UNIVERSITY OF LOUISIANA AT LAFAYETTE

CONTROL SYSTEMS

MCHE 474

Lab 1

Author:
MATTHEW J. BEGNEAUD

Professor:
DR. MOSTAFA A. ELSAYED

FEBRUARY 16, 2016

Contents

List of Figures

Introduction

This lab was conducted by utilizing MATLAB, specifically the Simulink package, in order to analyze control signal diagrams. These diagrams consisted of gains, summation points, transfer functions, and feedback loops. These systems were analyzed with the main goal of obtaining the time response. The systems to be analyzed are shown in Figure 1.
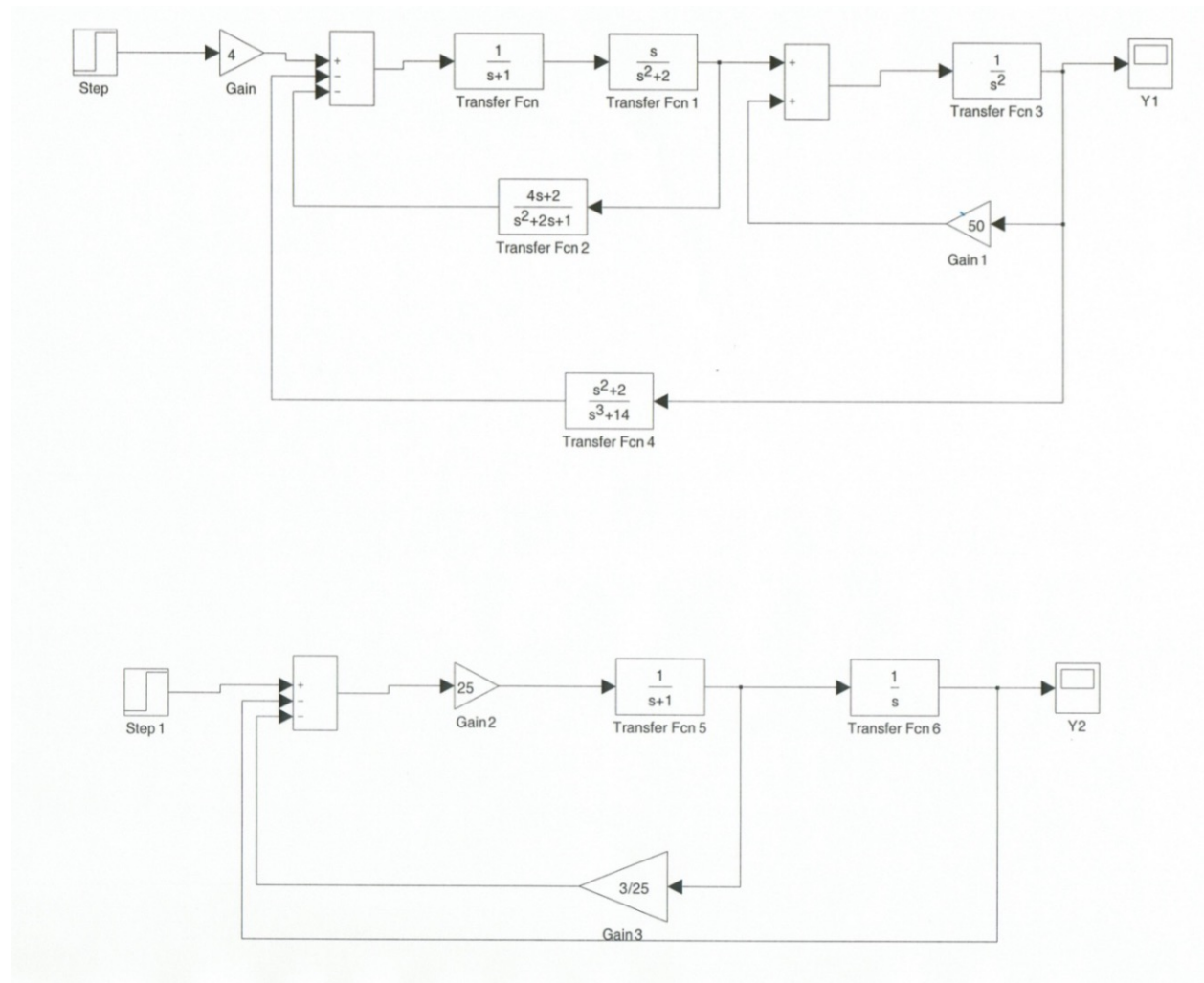

Figure 1: Signal Diagrams

Theory

Signals in control systems are manipulated by various means. Passing a signal through a gain simply increases, or decreases, a signal strength, which is quite useful. Signals can also go through feedback loops. Feedback loops can make the system more robust. For example, passing a signal to an actuator without any sensing cannot provide any feedback as to what the outcome is. This is detrimental if the actuator is not receiving enough power in order to carry out its desired task, or if the actuator has reached a goal position and needs to stop. Having sensors gives the system feedback which can ensure the system performs as intended. Feedback loops can be negative or positive depending on the desired function of the loop.

