}$, $B_u = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$, $B_d = \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix}$, and $C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$.

2.2 Plant Dynamics Analysis and Simulation

At this section, the obtained state-space plant will be analyzed and simulated with non-zero initial state. The bode plot and zero input response with the initial condition of $x_1 = x_2 = x_3 = 1$ are shown in Fig. 2.2 and Fig. 2.3 respectively. From the zero input response plot, we can see that the output does not go back to zero, which tell us the system is not asymptotically stable. The three system poles are at 0, $-3.585 \pm 2.2523i$. We see that two of them are stable as their real part are negatives, but the zero pole might cause instability to the system. However, since the zero eigenvalue (pole) has no Jordan block of order 2 or higher, it is known that the system is stable in the sense of Lyapunov.

The output response to a step input and the output response to a constant disturbance is shown in Fig. 2.4 and Fig. 2.5. They are both unbounded as time goes to infinity. Thus, the system is not bounded-input-bounded-output (BIBO) stable. In order to make the system stable and maneuverable, state feedback control is needed. In the next chapter, we will investigate the state feedback control law using LQR method in order to make the overall closed-loop system stable and meet design requirements.

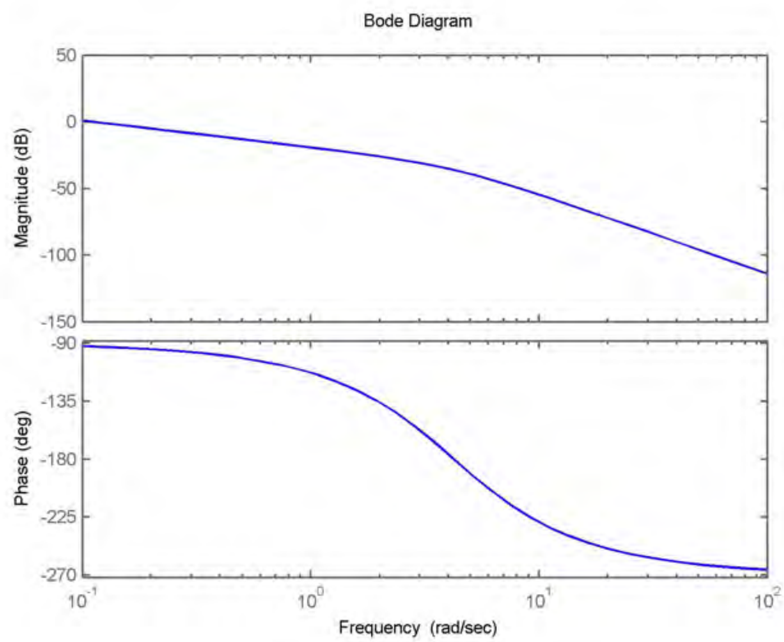

Figure 2.2: Bode plot of the original open-loop system

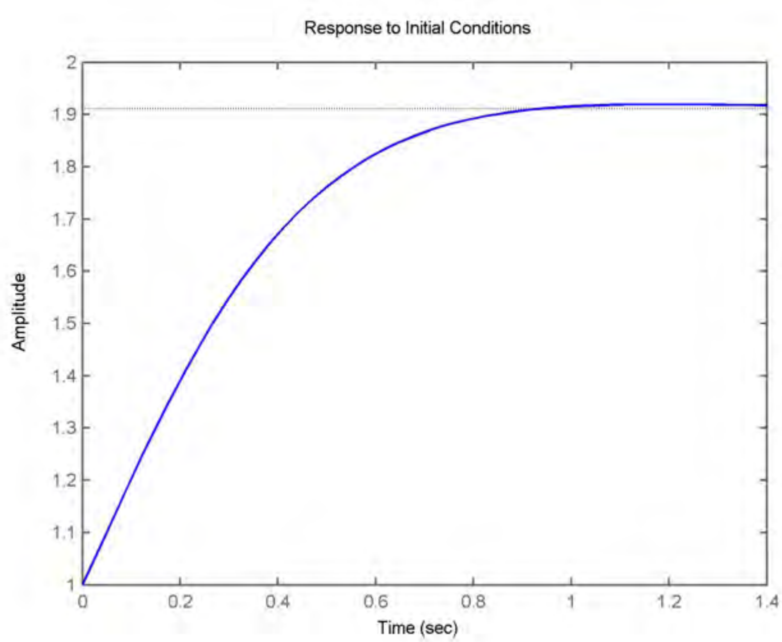

Figure 2.3: Zero input response with initial condition [1 1 1]

