E324, Control Systems Lab, Problem sheet 2

Dhruv Shah 190020039

Q1]

Here is the Matlab Code:

%% Q1 1st order system

sys = tf(38,[1 4]);

% step(sys);

S = stepinfo(sys);

for a = 38:38\*100

sys = tf([a],[1 4]);

S = stepinfo(sys);

x(a-37) = S.RiseTime;

end

for b = 4:4\*100

sys = tf([38],[1 b]);

S = stepinfo(sys);

y(b-3) = S.RiseTime;

end

plot(38:38\*100,x);

title(" Variation of Rise time wrt a");

xlabel(" a ");

ylabel("Rise Time");

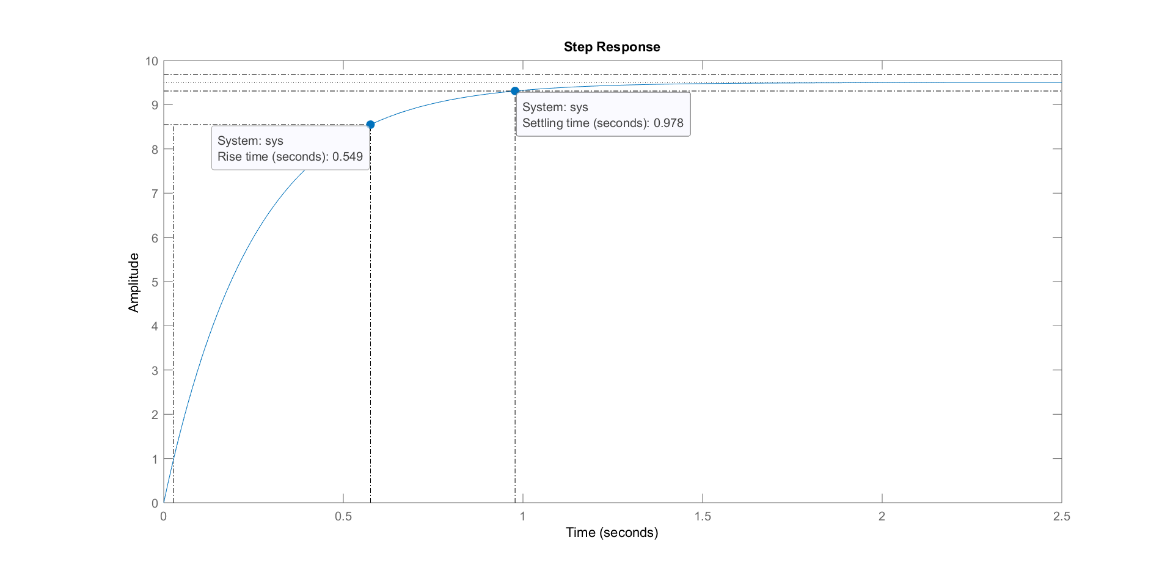
plot(4:4\*100,y);

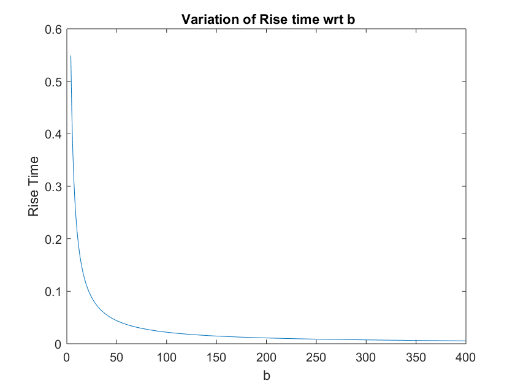
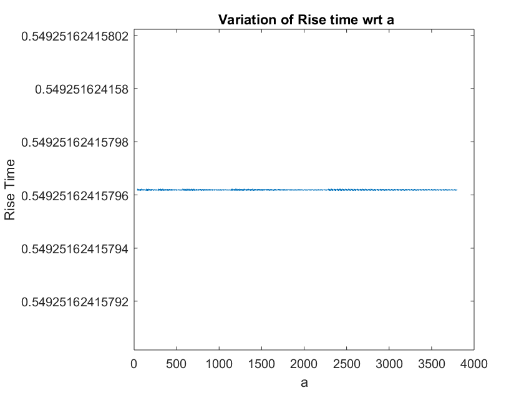
title(" Variation of Rise time wrt b");

xlabel(" b ");

ylabel("Rise Time");