Signals are also commonly passed through transfer functions. A transfer function compares the output to the input of a linear system, time-variant systems. Transfer functions are in the frequency domain, which correlates to the Laplace domain. The time domain expression of the output is typically of interest in control systems, and using the transfer function of a system can help to obtain the time-response of the system. The general form of a transfer function for a system with input $R(s)$ and output $Y(s)$ is shown in Equations 1 and 2 as follows,

$$H(s) = \frac{Y(s)}{R(s)} \tag{1}$$

$$H(s) = \frac{L[y(t)]}{L[r(t)]} \tag{2}$$

where the time domain representations of the system input and output are $r(t)$ and $y(t)$, respectively.

Systems can also be described by their status as either stable or unstable. A stable system eventually approaches and tends to stay at some constant settling value known as a steady state value, as shown in Figure 2, until some external disturbance acts on it. Once that external disturbance is removed, the system returns to the steady state value. Unstable systems will not settle on their own and will continue to change even when there are no external disturbances acting on the system.

Systems also have other characteristics, such as overshoot. Overshoot is when the system output exceeds the steady state value, and can be seen in Figure 2 when the system output exceeds the step input at first. The overshoot is often described as a percentage, which can be calculated with Equation 3 as follows:

$$O.S. = \frac{MaxValue - SteadyStateValue}{SteadyStateValue} * 100 \tag{3}$$

Systems can also be described by rise time, which is defined as the time taken to rise from a low value state to a higher value state, and helps describe a system's ability to respond to sudden inputs. A system also has a characteristic settling time which is the time taken for the system to reach a state that lies within a specified range of error after responding to a step input. The system depicted in Figure 2 can be seen to reach a state which lies within a specified range of error after some time, for example.
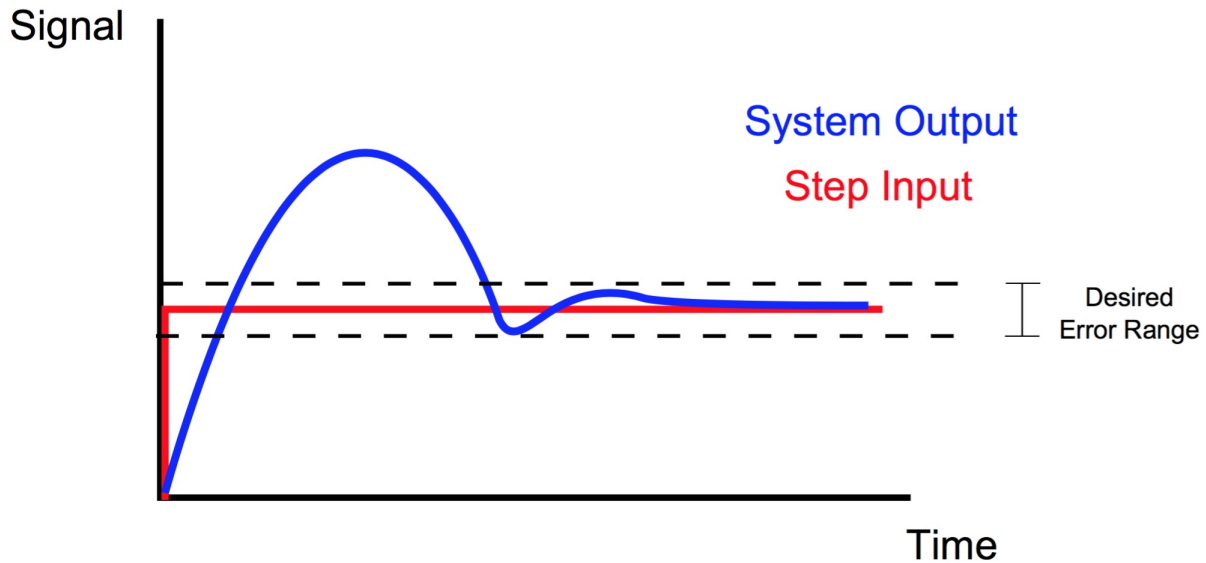
Figure 2: Example of Stable System

Procedure & Analysis

The first part of the lab was to obtain the time response of the two given systems. To do so, the systems were first modeled in Simulink, as shown in Figure 3. MATALB was then used to obtain the time response to these systems. The time response to the first system is shown in Figure 4, and the time response to the second system is shown in Figure 5. A MATLAB script was also written to calculate the percent overshoot of the systems. As can be seen, the first system is unstable because it does not approach a steady state value; therefore, the first system has no percent overshoot as indicated after running the MATLAB script, the results of which are shown in Figure 6. It can be seen that the second system is stable because it reaches a steady state value. The system has a percent overshoot of about 25%.