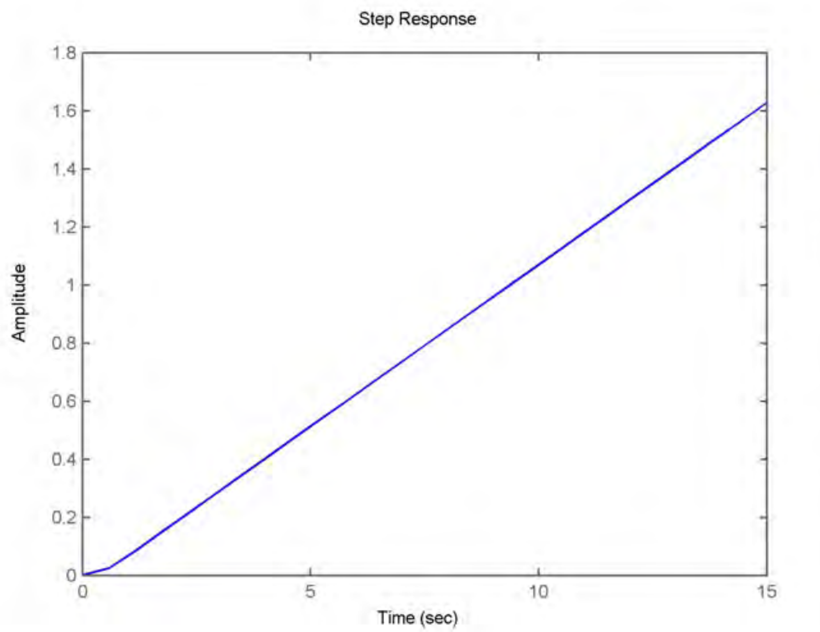

Figure 2.4: Output response to a constant input, $U(s) = \frac{1}{s}$

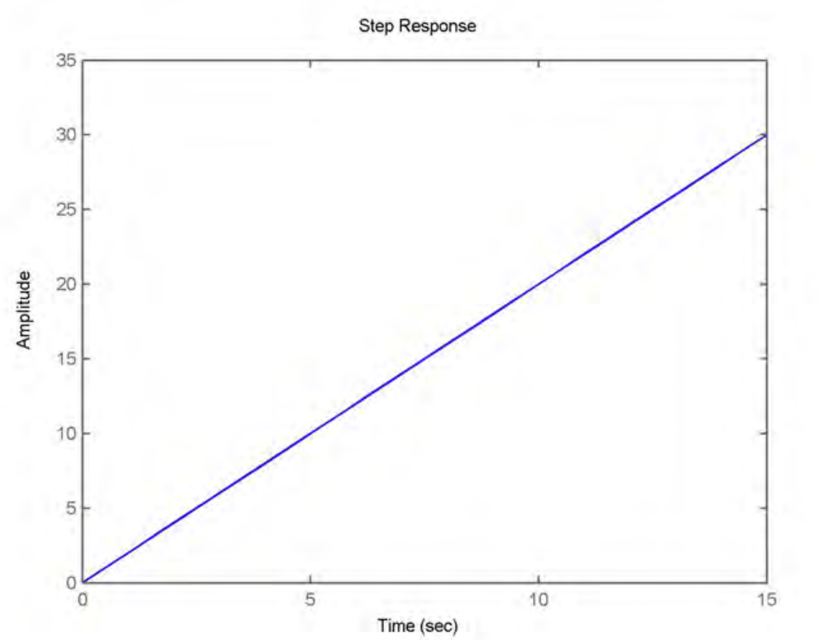

Figure 2.5: Output response to a constant disturbance, $D(s) = \frac{1}{s}$

Chapter 3

State Feedback Design using Linear Quadratic Regulator (LQR)

In this chapter, a state feedback controller is to be implemented using the LQR method. By adopting LQR method, our target is to minimize the cost function,

$$J = \frac{1}{2}\int_0^\infty (x^T Q x + u^T R u)\, dt \tag{3.1}$$

and

$$u = -Kx \tag{3.2}$$

Here, K is called the state feedback gain. With a non-zero initial condition, the state variables are expected to go from their initial values to zero as time goes to infinity. However, to make sure that the optimal control system is stable and realizable, three criteria need to be met:

1. The original system, (A, B), must be controllable;
2. The pair (A, H) must be observable, where $HH^T = Q$;
3. The weighting matrices Q and R must be both symmetric, semi-positive definite and positive definite respectively.

Then, the control law will be

$$u = -Kx = -R^{-1}BPx \tag{3.3}$$

where P is the solution of the Riccati equation

$$A^T P + PA + Q - PBR^{-1}B^T P = 0 \tag{3.4}$$

3.1 State Feedback Controller Design

In order to implement LQR state feedback controller, first of all, the original system must be controllable. Thus, we check the controllability matrix,

$$[B_u, AB_u, A^2B_u] = \begin{bmatrix} 0 & 0 & 2 \\ 0 & 1 & -7.17 \\ 1 & -7.17 & -33.4839 \end{bmatrix} \tag{3.5}$$

which is of full rank, and thus controllable. Since Q and R are under the choice of the designer, we just need to choose them wisely to fulfill the requirements. For this case, the simplest format of Q is a diagonal matrix with positive entries, and R is just a one dimensional constant. We set up the base case for this problem as

$$Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.6}$$

$$R = 1 \tag{3.7}$$

By solving the Riccati equation shown above, or by substituting A, B_u, Q, R into the MATLAB function *LQR*, we obtained the state feedback gain,

$$K = \begin{bmatrix} 1 & 0.8291 & 0.183 \end{bmatrix} \tag{3.8}$$

In addition, we set the initial state variables $x = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^T$, so that we can examine how the three state variables drop to zero dynamically.

The simulation result is plotted in Fig. 3.1. As expected, all three state variables converge to zero at about $t = 30$s. Thus, the closed-loop system is asymptotically stable which meets the requirement. Note that the control signal, u, is also plotted for comparison with the later cases.

3.2 Change Weighting Factor R

In this section, two new cases with small adjustments to the original controller have been tested. For the first controller, R is decreased by 10 times from its original value. As for the second controller, R is increased by 10 times from the original value. Table 3.1 shows the computed feedback gains K with different value of R. The simulation results are shown in Fig. 3.2 and Fig. 3.3 respectively.

Weighting factor, R	Controller gains, K
0.1	[3.1623 2.9330 1.0321]
1	[1.0000 0.8291 0.1830]
10	[0.3162 0.2555 0.0425]

Table 3.1: Different control gains K with different weighting factor R

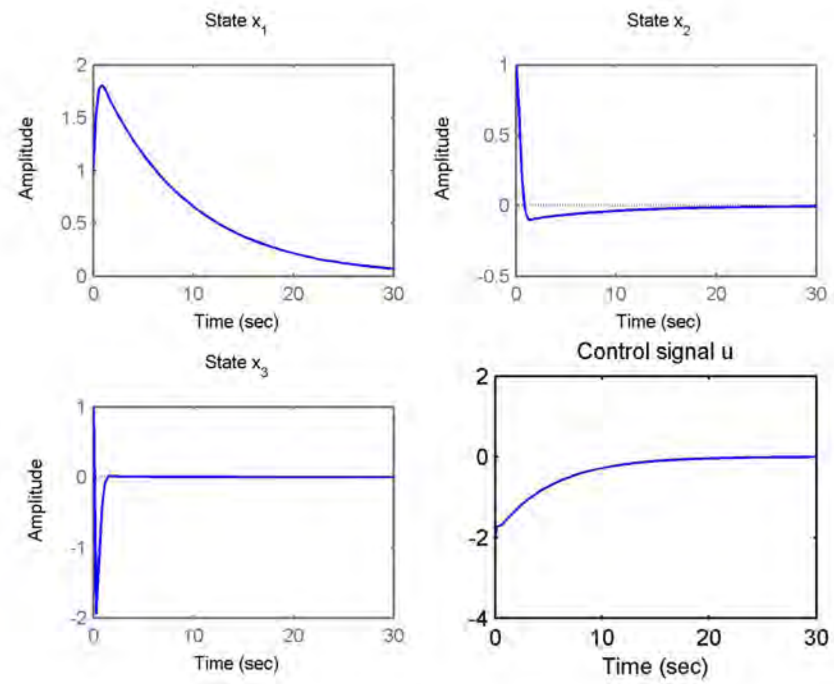

Figure 3.1: State response for the base case

The simulation results shows that when R is much smaller than the original value, the states x are able to converge to zero at a faster response. However, control signal, u, is large as compared to the original simulated results.

