George Mason University

Position Control for a Servo Motor
Designing Controller

Report from laboratory experiment 8 conducted on 5 April 2016
As part of ECE 429 Control Systems Lab
Course Instructor: Dr. Daniel M. Lofaro

Michael Kepler
G00804828

12 April 2016

The Volgenau School of Engineering

Objective

The objective of this lab is to design a Proportional-Derivative (with Derivative on Output Only), PD-DOO, controller for the system identified in the previous lab, unit C.1 (Lab 7).

Overview

For this lab a suitable response was decided by the student, namely specifications for peak overshoot and time to peak. Using these specifications, the gains for the controller were found. Before implementing the system in hardware, it was first simulated in MATLAB.

Implementation

Task 1

The system to be controlled is modeled by the equation:

$$T(t) = J\ddot{\theta}(t) + c\dot{\theta}(t) \tag{1}$$

Where

J – Moment of Inertia of the Disc
Θ – Angular Position of the Disc (Output)
T- Torque Applied by the Motor (Input)

The transfer function is defined as the output divided by the input:

$$G_p(s) = \frac{\theta(s)}{T(s)} \tag{2}$$

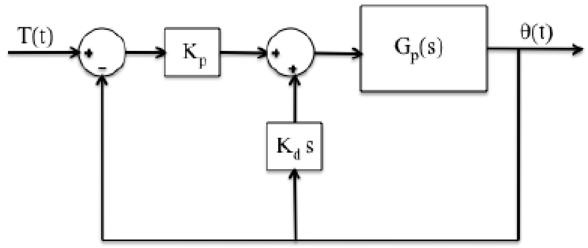
Applying the Laplace transform to eqn. (1):

$$T(s) = Js^2\theta(s) + cs\theta(s) \tag{3}$$

Solving for eqn. (2), the open-loop transfer function of the plant is obtained:

$$G_p(s) = \frac{\theta(s)}{T(s)} = \frac{1}{s(Js + c)} \tag{4}$$

The controller used on the system is a Proportional-Derivative, Derivative on the Output Only (PD-DOO) controller, which is modeled by the following block diagram:



For an open-loop plant function, $G_{ol}(s)$, with a feedback function $H(s)$, the closed loop equation can be found using the following equation:

$$G_{cl} = \frac{G_{ol}(s)}{1 + G_{ol}(s)H(s)} \tag{5}$$

The first step in solving for the closed-loop equation of the total system is to find the closed-loop equation of $G_p(s)$ with the feedback function $K_d s$ using eqn. (5):

$$G_{pCL1} = \frac{\left(\frac{1}{s(Js+c)}\right)}{1 + \left(\frac{1}{s(Js+c)}\right)K_d s}$$

$$G_{pCL1} = \frac{1}{Js^2 + (c + K_d)s} \tag{6}$$

To solve for the total closed-loop equation of the system, $G_{pCL1} * K_p$ will act as the open-loop function, and since unity feedback is applied, $H(s)$ is simply 1. Thus using eqn. (5) the total closed-loop equation is obtained:

$$G_{pCLtotal} = \frac{K_p\left(\frac{1}{Js^2 + (c + K_d)s}\right)}{1 + K_p\left(\frac{1}{Js^2 + (c + K_d)s}\right)}$$

$$G_{pCLtotal} = \frac{K_p}{Js^2 + (C + K_d)s + K_p}$$

$$\boldsymbol{G_{pCLtotal} = \frac{\left(\frac{K_p}{J}\right)}{s^2 + \left(\frac{C + K_d}{J}\right)s + \left(\frac{K_p}{J}\right)}} \tag{7}$$

Task 2

For a suitable response, peak percent overshoot (PO) was selected as 20% and desired time-to-peak was selected as 0.5 s.

