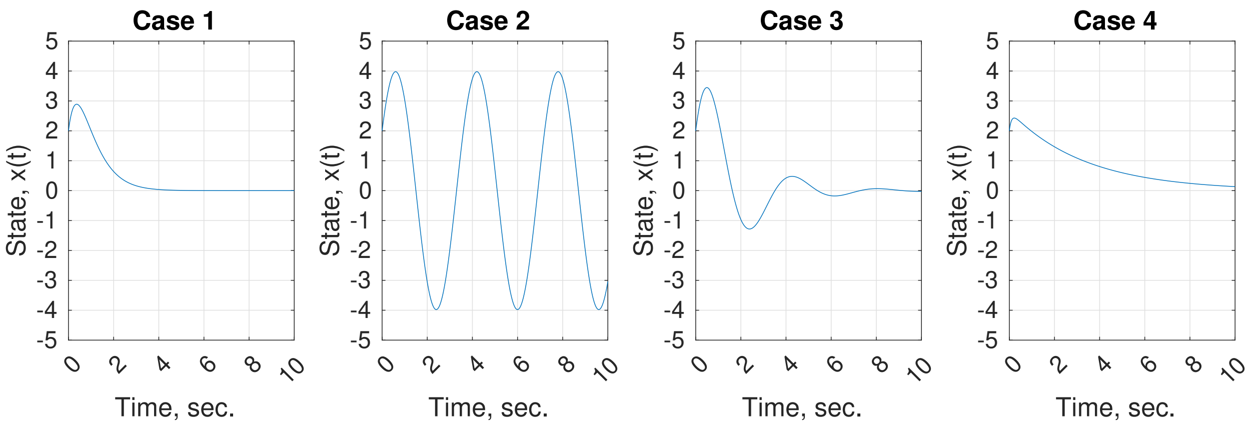
Homework 5

1 Problem



Match each of the resposes above to one of the following second-order system types:

- undamped
- underdamped
- critically damped
- overdamped

2 Problem

Consider the following transfer function:

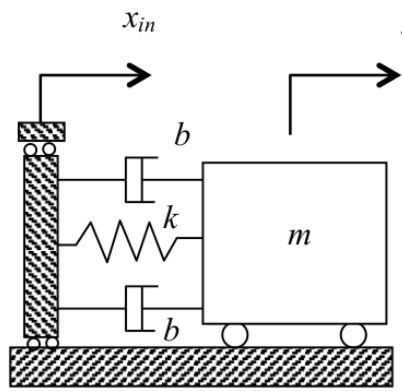
$$G(s) = \frac{X(s)}{U(s)} = \frac{100}{s^2 + 5s + 100} \, .$$

By comparing the characteristic equation in the denominator of $G(s)$ to that of a damped harmonic oscillator

$$G_{\text{DHO}}(s) = \frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

determine:

- the undamped natural frequency of the system (in both rad/s and Hz)
- the damping ratio of the system
- the damped natural frequency of the system (in both rad/s and Hz)



3 Problem

Consider the following model of a mechanical system:
The system has a mass m, a linear spring with stiffness k, and two identical dampers with damping constant b. The left wall generates an input motion $x_{in}(t)$ that causes the mass to undergo a displacement $x(t)$ from its equilibrium position. The initial position and velocity are zero.

- Find the ODE describing the motion of the system by drawing a free-body diagram and applying Newton's 2nd Law. Your answer should be in terms of the following variables:

$$x_{in}, \dot{x}_{in}, x, \dot{x}, \ddot{x}, b, k, m$$

- Show that the transfer function is

$$G(s) = \frac{X(s)}{X_{\text{in}}(s)} = \frac{\frac{2b}{m}s + \frac{k}{m}}{s^2 + \frac{2b}{m}s + \frac{k}{m}}$$

  When taking the Laplace transform $\mathcal{L}[\dot{x}_{in}(t)]$ you may assume the initial (input) condition $x_{in}(0) = 0$.

4 Problem

Define the transfer function from the previous problem in MATLAB using the `tf` command. The values of the constants are: $m = 1$ kg, $k = 97$ N/m, and $b = 4$ N-s/m.

