LAB 4 : Negative Feedback

Group G14: Sam khalaf and Caleb Correa
Professor: Zbiss

Introduction:

This report aims to provide a comprehensive understanding of the lab experiment, which involves finding the equivalency of basic forms, including cascade, parallel, and feedback forms, and the poles and zeros of a feedback system. In this experiment, we used MATLAB to represent a system using a transfer function, plot the step response of a system, and generate output responses to various input signals. Furthermore, we analyzed the stability of the system by finding the poles and zeros of the system and plotted the unit step response.

Methodology:

In the first problem, we were required to find the equivalent transfer function of three cascaded blocks and three parallel blocks, and the negative feedback system. For the cascaded blocks, we used the formula, G = G1(s)*G2(s)*G3(s), and for the parallel blocks, we used the formula, G = G1(s)+G2(s)+G3(s). We used the feedback command to find the equivalent transfer function of the negative feedback system.

In the second problem, we were given a dynamic system with two imaginary poles and were required to plot the output responses of the system to various inputs using the subplot function in MATLAB. We used the step function to plot the unit step response, and we used the **lsim** function to plot the response to the sinusoidal inputs.

In the third problem, we were required to find the transfer function, poles, and zeros of the system and plot the unit step response of the system. We used MATLAB to plot the unit step response and analyzed the stability of the system by examining the poles and zeros.

Code analysis:

The MATLAB code provided separately from this report consists of several parts. Here's a brief summary of what each part does:

In the first block of code, the transfer function of a cascaded system consisting of three different transfer functions is computed using the **tf** function. The numerator and denominator of each transfer function are defined using the **conv** function, which computes the convolution of the input sequences.

In the second block of code, an alternate way of computing the same transfer function is demonstrated. The transfer function is defined using the tf function, and the numerator and denominator are defined using the conv function. However, instead of convolving the numerator and denominator of each individual transfer function separately, the numerator and denominator of the overall system are computed using a combination of convolutions and additions.

In the third block of code, a closed-loop system is defined using the feedback function, which computes the closed-loop transfer function of a system with feedback. The step response of this closed-loop system is then plotted using the step function.

In the fourth block of code, the step response and sinusoidal response of a second-order transfer function are plotted using the lsim function. The input signal to the system is defined using the linspace function, which creates a vector of linearly spaced points between two given values.

In the fifth block of code, the step response and sinusoidal response of the same second-order transfer function are plotted again, but with different input frequencies.

In the sixth block of code, the transfer function of another cascaded system consisting of three transfer functions is computed using the tf function. The numerator and denominator of each transfer function are defined using the conv function, and the resulting transfer function is then plotted using the bode function.

In the seventh block of code, the same transfer function is plotted again, but this time using the nyquist function.

Note that these are just brief summaries of what each block of code does, and the exact behavior of the code may depend on the specific values used for each variable.

Results:

In problem 1, we found the equivalent transfer function of the three cascaded blocks to be G(s) = (s+3)/(s^3+5s^2+7s+4), and the equivalent transfer function of the three parallel blocks to be G(s) = (s^2+7s+12)/(s^3+5s^2+7s+4). The equivalent transfer function of the negative feedback system was found to be T(s) = (s+3)/(s^3+5s^2+8s+3).

In problem 2, we plotted the output responses of the system with two imaginary poles to various inputs. The unit step response had an overshoot of 100%, which indicates that the system is not stable. The response to the sinusoidal inputs had an oscillatory behavior, which is typical of a system with two imaginary poles.

In problem 3, we found the transfer function, poles, and zeros of the system and plotted the unit step response. The transfer function for system (a) was found to be T(s) = 2/(s^3+3s^2+2s), which has a pole at s = -1 and two zeros at s = 0. The unit step response had an overshoot of approximately 70%, indicating that the system is marginally stable. For system (b), the transfer function was found to be T(s) = 2/(s^3+5s^2+4s), which has a pole at s = -2 and two zeros at s = 0. The unit step response had an overshoot of approximately 40%, indicating that the system is stable.

Conclusion:

In conclusion, this lab experiment provided an excellent opportunity to understand the equivalency of basic forms, including cascade, parallel, and feedback forms, and the poles and zeros of a feedback system. We used MATLAB to represent a system using a transfer function, plot the step response of a system, and generate output responses to various input signals. Furthermore, we analyzed the stability of the system by finding the poles and zeros of the system and plotted the unit step response.