Problem 2

11 January 2022 09:59 AM

35/35

Q2) (i) Pseudo Acceleration Response Spectra:

EQ_HW3...


```
%Assignment #3 P2A-Constant-Meu Pseudo Acceleration Response Spectra EPP system with
5% damping
%Central Difference Scheme
clc
fid = fopen('El Centro Ground Motion data.txt') ;  % open the text file
S = textscan(fid,'%s');   % text scan the data
fclose(fid) ;      % close the file
S = S{1} ;
a_g = cellfun(@(x)str2double(x), S);  % convert the cell array to double
% Remove NaN's which were strings earlier
a_g(isnan(a_g))=[];
col = 2;
count = 0;
temp_arr =[];
temp_row = [];
for i = 1:length(a_g)
    if count == col
        temp_arr = [temp_arr;
                    temp_row];
        count = 0;
        temp_row = [];
    end
    temp_row = [temp_row,a_g(i)];
    count = count +1;
end
temp_arr = [temp_arr;
            temp_row];
a_g = temp_arr(:,2:end);
a_g=a_g.*386.09;
clear temp_arr temp_row S;
% Creating Time axis with zero padding of 20 sec
t=zeros(length(a_g),1);
for i=2:length(a_g)+(20/0.02)
    t(i)=t(i-1)+0.02;
end
del_t=0.005;
dt=0.005; % Time step for EPP analysis
% Refning the time axis with dt=0.005
t1=0:0.005:51.180;
% Adding zero padding to the given Earthquake excitation data
a_g=[a_g;zeros((20/0.02),1)]; % appneding the a_g vector with zeros for the next 20
sec.
% interpolating the acceleration values within the refined time range
a_g1=interp1(t,a_g,t1);
meu=[1,2,4,6,8]; % Array containg the Yield Strength reduction factors
Ry=[1:1:500]'; %Trial values of Ry considered for linear interpolation with meu
tn=0.02:0.02:3;
Z=0.05; %Damping ratio
m=1; %Considering unit mass
meu_prime=zeros(length(Ry),1); %Matrix to store the ductility demands for each Tn
against assumed Ry
ur_prime=zeros(length(Ry),1);
u_r=zeros(length(Ry),1); %Matrix to store the residual displacement for each Tn
againts given Ry
PSA=zeros(length(Tn),length(meu));
```

