Lab 02

Emre Can Şen - 21902516

Department of Electrical and Electronics Engineering, Bilkent University, 06800 Ankara, Turkey

1. Introduction

In this lab, system identification will be done on a DC motor in frequency domain. Firstly, the theoretical plot will be obtained with the given code and derived transfer function approximation from the previous lab. Then sinusoidal inputs with different frequencies will be applied to the system and by comparing the output and the input of the system, bode plot of the transfer function will be plotted. Finally, a delayed version of the transfer function will be used to plot the bode plot in order to account for the processing delays that become apparent in higher frequencies.

2. Laboratory Content

Part 1

In the first part, the transfer function that was used in the first lab was also used for this lab which was:

$$G(s) = \frac{13.583}{0.191989s + 1}$$

Using the given code Bode plot of the transfer function is plotted:

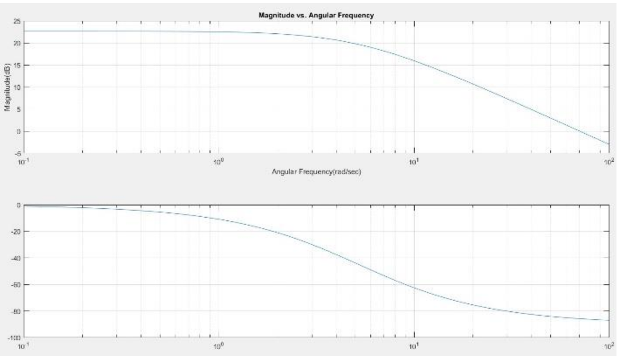

Figure 1- Bode Plot of Transfer Function

Part 2

In the second part, same as the pre-lab, the aim was to obtain the Bode plot of the transfer function by determining the response of the transfer function to different frequencies by plugging in a sinusoidal signal.

The form of the sinusoid was:

$$r(t) = 10 * \sin(\omega t)$$

Then sinusoidal with differing angular frequencies and durations were inputted to the system and the outputs and the input sinusoids were plotted in the same figure. The plots for given angular frequencies and durations were as follows:

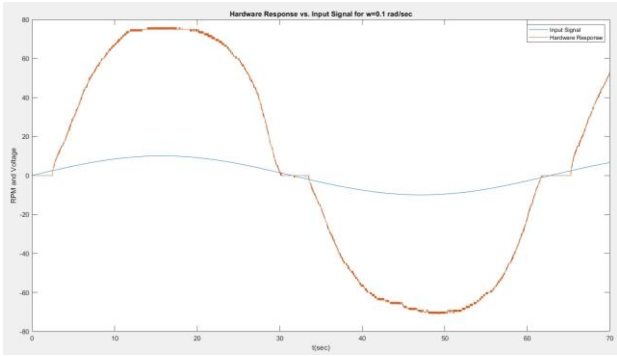

Figure 2- Hardware Response vs the Sinusoidal Input Where ω =0.1 rad/s

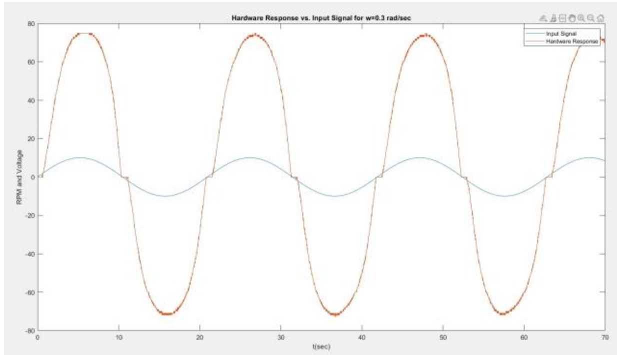

Figure 3- Hardware Response vs the Sinusoidal Input Where ω =0.3 rad/s

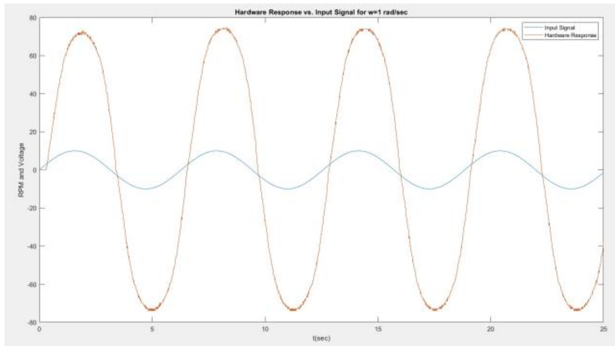

Figure 4- Hardware Response vs the Sinusoidal Input Where ω =1 rad/s

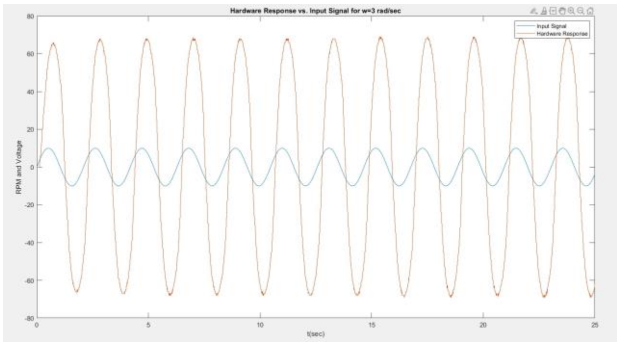

Figure 5- Hardware Response vs the Sinusoidal Input Where ω =3 rad/s

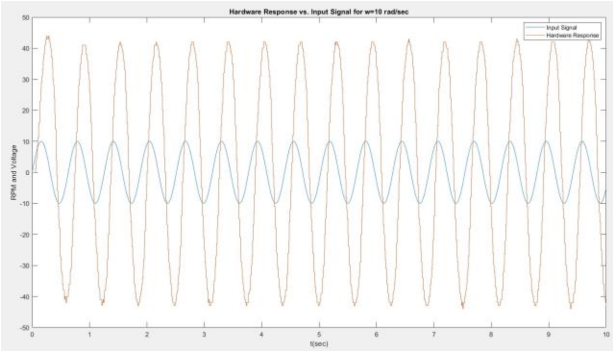

Figure 6- Hardware Response vs the Sinusoidal Input Where ω =10 rad/s

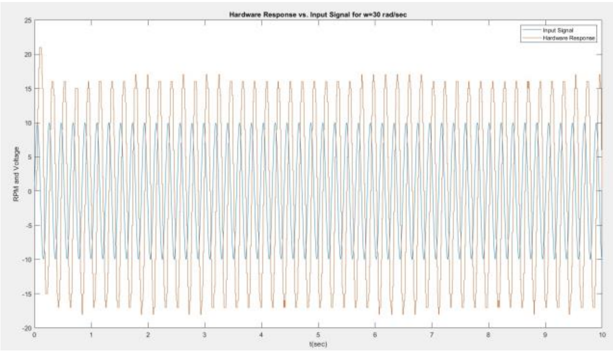

Figure 7- Hardware Response vs the Sinusoidal Input Where ω =30 rad/s

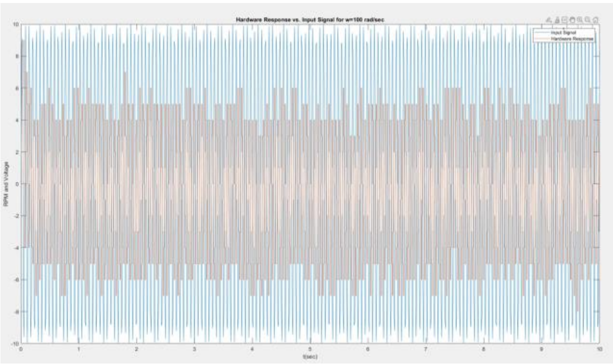

Figure 8- Hardware Response vs the Sinusoidal Input Where ω =100 rad/s

The same idea that was used in the pre-lab is used in this part and in the pre-lab the output of an LTI system when a sinusoidal signal is inputted is:

$$y(t) = |G(j\omega)| * \cos(\omega t + \angle G(j\omega))$$

This result shows that by comparing the input and output of the signal for a given frequency, the magnitude and phase can be obtained for a sinusoidal input. Then using the "fft" command in MATLAB, the maximum value of both input and output are divided to get the magnitude of the transfer function. Similarly using the "fft" command the angle values of the maximum values indexes' are subtracted to get the phase.