It could be explained theoretically, as R is the weighting factor of input u with respect to the cost function, if R is small, u can be very large because it is no longer the major cost concern (see Fig. 3.2). On the other hand, if R is large, u becomes very costly and the controller will try to keep it small (see Fig. 3.3).

If we compare the feedback gains calculated for the above two cases, the former has a much larger gain compared to the latter. Without losing generality, we can say that a larger feedback gain will cause faster system response. However, non-ideal conditions such as noises, non-linearity and saturation will pose problems if the feedback gain is too large.

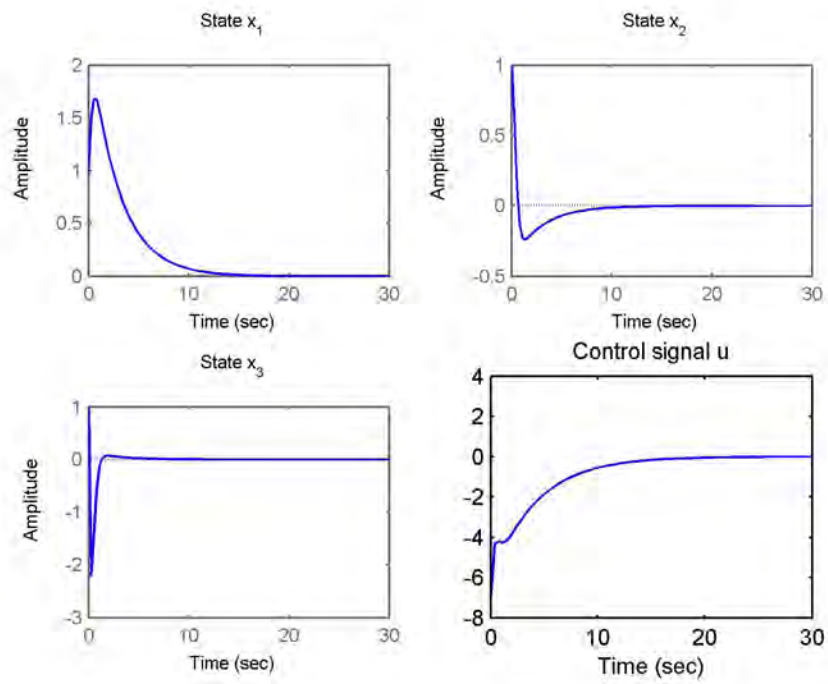

Figure 3.2: State response when $R = 0.1$

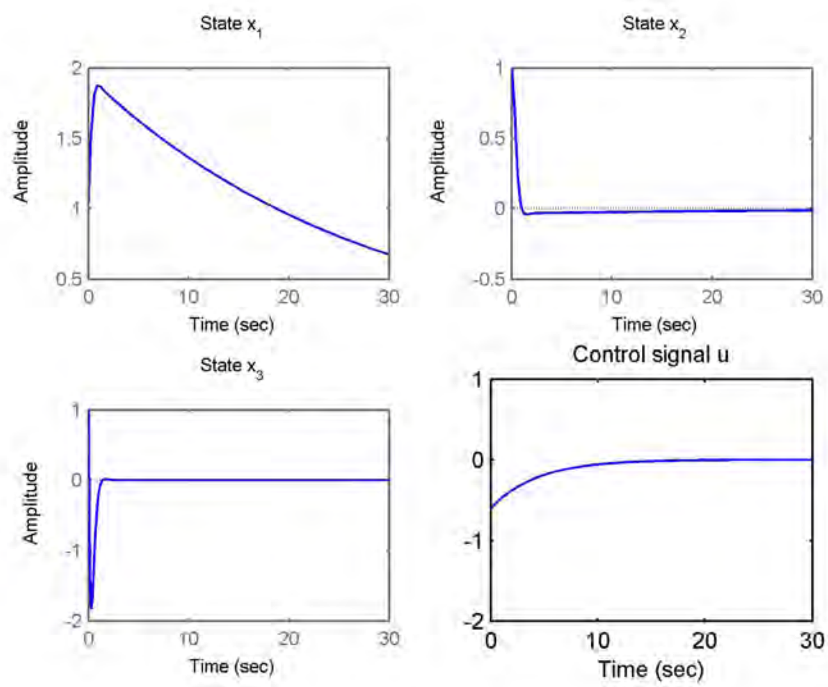

Figure 3.3: State response when $R = 10$

3.3 Change weighting factor Q

The values on the diagonal of matrix Q represent the weighting factors of their corresponding state variables. For example, putting 50 times more weight on q_{11} would means that the first state variable is 50 times more important than the other two state variables in the cost function.

Comparing the system response against the original controller where $Q = I_3$ (see Fig. 3.1), three new controllers are design with 100 times weighting on each states. The controller gains correspond to each of the controller are listed in Table 3.2.

Weighting factor, Q	Controller gains, K
$diag(100, 1, 1)$	[10.000 7.5930 1.0516]
$diag(1, 100, 1)$	[1.0000 3.3365 0.5165]
$diag(1, 1, 100)$	[1.0000 1.3614 5.2450]

Table 3.2: Different control gains K with different weighting factor Q

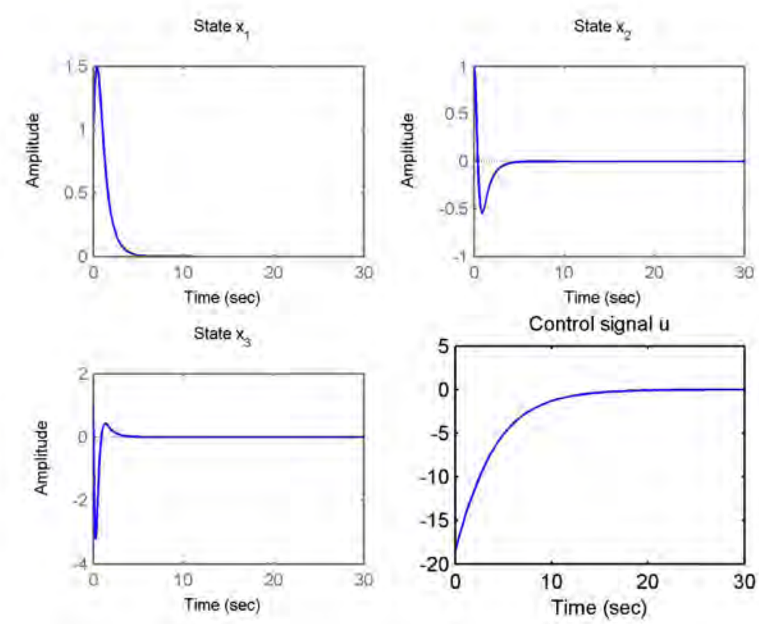

Figure 3.4: State response when $Q = diag(100, 1, 1)$

Based on the simulation results shown in Fig. 3.4, we see that by amplifying the weight of state x_1, the system's states converged to the zero states rather quickly compared to the original controller. The control signal is larger, due to the fast system response.

