EEE342 Feedback Control Systems - Laboratory Work 3

Ezgi Demir

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

1. Introduction

In this lab, we determined the gain, phase, and delay margins of our DC motor system using the mathematical model developed earlier. We then validated these margins on the actual hardware by varying the controller gain \(K\) to identify the thresholds between stable and unstable behavior. Finally, we adjusted the value h and repeated the tests to see how it affects stability. By comparing the experimentally observed margins with those computed by MATLAB's `allmargin` command, we confirmed the accuracy of our model and margin analysis.

2. Laboratory Content

Part 1

In the lab manual we are given a transfer function of our controller.

$$G_c(s) = \left(\frac{1}{s+\tau_{LPF}}\right)\left(\frac{K_c(s+80)}{s}\right) \quad (Eq.1)$$

In the formula,

$$K_c = \frac{2}{K_g} \approx 0.1437\,, \quad \tau_{LPF} = \frac{3}{\tau_p} \approx 33.37 \quad (Eq.2)$$

Where K_g and τ_p values are the values where we found in lab 1 [1]. We are asked to use first order Pade Approximation we used in lab 2 [2]. Therefore our controller transfer function turns out,

$$G_c(s) = \frac{0.1436s+11.4917}{s^2+32.967s} \quad (Eq.3)$$

$$G_{paded}(s) = G_p(s) * \frac{1-0.005s}{1+0.005s} \quad (Eq.4)$$

$$= \left(\frac{13.923}{0.0899s+1}\right) * \frac{1-0.005s}{1+0.005s} \quad (Eq.5)$$

$$= \frac{13.923-0.069615s}{0.000495s^2+0.0949s+1}$$

Overall transfer function will be,

$$G(s) = G_c(s)G_{paded}(s) \quad (Eq.6)$$

After we plug in our parameters the result of our G(s) values will be;

```
All margins structure:
    GainMargin: 9.7746
   GMFrequency: 20.0667
   PhaseMargin: 60.8235
   PMFrequency: 4.4629
   DelayMargin: 0.2379
   DMFrequency: 4.4629
        Stable: 1
```

Fig. 1: allmargin() Results of the Transfer Function

Now, we will compare our Matlab results from allmargin() command from the plot we observe

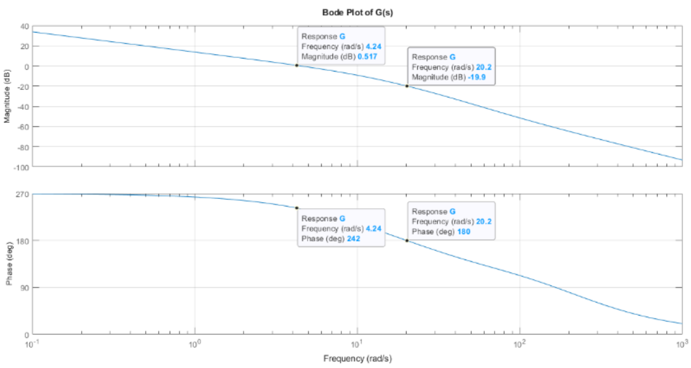


Fig. 2: The Bode Plot of our Overall Transfer Function

We saw a 19.9 dB from 0 dB point in magnitude plot at the same frequency where we see 180° degree art the phase plot. This is what we call gain margin. From Fig.1,

$$20\log_{10} 9.7746 = 19.8 \text{ dB} \quad (Eq.7)$$

Also, at the frequency we saw 0 dB in the magnitude plot, we get a 242 degree phase from Fig.2 where the phase difference is 62 degrees from 180 degree,

$$242 - 180 = 62° \quad (Eq.8)$$

Which is our phase margin. allmargin() result shows 60.8 degrees. This difference may occur because in the plot, we couldn't exactly catch 0dB point, but rather 0.5. Therefore, we slightly get a bigger phase difference. After we found the phase margin we can also compute delay margin from the formula,

$$DM = \left(\frac{\pi}{180}\right)\left(\frac{phase\ margin}{crossover\ frequency}\right) \quad (Eq.9)$$

The frequency at which we find the phase margin is our crossover frequency which is 4.24 rad/s.

$$DM_{calculated} = \left(\frac{\pi}{180}\right)\left(\frac{62}{4.24}\right) \approx 0.25$$

From Fig.1 we can see the delay margin matlab derived is about 0.24 which is very close to what we calculated.

