Control Systems

G V V Sharma*

Contents

***Abstract*—The objective of this manual is to introduce control system design at an elementary level.**

Download python codes using

```
svn co https://github.com/gadepall/school/trunk/
    control/ketan/codes
```

1 Designing a control system

1.1 Example

1.1. A position control system is shown in the figure below. The proportional controller used presently (k_a) is not providing satisfactory performance, Design appropriate controller to achieve maximum overshoot less than 15%. Further, the load should be positioned with 1% accuracy. For the motor to be used, load ad torque-speed curve is shown in figure below. System parameters are J_m = 2 kg-m^2, J_l = 10 kg-m^2, f_m = 2 N-m-s/rad , k = 12 N-m/rad, sensitivity of error detector is $1/\pi$

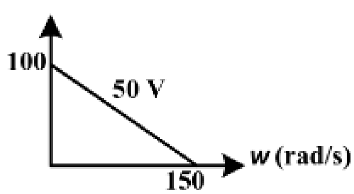

Fig. 1.1

Solution: The following paragraph tells us about the components in the given position control system.
We have,

*The author is with the Department of Electrical Engineering, Indian Institute of Technology, Hyderabad 502285 India e-mail: gadepall@iith.ac.in.

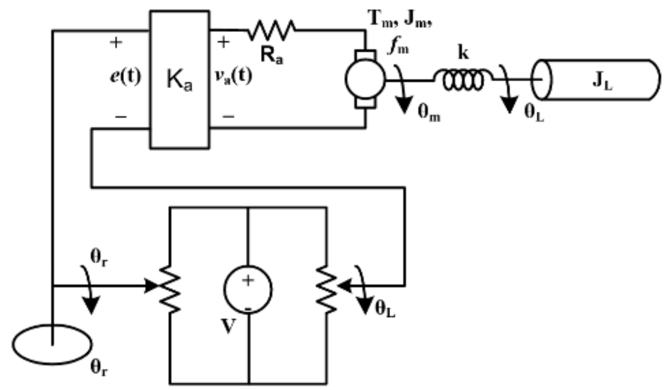

Fig. 1.1

Input, it gives constant input = θ_r
Plant, Our original system
Potentiometer, It convert position input and output to voltage.
Controller, to get desired output.

Below is the figure showing them,
Below is the block diagram, for the equivalent

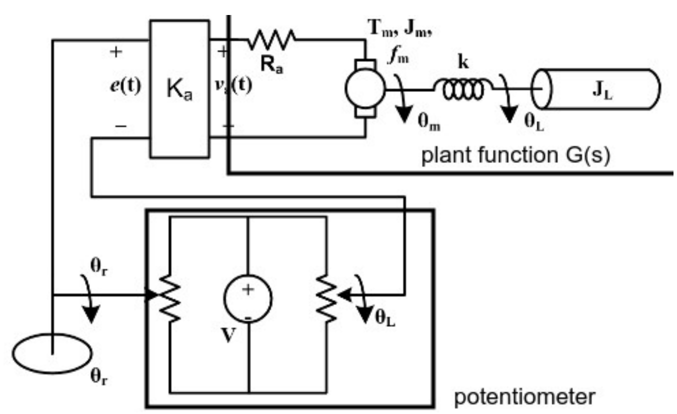

Fig. 1.1

position control system.

Clearly from the question its negative feedback system and given sensitivity of system, it basically gets multiplied after error (e(t)), thus giving us a block of gain $\frac{1}{\pi}$

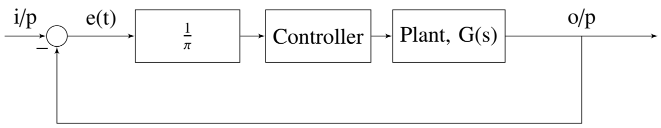

Fig. 1.1: Block Circuit diagram

Calculating plant transfer function G(s):