Theoretically, if the disturbance $d(t)$ is ignored, the system output is equivalent to the first state variable, $x_1(t)$. Thus, giving the first state variable a higher cost weighting is indeed a good solution since the output dynamic performance is our main concern. Thus, it is

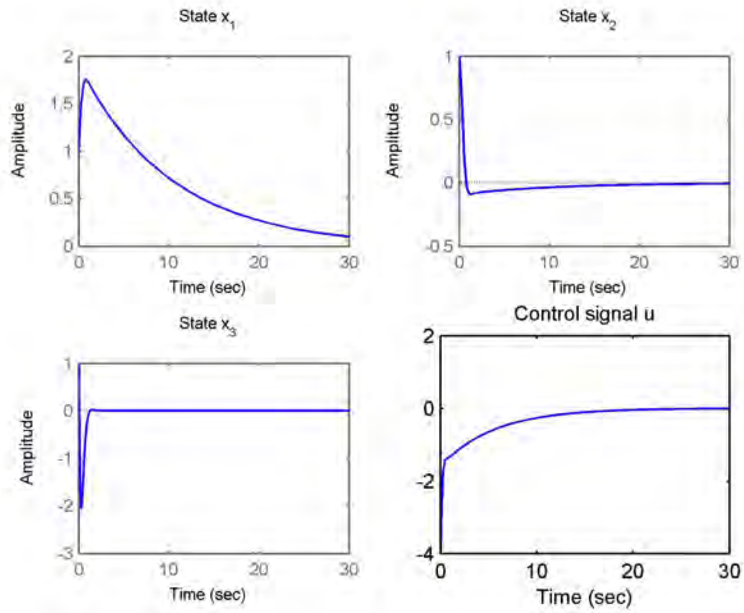

Figure 3.5: State response when $Q = diag(1, 100, 1)$

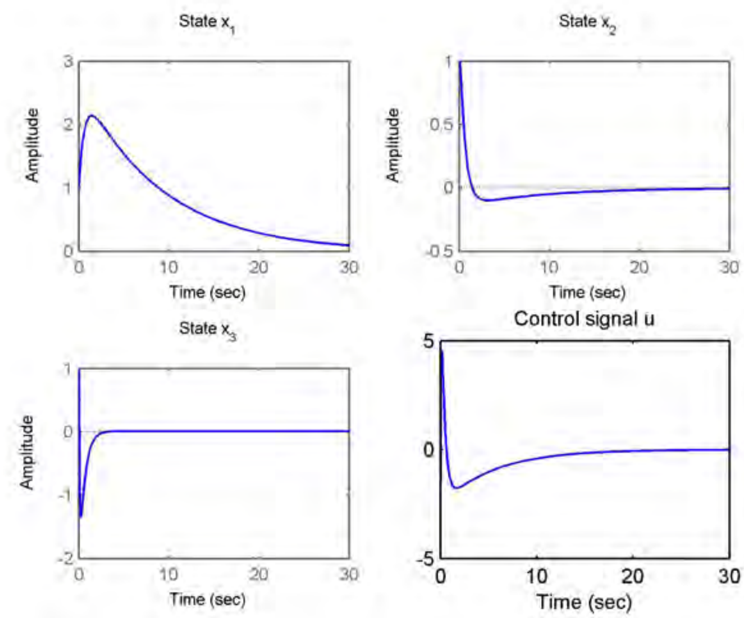

Figure 3.6: State response when $Q = diag(1, 1, 100)$

reasonable and expected to see a better (faster) system response in the simulation.

According to Fig. 3.5 and Fig. 3.6, we see that for $Q = diag(1, 100, 1)$ and $Q = diag(1, 1, 100)$, the system response are slower compared to the previous controllers. This is due to the increment of the weighting of state x_2 or x_3 which in turn increases the damping of the system. The controller efforts are smaller accordingly, too.

Chapter 4

State Observer Design using Pole Placement

In the previous chapter, we are assuming that the state variables x_1, x_2 and x_3 are completely measurable. However, in most practical applications, not all state variables can be measured directly. In this chapter, the idea of state estimation is implemented. A state observer will be built to estimate the state variables based on our knowledge of input, output, and plant model.

For the ease of investigation, we need to first design a state feedback controller using the pole placement method. As the plant we are studying now is a 3rd order system, the desired closed-loop characteristic equation, $\phi_d(s)$ must be 3rd order, too. The choice of pole placement can be done based on the 2nd order dominant pole strategy, with two conjugate stable poles are firstly selected, then the third pole is arbitrary chosen to be far away from the origin, so that it decays fast enough to be ignored.

we choose $\lambda_1 = -1+j$, $\lambda_2 = -1-j$ and $\lambda_3 = -10$. Thus, the desired closed-loop polynomial,

$$\phi_d(s) = (s+1-j)(s+1+j)(s+10) = s^3 + 12s^2 + 22s + 20$$

Since the controllability matrix has already been calculated in the previous chapter, we can apply Ackermann's formula to obtain the feedback gain:

$$\begin{aligned} K^T &= \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 2 \\ 0 & 1 & -7.17 \\ 1 & -7.17 & -33.4839 \end{bmatrix}^{-1} \phi_d(A) \\ &= \begin{bmatrix} 10 \\ 4.075 \\ 4.83 \end{bmatrix} \end{aligned} \tag{4.1}$$

4.1 Design of Observer by Pole Placement

We consider a full-order observer:

$$\dot{\hat{x}}(t) = A\hat{x}(t) + B_u(t) + L[y - C\hat{x}] \tag{4.2}$$

where $\hat{x}(t)$ denotes the estimation of x(t) and L is the observer feedback gains. Let the estimation error be $\tilde{x} = x - \hat{x}$. Then,

$$\begin{aligned}\dot{\tilde{x}} = \dot{x} - \dot{\hat{x}} &= Ax + B_u u - \{A\hat{x} + B_u u + L(y - C\hat{x})\} \\ &= A(x - \hat{x}) - L(Cx - C\hat{x}) \\ &= A(x - \hat{x}) - LC(x - \hat{x}) \\ &= A(\tilde{x}) - LC(\tilde{x}) \end{aligned} \tag{4.3}$$

If we consider

$$\begin{aligned}\det[sI - (A - LC)] &= \det[sI - (A - LC)]^T \\ &= \det[sI - (A - LC)^T] \\ &= \det[sI - (A^T - C^T L^T)] \\ &= \det[sI - (\tilde{A} - \tilde{B}\tilde{K})] \end{aligned} \tag{4.4}$$

where $\tilde{A} = A^T$, $\tilde{B} = C^T$, and $\tilde{K} = L^T$. It has changed to the same form as the normal state feedback using pole placement problem, where the state x is replaced by the estimation error $\tilde{x}$.

