EE324 CONTROL SYSTEMS LAB

Problem Sheet 2

Yatish Vaman Patil | 190070076

Question 1 :

Part A)

$$\mathbf{G(s)} = \frac{\mathbf{76}}{\mathbf{s+25}} = \frac{a}{s+b}$$

Here we have a continuous-time LTI system with transfer function G(s).

Scilab code for the same:

```
s = poly(0,'s');
Gs = 76/(s+25);
Gs = syslin('c',Gs);
```

Part B)

Here we have to find the step response of the given continuous-time LTI system. Let the C(s) be output and R(s) be input. Here R(s)=1/s. Using inverse Laplace transform, we can find output c(t) as follows.

$$\mathbf{C(s) = R(s)G(s)}$$

$$\mathbf{C(s)} = \frac{\mathbf{76}}{\mathbf{(s+25)(s)}}$$

By taking inverse Laplace transform, we get

$$\mathbf{c(t)} = \frac{\mathbf{76}}{\mathbf{25}} - \frac{\mathbf{76}e^{-25t}}{\mathbf{25}}$$

This equation is an exponential plateau function with a steady-state value of 76/25.

We know that, for a first-order system, rise time, settling time and time constant is calculated

- Time constant = τ = 1/b

τ = 1/25 = 0.04 s

- Rise time = $t_{90\%}$ - $t_{10\%}$

$$t_{rise} = \frac{\ln(10) - \ln\left(\frac{10}{9}\right)}{b} = \frac{\ln(9)}{b} = 0.0879 \text{ s}$$

- Settling time = $\frac{\ln(50)}{b}$ = 0.156 s

a = 76 & b=25

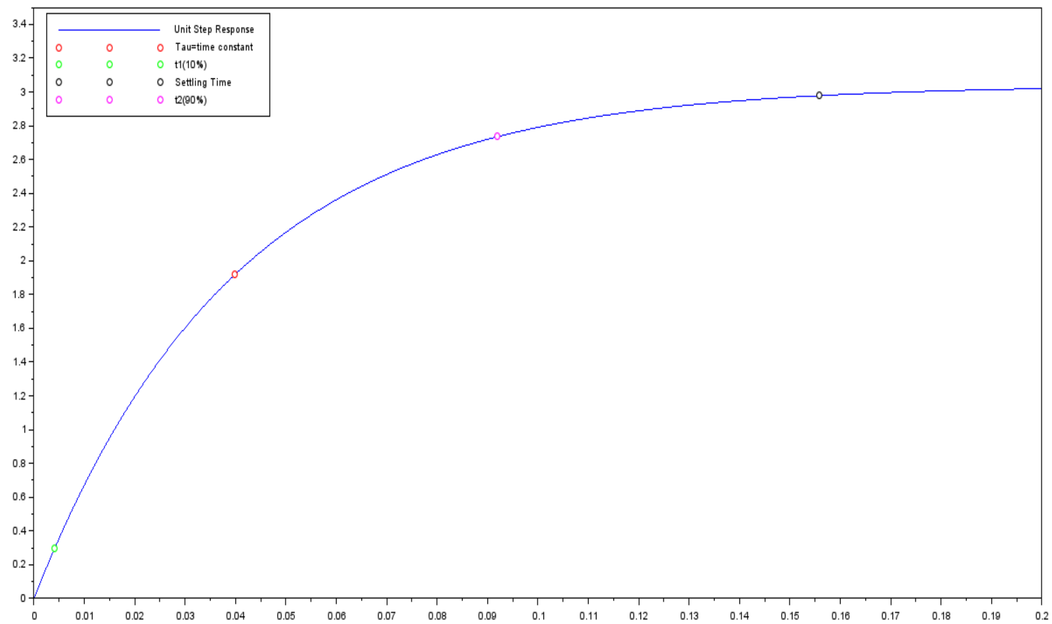


Figure: Unit Step Response of System

Scilab code for the same:

```
s = poly(0,’s’);
Gs=syslin('c',76/(s+25));
tau=1/25;
t1=log(10/9)*tau;
t2=log(10)*tau;
trise=t2-t1;
tsettle=log(50)*tau;
time=0:0.0001:0.2;
f=csim('step',time,Gs);
plot(time,f,1);
plot(time(400),f(400),'ro');
plot(time(42),f(42),'go');
plot(time(921),f(921),'mo');
plot(time(1560),f(1560),'ko');
legend('Unit Step Response','Tau=time constant', 't1(10%)', 't2(90%)',
'Settling Time',2);
```

Part C)

In this part, we have to vary the value of 'a' from 76 to 7600 in steps of 76. We have to analyse its effect on rising time.