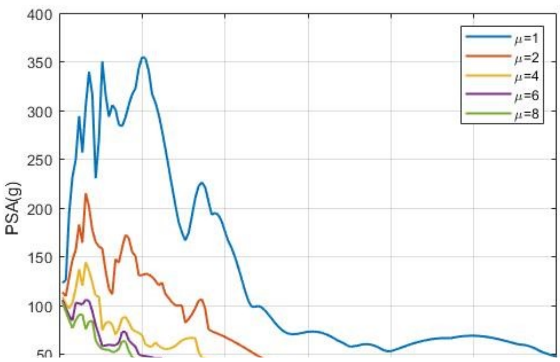


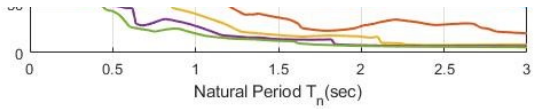
```
PSV=zeros(length(Tn),length(meu));
for j=1:length(meu)
    for x=1:length(tn)
        %Producing System response data for Equivalent Linear Elastic system
        Wn=(2*pi)/tn(x); %Natural Frequency
        k=m*Wn^2; %Linear elastic Stiffness
        Wd=Wn*sqrt(1-Z^2); %Damped Natural Frequency
        %Defining Parameters required A,B,C,D & A1,B1,C1,D1
        A=exp(-Z*Wn*del_t)*((Z/sqrt(1-Z^2))*sin(Wd*del_t)+cos(Wd*del_t));
```

```
        B=exp(-Z*Wn*del_t)*(sin(Wd*del_t)/Wd);
        C=(((2*Z)/(Wn*del_t))+exp(-Z*Wn*del_t)*(((1-2*Z^2)/(Wd*del_t)-(Z/sqrt(1-Z^2)))↙
*sin(Wd*del_t)-(1+((2*Z)/(Wn*del_t)))*cos(Wd*del_t)))/Wn^2;
        D=(1-((2*Z)/(Wn*del_t))+exp(-Z*Wn*del_t)*(((2*Z^2-1)/(Wd*del_t))*sin(Wd*del_t)↙
+((2*Z)/(Wn*del_t))*cos(Wd*del_t)))/Wn^2;
        A1=-exp(-Z*Wn*del_t)*((Wn/sqrt(1-Z^2))*sin(Wd*del_t));
        B1=exp(-Z*Wn*del_t)*(cos(Wd*del_t)-(Z/sqrt(1-Z^2))*sin(Wd*del_t));
        C1=((-1/del_t)+exp(-Z*Wn*del_t)*(((Wn/(sqrt(1-Z^2)))+(Z/(del_t*sqrt(1-Z^2))))↙
*sin(Wd*del_t)+(cos(Wd*del_t)/del_t)))/Wn^2;
        D1=(1-exp(-Z*Wn*del_t)*((Z/sqrt(1-Z^2))*sin(Wd*del_t)+cos(Wd*del_t)))/↙
(Wn^2*del_t);
        u=zeros(length(a_g1),1); %Initialising displacement response vector of the↙
SDOF system
        v=zeros(length(a_g1),1); %Initialising velocity response vector of the SDOF↙
system
        acc=zeros(length(a_g1),1);
        for i=1:length(a_g1)-1
            u(i+1)=A*u(i)+B*v(i)-C*a_g1(i)-D*a_g1(i+1);
            v(i+1)=A1*u(i)+B1*v(i)-C1*a_g1(i)-D1*a_g1(i+1);
            acc(i+1)=-a_g1(i+1)-2*Z*Wn*v(i+1)-Wn^2*u(i+1);
        end
        a_t=a_g1+acc;
        %plot(t(1:1560),u);
        umax=max(abs(u));
        f_0=k*umax; %Max. Force for system to remain Linear Elastic
        if meu(j)==1 %For Linear Elastic system with Meu=1,
            PSA(x,j)=Wn^2*umax;
            PSV(x,j)=Wn*umax;
        else
            Ry=[1:1:500]';
            meu_prime=zeros(length(Ry),1);
            ur_prime=zeros(length(Ry),1);
            for ct=1:length(Ry)
            % Performing Inelastic Response Analysis for EPP system
            fy=f_0/Ry(ct);
            u_epp=zeros(length(a_g1),1);
            v_epp=zeros(length(a_g1),1);
            a_epp=zeros(length(a_g1),1);
            fs_epp=zeros(length(a_g1),1);
            [meu_prime(ct,1),ur_prime(ct,1)]=ElastoPlastic(m,Z,Wn,dt,a_g1,k,fy,u_epp,↙
v_epp,a_epp,fs_epp);
            end
            meu_diff=1;
            while meu_diff>0.01
                row=find(meu(j)>meu_prime,1,'last');
                slope=(log(Ry(row+1,1))-log(Ry(row,1)))/(log(meu_prime(row+1,1))-log↙
```



```
(meu_prime(row,1)));
                Ry(row+1,1)=exp((slope*(log(meu(j))-log(meu_prime(row,1)))+log(Ry(row,↙
1))));
                fy=f_0/Ry(row+1,1);
                [meu_prime(row+1,1),ur_prime(row+1,1)]=ElastoPlastic(m,Z,Wn,dt,a_g1,k,↙
fy,u_epp,v_epp,a_epp,fs_epp);
                meu_diff=abs(meu_prime(row+1,1)-meu(j));
            end
            uy=(fy/f_0)*umax;
            PSA(x,j)=Wn^2*uy;
            PSV(x,j)=Wn*uy;
        end
    end
end
plot(tn,PSA,'linewidth',1.5)
xlabel('Natural Period T_{n}(sec)');
ylabel('PSA(g)');
legend('\mu=1','\mu=2','\mu=4','\mu=6','\mu=8');
grid on
```





For μ=1, it basically represents the Linear Elastic case and thus coincides with what we obtained previously in assignment 1. With increase in the ductility demand, the PSA response spectrum goes on decreasing for a given natural period.