This all proves that we can derive the margins from the plots and analytically calculate it.

	Plot Results	allmargin() results
GM	19.8dB	19.082dB
PM	62(degree)	60.8235(degree)
DM	0.25(s)	0.2379(s)

Table 1: Comparison of Margins

Part 2

We set the input as *r(t)=40u(t),* we start with the GM we derived, but we ca see it is not stable so we continue to check K(gain), and try to find the exact value of K_f that makes system stable.

Again we start with the value of K we find in part 1, where we can see unstability, therefore, we start to decrease the value in order to find where our system is marginally stable.

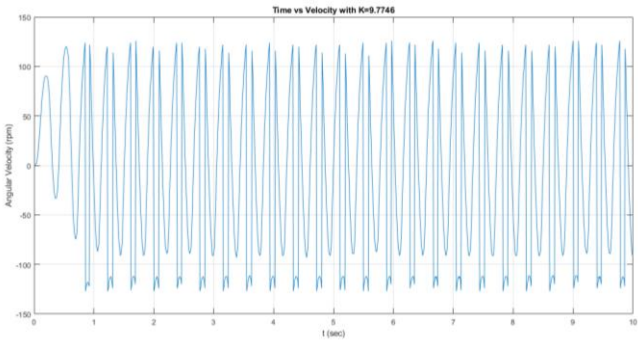


Fig. 3: DC Motor output when K=9.7746

At the point K = 4.5 We observe a wave whose amplitude increases over time. After this point we see a stable wave.

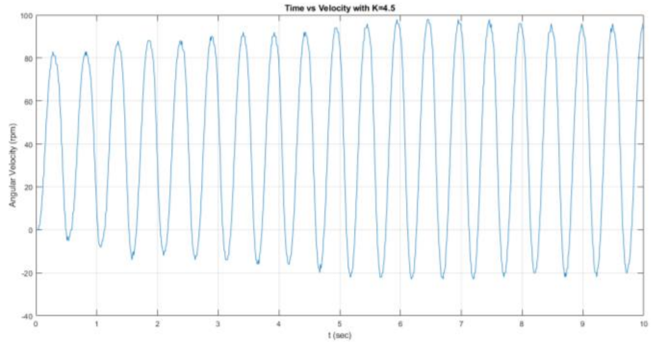


Fig. 4: DC Motor output when K=4.5

Then we try to find the highest K where again we see a wave whose amplitüde slighlty decreases over time where this is a point of marginally stable [3].