a)Time Constant = b = ¼

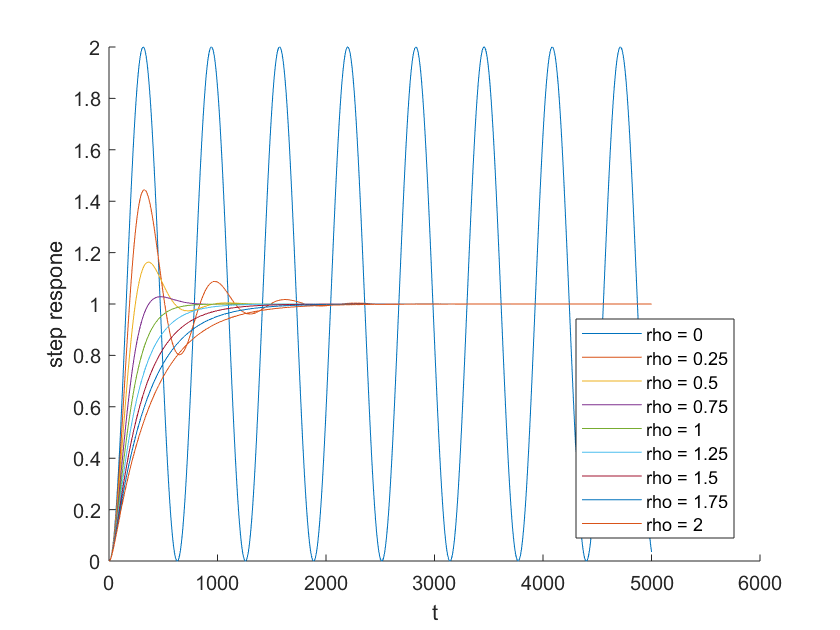
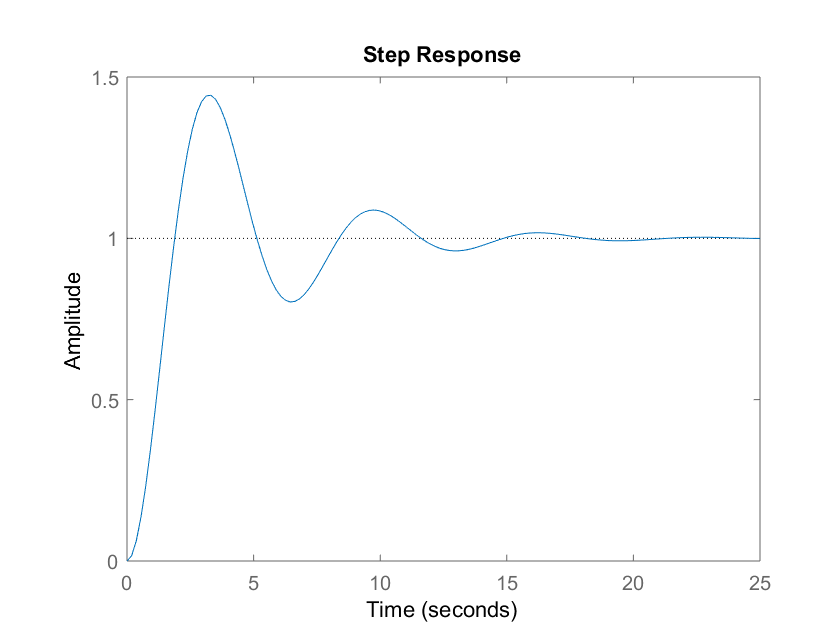




Conclusions:

* Rise Time does not depend on a
* Rise Time has an inverse relationship with b. (of the form 1/b)

Q2]



All the required information and steps required to reach the plots are mentioned in the code :

MATLAB Code:

%% Second Order System Time Domain Analysis

w = 1;

rho = 0.25;

t = 0:0.01:50;

sys = tf(w^2,[1 2\*rho\*w w^2]);

x = step(sys,t);

rho = 0:0.25:2;

Y = ones(length(rho),length(t));

hold on

for i = 1:length(rho)

sys = tf(w^2,[1 2\*rho(i)\*w w^2]);

Y(i,:) = step(sys,t);

S = stepinfo(sys);

rise\_time(i) = S.RiseTime;

Overshoot(i) = S.Overshoot;

PeakTime(i) = S.PeakTime;