By the pole placement theorem, L can be found by arbitrarilty chosen eigenvalues of $\det[sI - (A - LC)] = 0$. The choice of observer pole placement is a trade-off between speed and constraints imposed by noise, saturation and non-linearity. A simple guideline is to place the observer poles 3-5 times faster than the control poles. Thus, we adapted this guideline and the observer closed-loop characteristic equation is chosen as follows:

$$\begin{aligned}\phi_o(\lambda) &= (\lambda + 3 + 3i)(\lambda + 3 - 3i)(\lambda + 30) \\ &= \lambda^3 + 36\lambda^2 + 198\lambda + 540 \end{aligned} \tag{4.5}$$

Using Ackermann's formula, we obtain

$$L^T = \begin{bmatrix} 28.8300 & -13.3180 & 107.1015 \end{bmatrix} \tag{4.6}$$

By using this feedback gain for the observer control loop with initial condition $x(0) = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^T$ and $\tilde{x}(0) = \begin{bmatrix} -1 & -1 & -1 \end{bmatrix}^T$, the zero input response, the state estimation error and the control effort of the system are plotted in Fig. 4.1. We can see that the estimation error for all three state variables converges to zero, and its dynamics is much faster than the output response.

4.2 Effects of Observer Pole Position

Next, we will investigate the effects of observer pole position on state estimation error and closed-loop control performance. In this section, two different pole placements wil be simulated. First, we try to put the observer pole at a position at the exact closed-loop pole

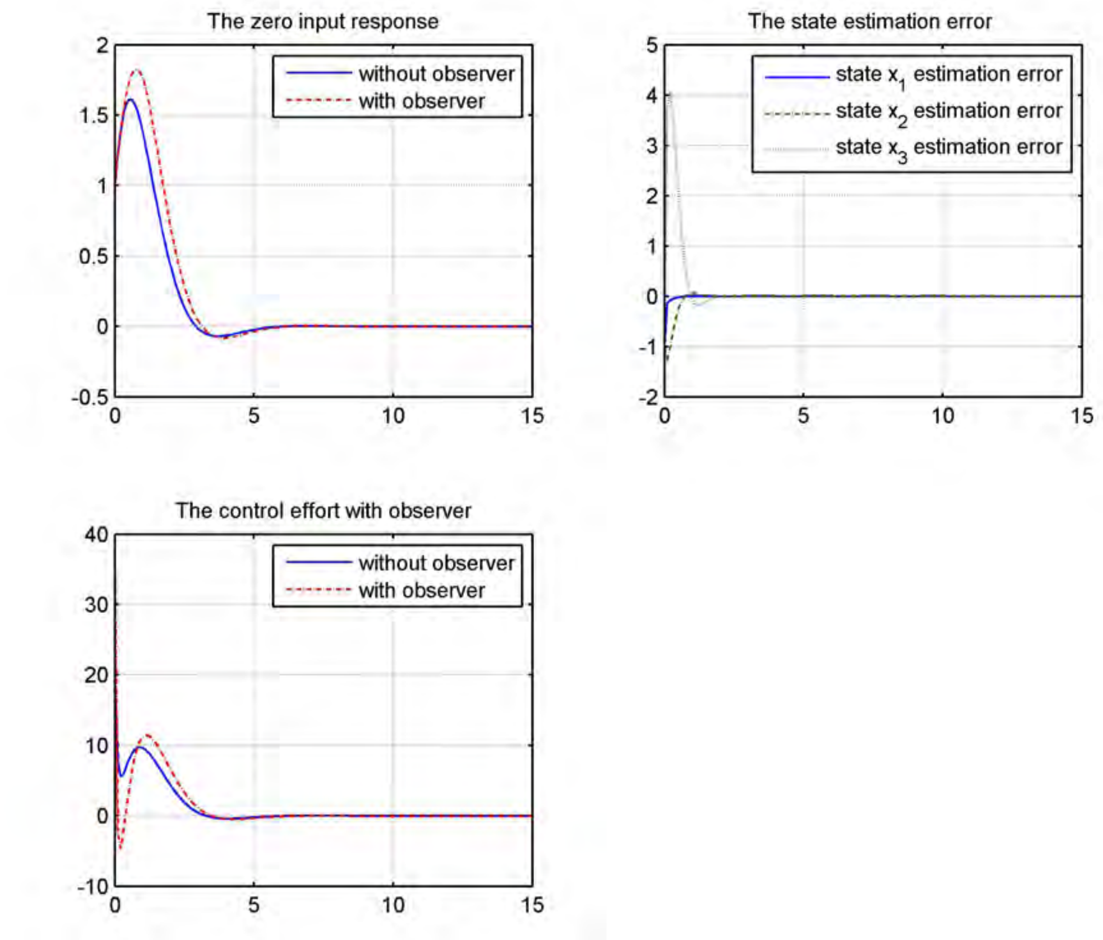

Figure 4.1: Zero input response, state estimation error and control effort of the system with observer

position, where $\lambda_1 = -1 + i$, $\lambda_2 = -1 - i$, and $\lambda_3 = -10$. Again, using the Ackermann's formula, we can get

$$L_1^T = \begin{bmatrix} 4.8300 & -15.2781 & 76.2547 \end{bmatrix} \tag{4.7}$$

Next, we arbitrary selected fast poles, in this case, 10 times faster than the exact closed-loop pole position, where $\lambda_1 = -10 + 10i$, $\lambda_2 = -10 - 10i$, and $\lambda_3 = -100$. Then we have

$$L_2^T = \begin{bmatrix} 112.8 & 686.5 & 4066.3 \end{bmatrix} \tag{4.8}$$

Both controller with observer feedback gains L_1^T and L_2^T are simulated, and the results are compared to the original feedback gains L^T in Fig. 4.2 and Fig. 4.3.