We can observe that rise time is independent of the value of a

$$t_{rise} = \frac{\ln(10) - \ln\left(\frac{10}{9}\right)}{b} = \frac{\ln(9)}{b} = 0.0879 \text{ s}$$

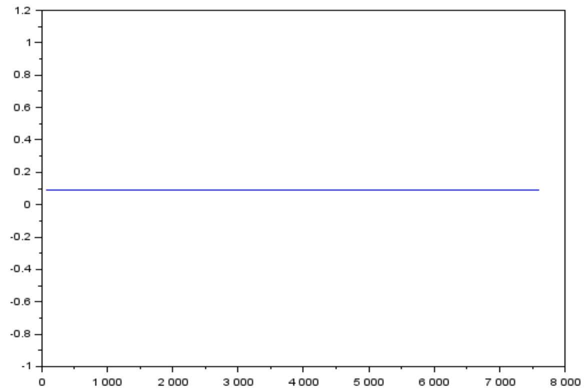


Figure: Variation of Rising time with 'a'

Scilab code for the same:

```
ax=[76:76:7600];
plot(ax,log(9)/25);
```

Part D)

In this part, we have to vary the value of 'b' from 25 to 2500 with the steps of 25. We also have to analyse its effect on the rising time of unit step response.

We can observe that rising time is inversely proportional to the value of 'b'

$$t_{rise} = \frac{\ln(10) - \ln\left(\frac{10}{9}\right)}{b} = \frac{\ln(9)}{b}$$

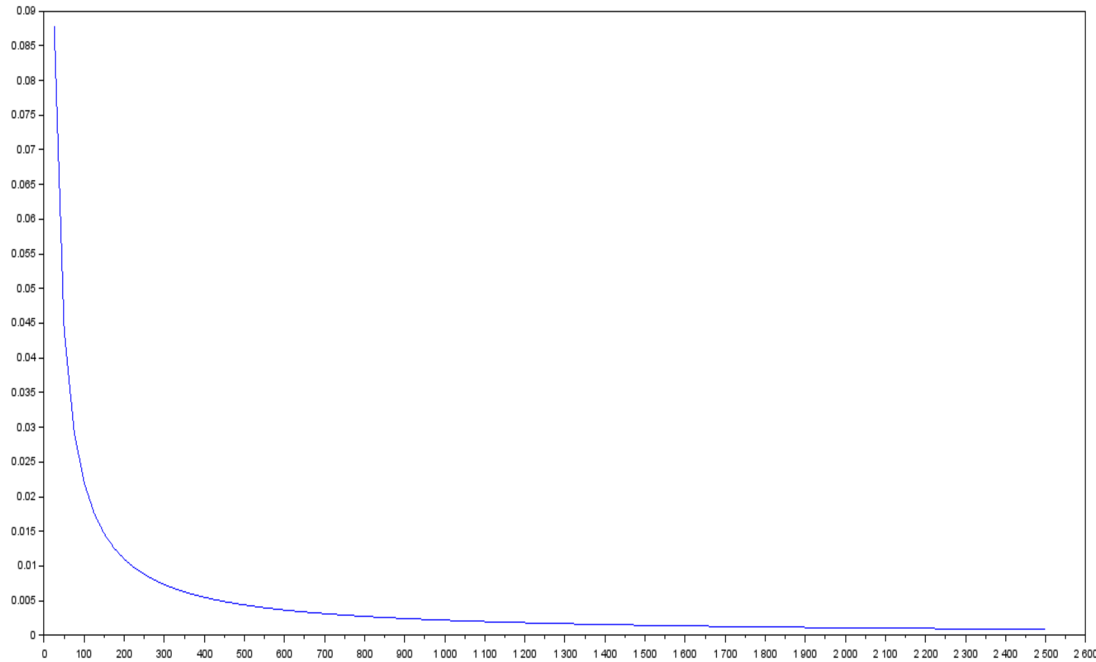


Figure: Variation of the Rising time with 'b'