SettlingTime(i) = S.SettlingTime;

plot(Y(i,:))

xlabel('t');

ylabel('step respone');

end

legend( 'rho = 0','rho = 0.25','rho = 0.5','rho = 0.75','rho = 1','rho = 1.25','rho = 1.5','rho = 1.75' ,'rho = 2' );

hold off

Conclusions:

As the damping ratio is increased the following trends are observed from the plot

* Percentage Overshoot reduces as rho increases from 0 to 1 and is zero for all rho >= 1.
* The peak time and rise time increase as rho increases
* The settling time reduces as rho goes from 0 to 1 but increases when rho > 1

Q3]

MATLAB Code:

%% Differences in First and Second Order Responses

sys\_first\_order = tf(4,[1 4]);

w = 1;

rho = 1.5;

sys\_second\_order = tf(w^2,[1 2\*rho\*w w^2]);

rho = 1;

sys\_second\_order\_critical = tf(w^2,[1 2\*rho\*w w^2]);

x = step(sys\_first\_order);

y = step(sys\_second\_order);

z = step(sys\_second\_order\_critical);

hold on

plot(x);

plot(y);

plot(z);

xlabel('time (t)');

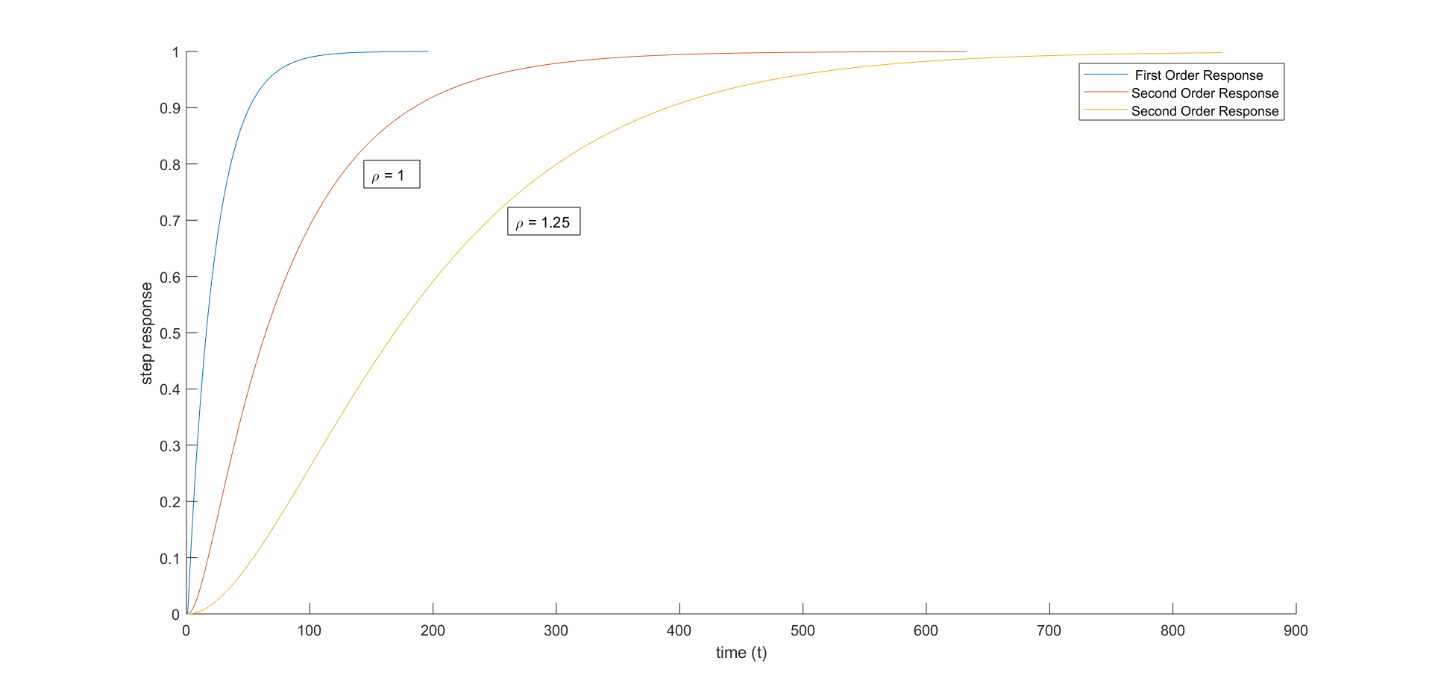
ylabel('step response');

legend(' First Order Response' ,'Second Order Response' ,'Second Order Response');

hold off

S = stepinfo(sys\_second\_order\_critical);

