

up

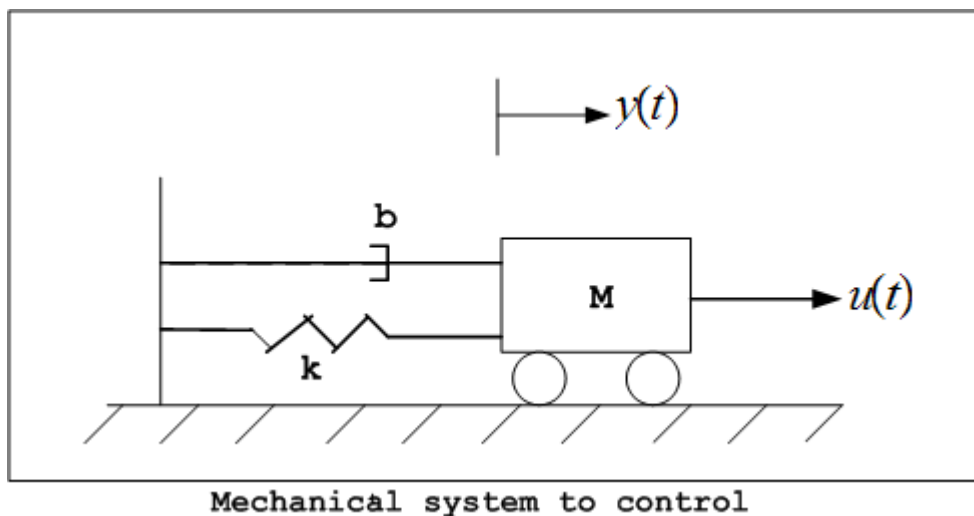
Determinatin of PID controller parameters based on step response specifications

Nasser M. Abbasi, May 2012

May 11, 2012

This paper describes how to design a PID controller for a system represented by a second order differential equation based on the user requirments for a step response given by the rise time and the settling time. The goal is to determine the 3 PID parameters (K_i, K_d, K_p) given the plant transfer function and t_r, t_s (rise time and settling time).

Consider the following mechanical system



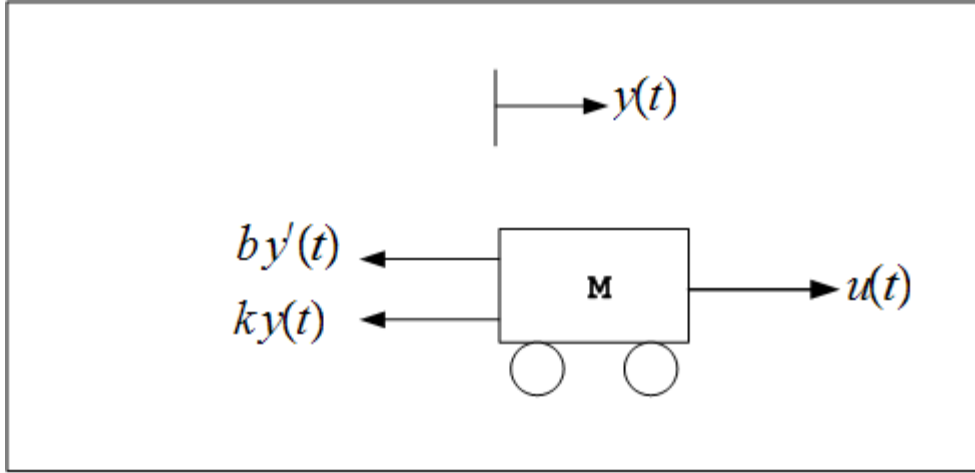
Where M is the mass of the car, b is the damping coefficient, k is the spring constant.

Assume, using standard SI units:

$$\begin{aligned}M &= 1 \text{ kg} \\ b &= 10 \text{ Ns/m} \\ k &= 20 \text{ N/m}\end{aligned}$$

The first step is to derive a mathematical model for the system. This means to derive a differential equation which relates the output (the displacment $y(t)$) to the input (which is the applied force $u(t)$). Note that we ignored the friction force between the mass M and the ground. We assume there is no friction.

