Lab-1 Preliminary Work

Ayhan Okuyan

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

1. Introduction

This preliminary work consisted of modelling and finding the parameters of a DC Motor block, in a given test environment. Overall, we were given a Simulink model and a MATLAB data to guide us through the work.
Hence, we have first modelled the DC Motor block given by placing the general gain formula to find the output velocity(rpm) by $G_{DC} = {}^{K}/_{\tau s + 1}$. Then, we model the position(deg) output by passing the output through $G_{Int} = {}^{6}/_{s}$. The 6 in the numerator represents the unit transition from rpm to degrees. The denominator represents the integrator from velocity to position since velocity is the derivative of position.
After modeling the DC Plant Section, we have moved onto finding the transfer function of the position output. After completing that, we have separated the transfer function into the multiplication pf a first order and a second order transfer function. Then, solved the equation believing that the second order is dominant. To do so, we have estimated the second order transfer function coefficient by using the certain approximations that we have gone through in class, which will be discussed in the following sections. Then, we have used MATLAB to solve the equations that we have obtained to find the DC Motor Plant coefficients and concluded the assignment by discussing further on the discrepancies between the given position data and the one that we have found.

2. Laboratory Content

After the modeling is done, the next part was to configure the transfer function of the feedback system. The total configuration of the block diagram is given below.

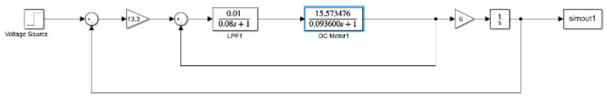

Fig. 1: Block Diagram of the System

Then, by using the block diagram simplification techniques, we have simplified the system, and found the transfer function as:

$$T(s) = \frac{0.06KG_c}{s^3(0.08\tau) + s^2(0.08 + \tau) + s(0.01K) + 0.06KG_c},$$

$$G_c = 13.3$$

In the manual, we were told to separate this transfer function into a second order and a first order transfer function. Instead, I have chosen to multiply the given set of transfer functions and the work with the equivalent transfer functions in both sides. Hence, when multiplied, the equation appears as:

$$Tr(s) = \frac{\omega_0^2}{s^2 + 2\zeta\omega_0 s + \omega_0^2}\frac{1}{\tau_x s + 1}$$

$$= \frac{\omega_0^2}{s^3(\tau_x)+s^2(2\zeta\omega_0\tau_x+1)+s(\tau_x\omega_0^2+2\zeta\omega_0)+\omega_0^2}$$

Hence, we know that there is direct relation between those two equations. However, there is also a hint given in the manual that said numerator of Tr(s) not being equal to $0.06KG_c$. Hence, I have instead equating the equations themselves, I have equated the ratios of the equations separately.
Since I have found the transfer function, the next phase was to approximate the second order characteristic parameters. In the lectures, we have learned that we can estimate the ζ value by using the maximum overshoot estimation. However, in the lectures, the steady state value of the output was one, hence I have generalized the formula like below;

$$\frac{M(0) - y(\infty)}{y(\infty)} = e^{\frac{-\pi\zeta}{\sqrt{1-\zeta^2}}}$$

Here, $M(0)$ stands for the maximum peak value, $M(\infty)$ stands for the steady state solution. These values are obtained from the given MATLAB data, easily by using data cursors, or the max() function. Then, the equation becomes;

$$\frac{151 - 90}{90} = 0.677 = e^{\frac{-\pi\zeta}{\sqrt{1-\zeta^2}}}$$

Hence, the ζ value becomes 0.122864. Then, knowing this, we can also estimate the value of ω_0 by using the settling time approximation. This approximation creates a relation between the settling time, ζ and ω_0 in its simples form which is given below.

$$T_{Set} = \frac{4}{\zeta\omega_0}$$

By plugging the value found for ζ, we see that the ω_0 becomes 14.469461.
This portion of the lab assignment is done by using a MATLAB code, that solves these equations and uses these for further analysis. The code is also given below.

```
syms tau tau_x K zeta w
t_set = 2.25;
m_p = 151;
ss_sol = 90;

% Crude Approximation
%eqn_zeta = (m_p-ss_sol)/ss_sol == 1-(zeta/0.6);
eqn_zeta = (m_p-ss_sol)/ss_sol == exp((-1*pi*zeta)/sqrt(1-zeta^2));
res_zeta = solve(eqn_zeta, zeta);
fprintf("%f\n",res_zeta(1));
zeta1 = res_zeta(1);

eqn_w = t_set == 4/(w*zeta1);
res_w = solve(eqn_w, w);
fprintf("%f\n",res_w(1));
w1 = res_w(1);
```

As we have found the coefficients and the transfer function, we can deliver the ratios of the three different equations that we have found by dividing the respective parts of both equations.

$$\frac{0.06KG_c}{0.01K} = \frac{\omega_0^2}{\tau_x\omega_0^2 + 2\zeta\omega_0}$$

$$\frac{0.06KG_c}{0.08 + \tau} = \frac{\omega_0^2}{2\zeta\omega_0\tau_x + 1}$$

$$\frac{0.06KG_c}{0.08\tau} = \frac{\omega_0^2}{\tau_x}$$

Then, we put in the numbers that we have found and create a MATLAB code to solve the equation system with three unknowns and three equations. The code that I have written is given below.

```
eqn1 = (0.798*K)/(1+0.01*K) ==
(w1^2)/(2*zeta1*w1 + w1^2*tau_x);
eqn2 = (0.798*K)/(0.08+tau) ==
(w1^2)/(1+2*zeta1*w1*tau_x);
eqn3 = (0.798*K)/(0.08*tau) == (w1^2)/(tau_x);
[res_k, res_tau, res_tau_x] =
solve(eqn1,eqn2,eqn3, K, tau, tau_x);
fprintf("%f\n",res_k)
fprintf("%f\n",res_tau)
fprintf("%f\n",res_tau_x)
```