S.Overshoot % Answer = 0



Conclusions:

* The second order responses are slower than the first order responses.
* The fastest response for the second order system is given by the critically damped system (rho = 1) (repeated poles) and the response becomes slower as we increase the damping constant.
* The percentage overshoot for rho = 1 response was calculated and the output was zero. Hence, the output is monotonic

Q4]

MATLAB Code:

%% Q4 Discrete and Continuous Time Transfer Functions

s = tf('s');

G = 1/s;

% step(G);

ts = 0.1;

z = tf('z',ts);

H = 1/z;

% step(H)

T = 1/(s+z); % This line gives an Error

step(T);

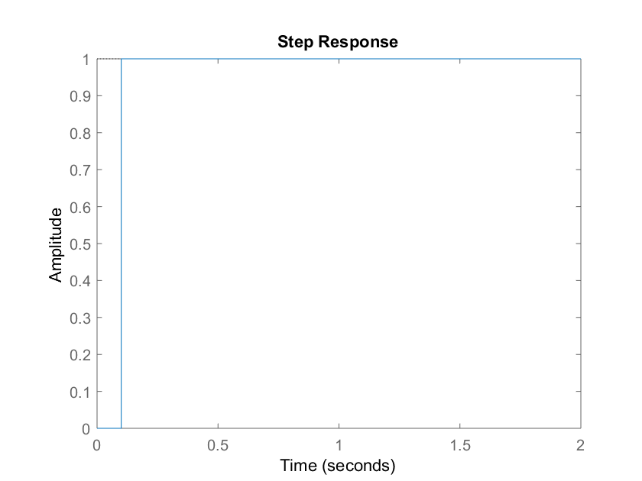
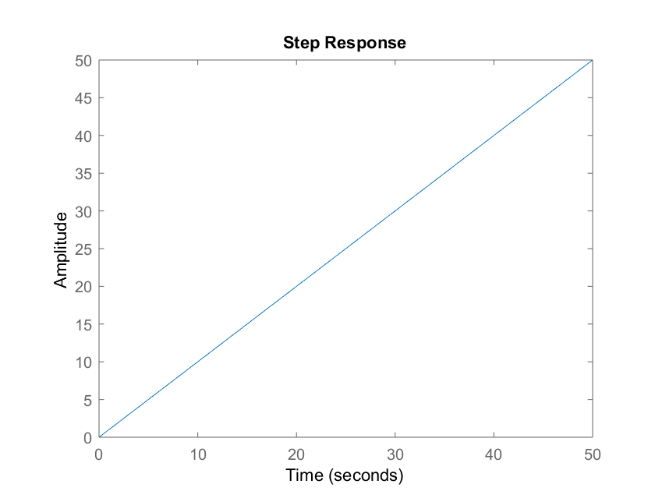
Scilab Code:

z = poly ( 0 , ’ z ’ ) ;

G = 1/ z ; s1 = tf2ss(G) ;

x = dsimul(s1,ones( 1 , 1 0 ) ) ;

plot( x ) ;



Conclusions:

* The plots are different because in the discrete case we have chosen a sampling time = 0.1
* On providing discrete as well as continuous variables in a single function, scilab throws the warning ”WARNING: csim: Input argument 1 is assumed continuous time.” and forces the other argument to continuous time
* However in Matlab , there is an error and the program doesn’t run.

Q5]

MATLAB Code:

%% Q5 Order of Blocks

G1 = tf([1 5],[1 6 8]);

H1 = tf([1 5],[1 4]);

H2 = tf(1,[1 2]);

% tau = 0.1;

% tau = 0.5;

tau = 2;

t = [0:tau:10];

y1 = step(G1,t);

u1 = step(H1,t);

u2 = step(H2,t);

y2 = lsim(H2,u1,t);

y3 = lsim(H1,u2,t);

hold on

plot(t,y1);