This process is applied for each of the frequencies given above to obtain the Bode plot:

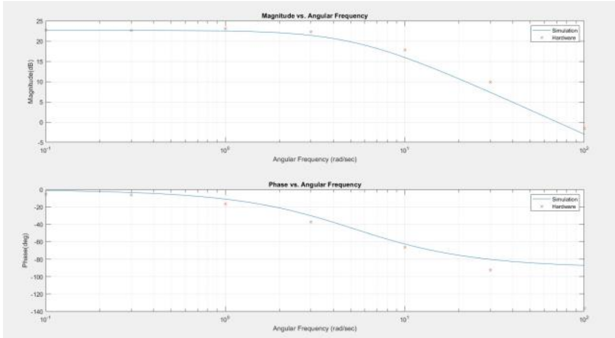

Figure 9- Theoretical(Blue) and Hardware(Orange Dots) Bode Plots

From the graph it can be seen that, both the phase and magnitude points start at similar values to the theoretical value but deviate with higher frequency values. For the magnitude case, this is due to the imperfectness of the DC motor. When the motor first starts to move from stance, there is a jerky motion and first sinusoidal waves spike. Those spikes become clearer as frequency increases. Later the wave stabilizes but still the waves aren't perfect because the sampling rate stays the same but frequency increases, increasing the change and lost data between samples. Additionally, the gain was multiplied by 2. For the phase case, time delay of hardware system comes into play, which will be worked on next part.

Part 3

In the last part, the delay due to processing time will be accounted for in the transfer function. This delay doesn't cause noticeable difference in smaller frequencies but it becomes noticeable as frequencies increase. So the transfer function of the system is updated as:

$$G_{\text{delayed}}(s) = G(s)\frac{1 - 0.005s}{1 + 0.005s}$$

Then three bode plots are plotted together: theoretical, hardware and delayed bode plots:

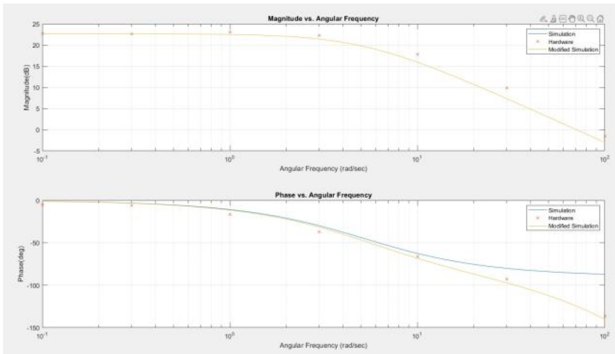

Figure 10- Theoretical(Blue) and Hardware(Orange Dots) and Delayed(Orange Line) Bode Plots

With the delayed plot, it is observed that higher frequency values fit better to this plot and there is no difference in the magnitude plot because the modification made delays the system which doesn't affect the magnitude.

3. Conclusion

In the end, bode plot of a transfer function was generated in three methods. Firstly, by utilizing the transfer function from the previous lab theoretical bode plot was plotted with the given code. In the second part, a sinusoidal signal is applied to the system. Then a sinusoidal output is obtained and by inspecting differences between input and output magnitude and phase are obtained. Obtained phase and magnitudes are for that input sinusoid's frequency and sinusoids with different frequencies are applied to obtain the bode plot point values. Then the theoretical and hardware(sinusoid method) bode plots are plotted together. It is observed that, there is a discrepancy in two plots for higher frequencies. Then Pade approximation is applied to the transfer function to account for processing delay. Then all three of these bode plots are plotted together: theoretical, hardware and delayed. Finally, it was observed that the hardware values obtained best fitted the delayed bode plot. This was the expected result as the delayed transfer function was made to best fit the real world case. The phase bode plot fitted well and there was a slight difference in the magnitude plot in the end. This slight difference might be due to the imperfections on the DC motor. All in all, the lab work was completed successfully and the wanted expected were observed.