Scilab code for the same:

```
bx=[25:25:2500];
yx=log(9)*bx^-1;
plot(bx,yx);
```

Question 2 :

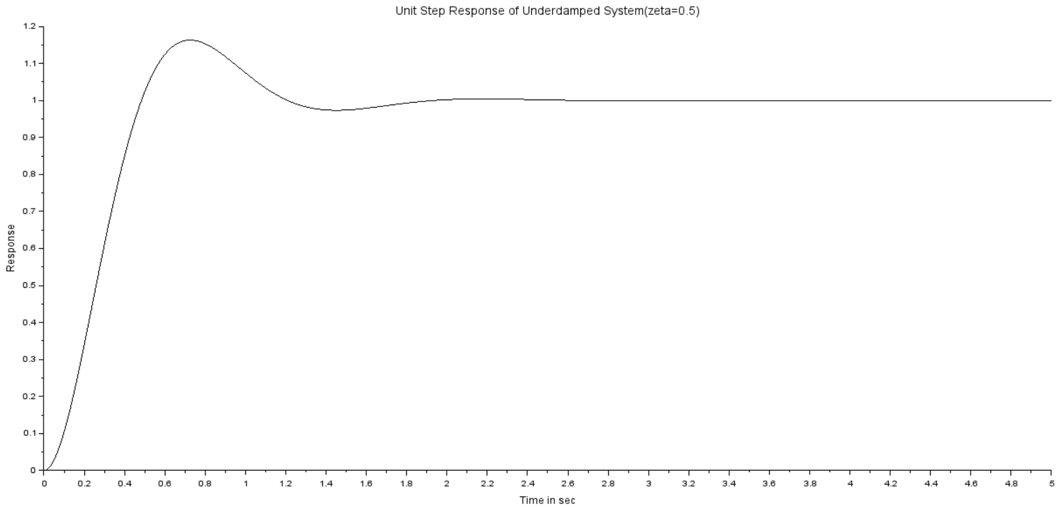
Part A)

In this part, we have to find step responses for the underdamped second-order system. The General second-order system is

$$G(s) = \frac{w^2}{s^2 + 2s\zeta w + w^2}$$

For an underdamped system, let's take ζ=0.5 and w=5

$$G(s) = \frac{25}{s^2 + 5s + 25}$$



Scilab code for the same:

```
s=%s;
Gs=25/(s^2+5*s+25);
Gs=syslin('c',Gs);
time=0:0.001:5;
PlotY=csim('step',time,Gs);
plot2d(time,PlotY);
xtitle('Unit Step Response of Underdamped System(Zete=0.5)',' Time in
sec',' Response');
```

Part B)

