POLITECNICO
MILANO 1863

Politecnico di Milano, Master of Science in Civil Engineering for Risk Mitigation

Polo Regionale di Lecco, Course Structural Dynamics - A.A. 2022/2023

Dynamics of Structures – Individual Project

Modal Analysis

Direct Integration

Professor
Giacomo Boffi

Student
Jasem Avaz Nasab
(996310)

September 2023

1. Modal Analysis

Introducing $q(t)$ and Ψ as the vector of Modal Coordinates and the Eigenvector matrix, respectively, Equation of Motion (EoM) can be expressed (in starred form) for an undamped system as follow,

$$M^*\ddot{q} + K^*q = P^*(t)$$

where $M^* = \Psi^T M\Psi$ and $K^* = \Psi^T K\Psi$, and Load vector is $P^*(t) = \Psi^T p(t)$. Here, $p(t) = rp_0$ where r is the vector of load shape (composed of " 1 " for loaded degrees of freedom, and "0" for others). p_0 is equal to unit load which is applied to the horizontal degree of freedom:

Condition 1: a single unit load $p_0 = +1$ N, acting on point B, applied instantaneously at time $t = 0$,
Condition 2: two-unit loads, $p_1 = +1$ N acting on point B and $p_2 = -1$ N acting on point A, applied instantaneously at time $t = 0$.

The modal participation factor for mode (i) is calculated according to:

```
[nn,nMod]=size(evecs);
M_star=diag(evecs'*dM*evecs).*eye(nMod);

GammaVec_cond1=zeros(nMod,1);
GammaVec_cond2=zeros(nMod,1);

for i=1:nMod
    GammaVec_cond1(i)=(evecs(:,i)'*dM*r_cond1)/M_star(i,i);
    GammaVec_cond2(i)=(evecs(:,i)'*dM*r_cond2)/M_star(i,i);
end

GammaVec_cond1=GammaVec_cond1'*eye(nMod);
GammaVec_cond2=GammaVec_cond2'*eye(nMod);
```

In here evecs is the matrix of modal shapes, dM is the mass matrix, r_cond1 and r_cond2 are the load vectors assigning unit load to the corresponding degrees of freedom for conditions 1 and 2. Then the modal displacement is calculating as a function of time as:

$$q(t) = Gamma_i(1 - Cos(\omega_i\, t))$$

Where $Gamma_i$ is the modal participation factor and ω_i is natural frequency of mode (i) .

According to the property of modal coordinates $q(t)$, and as far as the displacement is concerned, the displacement of j^{th} degree of freedom (as total response) can be defined as,

$$S_j = \sum_{i=1}^{N} \Psi_{ji} q_i(t)$$

in which each term $\Psi_{ji} q_i(t)$ is the modal response of j^{th} degree of freedom related to i^{th} mode.

As the final step the static correction has been calculated by imposing the modal load into the linear system solving procedure for only mode 5 as follows:

```
i=5;
 R_cond1 =GammaVec_cond1(i) * dM * evecs (:,i);
 R_cond2 =GammaVec_cond2(i) * dM * evecs (:,i);

 [du_cond1,~, ~, ~]=syssol(dK,dM,R_cond1,nUu,nUs,dUs,nDofTot);
[du_cond2,~,~, ~]=syssol(dK,dM,R_cond2,nUu,nUs,dUs,nDofTot);
```

Which the results in terms of modal displacements can be used for static correction purposes.