Start by drawing the free body diagram, and then apply Newton second law:



Free body diagram

Apply Newton law. we obtain

$$F = ma$$

$$u(t) - by'(t) - ky(t) = My''(t)$$

or

$$My''(t) + by'(t) + ky(t) = u(t)$$

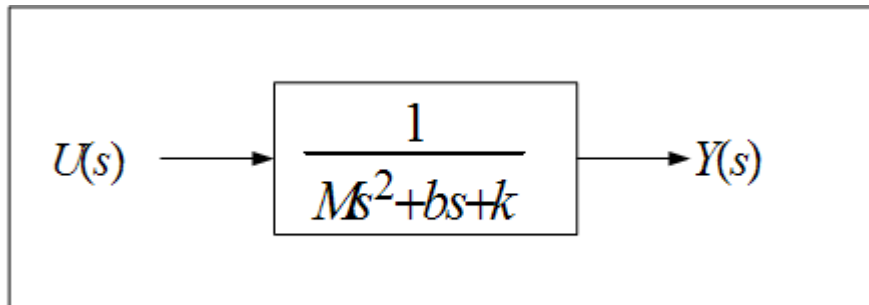
Take laplace transform, and assume initial conditions, we obtain

$$Ms^2Y(s) + bsY(s) + kY(s) = U(s)$$

The transfer function is defined as the ratio of the output to the input in the Laplace domain. Here the input is $u(t)$, which is the force on the body, and the output is $y(t)$ which is the displacement. Hence the transfer function is

$$sys(s) = \frac{Y(s)}{U(s)} = \frac{1}{Ms^2 + bs + k}$$

Using block diagram, the above can be draw as follows.



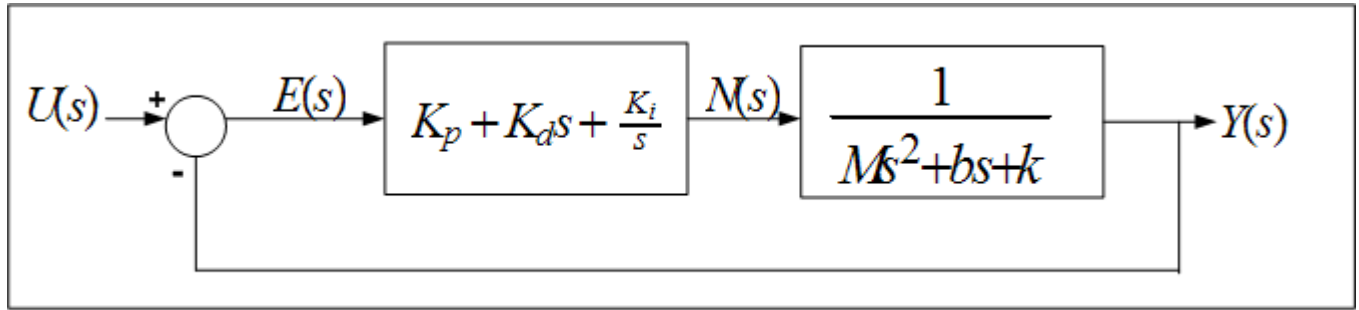
System differential equation as a transfer function

This is the system dynamic equation. Next we add a PID controller.

The transfer function of a PID controller is

$$K_p + K_d s + \frac{K_i}{s}$$

We add the above controller to the system and close the loop. The following diagram represents the updated system with this controller in place