Appendix

MATLAB:

```
A=70.74884499;
B=5.208631745;
w = logspace(-1,2,100);
for k = 1:100
s = 1i * w(k);
G(k) = A / (s+B);
end
subplot(2,1,1)
semilogx(w,20*log10(abs(G)));
title("Magnitude vs. Angular Frequency")
ylabel("Magnitude(dB)")
xlabel("Angular Frequency(rad/sec)")
grid on
subplot(2,1,2)
semilogx(w,angle(G)*180/pi)
title("Phase vs. Angular Frequency")
ylabel("Phase(deg)")
xlabel("Angular Frequency(rad/sec)")
grid on
%Step 6%
magnitudes=zeros(7,1);
phases=zeros(7,1);
%W=0.1%
figure
angular_frequency = 0.1;
duration = 70;
t = 0:0.01:duration;
input = 10*sin(angular_frequency * t);
plot(t,input)
hold on
plot(angf_01)
title('Hardware Response vs. Input Signal
for w=0.1 rad/sec')
legend('Input Signal','Hardware Response')
xlabel('t(sec)')
ylabel('RPM and Voltage')
hold off

[maxVel_01, maxVelIndex_01] =
max(abs(fft(angf_01.data)));
[maxIn_01, maxIn_Index_01] =
max(abs(fft(input)));
K_01=maxVel_01/maxIn_01;
K_01=20*log10(2*K_01);
fft_y01=fft(angf_01.data);
fft_In01=fft(input);
Phase_01=angle(fft_y01(maxVelIndex_01))-
angle(fft_In01(maxIn_Index_01));
magnitudes(1)=K_01;
phases(1)=Phase_01;
%W=0.3%
figure
angular_frequency = 0.3;
duration = 70;
t = 0:0.01:duration;
input = 10*sin(angular_frequency * t);
plot(t,input)
hold on
plot(angf_03)
title('Hardware Response vs. Input Signal
for w=0.3 rad/sec')
legend('Input Signal','Hardware Response')
xlabel('t(sec)')
ylabel('RPM and Voltage')
hold off
[maxVel_03, maxVelIndex_03] =
max(abs(fft(angf_03.data)));
[maxIn_03, maxIn_Index_03] =
max(abs(fft(input)));
K_03=maxVel_03/maxIn_03;
K_03=20*log10(2*K_03);
fft_y03=fft(angf_03.data);
fft_In03=fft(input);
Phase_03=angle(fft_y03(maxVelIndex_03))-
angle(fft_In03(maxIn_Index_03));
magnitudes(2)=K_03;
phases(2)=Phase_03;
%W=1%
figure
angular_frequency = 1;
duration = 25;
t = 0:0.01:duration;
input = 10*sin(angular_frequency * t);
plot(t,input)
hold on
plot(angf_1)
title('Hardware Response vs. Input Signal
for w=1 rad/sec')
legend('Input Signal','Hardware Response')
xlabel('t(sec)')
ylabel('RPM and Voltage')
hold off
```

```
[maxVel_1, maxVelIndex_1] =
max(abs(fft(angf_1.data)));
[maxIn_1, maxIn_Index_1] =
max(abs(fft(input)));
K_1=maxVel_1/maxIn_1;
K_1=20*log10(2*K_1);
fft_y1=fft(angf_1.data);
fft_In1=fft(input);
Phase_1=angle(fft_y1(maxVelIndex_1))-
angle(fft_In1(maxIn_Index_1));
magnitudes(3)=K_1;
phases(3)=Phase_1;
%W=3%
figure
angular_frequency = 3;
duration = 25;
t = 0:0.01:duration;
input = 10*sin(angular_frequency * t);
plot(t,input)
hold on
plot(angf_3)
title('Hardware Response vs. Input Signal
for w=3 rad/sec')
legend('Input Signal','Hardware Response')
xlabel('t(sec)')
ylabel('RPM and Voltage')
hold off
[maxVel_3, maxVelIndex_3] =
max(abs(fft(angf_3.data)));
[maxIn_3, maxIn_Index_3] =
max(abs(fft(input)));
K_3=maxVel_3/maxIn_3;
K_3=20*log10(2*K_3);
fft_y3=fft(angf_3.data);
fft_In3=fft(input);
Phase_3=angle(fft_y3(maxVelIndex_3))-
angle(fft_In3(maxIn_Index_3));
magnitudes(4)=K_3;
phases(4)=Phase_3;
%W=10%
figure
angular_frequency = 10;
duration = 10;
t = 0:0.01:duration;
input = 10*sin(angular_frequency * t);
plot(t,input)
hold on
plot(angf_10)
title('Hardware Response vs. Input Signal
for w=10 rad/sec')
legend('Input Signal','Hardware Response')
xlabel('t(sec)')
ylabel('RPM and Voltage')
hold off
[maxVel_10, maxVelIndex_10] =
max(abs(fft(angf_10.data)));