Another system was then described by only its transfer function in the second part of the lab. The transfer function was then analyzed using a MATLAB script in order to obtain various system characteristics. The characteristics obtained after running the MATLAB script are shown in Figure 7 and the plot of the system time response is shown in Figure 8. The system was calculated by the MATLAB script to have a percent overshoot of about 10.8%, while the percent overshoot determined by analyzing the plot was found to be about [].
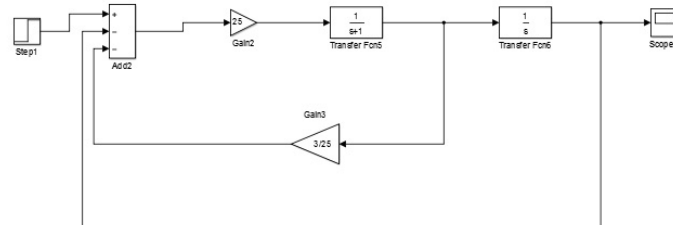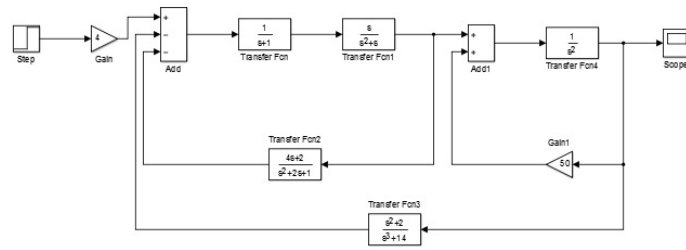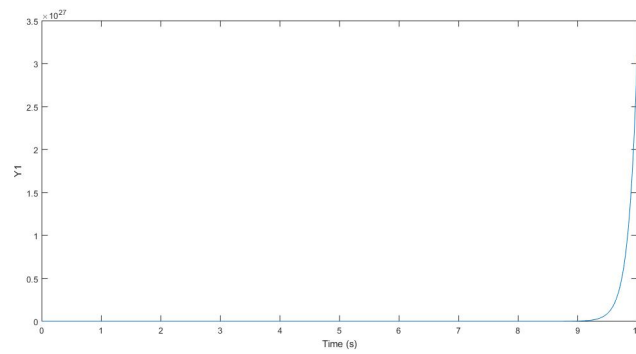
Figure 3: Simulink Models

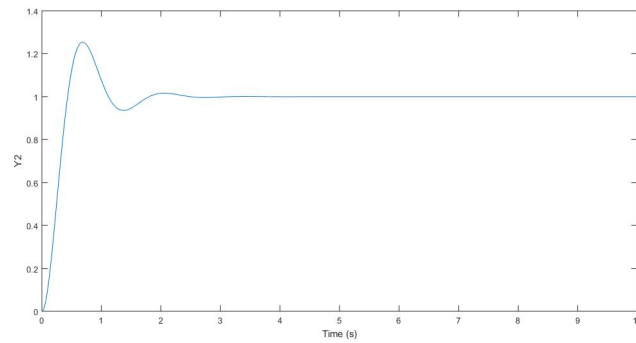
Figure 4: Lab Part 1 System 1 Time Response

Conclusion

Figure 5: Lab Part 1 System 2 Time Response

Figure 6: Lab Part 1 Commandline

```
ans =

        RiseTime: 0.3121
    SettlingTime: 1.0213
     SettlingMin: 3.2599
     SettlingMax: 3.9904
       Overshoot: 10.8433
      Undershoot: 0
            Peak: 3.9904
        PeakTime: 0.6631

>>
```
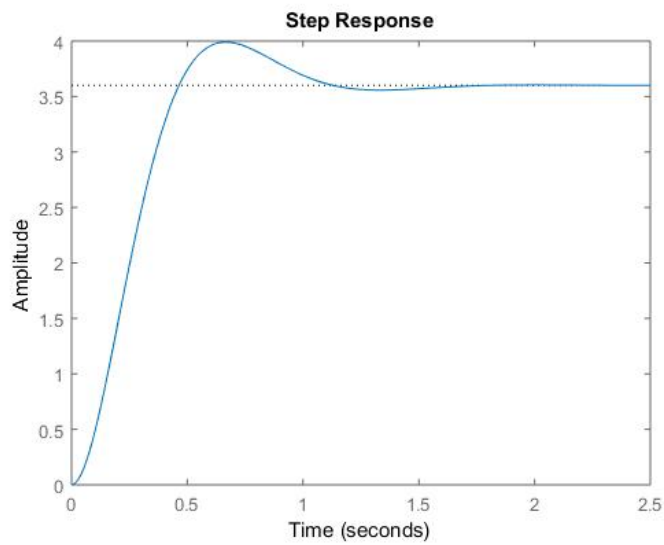
Figure 7: Lab Part 2 Commandline

Figure 8: Lab Part 2 Time Response