$V_a(t)$ is the input to the plant and output is θ_L
We know that,

$$T_m(s)\frac{R_a}{K_t} + K_b S\theta_m(s) = V_a(s) \tag{1.1.1}$$

$$(J_m S^2 + f_m S + k)\theta_m(s) - k\theta_L(s) = T_m(s) \tag{1.1.2}$$

$$(J_L S^2 + k)\theta_L(s) = k\theta_m(s) \tag{1.1.3}$$

Where,

$$\frac{R_a}{K_t} = \frac{V}{T_{stall}} = \frac{50}{100} = \frac{1}{2} \tag{1.1.4}$$

$$k_b = \frac{V}{w_{no-load}} = \frac{50}{150} = \frac{1}{3} \tag{1.1.5}$$

On Solving we get,

$$G(s) = \frac{\theta_L(s)}{V_a(s)} \tag{1.1.6}$$

$$G(s) = \frac{1}{\frac{5}{6}S^4 + \frac{10}{9}S^3 + 6S^2 + \frac{4}{3}S} \tag{1.1.7}$$

Now, lets design a controller for the following plant function to get desired output.
1) Steady state error is 1%
2) Peak overshoot is less than 15%

Steady state error condition:
Considering the closed-loop gain of plant function.
Steady state error is given by,

$$= \frac{G(s)}{1 + G(s)} \tag{1.1.8}$$

$$\tag{1.1.9}$$

We know that, steady state output is given by

$$\lim_{s\to 0} S\,R(s)\left(\frac{G(s)}{1+G(s)}\right) \tag{1.1.10}$$

$$R(s) = \frac{1}{s} \tag{1.1.11}$$

$$\text{steady state output} = 1 \tag{1.1.12}$$

Given we have 1% error,
Steady state error is given by,

$$e_{ss} = \lim_{s\to 0}\left(\frac{S\,R(s)}{1+G(s)}\right) \tag{1.1.13}$$

Here, as we see we get 0 steady state error, therefore to make it finite we cascade a differentiator to the plant function
Below is the block representation

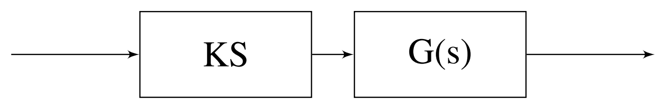

Fig. 1.1: Cascading a differentiator

As, $\frac{1}{\pi}$ is casacde to our plant, so directly multiplying for sake of simplicity.

$$\lim_{s\to 0}\left(\frac{S\,R(s)}{1+G'(s)}\right) = 0.01 \tag{1.1.14}$$

$$\lim_{s\to 0}\frac{1}{1 + \frac{1}{\pi}(KS)G(S)} = 0.01 \tag{1.1.15}$$

Solving,

$$K = 132\pi \tag{1.1.16}$$

Therefor, the overall system as can be written as,

$$G(s) = \frac{132}{\frac{5}{6}S^3 + \frac{10}{9}S^2 + 6S + \frac{4}{3}} \tag{1.1.17}$$

Peak overshoot:
Lets use a PID-controller for this purpose, So, as we want peak overshoot to be less than 15%.
So, this is the final block diagram

General expression for PID-controller given

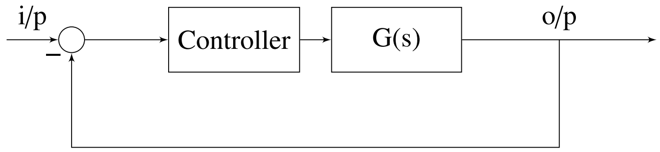

Fig. 1.1: Block diagram so far

by:

$$K_p\left(1 + T_d s + \frac{1}{T_i s}\right) \tag{1.1.18}$$

For it to statisfy our steady state condition, we will take $K_P = 1$.
And also **cascading it with a differentiator of gain =** T_i, to remove the pole and thus maintaining or steady state value.
Final transfer function is given by,

$$= \frac{132\left(S + T_d S^2 + \frac{1}{T_i}\right) T_i}{\frac{5}{6}S^3 + \frac{10}{9}S^2 + 6S + \frac{4}{3}} \tag{1.1.19}$$