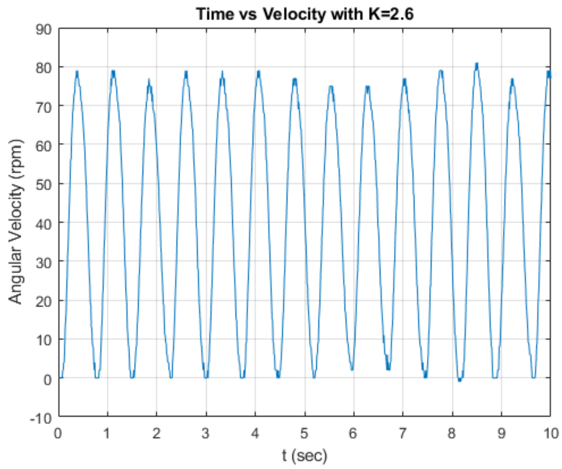


Fig. 5: DC Motor output when K=2.6

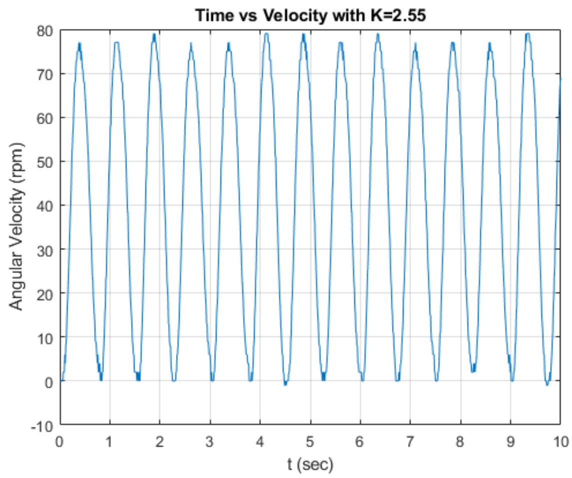


Fig. 6: DC Motor output when K=2.55

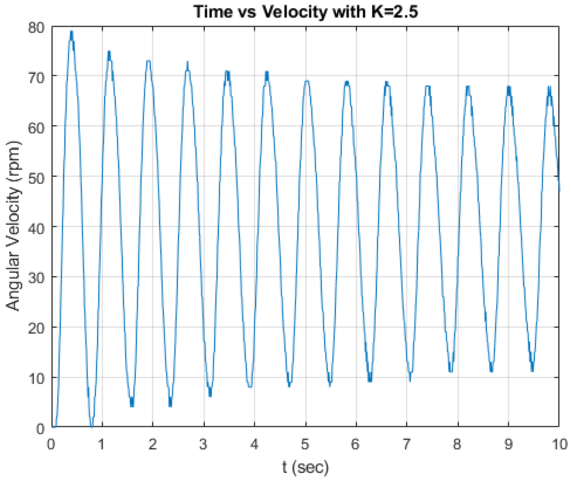


Fig. 7: DC Motor output when K=2.5

K=2.55 is a point where stability starts. When K<2.55

$$GM = 20log(k) = 20log(2.55) = 8.13\ dB \ (Eq.10) \quad [4]$$

The discrepancy in the experimentally measured gain margin arises because the actual system cannot tolerate as large a gain increase as theoretical models predict—this is due to unmodeled dynamics, sensor noise, and implementation delays. As a result, the observed gain margin is lower than expected [3].

Part 3

We will continue finding the point where stability begins. This time we find the point by looking at delay value h.

The ouptut result is below, where delay= 0.2379, system is unstable.

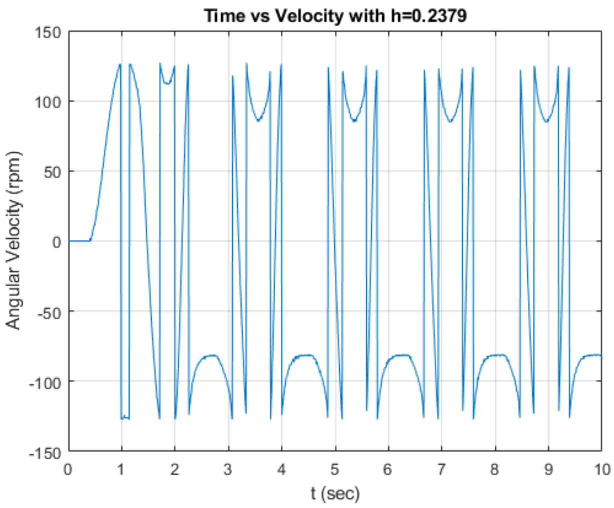


Fig. 8: DC Motor output when h=9.2379

When we decrease h and try to find the exact point we reach stability we called marginally stable. This occurs when h=0.04