According to Fig. 4.2, the zero input response of the system for using the exact closed-loop pole positions is very slow and with large overshoot. However, as the observer pole positions moves further from the origin (more negative poles), we notice that the performance of the

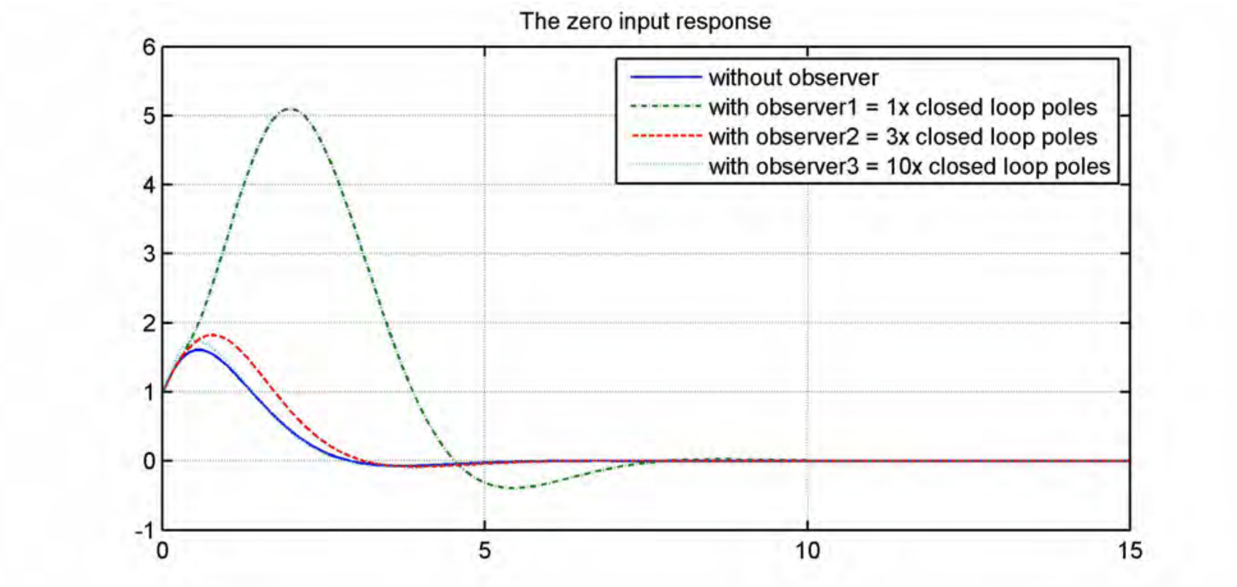

Figure 4.2: Zero input response of the system for different observer poles

observer is getting nearer to the real system without observer.

We notice from Fig. 4.3 that the error dynamics of the system performed better (errors decay faster) when the observer poles are further away from the origin. Here we can deduce that the observer poles have to be fast enough so that the control performance can be assured. It seems that the faster the observer poles, the better the performance.

In practical applications, however, sensor or computer sampling frequency may not be fast enough for the designed dynamics, and noises and measurement errors will be scaled up by large observer feedback gain, which could cause bad performance and instability. Thus, placing observer poles 3-5 times faster than the control poles is good enough based on many researchers' experience.

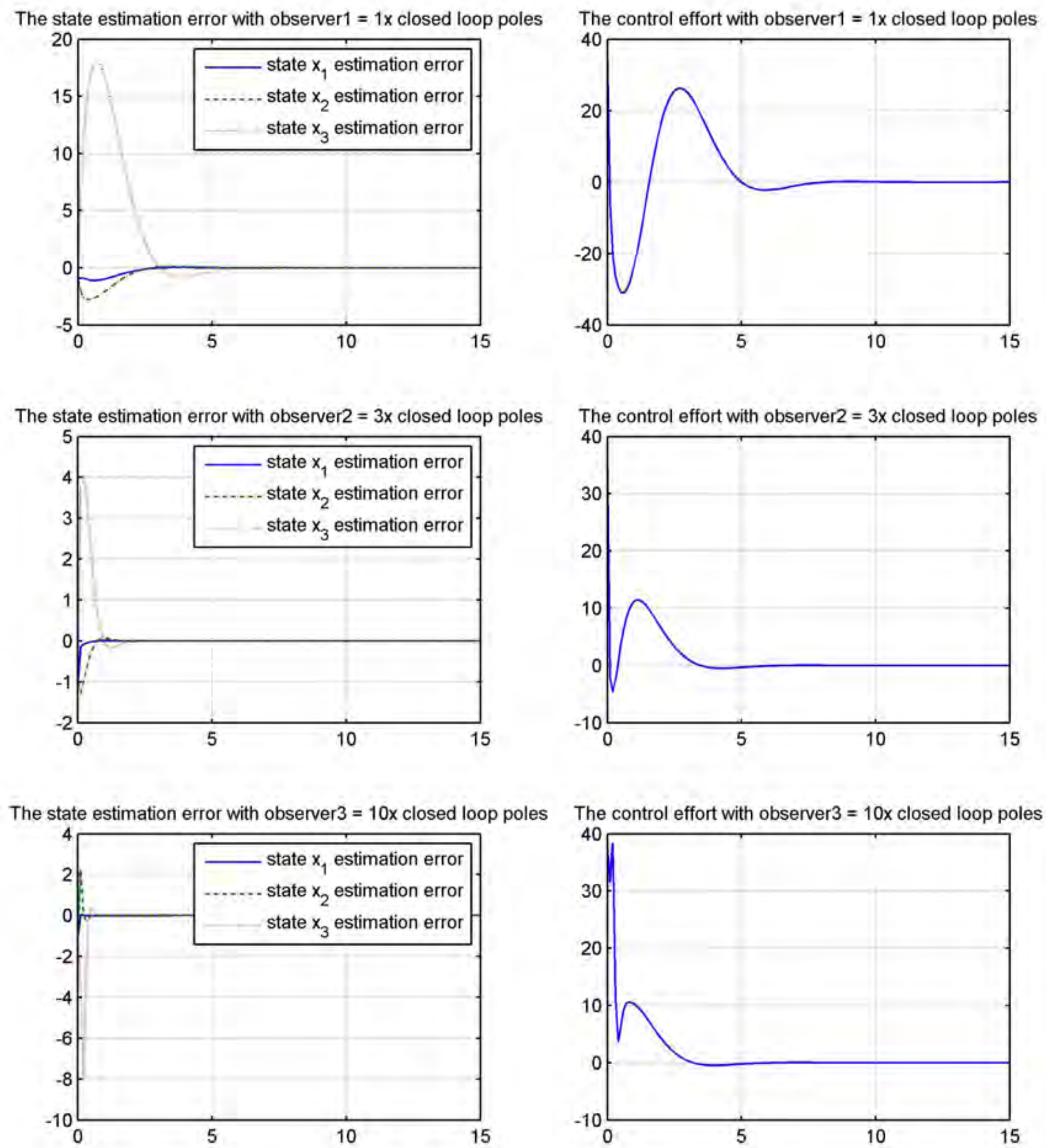

Figure 4.3: State estimation error and control effort of the system for different observer poles

Chapter 5

Servo Control

From the previous chapter, the system transient performance is adjusted by state feedback and state feedback is designed by LQR or pole placement methods. However, all these have not taken care of the steady-state performance of the system. In this chapter, our target is to make the output of the system, $y(t)$, tracks a step reference input $r(t)$ (asymptotic tracking), despite a step disturbance $d(t)$ (disturbance rejection).