Task 3

Using the value for percent overshoot, the damping factor (ζ) can be calculated:

$$\zeta = \frac{\left|\ln \frac{PO}{100}\right|}{\sqrt{\pi^2 + \ln^2 \frac{PO}{100}}} \tag{8}$$

$$\zeta = 0.4559$$

The natural frequency can be computed using the damping factor and the time to peak result:

$$\omega_n = \frac{\pi}{t_p\sqrt{1-\zeta^2}} \tag{9}$$

$$\omega_n = 7.0597\,\frac{rad}{s}$$

Task 4

The equation for a general second order system is:

$$2nd\ Order\ System = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{10}$$

Comparing eqn. (14) with eqn. (9) we see that:

$$\frac{K_p}{J} = \omega_n^2 \tag{11}$$

And

$$\frac{C + K_d}{J} = 2\zeta\omega_n \tag{12}$$

Thus using eqns. (7) and (10), K_p and K_d, respectively, can be solved for:

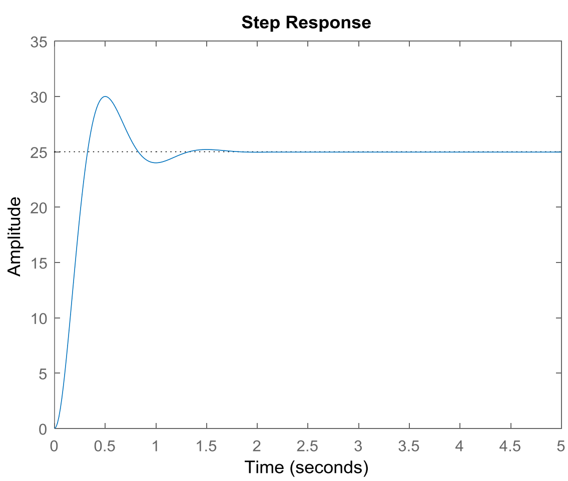
$$\boldsymbol{K_p = 0.0397}$$
$$\boldsymbol{K_d = 0.0032}$$

Task 5

The system was then simulated in MATLAB with the following code:

```
clear all

%Specifications
PO = 20;
tp = 0.5;

zeta = sqrt(log(PO/100)^2/(pi^2+log(PO/100)^2))
wn = pi/(tp*sqrt(1-zeta^2))

%Define Plant
s = tf('s');
J = 0.000796;
C = 0.00192;
gp_ol = 1/(s*(J*s+C));

%Calculate controller gains
kp = wn^2*J
kd = J*2*zeta*wn - C

%Compute the total system
Hd = kd*s;
g_cl1 = feedback(gp_ol,Hd);
g_cl = feedback(kp*g_cl1, 1);

%Step response
stepinfo(25*g_cl)
step(25*g_cl)
xlim([0,5])
ylim([0,35])
```

Which yielded the following simulation results:

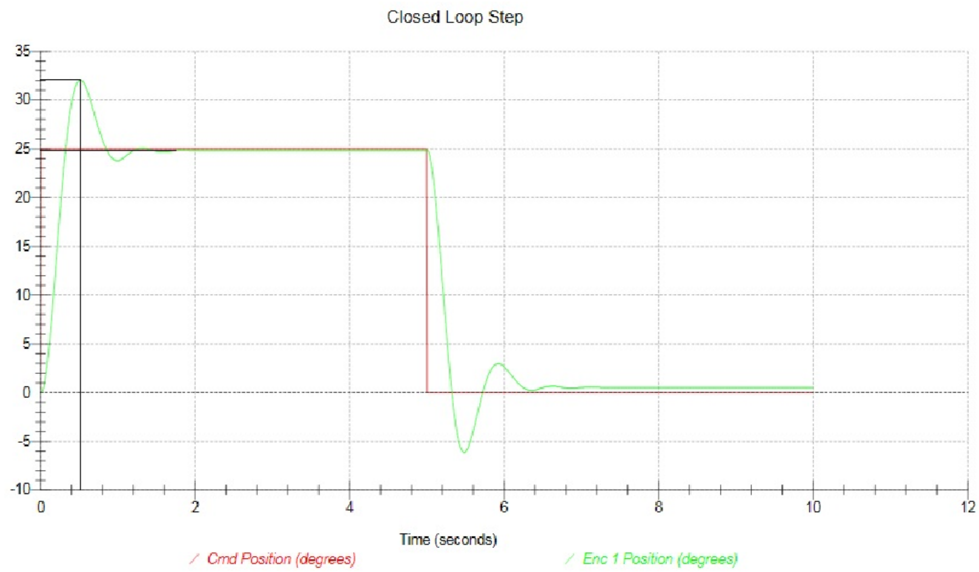



```
ans =

        RiseTime: 0.2207
    SettlingTime: 1.1796
     SettlingMin: 23.3355
     SettlingMax: 29.9999
       Overshoot: 19.9997
      Undershoot: 0
            Peak: 29.9999
        PeakTime: 0.5000
```

The system satisfies the percent overshoot requirement (<20%) and peak time (<= 0.5s).