[maxIn_10, maxIn_Index_10] =
max(abs(fft(input)));
K_10=maxVel_10/maxIn_10;
K_10=20*log10(2*K_10);
fft_y10=fft(angf_10.data);
fft_In10=fft(input);
Phase_10=angle(fft_y10(maxVelIndex_10))-
angle(fft_In10(maxIn_Index_10));
magnitudes(5)=K_10;
phases(5)=Phase_10;
%W=30%
figure
angular_frequency = 30;
duration = 10;
t = 0:0.01:duration;
input = 10*sin(angular_frequency * t);
plot(t,input)
hold on
plot(angf_30)
title('Hardware Response vs. Input Signal
for w=30 rad/sec')
legend('Input Signal','Hardware Response')
xlabel('t(sec)')
ylabel('RPM and Voltage')
hold off
[maxVel_30, maxVelIndex_30] =
max(abs(fft(angf_30.data)));
[maxIn_30, maxIn_Index_30] =
max(abs(fft(input)));
K_30=maxVel_30/maxIn_30;
K_30=20*log10(2*K_30);
fft_y30=fft(angf_30.data);
fft_In30=fft(input);
Phase_30=angle(fft_y30(maxVelIndex_30))-
angle(fft_In30(maxIn_Index_30));
magnitudes(6)=K_30;
phases(6)=Phase_30-2*pi;
%W=100%
figure
angular_frequency = 100;
duration = 10;
t = 0:0.01:duration;
input = 10*sin(angular_frequency * t);
plot(t,input)
hold on
plot(angf_100)
title('Hardware Response vs. Input Signal
for w=100 rad/sec')
legend('Input Signal','Hardware Response')
xlabel('t(sec)')
ylabel('RPM and Voltage')
hold off
[maxVel_100, maxVelIndex_100] =
max(abs(fft(data1)));
[maxIn_100, maxIn_Index_100] =
max(abs(fft(input)));
K_100=maxVel_100/maxIn_100;
K_100=20*log10(2*K_100);
fft_y100=fft(data1);
fft_In100=fft(input);
Phase_100=angle(fft_y100(maxVelIndex_100))
-angle(fft_In100(maxIn_Index_100));
```

```matlab
magnitudes(7)=K_100;
phases(7)=Phase_100;
figure
Angles=[0.1 0.3 1 3 10 30 100];
subplot(2,1,1)
semilogx(w,20*log10(abs(G)));
hold on
semilogx(Angles,magnitudes,'x')
title('Magnitude vs. Angular Frequency')
ylabel('Magnitude(dB)')
xlabel('Angular Frequency (rad/sec)')
legend('Simulation','Hardware')
grid on
hold off
subplot(2,1,2)
semilogx(w,angle(G)*180/pi)
hold on
semilogx(Angles,phases*180/pi,'x')
title('Phase vs. Angular Frequency')
ylabel('Phase(deg)')
xlabel('Angular Frequency (rad/sec)')
legend('Simulation','Hardware')
grid on
hold off
%Question 3%
A=70.74884499;
B=5.208631745;
w = logspace(-1,2,100);
for k = 1:100
s = 1i * w(k);
T(k) = (A*(1-
0.005*s))/((1+0.005*s)*(s+B));
end
subplot(2,1,1)
semilogx(w,20*log10(abs(G)));
hold on
semilogx(Angles,magnitudes,'x')
semilogx(w,20*log10(abs(T)));
title('Magnitude vs. Angular Frequency')
ylabel('Magnitude(dB)')
xlabel('Angular Frequency (rad/sec)')
legend('Simulation','Hardware','Modified
Simulation')
grid on
hold off

subplot(2,1,2)
semilogx(w,angle(G)*180/pi)
hold on
semilogx(Angles,phases*180/pi,'x')
semilogx(w,angle(T)*180/pi)
title('Phase vs. Angular Frequency')
ylabel('Phase(deg)')
xlabel('Angular Frequency (rad/sec)')
legend('Simulation','Hardware','Modified
Simulation')
grid on
hold off
```

```matlab
figure
Angles=[0.1 0.3 1 3 10 30 100];
subplot(2,1,1)
semilogx(w,20*log10(abs(G)));
hold on
semilogx(Angles,magnitudes,'x')
title('Magnitude vs. Angular Frequency')
ylabel('Magnitude(dB)')
xlabel('Angular Frequency (rad/sec)')
legend('Simulation','Hardware')
grid on
hold off
subplot(2,1,2)
semilogx(w,angle(G)*180/pi)
hold on
semilogx(Angles,phases*180/pi,'x')
title('Phase vs. Angular Frequency')
ylabel('Phase(deg)')
xlabel('Angular Frequency (rad/sec)')
legend('Simulation','Hardware')
grid on
hold off
```