System Block diagram with a PID controller

Let $L(s)$ be the open loop transfer function, hence

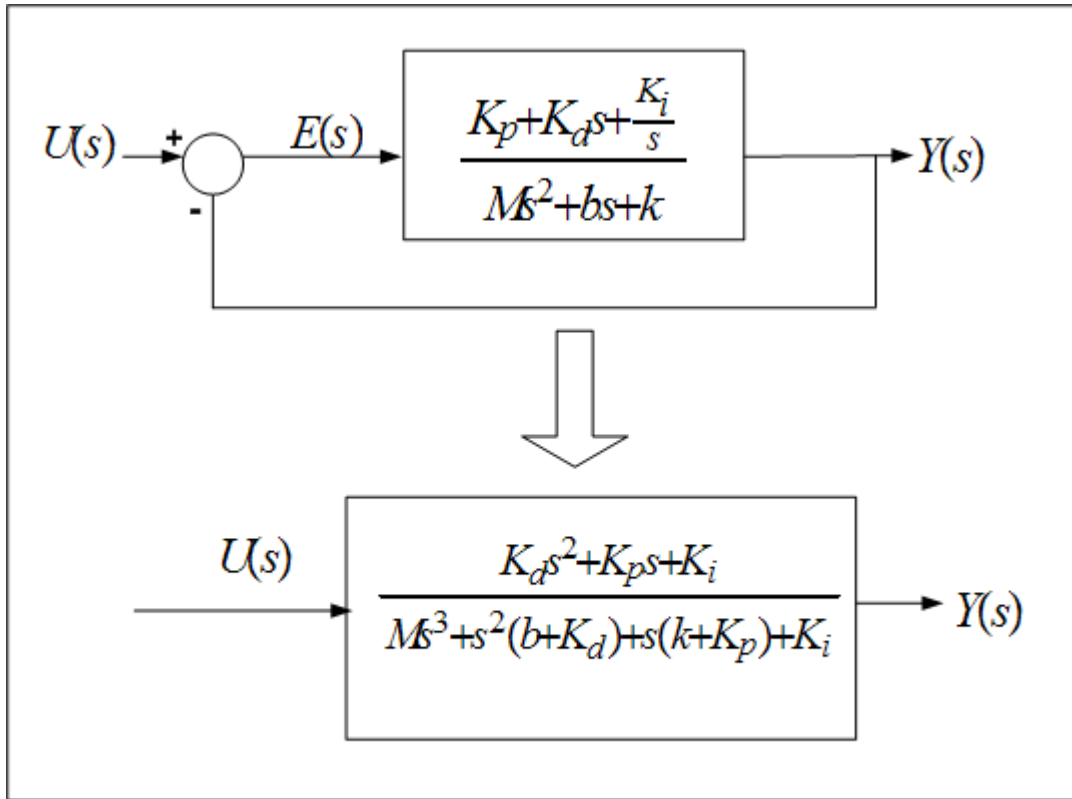
$$L(s) = \frac{K_p + K_d s + \frac{K_i}{s}}{Ms^2 + bs + k} = \frac{K_p s + K_d s^2 + K_i}{Ms^3 + bs^2 + ks}$$

Hence the closed loop transfer function is

$$G(s) = \frac{L(s)}{1 + L(s)} = \frac{K_p s + K_d s^2 + K_i}{Ms^3 + bs^2 + ks + K_p s + K_d s^2 + K_i}$$