Q2) (ii) PSA, PSV, SD response spectra in Tripartite Plot

EQ_HW3...


```
%Assignment #3 P2B-Constant-Meu Response Spectra EPP system with 5% damping
%Tripartite Plot
%Central Difference Scheme
clc
fid = fopen('El Centro Ground Motion data.txt') ;  % open the text file
S = textscan(fid,'%s');   % text scan the data
fclose(fid) ;      % close the file
S = S{1} ;
a_g = cellfun(@(x)str2double(x), S);  % convert the cell array to double
% Remove NaN's which were strings earlier
a_g(isnan(a_g))=[];
col = 2;
count = 0;
temp_arr =[];
temp_row = [];
for i = 1:length(a_g)
    if count == col
        temp_arr = [temp_arr;
                    temp_row];
        count = 0;
        temp_row = [];
    end
    temp_row = [temp_row,a_g(i)];
    count = count +1;
end
temp_arr = [temp_arr;
            temp_row];
a_g = temp_arr(:,2:end);
a_g=a_g.*386.09;
clear temp_arr temp_row S;
% Creating Time axis with zero padding of 20 sec
t=zeros(length(a_g),1);
for i=2:length(a_g)+(20/0.02)
    t(i)=t(i-1)+0.02;
end
del_t=0.005;
dt=0.005; % Time step for EPP analysis
% Refning the time axis with dt=0.005
t1=0:0.005:51.180;
% Adding zero padding to the given Earthquake excitation data
a_g=[a_g;zeros((20/0.02),1)]; % appneding the a_g vector with zeros for the next 20↙
sec.
% interpolating the acceleration values within the refined time range
a_g1=interp1(t,a_g,t1);
meu=[1,2,4,6,8]; % Array containg the Yield Strength reduction factors
Ry=[1:1:500]'; %Trial values of Ry considered for linear interpolation with meu
tn1=[0.02:0.01:0.1]'; % 1st segment of Tn axis
tn2=[0.12:0.02:1]'; %2nd segment of Tn axis
tn3=[1.05:0.05:50]'; %3rd segment of Tn axis
tn=zeros(length(tn1)+length(tn2)+length(tn3),1); %Natural Period Range
tn(1:length(tn1),1)=tn1;
tn(length(tn1)+1:length(tn1)+length(tn2),1)=tn2;
tn(length(tn1)+length(tn2)+1:length(tn1)+length(tn2)+length(tn3),1)=tn3;
Z=0.05; %Damping ratio
m=1; %Considering unit mass
```



```
meu_prime=zeros(length(Ry),1); %Matrix to store the ductility demands for each Tn↙
against assumed Ry
ur_prime=zeros(length(Ry),1);
u_r=zeros(length(Ry),1); %Matrix to store the residual displacement for each Tn↙
againts given Ry
```

```
PSA=zeros(length(Tn),length(meu));
PSV=zeros(length(Tn),length(meu));
for j=1:length(meu)
    for x=1:length(tn)
        %Producing System response data for Equivalent Linear Elastic system
        Wn=(2*pi)/tn(x); %Natural Frequency
        k=m*Wn^2; %Linear elastic Stiffness
        Wd=Wn*sqrt(1-Z^2); %Damped Natural Frequency
        %Defining Parameters required A,B,C,D & A1,B1,C1,D1
        A=exp(-Z*Wn*del_t)*((Z/sqrt(1-Z^2))*sin(Wd*del_t)+cos(Wd*del_t));
        B=exp(-Z*Wn*del_t)*(sin(Wd*del_t)/Wd);
        C=(((2*Z)/(Wn*del_t))+exp(-Z*Wn*del_t)*(((1-2*Z^2)/(Wd*del_t)-(Z/sqrt(1-Z^2)))
*sin(Wd*del_t)-(1+((2*Z)/(Wn*del_t)))*cos(Wd*del_t)))/Wn^2;
        D=(1-((2*Z)/(Wn*del_t))+exp(-Z*Wn*del_t)*(((2*Z^2-1)/(Wd*del_t))*sin(Wd*del_t)
+((2*Z)/(Wn*del_t))*cos(Wd*del_t)))/Wn^2;
        A1=-exp(-Z*Wn*del_t)*((Wn/sqrt(1-Z^2))*sin(Wd*del_t));
        B1=exp(-Z*Wn*del_t)*(cos(Wd*del_t)-(Z/sqrt(1-Z^2))*sin(Wd*del_t));
        C1=((-1/del_t)+exp(-Z*Wn*del_t)*(((Wn/(sqrt(1-Z^2)))+(Z/(del_t*sqrt(1-Z^2))))
*sin(Wd*del_t)+(cos(Wd*del_t)/del_t)))/Wn^2;
        D1=(1-exp(-Z*Wn*del_t)*((Z/sqrt(1-Z^2))*sin(Wd*del_t)+cos(Wd*del_t)))/
(Wn^2*del_t);
        u=zeros(length(a_g1),1); %Initialising displacement response vector of the
SDOF system
        v=zeros(length(a_g1),1); %Initialising velocity response vector of the SDOF
system
        acc=zeros(length(a_g1),1);
        for i=1:length(a_g1)-1
            u(i+1)=A*u(i)+B*v(i)-C*a_g1(i)-D*a_g1(i+1);
            v(i+1)=A1*u(i)+B1*v(i)-C1*a_g1(i)-D1*a_g1(i+1);
            acc(i+1)=-a_g1(i+1)-2*Z*Wn*v(i+1)-Wn^2*u(i+1);
        end
        a_t=a_g1+acc;
        %plot(t(1:1560),u);
        umax=max(abs(u));
        f_0=k*umax; %Max. Force for system to remain Linear Elastic
        if meu(j)==1 %For Linear Elastic system with Meu=1,
            PSA(x,j)=Wn^2*umax;
            PSV(x,j)=Wn*umax;
        else
            Ry=[1:1:500]';
            meu_prime=zeros(length(Ry),1);
            ur_prime=zeros(length(Ry),1);
            for ct=1:length(Ry)
            % Performing Inelastic Response Analysis for EPP system
            fy=f_0/Ry(ct);
            u_epp=zeros(length(a_g1),1);
            v_epp=zeros(length(a_g1),1);
            a_epp=zeros(length(a_g1),1);
            fs_epp=zeros(length(a_g1),1);
            [meu_prime(ct,1),ur_prime(ct,1)]=ElastoPlastic(m,Z,Wn,dt,a_g1,k,fy,u_epp,
```