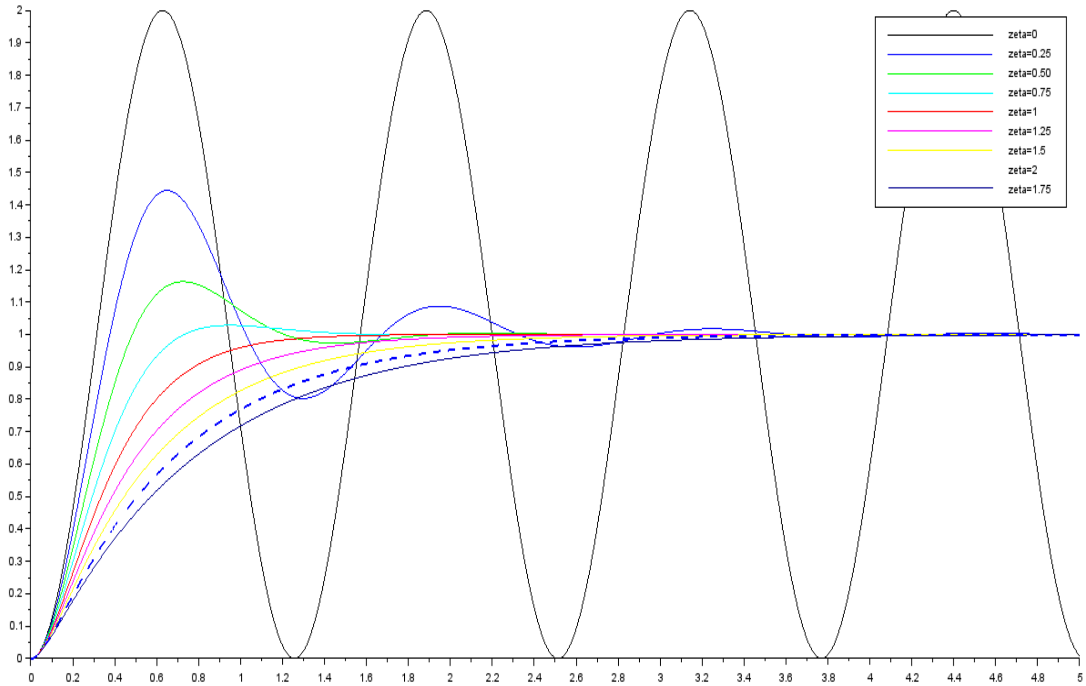
Here we have to find a time-domain response for the second-order system mentioned above for step input. We will vary the value ζ from 0 to 2 such that

For ζ=0 ==> Undamped System

0<ζ<1 ==> Underdamped System

ζ=1 ==> Critically damped System

ζ>1 ==> Overdamped System



Scilab code for the same:

```
s=%s;
zeta=0:0.25:2;
time=0:0.0001:5;
for n=1:9
den1=s^2 + 2*zeta(n)*5*s + 25;
Gs=syslin('c',25,den1);
PlotY=csim('step', time,Gs);
plot2d(time,PlotY,n);
f=gcf();
f.children.children(1).children.line_style = n;
f.children.children(1).children.thickness = 2;
end
leg=legend('zeta=0','zeta=0.25','zeta=0.50','zeta=0.75','zeta=1','zeta
=1.25','zeta=1.5','zeta=2','zeta=1.75',1);
```

Question 3 :

Part A)

Here we have two systems, first and second order, respectively. As stated, both systems have their step response increasing monotonically. For this reason, I have used an overdamped second-order system.

$$\text{First Order System: } G1(s) = \frac{5}{s+5} = \frac{a}{s+a}$$

$$\text{Second-Order System: } G2(s) = \frac{36}{s^2+18s+36} = \frac{w^2}{s^2+2\zeta ws+w^2}$$

Let c(t) be the time domain response

Salient Features Between First and Second-Order system

First Order System	Second-Order System
The first-order system can not overshoot	Second-order system overshoots for underdamped system
A single parameter(a) describes the First-order system	The Second-order system needs two parameters(w and ζ) for it to be defined
At t=0, the slope of c(t) is a i.e. c’(t)=a	At t=0, the slope of c(t) is zero i.e. c’(t)=0