The solution for this system is given as,

$$K = 30.964762$$
$$\tau = 0.053137$$
$$\tau_x = 0.036018$$

This concludes the analytical section of the prelab assignment. The next process is to design a Simulink schematic and put the values that we have obtained. Then to plot the observed values together with the data that we are given.
Hence, first thing we do is to create the schematic given in the manual and by following the instructions and put the values that are obtained. The final block diagram is given below.

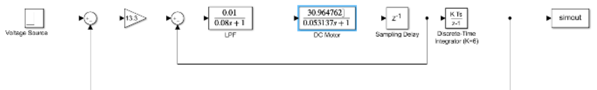

Fig. 2: Block Diagram with the Calculated Coefficients

One thing to observe is that, instead of a pure Laplacian operation, we have discretized the output by sampling it with 10ms, which is the reasonable modeling since this is a physical system model and not a hypothetical one.
In this diagram, we have used the step, transfer function, gain, sum, delay and discrete -integrator blocks of Simulink and then we added a sink to move the output to MATLAB workspace.
After it is done, I have plotted the given data and the calculated data on top of each other.

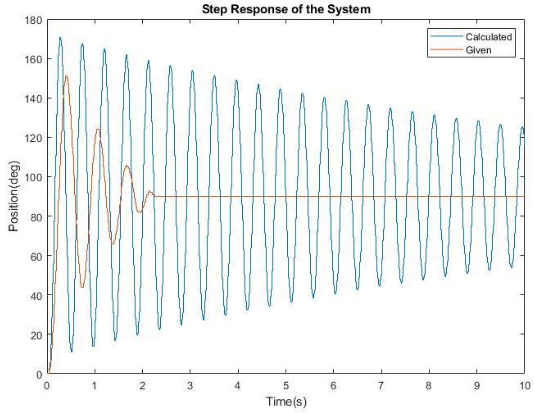

Fig. 2: Step Response of the System

While the red line is the given data by the instructor, the blue line is the data that we have calculated. While the two should have looked like each other, we see that they do not at all. The reasons for that will be conveyed in the Conclusion section. Furthermore, there is one more option in order to understand the validity of the data, which is to change the approximations that are made. The easiest one to do is to change the approximation of ζ. There is a cruder but easy to implement approximation of maximum overshoot that is related to ζ which is given below.

$$\frac{M(0) - y(\infty)}{y(\infty)} = 1 - \frac{\zeta}{0.6}$$

If we use this approximation instead of the other one, there is a possibility that the data is more interpretable.
Changing this and running the same calculations over, we get the results;

$$\zeta = 0.193333$$
$$\omega_0 = 9.195402$$
$$K = 15.573476$$
$$\tau = 0.093600$$
$$\tau_x = 0.050947$$

Then, by plugging the covered results, we can observe the output.

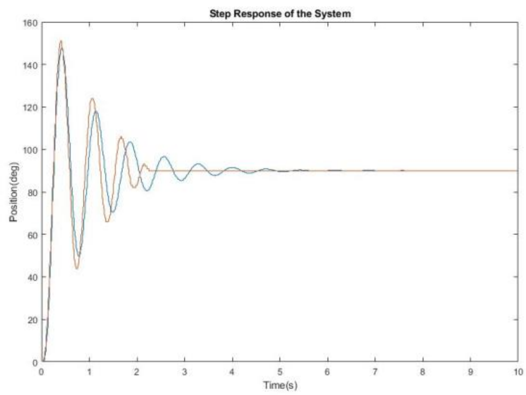

Fig. 4: Step Response due to the Crude Zeta Approximation

Here, we can see that the graphs look like each other quite nicely, with longer settling time and more or less the same maximum overshoot. Also, here, we can clearly observe the effect of delay to the sinusoidal as the graph tends to move slightly further to the positive time axis. Hence, we can clearly see that the work that we have done is credible.

3. Conclusion

As we observed in the previous section, the solution that we have found was inconsistent with the output that was given to us. There are many reasons for that to occur.

Firstly, we have made all our computations assuming the dominance of the second-order transfer function. Hence, we assumed that the first-order system's transfer function wouldn't matter to the time constant. However, when we ran the equations, we have seen that τ_x and τ were in fact not as different. From this, we see that we cannot ignore the first order transfer function.

Furthermore, there are some certain nonlinearities that effected the outcome, like friction for example and we didn't model these nonlinearities.

Overall, this was an illuminating lab assignment. I have learned a lot about simplifying block diagrams, finding transfer functions. Also, I have further worked on MATLAB and Simulink. I have seen that Simulink can be a very practical tool in terms of solving and simulating feedback systems.

REFERENCES

1. Modern Control Systems, Dorf R.C., Bishop R.H., 4.ed., pp. 90-110 & 271-275. Pearson Intl
2. Lecture Notes on Control Systems/D. Ghose/2012 (https://nptel.ac.in/courses/101108056/module7/lecture20.pdf)