Problem 3

11 January 2022 09:59 AM

34/35

Q3) (i) Time Variation of Energy for Linear Elastic System

EQ_HW3...

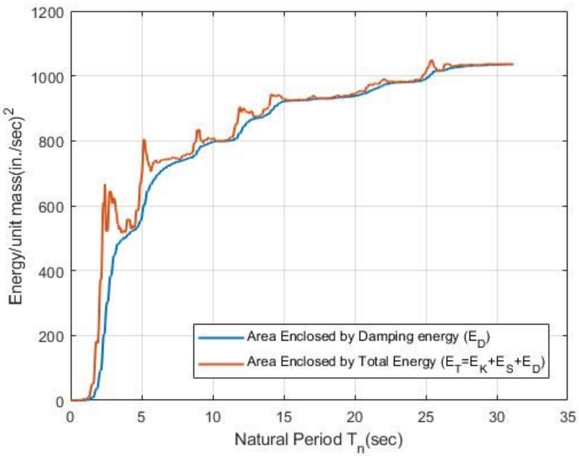
12/2/22 7:06 PM EQ_HW3_P3a.m 1 of 2

```
%Assignment #3 Problem 3a- Time History of Energy Dissipation for Linear Elastic
system with Tn=0.5s & Damping=5%
%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)
    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:t(end);
% 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);
Tn=0.5; %Natural Period of the system
Z=0.05; %Damping ratio
m=1; %Considering unit mass
Wn=(2*pi)/Tn; %Natural Frequency
k=m*Wn^2; %Linear elastic Stiffness
c=2*Z*Wn*m; %Damping coefficient
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
```

12/2/22 7:06 PM EQ_HW3_P3a.m 2 of 2

```
(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
fs=zeros(length(a_g1),1); %Resistive force
Es=zeros(length(a_g1),1); %Recoverable strain energy
Et=zeros(length(a_g1),1); %Total Resistive energy
Ed=zeros(length(a_g1),1); %Energy dissipated due to viscuous damping
Ek=zeros(length(a_g1),1); %Kinetic energy of the mass relative to the ground
for i=1:length(u)-1
    du=u(i+1)-u(i);
    dv=v(i+1)-v(i);
    da=acc(i+1)-acc(i);
    dt=t1(i+1)-t1(i);
    fs(i+1)=k*du+fs(i);
    Es(i+1)=(fs(i+1)^2)/(2*k);
    Ek(i+1)=Ek(i)+0.5*m*(acc(i+1)*v(i+1)+acc(i)*v(i))*dt;
    Ed(i+1)=Ed(i)+0.5*c*(v(i+1)^2+v(i)^2)*dt;
    Et(i+1)=Ek(i+1)+Ed(i+1)+Es(i+1);
end
plot(t1,Ed)
% comet(u,fs)
grid on
hold on
plot(t1,Et)
```



15

Q3) (ii) For EPP system

EQ_HW3...