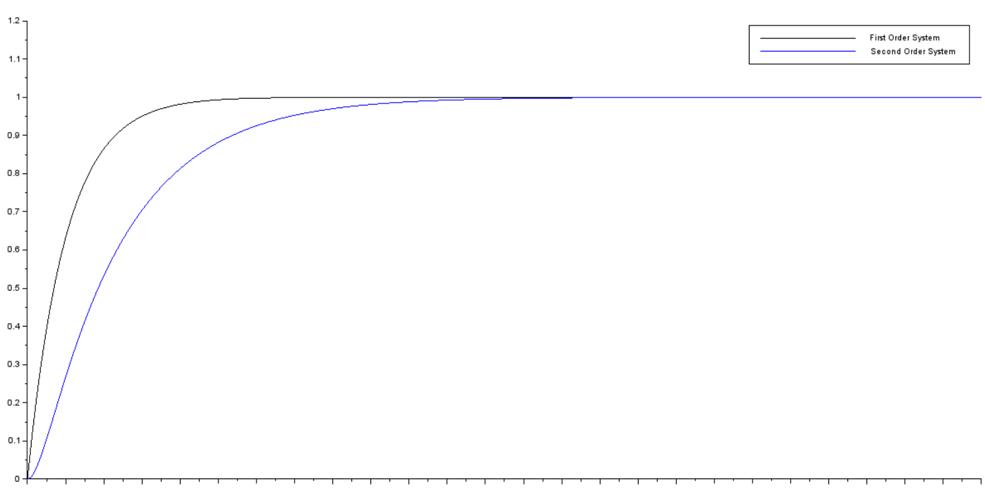


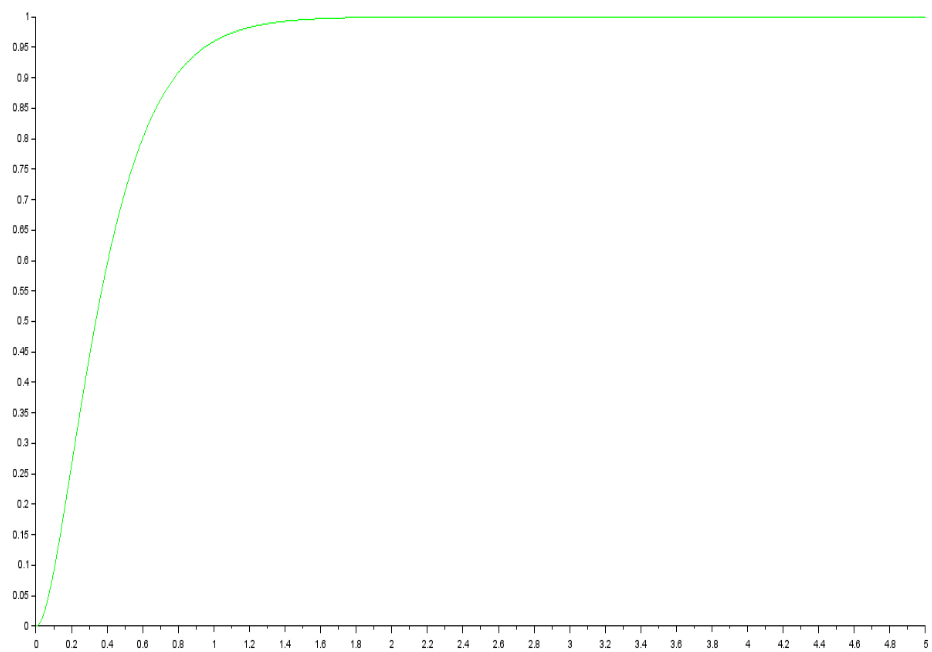
Figure: Step response of First and Second-Order system

Scilab code for the same:

```
s=%s;
Gs1=5/(s+5);
Gs2=36/(s^2+18*s+36);
Gs1=syslin('c',Gs1);
Gs2=syslin('c',Gs2);
time=0:0.0001:5;
Ps1=csim('step',time,Gs1);
Ps2=csim('step',time,Gs2);
plot2d(time,Ps1,1);
plot2d(time,Ps2,2);
leg1=legend('First Order System','Second Order System',1);
```

Part B)

We have no overshooting for the repeated poles(real) in a second-order system, i.e. critically damped system. The system reaches its final value monotonically



Question 4 :