5.1 Servo Controller Design

Based on the general single input single output servo control methodology, we need to determine the servo mechanism followed by the stabilization of the system. In our simulation, the reference signal and the disturbance signal are both step function. Thus, their Laplace representations are of $\frac{1}{s}$ based. However, one challenge is that $d(t)$ or $D(s)$ does not go directly into the output in our case. Thus, modification can be made by first converting the plant block diagram into an equivalent plant that we are more familiar with as shown below:

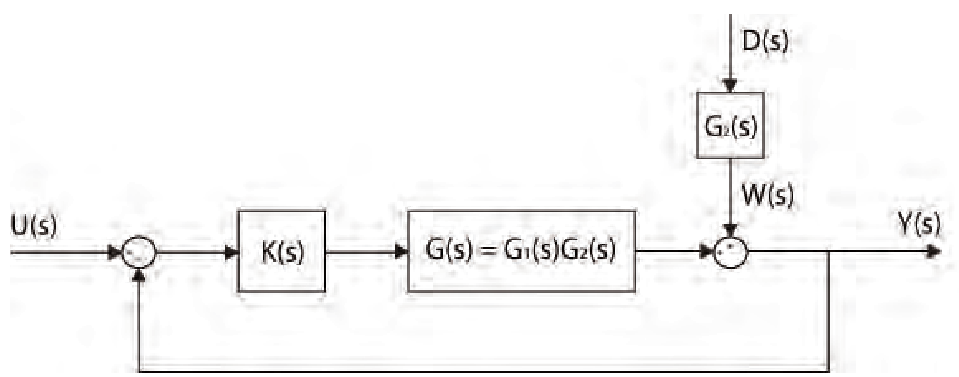

Figure 5.1: Servo control loop and blocks

Now, the equivalent disturbance $W(s) = G_2(s)D(s)$ is added directly at the output of the system with a Laplace representation of $\frac{1}{s^2}$ type. The standard procedures for servo control can be applied. The plant transfer function is given by

$$G(s) = G_1(s)G_2(s) = \frac{2}{s^3 + 7.17s^2 + 17.925s} = \frac{N(s)}{D(s)} \tag{5.1}$$

The servo mechanism is

$$\frac{1}{Q(s)} = \frac{1}{s^2} \tag{5.2}$$

which takes the least common denominator of unstable modes of $R(s)$ and $W(s)$. We cascade the servo mechanism with the plant $G(s)$ to form the generalized plant, $\tilde{G}(s)$:

$$\tilde{G}(s) = \frac{1}{Q(s)}G(s) = \frac{N(s)}{Q(s)D(s)} = \frac{2}{s^5 + 7.17s^4 + 17.925s^3} \tag{5.3}$$

We can now design a proper compensator to stabilize the generalized plant since G(s) is a strictly proper with $N(s)$ and $D(s)$ coprime, and no roots of $Q(s)$ is a zero of the plant G(s). Since the generalized plant is of order 5, the controller $\tilde{K}(s)$ needs to be of order 4. Let

$$\tilde{K}(s) = \frac{\tilde{B}(s)}{\tilde{A}(s)} = \frac{B_4s^4 + B_3s^3 + B_2s^2 + B_1s + B_0}{A_4s^4 + A_3s^3 + A_2s^2 + A_1s + A_0} \tag{5.4}$$

and the desired characteristic polynomial be

$$P_c = (s+1)^9 \tag{5.5}$$

Thus, controller $\tilde{K}$ can be calculated from:

$$\tilde{B}N + \tilde{A}DQ = P_c \tag{5.6}$$

The solution of the above equation is

$$\begin{aligned}\tilde{A}(s) &= s^4 + 1.83s^3 + 4.9539s^2 + 15.6778s - 75.2084\\ \tilde{B}(s) &= 192.1099s^4 + 716.0552s^3 + 18s^2 + 4.5s + 0.5\\ \tilde{K}(s) &= \frac{192.1099s^4 + 716.0552s^3 + 18s^2 + 4.5s + 0.5}{s^4 + 1.83s^3 + 4.9539s^2 + 15.6778s - 75.2084}\end{aligned}$$

Thus, the complete controller is

$$K(s) = \frac{\tilde{K}(s)}{s^2} = \frac{192.1099s^4 + 716.0552s^3 + 18s^2 + 4.5s + 0.5}{s^6 + 1.83s^5 + 4.9539s^4 + 15.6778s^3 - 75.2084s^2} \tag{5.7}$$

Substitute this $K(s)$ into Fig. 5.1, assign a constant reference of magnitude 1 and step disturbance of magnitude 0.1 at time $t = 50$s, the system is simulated as shown in Fig. 5.2

From the simulation, we see that the output signal goes to the reference point '1' precisely at steady-state, after both the step reference and disturbance are applied. However, the dynamic response of the system is not satisfying as large overshoots occurs. Theoretically, the occurance of large overshoots are due to the high order controller design. Thus, in order to design a good servo controller, the dynamic performance can be improved by using an alternate design method which lead to the following section.