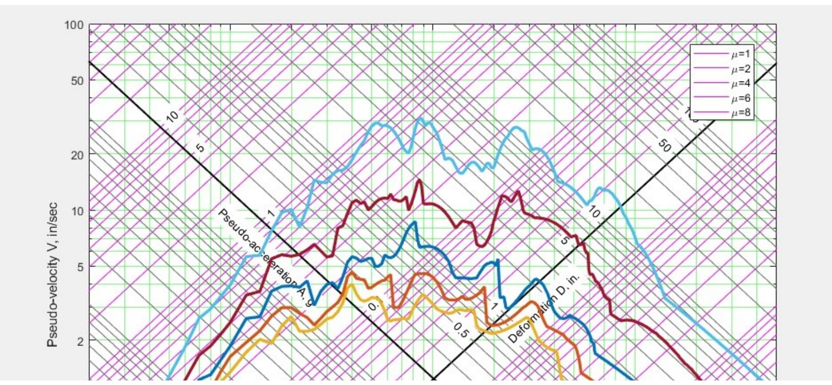


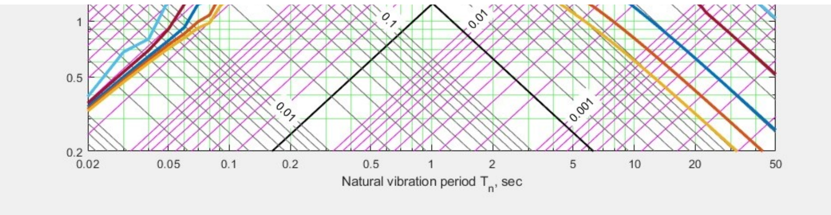
```
v_epp,a_epp,fs_epp);
            end
            meu_diff=1;
            while meu_diff>0.01
                row=find(meu(j)>meu_prime,1,'last');
                slope=(log(Ry(row+1,1))-log(Ry(row,1)))/(log(meu_prime(row+1,1))-log
(meu_prime(row,1)));
                Ry(row+1,1)=exp((slope*(log(meu(j))-log(meu_prime(row,1)))+log(Ry(row,
1))));
                fy=f_0/Ry(row+1,1);
                [meu_prime(row+1,1),ur_prime(row+1,1)]=ElastoPlastic(m,Z,Wn,dt,a_g1,k,
fy,u_epp,v_epp,a_epp,fs_epp);
                meu_diff=abs(meu_prime(row+1,1)-meu(j));
            end
            uy=(fy/f_0)*umax;
            PSA(x,j)=Wn^2*uy;
            PSV(x,j)=Wn*uy;
        end
    end
end
Tnmin = 0.02; Tnmax = 50; TnTicks = [0.02 0.05 0.1 0.2 0.5 1 2 5 10 20 50];
Vmin = 0.2; Vmax = 100; VTicks = [0.2 0.5 1 2 5 10 20 50 100];
Amin = 0.0001; Amax = 100; ATicks = [0.001 0.01 0.1 1 5 10];
Dmin = 0.0001; Dmax = 1000; DTicks = [0.01 0.1 0.5 1 5 10 50 100];
g = 386; % in/sec^2;
AgridCol = [1 0 1]; DgridCol = [0.4 0.4 0.4];
AgridW = 0.5; DgridW = 0.5;
D_axis_A = 0.02; % Value of A grid line that represents D axis
A_axis_D = 0.2; % Value of D grid line that represents A axis

tickoffset = 1.15;

%% Constant A grid lines and D-axis
```

```
fp =[50 100 1000 750];
figure('position',fp)
Agrids_A = Amin;
for i = 1:log10(Amax/Amin)
    Agrids_A = [Agrids_A 2*max(Agrids_A):max(Agrids_A):10*max(Agrids_A)];
end
for j = 1:length(Agrids_A)
    Agrids_Tn(j,1) = 2*pi*Vmin/(Agrids_A(j)*g);
    Agrids_Tn(j,2) = 2*pi*Vmax/(Agrids_A(j)*g);
    if Agrids_A(j)== D_axis_A
        loglog([Agrids_Tn(j,1) Agrids_Tn(j,2)],[Vmin Vmax],'-k','Linewidth',1.5);
    else
        loglog([Agrids_Tn(j,1) Agrids_Tn(j,2)],[Vmin Vmax],'linestyle','-','color',↙
AgridCol,'Linewidth',AgridW); grid on; hold on
    end
end
xlabel('Natural vibration period T_n, sec');
ylabel('Pseudo-velocity V, in/sec')
axis([Tnmin Tnmax Vmin Vmax])
set(gca,'XTick',TnTicks,'YTick',↙