12/2/22 7:07 PM EQ_HW3_P3b.m 1 of 3

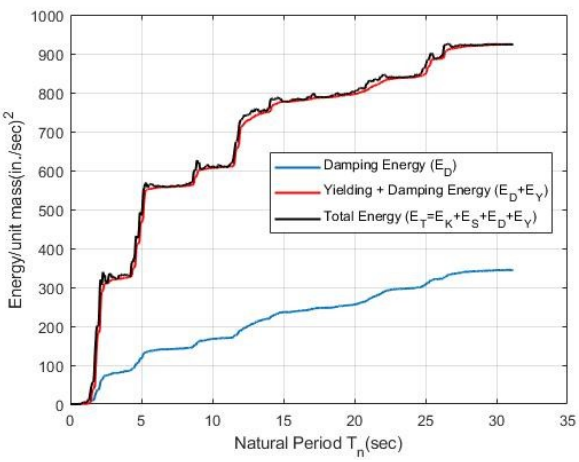
```
%Assignment #3 P3b-Time History of Energy Disssipation for EPP system with
%Tn=0.5, Damping=5% and Ry=4
%Central Difference Scheme
clc
fid = fopen('El Centro Ground Motion data.txt') ; % open the text file
S = textscan(fid,'%s');   % text scan the data
fclose(fid) ;      % close the file
S = S{1} ;
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)
    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:t(end);
% 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);
Tn=0.5; %Natural Period of the system
Z=0.05; %Damping ratio
m=1; %Considering unit mass
Wn=(2*pi)/Tn; %Natural Frequency
k=m*Wn^2; %Linear elastic Stiffness
c=2*Z*m*Wn; %Damping Coefficient
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;
umax=abs(max(u));
f_0=k*umax; %Max. Force for system to remain Linear Elastic
% Performing Inelastic Response Analysis
u_epp=zeros(length(a_g1),1);
v_epp=zeros(length(a_g1),1);
a_epp=zeros(length(a_g1),1);
%Initial calculations:
fs=zeros(length(a_g1),1);
fs(1)=k*u_epp(1);
Ry=4; %Yield Strength reduction factor
fy=f_0/Ry; %yield strength of the system
a_epp(1)=(-m*a_g1(1)-2*m*Z*Wn*v_epp(1)-fs(1))/m; %Initial acceleration
u_0=u_epp(1)-dt*v_epp(1)+0.5*dt^2*a_epp(1);
k_hat=(m/dt^2)+((m*Z*Wn)/dt); %effective stiffness
a=(m/dt^2)-((m*Z*Wn)/dt); %Integration parameter
b=(2*m)/dt^2; %Integration parameter
p_hat=0;du=0;fst=0;
for i=1:length(a_g1)-1
    if i==1
        p_hat=-m*a_g1(1)-a*u_0-fs(1)+b*u_epp(1);
        u_epp(2)=p_hat/k_hat;
        v_epp(1)=(u_epp(2)-u_0)/(2*dt);
        a_epp(1)=(u_epp(2)-2*u_epp(1)+u_0)/dt^2;
    else
        p_hat=-m*a_g1(i)-a*u_epp(i-1)-fs(i)+b*u_epp(i);
        u_epp(i+1)=p_hat/k_hat;
        v_epp(i)=(u_epp(i+1)-u_epp(i-1))/(2*dt);
        a_epp(i)=(u_epp(i+1)-2*u_epp(i)+u_epp(i-1))/dt^2;
    end
    du=u_epp(i+1)-u_epp(i);
    fst=fs(i)+k*du;
    if abs(fst)>fy
        fs(i+1)=sign(fst)*fy;
    else
        fs(i+1)=fst;
    end
end
Es=zeros(length(a_g1),1); %Recoverable strain energy
```

```
Et=zeros(length(a_g1),1); %Total energy of the system
Ed=zeros(length(a_g1),1); %Energy dissipated due to viscuous damping
Ek=zeros(length(a_g1),1); %Kinetic energy of the mass relative to the ground
Er=zeros(length(a_g1),1); %Total Resistive energy i.e Es+Ey
Ey=zeros(length(a_g1),1); %Energy lost due to yielding
for i=1:length(u_epp)-1
    du=u_epp(i+1)-u_epp(i);
    dv=v_epp(i+1)-v_epp(i);
    da=a_epp(i+1)-a_epp(i);
    dt=t1(i+1)-t1(i);
    Er(i+1)=Er(i)+0.5*(fs(i+1)+fs(i))*du; %Total resistive energy
    Es(i+1)=(fs(i+1)^2)/(2*k); %Recoverable Strain energy
    Ey(i+1)=Er(i+1)-Es(i+1); %Energy dissipated by yielding
    Ek(i+1)=Ek(i)+0.5*m*(a_epp(i+1)*v_epp(i+1)+a_epp(i)*v_epp(i))*dt; %Kinetic Energy
    Ed(i+1)=Ed(i)+0.5*c*(v_epp(i+1)^2+v_epp(i)^2)*dt; %Energy dissipated due to↙
viscuous damping
    Et(i+1)=Ek(i+1)+Ed(i+1)+Es(i+1)+Ey(i+1);
end
plot(t1,Ed)
% comet(u,fs)
grid on
hold on
plot(t1,Ey+Ed,'r')
hold on
plot(t1,Et,'k')
```



15

For Linear Elastic system, most of the energy dissipated is in the form of the damping energy, whereas in case of the Elastic Perfectly Plastic system, the damping plays a very little role is energy dissipation as is evident from the graph, in this case the predominant dissipator of energy comes in the form of yielding. The EPP system undergoes permanent deformation and dissipates much of the earthquake energy input into it. 4