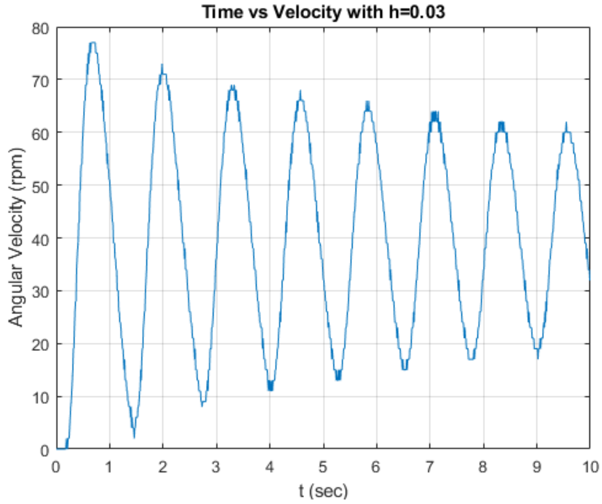


Fig. 9: DC Motor output when h=0.03

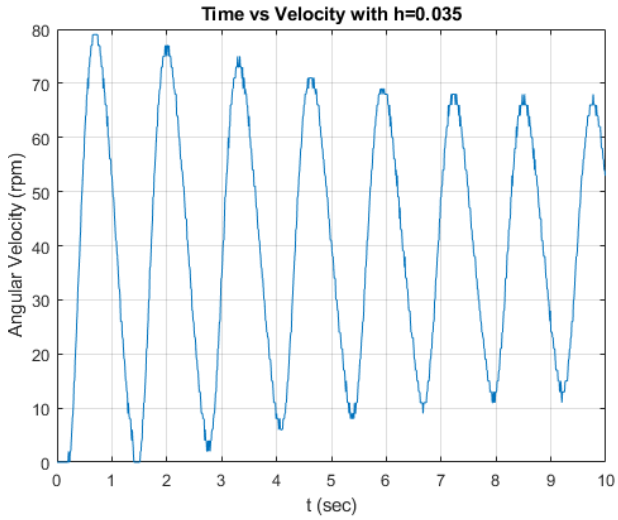


Fig. 10: DC Motor output when h=0.035

After h=0.04 we again reach a point where our system is unstable.