Considering 2^{nd} order approximation.
Relation between %OS = 15% and Damping ratio

$$\zeta = \frac{-\ln(\%OS/100)}{\sqrt{(\pi)^2 + (\ln(\%OS/100))^2}} \tag{1.1.20}$$

$$\implies \zeta = 0.517 \tag{1.1.21}$$

Phase Margin for a Damping ratio is given by Eq (1.1.22)

$$\phi_m = 90^\circ - \arctan\left(\frac{\sqrt{-2\zeta^2 + \sqrt{1 + 4\zeta^4}}}{2\zeta}\right) \tag{1.1.22}$$

$$\implies \phi_m = 53.17^\circ \tag{1.1.23}$$

Adding a correction factor, to compensate for the phase gain, to our phase margin we get,

$$\phi_m = 70^\circ \tag{1.1.24}$$

Now, solving for the values,

$$|G(j\omega)| = \frac{132T_i\sqrt{(1/T_i - T_d\omega^2)^2 + \omega^2}}{\sqrt{(4/3 - 10\omega^2/9)^2 + (6\omega^2 - 5\omega^3/6)}} \tag{1.1.25}$$

$$= 1 \tag{1.1.26}$$

And,

$$\arctan\left(\frac{\omega}{1/T_i - T_d\omega^2}\right) - \arctan\left(\frac{6\omega^2 - 5\omega^3/6}{4/3 - 10\omega^2/9}\right) \tag{1.1.27}$$

$$= 70^\circ - 180^\circ = -110^\circ \tag{1.1.28}$$

Solving one the best fits we get is,

$$T_i = \frac{100}{132} \tag{1.1.29}$$

$$T_d = 0.1\left(\frac{132}{100}\right) \tag{1.1.30}$$

$$\omega = 17.07 \tag{1.1.31}$$

Final transfer function is given by:

$$= \frac{13.2S^2 + 100S + 132}{5/6S^3 + 10/9S^2 + 6S + 4/3} \tag{1.1.32}$$

Below is the figure for the magnitude and phase plot of the function
Following's the code:

```
codes/ee18btech11019_1.py
```

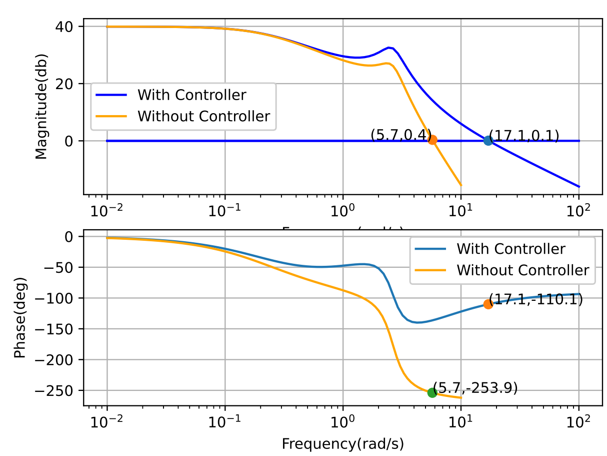

Fig. 1.1: Mag and phase plot comparison

Below is the figure for the output comparison in time domain for unit step input:
Following's the code:

codes/ee18btech11019_2.py

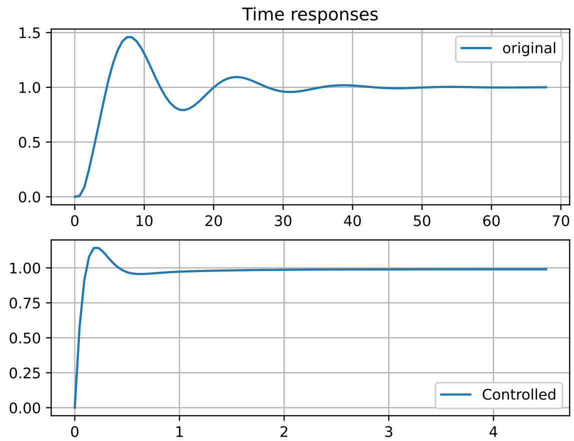

Fig. 1.1: Time response

Here, is the table of expected value and obtained value:
Originally, we had around 50% peak overshoot, without controller

Specification	Expected	Obtained
OS%	15%	14.25%
Steady state error	1%	1.007%

TABLE 1.1: Comparing the expected and Obtained results