VTicks,'MinorGridLineStyle','-','GridLineStyle','-',...
    'MinorGridColor','g','GridColor','g','MinorGridAlpha',0.5,'GridAlpha',0.5);
```



```
%% Constant D grid lines and A-axis
Dgrids_D = Dmin;
for i = 1:log10(Dmax/Dmin)
    Dgrids_D = [Dgrids_D 2*max(Dgrids_D):max(Dgrids_D):10*max(Dgrids_D)];
end
for j = 1:length(Dgrids_D)
    Dgrids_Tn(j,1) = 2*pi*Dgrids_D(j)/Vmin;
    Dgrids_Tn(j,2) = 2*pi*Dgrids_D(j)/Vmax;
    if Dgrids_D(j)==A_axis_D
        loglog([Dgrids_Tn(j,1) Dgrids_Tn(j,2)],[Vmin Vmax],'-k','Linewidth',1.5);
    else
        loglog([Dgrids_Tn(j,1) Dgrids_Tn(j,2)],[Vmin Vmax],'linestyle','-','color',↙
DgridCol,'Linewidth',DgridW); hold on
    end
end

%% Label the A and D axes
hA = text(0.09,10,'Pseudo-acceleration A, g');
set(hA,'Rotation',-45,'BackgroundColor','white')

hD = text(2.4,2,'Deformation D, in.');
set(hD,'Rotation',45,'BackgroundColor','white')

%% Add tick labels to A and D axes
for i = 1:length(ATicks)
    Tn = 2*pi*sqrt(tickoffset*A_axis_D/(ATicks(i)*g));
    V = Tn*ATicks(i)*g/(2*pi);
    ht = text(Tn,V,num2str(ATicks(i)));
    set(ht,'HorizontalAlignment','left','Rotation',45,'BackgroundColor','white')
end

for i = 1:length(DTicks)
    Tn = 2*pi*sqrt(DTicks(i)/(tickoffset*D_axis_A*g));
    V = 2*pi*DTicks(i)/Tn;
    ht = text(Tn,V,num2str(DTicks(i)));
    set(ht,'HorizontalAlignment','right','Rotation',-45,'BackgroundColor','white')
end
hold on
plot(tn,PSV,'linewidth',2.5)
```





In the acceleration sensitive region, for very small natural periods, we observe that all the responses converge close to the linear elastic case (I.e. for μ=1) and the PSA=PGA. This region plays important role in the Spectral deformation values and very less to the earthquake design forces.
In the Displacement sensitive region, for higher values of natural periods, The ductility demand nearly equals the yield strength reduction factor Ry and the response spectra has SD=PGD. This region is important for the earthquake design forces but not so much on the yield deformation.