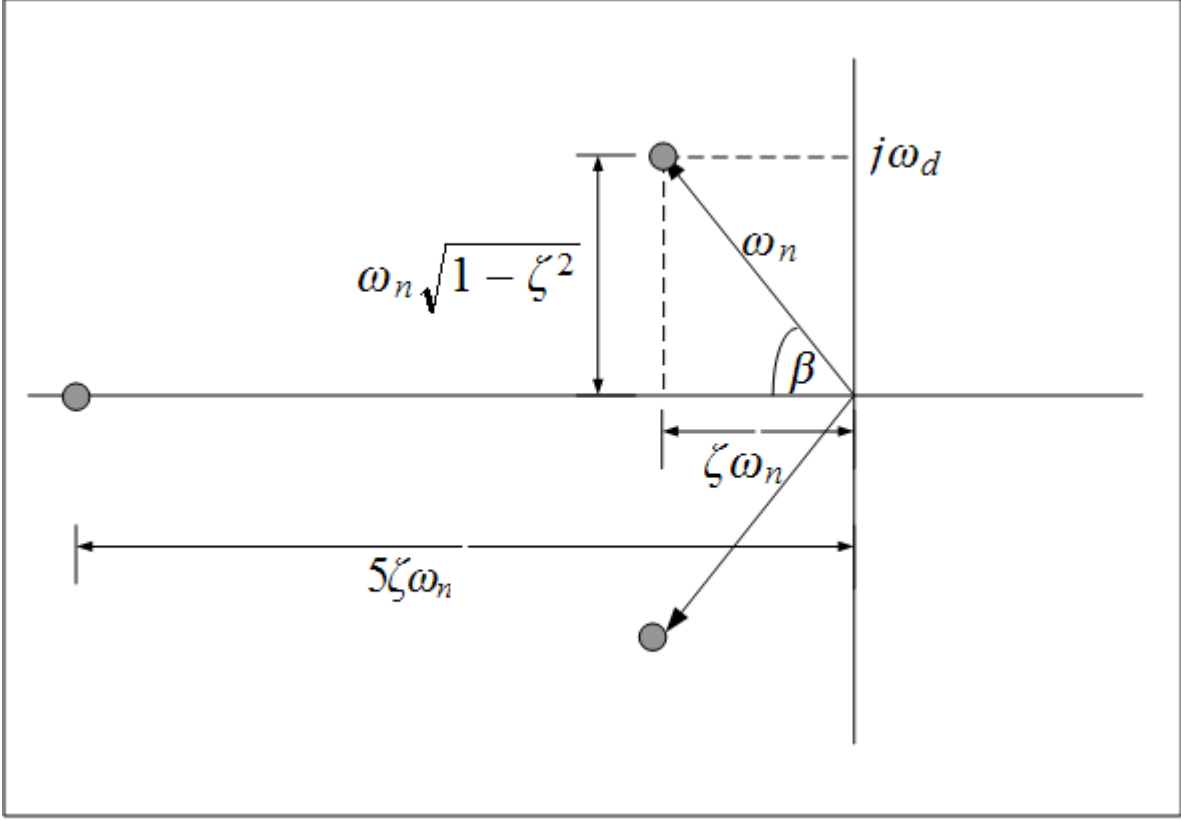
Hence

$$G(s) = \frac{\frac{1}{M}(K_d s^2 + K_p s + K_i)}{s^3 + s^2 \left(\frac{b + K_d}{M} \right) + s \left(\frac{k + K_p}{M} \right) + \frac{K_i}{M}} \quad (1)$$



Open loop transfer function using PID controller

Now looking at the closed loop transfer function (1), we see that there are 3 poles. We put one pole at a distance of $5\zeta\omega_n$ away from the imaginary axis, and the remaining 2 poles will be the dominant poles. Hence we obtain the following diagram



pole placement

Hence the denominator of equation (1) can be rewritten as

$$\begin{aligned}
 s^3 + s^2 \left(\frac{b + K_d}{M} \right) + s \left(\frac{k + K_p}{M} \right) + \frac{K_i}{M} &\equiv (s + 5\zeta\omega_n) (s^2 + 2\zeta\omega_n s + \omega_n^2) \\
 &\equiv s^3 + 2\zeta\omega_n s^2 + s\omega_n^2 + 5\zeta\omega_n s^2 + 10\zeta^2\omega_n^2 s + 5\zeta\omega_n^3 \\
 &\equiv s^3 + s^2 (7\zeta\omega_n) + s (10\zeta^2\omega_n^2 + \omega_n^2) + 5\zeta\omega_n^3
 \end{aligned}$$

equating coefficients we obtain

$$\begin{aligned}
 \frac{b + K_d}{M} &= 7\zeta\omega_n \\
 \frac{k + K_p}{M} &= 10\zeta^2\omega_n^2 + \omega_n^2 \\
 \frac{K_i}{M} &= 5\zeta\omega_n^3
 \end{aligned}$$

Solve for PID parameters, we obtain

$$\begin{aligned}
 K_d &= 7M\zeta\omega_n - b \\
 K_p &= M (10\zeta^2\omega_n^2 + \omega_n^2) - k \\
 K_i &= 5M\zeta\omega_n^3
 \end{aligned} \tag{2}$$

We have obtained the PID parameters as a function of ω_n, ζ .

