%% Transfer function definitions

z1 = -0.435; p1 = [-1.23 -0.113-1j\*0.0642 -0.113+1j\*0.0642]; k1 = -0.125;

z2 = -0.435; p2 = [-0.113-1j\*0.0642 -0.113+1j\*0.0642]; k2 = -0.1016;

z3 = []; p3 = [-0.113-1j\*0.0642 -0.113+1j\*0.0642]; k3 = -0.0442;

G1 = zpk(z1, p1, k1); % Represents original tf

G2 = zpk(z2, p2, k2); % Represents G1 + 5 times rule

G3 = zpk(z3, p3, k3); % Represents G1 + pole-zero cancellation

**Output**

G1 =

-0.125 (s+0.435)

---------------------------------

(s+1.23) (s^2 + 0.226s + 0.01689)

Continuous-time zero/pole/gain model.

G2 =

-0.1016 (s+0.435)

------------------------

(s^2 + 0.226s + 0.01689)

Continuous-time zero/pole/gain model.

G3 =

-0.0442

------------------------

(s^2 + 0.226s + 0.01689)

Continuous-time zero/pole/gain model.

%% Time domain representation

syms s;

G1\_syms = (-0.125\*(s+0.435))/((s+1.23)\*(s^2 + 0.226\*s + 0.01689))

ilaplace(G1\_syms)

**Output**

g1\_t =

(189012500\*exp(-(113\*t)/1000)\*(cos((4121^(1/2)\*t)/1000) + (1261088\*4121^(1/2)\*sin((4121^(1/2)\*t)/1000))/62313641))/70476903 - (331250\*exp(-(123\*t)/100))/5132421 - 181250/69249

%% Response of the system (with pole-zero reduction)

figure

step(G2)

grid

title('With pole-zero reduction')



%% Response of the system (with 5 times rule reduction)

figure

step(G3)

grid

title('With 5 times rule reduction')



%% Response of the system (complete system, without reduction)

figure

step(G1)

grid

title('Without reduction')



%% Comparison

figure

hold on

step(G1)

step(G2)

step(G3)

legend('Without reduction', 'With pole-zero reduction', ...

'With 5 times rule reduction')

grid on

title('Response Comparison')



%% Rise Time

syms wn\_t zeta;

zeta\_fit = linspace(0.1, 0.9, 5000);

c\_t = 1 - (1/sqrt(1-zeta^2))\*exp(-zeta\*wn\_t)\*cos(wn\_t\*sqrt(1-zeta^2)- ...

atan(zeta/sqrt(1-zeta^2)));

c\_fit = subs(c\_t, zeta, zeta\_fit);

norm\_time = zeros(1, 5000);

for i = 1:5000

t1 = vpasolve(c\_fit(i) == 0.9, wn\_t, [0 5]);

t2 = vpasolve(c\_fit(i) == 0.1, wn\_t, [0 5]);

norm\_time(i) = t1 - t2;

end

plot(zeta\_fit, norm\_time)

grid

title('Normalized time vs. Damping ratio')

xlabel('Damping ratio $\zeta$')

ylabel('Normalized time $\omega\_{n}t$')

% polyfit returns coefficients of nth (4 here) order polynomial

P = polyfit(zeta\_fit, norm\_time, 4);