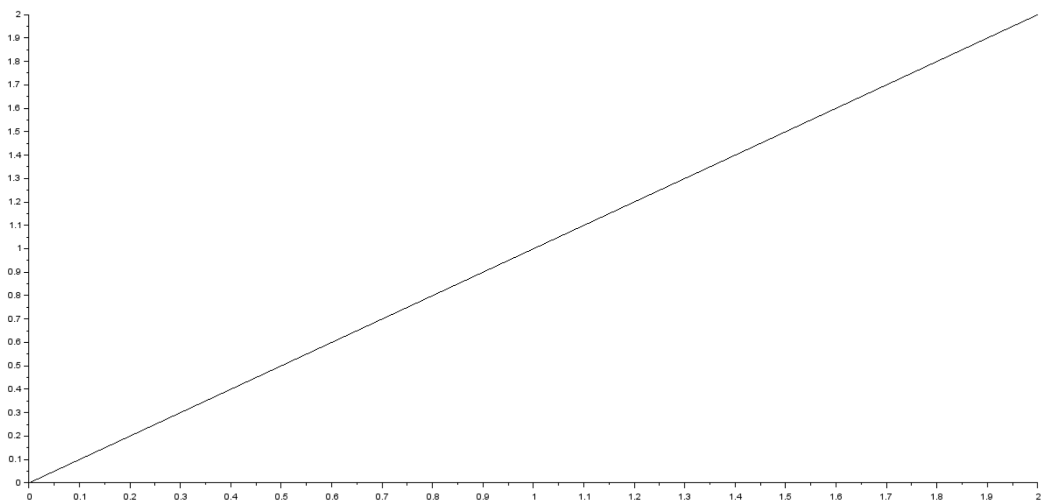
Part A)

In this case, we have been given an integrator as our transfer function. The integrator is 1/s, and we have to find its step response.

$$G(s) = \frac{1}{s}$$

Output= $C(s)=\frac{1}{s^2}$

c(t)=t

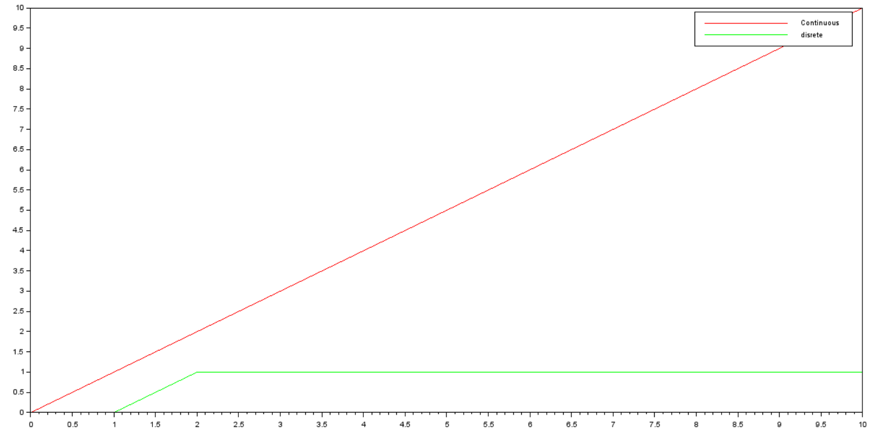


Scilab code for the same:

```
s=%s;
Gs=1/s;
Gs=syslin('c',Gs);
time=0:0.01:2;
PlotY=csim('step',time,Gs);
plot(time,PlotY);
```

Part B)

In this part, we have the discrete-time transfer function 1/z. We have to simulate this function using the discrete-time step function.

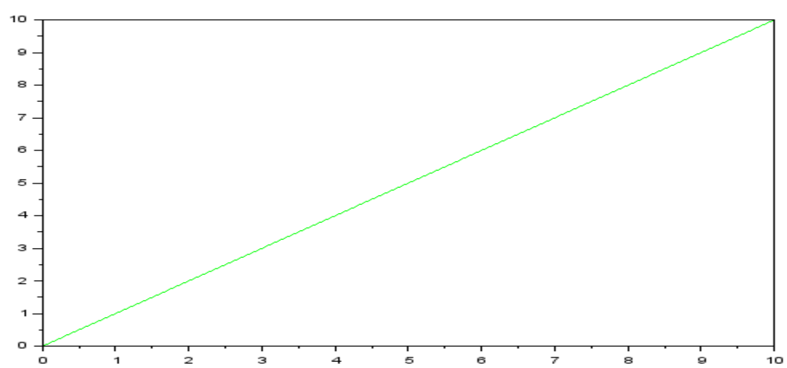


Scilab code for the same:

```
s = poly(0,'s');
z = poly(0,'z');
G = syslin('c', 1/s);
G2 = tf2ss(1/z);
t = 0:0.001:10;
continous = csim('step', t, G);
plot (t, continous, 'r');
u=ones(1,10);
discrete = dsimul(G2,u);
plot (1:10, discrete, 'g');
leg2=legend('Continuous','disrete',1);
```