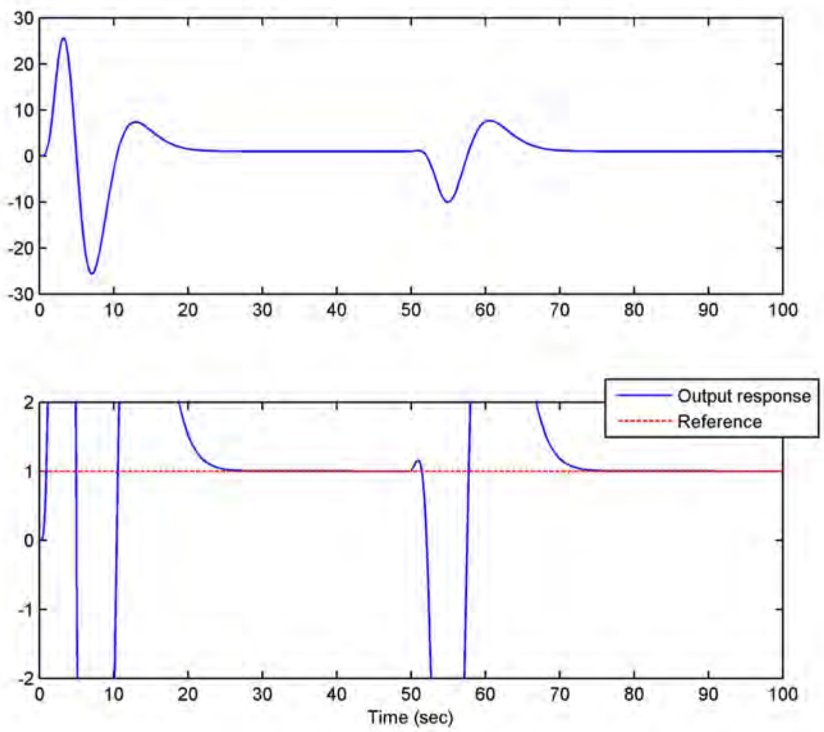

Figure 5.2: System response to step reference and disturbance

5.2 Reduced Order Servo Controller

In some system, we can find servo controllers in lower order than the one obtained by the general method shown in the previous section. Fortunately, for our system, since the transfer function $G(s)$ already has an integrator term, $\frac{1}{s}$, it means that the controller only needs another integrator, $\frac{1}{s}$, to realize the servo mechanism, $\frac{1}{s^2}$. Furthermore, there are two stable poles for $G(s)$ with characteristic polynomial $s^2 + 7.17s + 17.925$. We can therefore assign the same polynomial at the numerator of the controller such that they will cancel each other to reduce the order of the system. Note that stable pole zero cancellation will not affect system stability. Thus, we choose:

$$K(s) = \frac{B(s)}{A(s)} = \frac{(s^2 + 7.17s + 17.925)(B_1 s + B_0)}{s(A_3 s^3 + A_2 s^2 + A_1 s + A_0)} \tag{5.8}$$

This leads to the closed-loop characteristic polynomial to be

$$\phi_{cl}(s) = A_3 s^5 + A_2 s^4 + A_1 s^3 + A_0 s^2 + 2B_1 s + 2B_0 \tag{5.9}$$

Let the desired characteristic polynomial to be

$$(s+1)^5 = s^5 + 5s^4 + 10s^3 + 10s^2 + 5s + 1 \tag{5.10}$$

By comparing coefficients, We have

$$\begin{aligned} A_3 &= 1 \\ A_2 &= 5 \\ A_1 &= 10 \\ A_0 &= 10 \\ B_1 &= 2.5 \\ B_0 &= 0.5 \end{aligned}$$

Thus, Eqn. 5.8 becomes:

$$K(s) = \frac{B(s)}{A(s)} = \frac{2.5s^3 + 18.425s^2 + 48.3975s + 8.9625}{s^4 + 5s^3 + 10s^2 + 10s} \tag{5.11}$$

Substitute this $K(s)$ into Fig. 5.1 and assign a constant reference of magnitude 1 and a step disturbance of magnitude 0.1 at time equals to 50 s as previously simulated, the system is simulated as shown in Fig. 5.3.

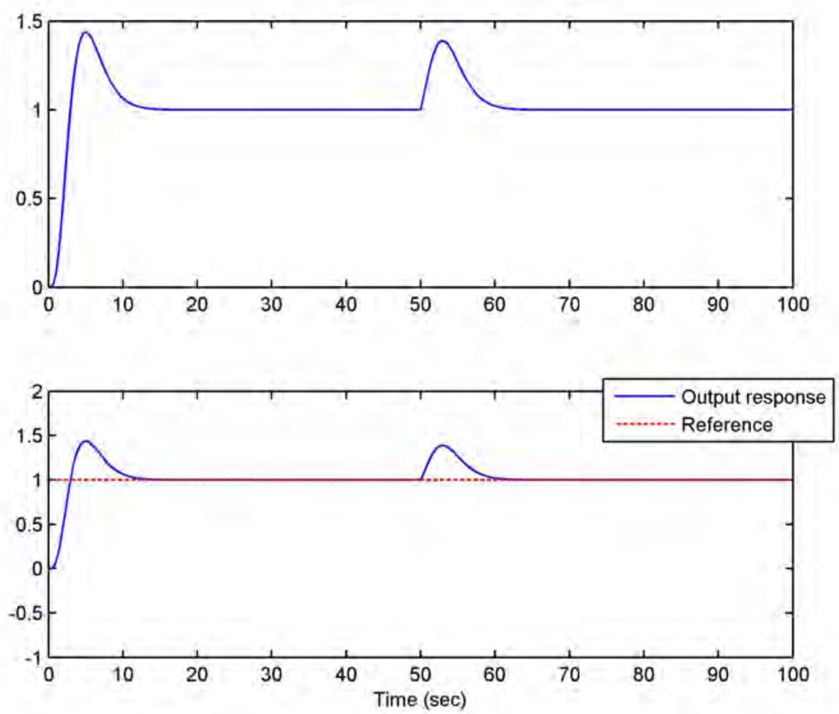

Figure 5.3: System response to step reference and disturbance

Comparing Fig. 5.3 and Fig. 5.2, the dynamic of the system shows a better response as it shows smaller overshoots and shorter settling time. The resultant behavior is due to the stable pole-zero cancellation which results in a lower-ordered closed-loop transfer function. Moreover, this method requires minimal amount of computation as the order of the controlled system is reduced, thus it is highly desirable and preferable.

Chapter 6

Conclusion

In this project, the given single input single output (SISO) plant is converted from a transfer function expression into the state-space representation. In chapter 2, the plant's open loop dynamics is analyzed and simulated. Simulation results show that it is neither asymptotically stable nor BIBO stable. Thus, state feedback is needed to stabilize the system. In chapter 3, we assume that all state variables are measurable. State feedback control using the LQR method is implemented and the effects of changing weighting factor Q and R are investigated. In chapter 4, we consider the case when the state variables are not measurable. A full order state observer controller is designed based on the pole placement method. The relationship between the observer pole location and system performance is discussed. In the last chapter, servo control which enable reference tracking and disturbance rejection is formulated. A general solution and a special reduced order solution are both proposed and simulated.