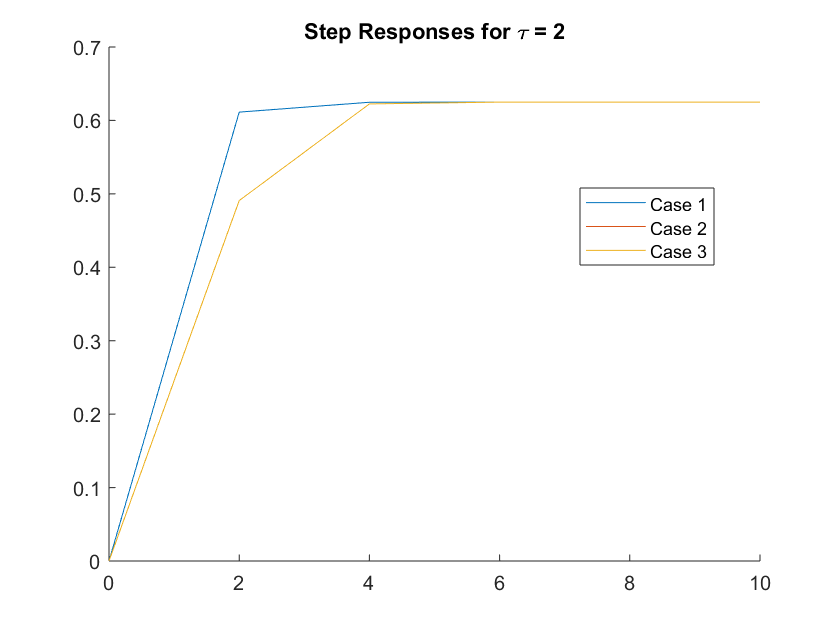
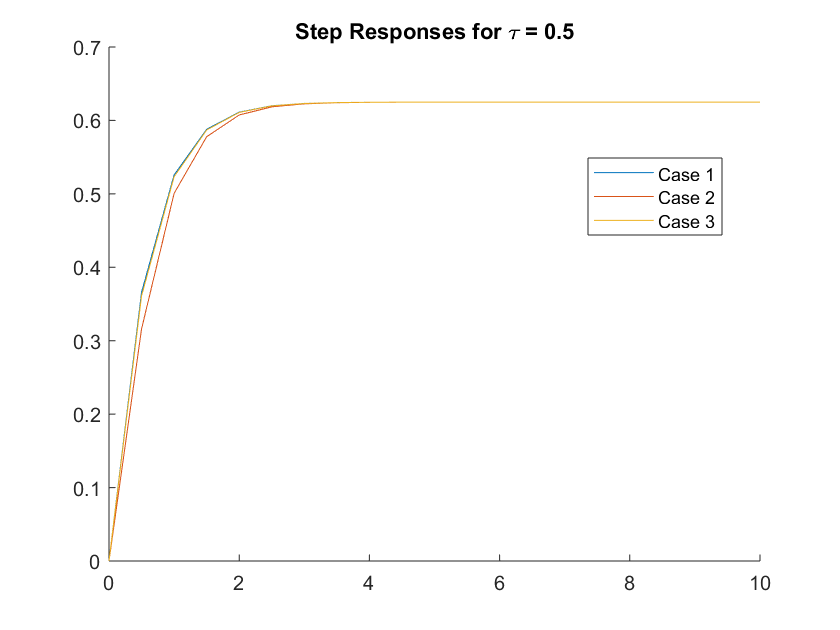
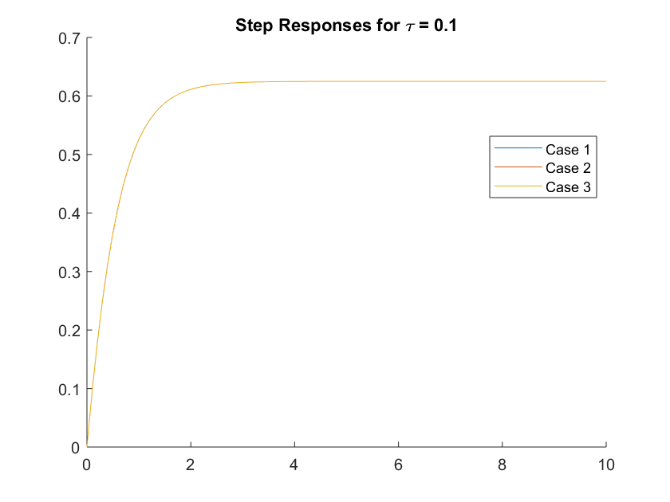
plot(t,y2);

plot(t,y3);

legend('Case 1','Case 2','Case 3');

title('Step Responses for \tau = 2');

hold off



Conclusions:

* We observe that there is almost no error when tau = 0.1 and all the plots coincide
* However upon increasing the value of tau more error creeps in and the plots do not coincide.