Now we need to do is determine ω_n, ζ , and that will give us the PID parameters.

We now need to introduce the time response specifications we are required to design for. Assume the following are the requirements

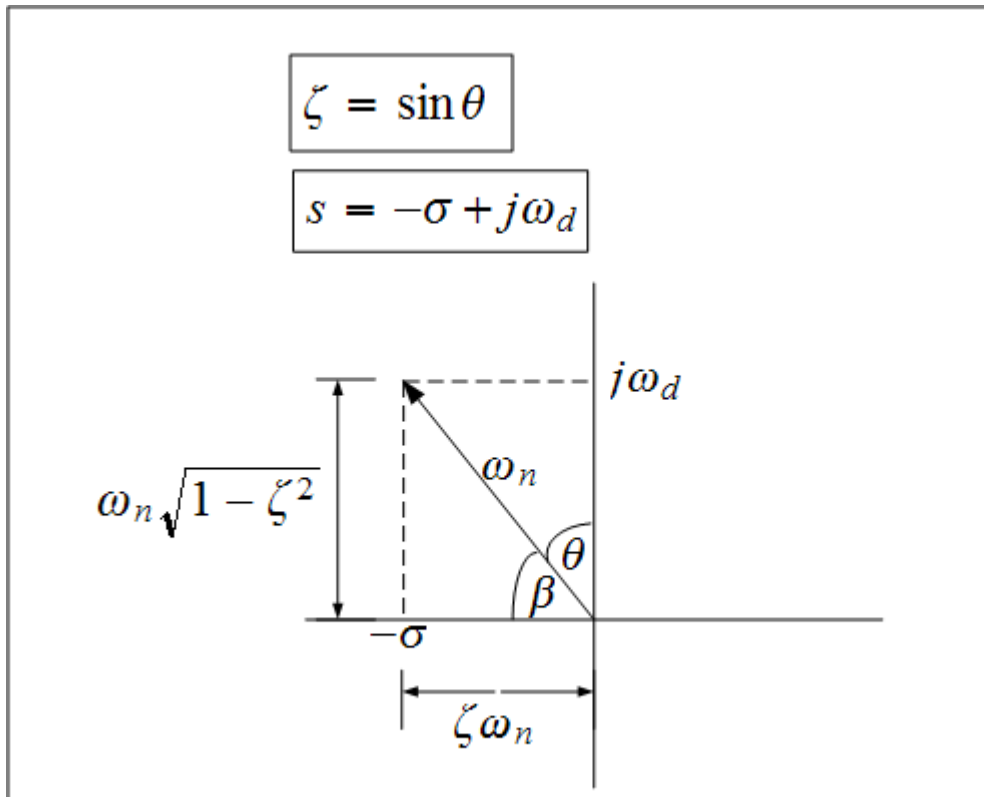
1. The settling time $t_s = 2 \text{ sec}$
2. The rise time $t_r = 0.1 \text{ sec}$

But

$$t_s = \frac{4}{\zeta\omega_n} \quad (3)$$

for 2% criterion. And the rise time is given by

$$t_r = \frac{\pi - \beta}{\omega_d}$$



Dominant pole parameters

but $\beta = \arctan\left(\frac{\omega_n\sqrt{1-\zeta^2}}{\zeta\omega_n}\right)$, hence

$$t_r = \frac{\pi - \arctan\left(\frac{\sqrt{1-\zeta^2}}{\zeta}\right)}{\omega_n\sqrt{1-\zeta^2}} \quad (4)$$

From (3) and (4) we can solve for ω_n, ζ , hence we get the following 2 equations in 2 unknowns

$$2 = \frac{4}{\zeta \omega_n}$$

$$0.1 = \frac{\pi - \arctan\left(\frac{\sqrt{1-\zeta^2}}{\zeta}\right)}{\omega_n \sqrt{1-\zeta^2}}$$