Part C)

Here we are giving general polynomials as input to the csim. Here it is giving the same output as Part A but with a warning. It assumes a ratio of the polynomial as a continuous-time function



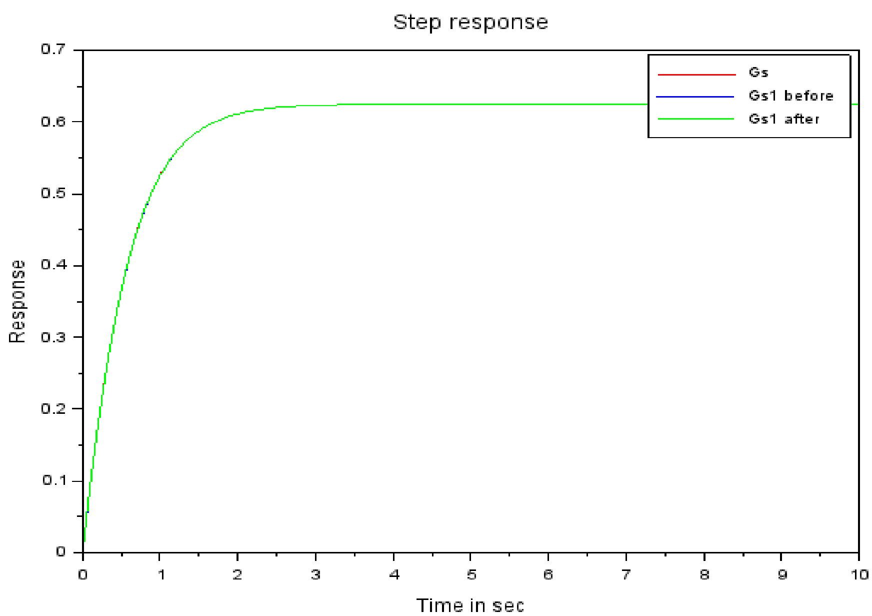
Scilab code for the same:

```
s = poly(0,'s');
Gs = 1/s
t = 0:0.01:10;
cont = csim('step', t,Gs );
plot (t, cont, 'g');
```

Question 5 :

