Contents


```
%{
Example of 2nd order system stability
%}

clear
clc
close all
```

System data for free response

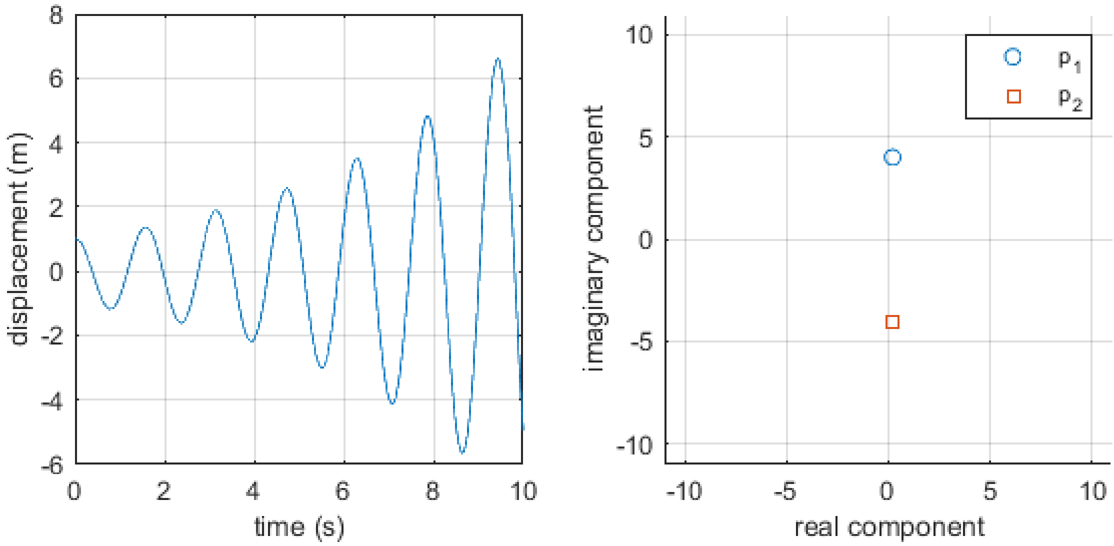
```
wn = 4;                       % natural frequency omega_n, rad/sec
z = -0.05;                       % damping z for a stable underdamped system
EOM = [1 2*z*wn wn^2];        % standard form of the EOM for free vibration
x_0 = 1;                   % initial displacement of the system
v_0 = 0;                      % initial velocity of the system
tt = linspace(0,10,1000);     % time vector of the system

% calculate the roots of the system
p=roots(EOM)

% select the damping case and solve for the temporal displacement
if abs(z) < 1 % underdamped
    A = sqrt(wn^2*x_0^2+v_0^2)/wn;
    phi = atan((x_0*wn)/v_0);
    xx = A*exp(-z*wn*tt).*sin(wn*tt+phi);
elseif abs(z) == 1 % critically damped
        a_1 = x_0;
        a_2 = v_0+wn*x_0;
        xx = (a_1 + a_2*tt).*exp(z*-wn*tt);
elseif abs(z) > 1 % overdamped
        a_1 = (-v_0 + (-z + sqrt(z^2-1) )* wn*x_0)/(2 * wn * sqrt(z^2 -1));
        a_2 = (v_0 + (z + sqrt(z^2-1) )* wn*x_0)/(2 * wn * sqrt(z^2 -1));
        xx = exp(-z*wn*tt) .* (a_1* exp(-wn*tt*sqrt(z^2-1)) + a_2 * exp(wn*tt*sqrt(z^2-1)));
end
```

```
p =

   0.2000 + 3.9950i
   0.2000 - 3.9950i
```

Plot the results

```
% plot the time-series response
figure('units','inch','position',[2,2,7,3])
subplot(1,2,1)
plot(tt,xx)
xlabel('time (s)')
ylabel('displacement (m)')
```

```
grid on

% plot the poles
subplot(1,2,2)
hold on
axis([-11,11,-11,11])
grid on
scatter(real(p(1)),imag(p(1)))
scatter(real(p(2)),imag(p(2)),'s')
legend('p_1','p_2')
xlabel('real component')
ylabel('imaginary component')
```



Published with MATLAB® R2021a