Solving the above numerically, we obtain the solution as shown below

```

restart;
> eq1:= 2 = 4 / (zeta*omega);
> k:= sqrt(1-zeta^2);
> eq2:= 0.1 = (Pi-arctan( k/zeta )) / (omega*k);
> fsolve({eq1,eq2});

```

$$eq1 := 2 = \frac{4}{\zeta \omega}$$

$$k := \sqrt{1 - \zeta^2}$$

$$eq2 := 0.1 = \frac{\pi - \arctan\left(\frac{\sqrt{1 - \zeta^2}}{\zeta}\right)}{\omega \sqrt{1 - \zeta^2}}$$

$$\{\omega = 17.00484998, \zeta = 0.1176135045\}$$

Solution using Maple

The solution is

$$\zeta = 0.1176$$

and

$$\omega_n = 17 \text{ rad/sec}$$

Substitute these values of ω_n and ζ in equation(2), and the values given for M, b and k , we obtain

$$K_d = 7(1)(0.1176)(17) - 10 = 3.9944 \tag{5}$$

$$K_p = 1(10(0.1176^2)(17^2) + (17)^2) - 20 = 308.97$$

$$K_i = 5(1)(0.1176)(17)^3 = 2888.8$$

To obtain the step response, using Matlab, this is the result

```

clear all; close all;
s=tf('s');

M=1;
b=10;
k=20;

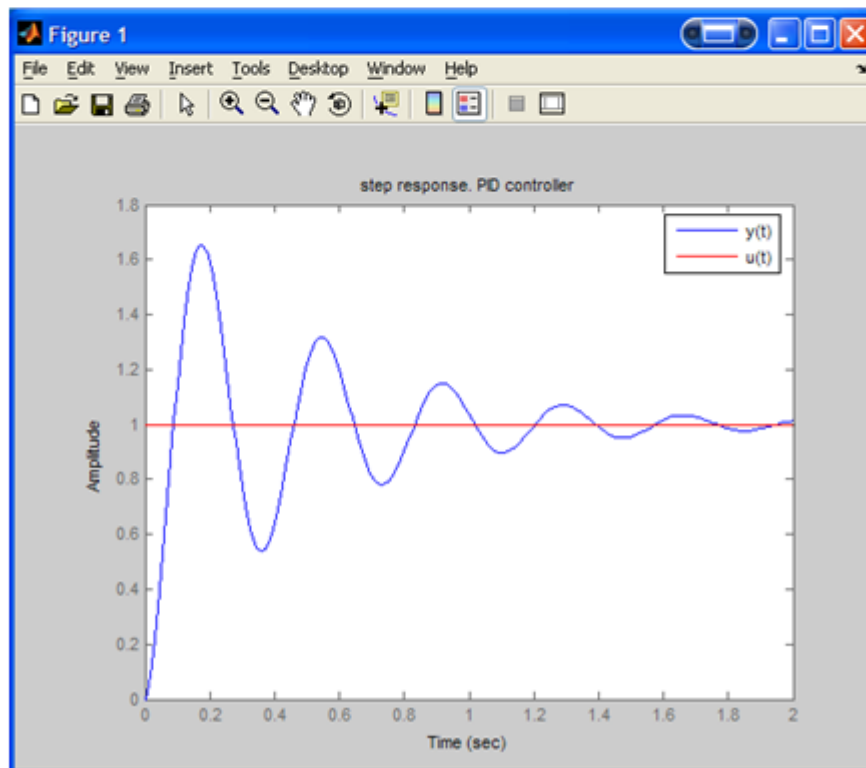
gama=17;
zeta=0.1176;

Kd=(M*7*zeta*gama)-b
Kp=M*(gama^2+10*zeta^2*gama^2)-k
Ki=5*zeta*gama^3*M

sys=(Kd*s^2+Kp*s+Ki)/(M*s^3+s^2*(b+Kd)+s*(k+Kp)+Ki);
t=[0:0.01:2];
step(sys,t);
hold on;
line([0 2],[1 1],'Color','r');
%Ylim([0 1.2]);

legend('y(t)','u(t)');
title('step response. PID controller')

```



Step response using PID controller