Task 6

Given that the system meets specifications in simulation, it can now be implemented on the physical torsional plant hardware:



Using pixel resolution to analyze the graph, the following data was extracted:

Final Value	Peak Value	PO	Peak Time (s)
24.7778	32.1111	0.2960	0.5185

Task 7

The system did not satisfy the percent overshoot requirement (<20%) and peak time requirement (<= 0.5s). The controller had to be redesigned. The controller was redesigned to satisfy percent overshoot requirement (<10%) and peak time requirement (<= 0.45s). Repeating the calculations in tasks 3 and 4, the following values were obtained:

Calculations for PO = 10% and t_p = 0.45s			
ω_n (rad/s)	ζ	k_p	k_d
8.6557	0.5912	0.0596	0.0062

These are the results from the system simulation after changing the gains:

Trial	Final Value	Peak Value	PO	Peak Time (s)
1	24.7272	28.9090	0.1691	0.4888

These values satisfied the specifications for percent overshoot of 20% and a peak time less than 0.5s. The system was implemented twice more.

The results from all three trials are tabularized below:

Trial	Final Value	Peak Value	PO	Peak Time (s)
1	24.7272	28.9090	0.1691	0.4888
2	24.6363	28.9090	0.1734	0.474
3	24.6363	28.909	0.1734	0.474
Average	24.6666	28.9090	0.1720	0.4789

The averages demonstrates the system achieves the desired percent overshoot and desired peak time (< 0.5s).

Task 8

Task 8 instructs the student to relocate the weights that are currently on the ends of the plate inwards by 2-3 cm and implement the system with the same step input of 25 degrees. For our implementations, the weights were brought in and placed 5 cm. from the center.


Closed Loop Step
30
25
20
15
10
5
0
-5
0
2
4
6
8
10
12
Time (seconds)
Cmd Position (degrees)
Enc 1 Position (degrees)


Closed Loop Step
30
25
20
15
10
5
0
-5
0
2
4
6
8
10
12
Time (seconds)
Cmd Position (degrees)
Enc 1 Position (degrees)

The results from all three trials are tabularized below:

Trial	Final Value	Peak Value	PO	Peak Time (s)
1	24.4545	27.5454	0.1264	0.4148
2	24.4545	27.7272	0.1338	0.4296
3	24.4545	27.7272	0.1338	0.4296
Average	24.4545	27.6666	0.1314	0.4247

The system response achieves a smaller time to peak than the previous implementation where the weights were placed at the ends of the plate. This is because the moment of inertia is decreased by moving the weights in closer. Looking at eqn. (11) we see that as the moment of inertia decreases, ω_n^2 increases. Looking at eqn. (9) we see that as ω_n^2 increases, the time to peak decreases, as was evident from the results.

Similarly the percent overshoot decreased as well. Looking at eqn. (12) we see that as the moment of inertia decreases, ζ increases. Looking at eqn. (8) we see that as ζ increases, the percent overshoot decreases, as was evident from the results.

The weights that were brought in were now moved back and placed on the ends of the plate. The other two weights corresponding to our G number were placed 3 cm. from the center. The system was then implemented three times:


Closed Loop Step
30
25
20
15
10
5
0
-5
0
2
4
6
8
10
12
Time (seconds)
Cmd Position (degrees)
Enc 1 Position (degrees)


Closed Loop Step
30
25
20
15
10
5
0
-5
0
2
4
6
8
10
12
Time (seconds)
Cmd Position (degrees)
Enc 1 Position (degrees)

The results from all three trials are tabularized below:

Trial	Final Value	Peak Value	PO	Peak Time (s)
1	24.3636	27.0909	0.1119	0.4148
2	24.4545	27.0909	0.1078	0.4148
3	24.4545	27.0909	0.1078	0.4148
Average	24.4242	27.0909	0.1092	0.4148

Once again the percent overshoot and time to peak decreased. This is because the moment of inertia decreased and affected the percent overshoot and peak time due to the same reasons stated for the previous implementation.

Conclusion

By comparing the denominator of the closed loop transfer function with the general form of a second order system we can compare coefficients to obtain the proper gain values. This was demonstrated by the system response of the torsional plant.