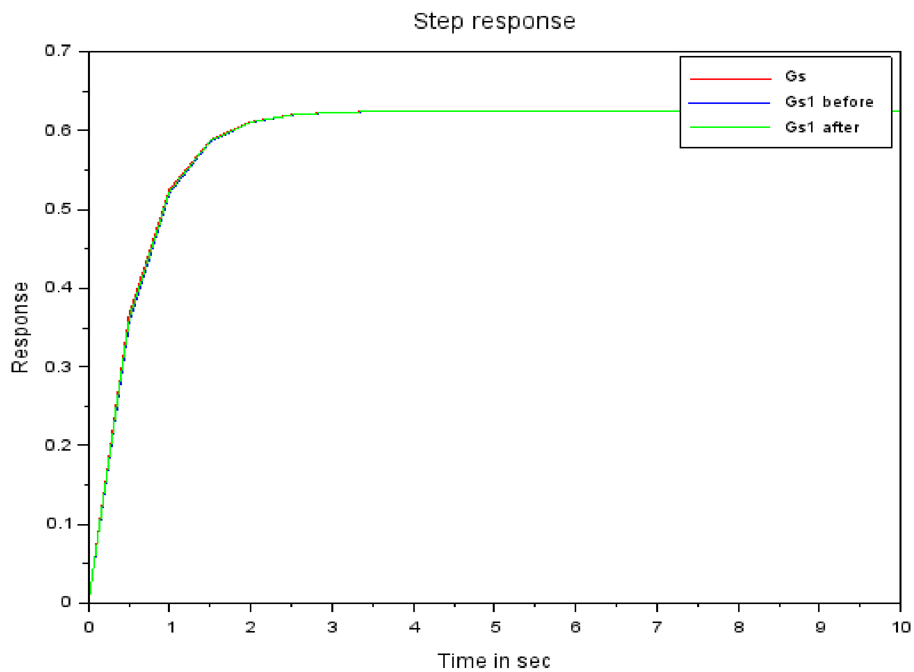
Part A) τ=0.1 sec

Here we are taking the sampling period as 0.1 sec. At this sampling period, all three graphs will be nearly superimposed as sampling time is short, so the error introduced will be low



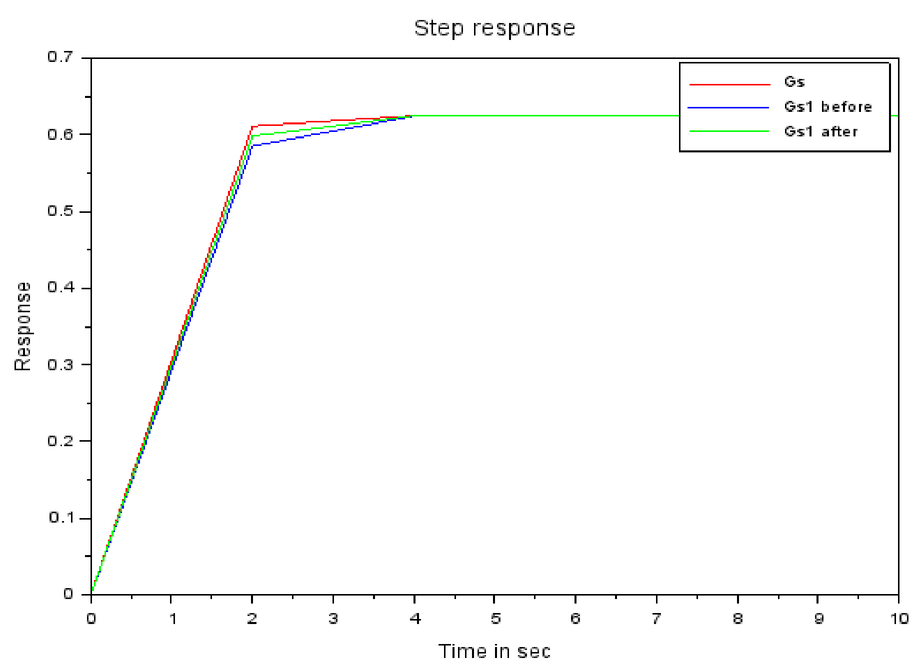
Part B) τ=0.5 sec

Here we are taking the sampling period as 0.5 sec. All three graphs will be moderately superimposed at this sampling period as sampling time is not sufficiently short, so the error introduced will be moderate.



Part C) τ=2 sec

Here we are taking the sampling period as 2 sec. All three graphs will be poorly superimposed at this sampling period as sampling time is considerable, so the error introduced will be high.



Scilab code for the same:

```
Tau=[0.1,0.5,2];
s=%s;
Gs=(s+5)/((s+2)*(s+4));
Gs1=(s+5)/(s+4);
Gs2=1/(s+2);

Gs=syslin('c',Gs);
Gs1=syslin('c',Gs1);
Gs2=syslin('c',Gs2);

time=0:Tau(n):10;

stepres1=csim('step',time,Gs);

stepres2=csim('step',time,Gs1);
stepres3=csim(stepres2,time,Gs2);

stepres2r=csim('step',time,Gs2);
stepres3r=csim(stepres2r,time,Gs1);

plot(time,stepres1,'r');
plot(time,stepres3,'b');
plot(time,stepres3r,'g');

leg1=legend('Gs','Gs1 before','Gs1 after');

xlabel('Time in sec');
ylabel('Response');
xtitle('Step response');
```