- Suppose the input is a unit step, $x_{in}(t) = u(t)$. Use your transfer function $G(s)$ to plot the response in MATLAB.

- Suppose the input is an impulse, $x_{in}(t) = \delta(t)$. Use your transfer function $G(s)$ to plot the response in MATLAB.

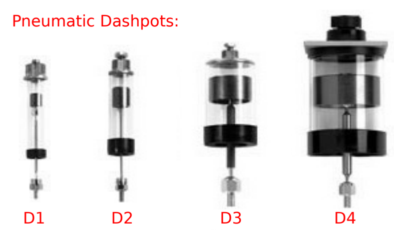
- Suppose the input is a chirp (a sinusoid of increasing frequency), $x_{in}(t) = \sin(5t^2)$. Use your transfer function $G(s)$ to plot the response in MATLAB from time 0 to 5 seconds. To ensure the input chirp is defined with sufficient resolution, use at least 1,000 time steps to represent the time interval requested.

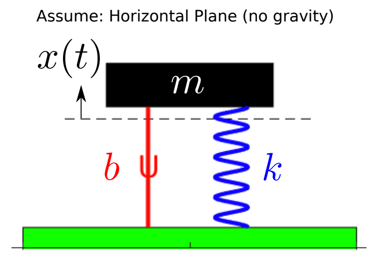
5 Problem

Consider the damped harmonic oscillator shown below and modeled by the system:

$$\ddot{x} + \left(\frac{b}{m}\right)\dot{x} + \left(\frac{k}{m}\right)x = 0 \tag{1}$$

with mass $m = 1$ kg, spring stiffness $k = 100$ N/m, and initial conditions $x(t_0) = 0.1$ m and $\dot{x}(t_0) = 0$, where $x(t)$ is the displacement of the mass from the nominal (unstretched) spring position. Four dashpots are being considered for this system: D1, D2, D3 and D4 have damping coefficients $b_1 = 3$, $b_2 = 10$, $b_3 = 20$, and $b_4 = 50$ N/(m/s), respectively.





You are tasked with selecting the correct dashpot that renders the system critically damped and to determine the response of the system over a two second interval from the initial condition. To perform your analysis, include the following:

- A *single plot with all five curves* showing the response of the system to the initial conditions for each dashpot over a two second interval, and for the case of no dashpot $b = 0$. Each curve should be plotted with a different color with time on the abscissa and the displacement $x(t)$ on the ordinate axis. Label your axes and include a legend.
- Determine the damping ratio (numerical value) for Case 0 (no dashpot) and for Cases 1-4 with dashpots D1, D2, D3, and D4. For each case specify whether it as critically damped, overdamped, undamped, or underdamped.
- Make your final recommendation on which dashpot to choose

MATLAB Hints (optional):

- To avoid repeating the same code for each of the five cases you are encouraged to encapsulate it in a function of the form below.

```
function xhist = simulateMassSpringDamper(tvals,x0,xdot0,k,m,b)

% simulate system (your code here)

end
```

This function can then be called five times in your main script to simulate each case (with a different input value for `b`).

```
b_vec = [0 3 10 20 50];
% Simulate (changing b value each time)
xhist0 = simulateMassSpringDamper(t,x0,xdot0,k,m,b_vec(1)); %
xhist1 = simulateMassSpringDamper(t,x0,xdot0,k,m,b_vec(2)); %
xhist2 = simulateMassSpringDamper(t,x0,xdot0,k,m,b_vec(3)); %
xhist3 = simulateMassSpringDamper(t,x0,xdot0,k,m,b_vec(4)); %
xhist4 = simulateMassSpringDamper(t,x0,xdot0,k,m,b_vec(5)); %
```

- You can animate/verify your solution by downloading the `playMassSpringDamper.m` file from Canvas, placing it in your working directory, and running the following commands once your simulation is complete:

  ```
  playMassSpringDamper(thist,xhist)
  ```

  The above command assumes `xhist` is a `1 x N` vector of displacements, and `thist` is a `1 x N` vector of corresponding times (from 0 to 2 seconds) where a value of around `N=100` was used to produce the animations posted on Canvas.