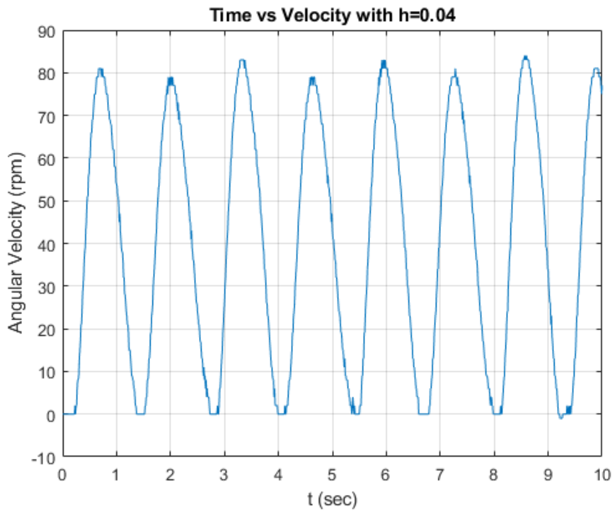


Fig. 11: DC Motor output when h=0.04

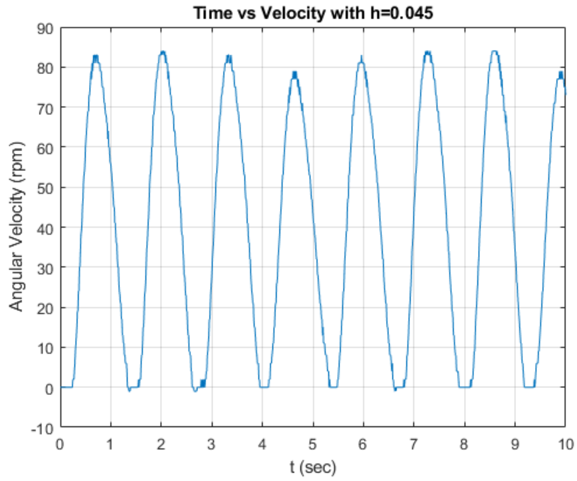


Fig. 12: DC Motor output when h=0.045

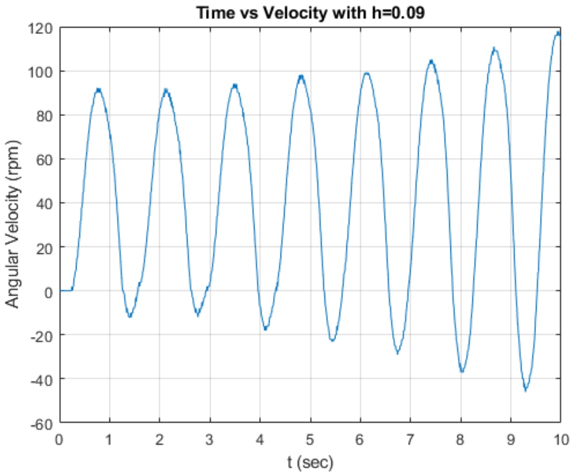
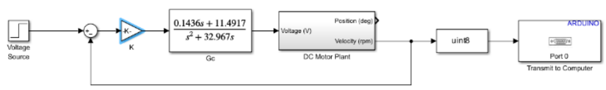


Fig. 13: DC Motor output when h=0.09

Fig. 13 shows the h value which our wave's amplitude start to increase till some point where we observe unstability easily.

	Experimental Results	allmargin() results
GM	8.13dB	19.082dB
PM	-	60.8235(degree)
DM	0.04(s)	0.2379(s)

Table 1: Comparison of Margins



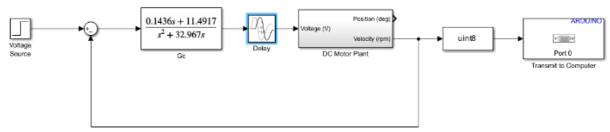


Fig. 14: Simulink Blocks where we manipulate h and K values

Matlab Code

```
clear; clc; close all;
tau_p = 0.091;
Kp    = 13.9231;

Kg       = Kp;
Kc       = 2/Kg;
tau_LPF   = 3/tau_p;

s = tf('s');

Gc = Kc * (s + 80) / ( s * (s + tau_LPF) );
Gp = Kp/(tau_p*s+1)*(1-0.005*s)/(1+0.005*s);

G = Gc * Gp;

figure;
bode(G,{0.1,1000});
grid on;
title('Bode Plot of G(s)');

disp('All margins structure:');
disp(allmargin(G)); %gm 20log

sim("lab3_read.slx");
vel1=squeeze(velocity.data(1,1,:));

% save("vel1.mat","vel1");
% load("vel1.mat","vel1")

vel1 = cast(vel1,"uint16");

vel1 = double(vel1);

for i=1:length(vel1)
    if(vel1(i)>=128)
        vel1(i)=vel1(i)- 255;%255
    end
end

figure;

plot(linspace(0, 10, length(vel1)), vel1);
title('Time vs Velocity with K=4.5');
xlim([0,10]);
xlabel('t (sec)');
ylabel('Angular Velocity (rpm)');
grid on;


%
% figure;
%
% plot(linspace(0, 10, length(vel1)), vel1);
% title('Time vs Velocity with h=0.09');
% xlim([0,10]);
% xlabel('t (sec)');
```

```
% ylabel('Angular Velocity (rpm)');
% grid on;
```

Conclusion

In this lab, I learned how to compute and interpret gain, phase, and delay margins using Pade approximation and MATLAB, then verify these stability boundaries experimentally on a DC-motor setup. I observed firsthand how unmodeled dynamics, sensor noise, and delays reduce real-world margins, highlighting the need for conservative controller design. Overall, this hands-on experience deepened my understanding of feedback stability and the practical steps required to ensure robust control performance .

REFERENCES

1. EE342 Spring 2025, Lab work 1- Ezgi Demir
2. EE342 Spring 2025, Lab Work 2- Ezgi Demir
3. R. H. Bishop and R. C. Dorf, Modern Control Systems: International Edition, 10th ed. Upper Saddle River, NJ: Pearson, 2005.
4. Desmos sceintific calculator, https://www.desmos.com/scientific?lang=tr