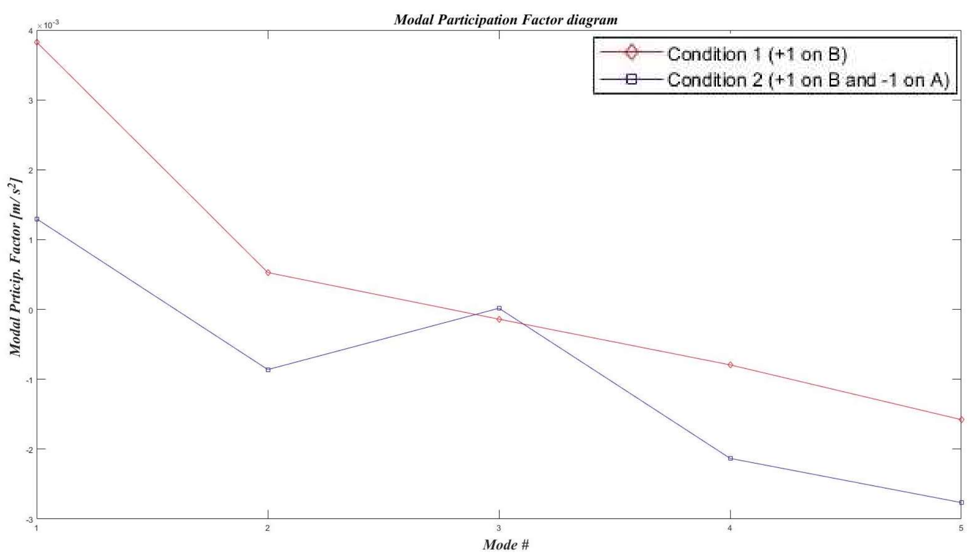
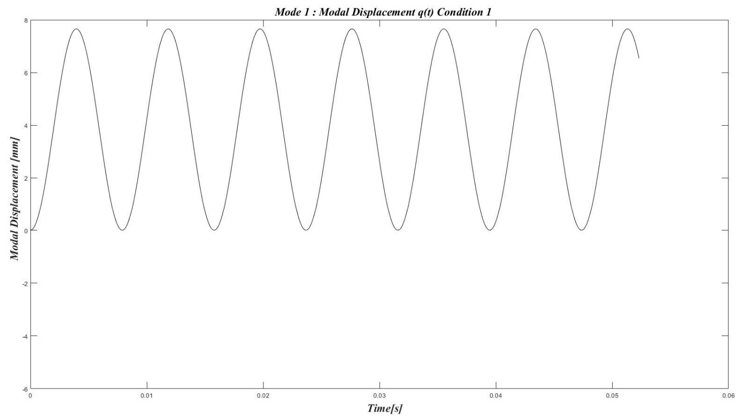


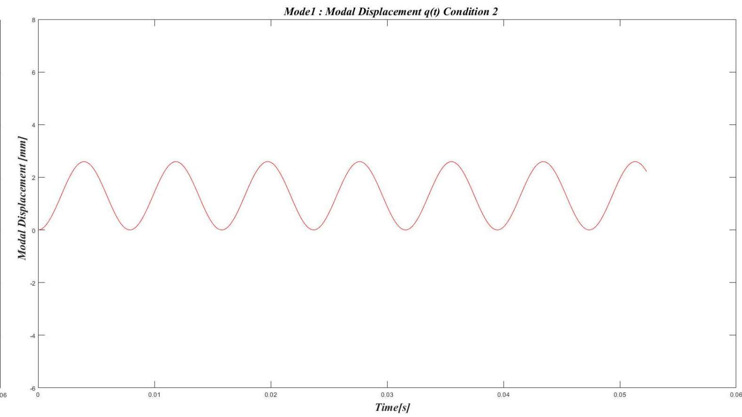
Table1.1: Modal participation values of 5 modes for 2 conditions

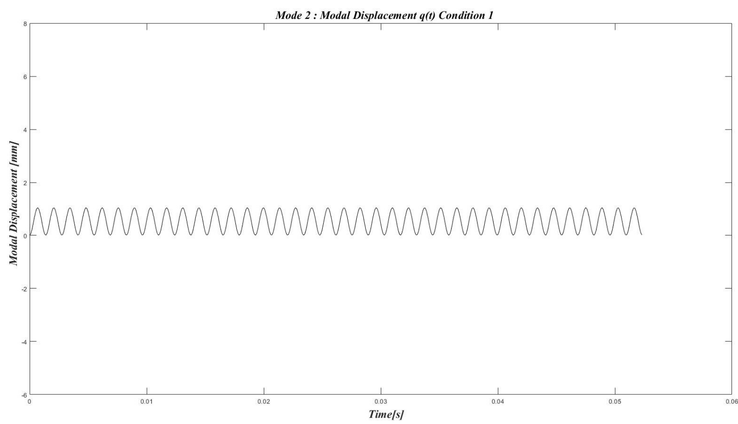
	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5
Condition1	3.8286e-3	5.2821e-04	-1.3936e-04	-7.9381e-04	-1.5778e-03
Condition2	1.2971e-03	-8.6104e-04	1.7335e-05	-2.1323e-03	-2.7652e-03

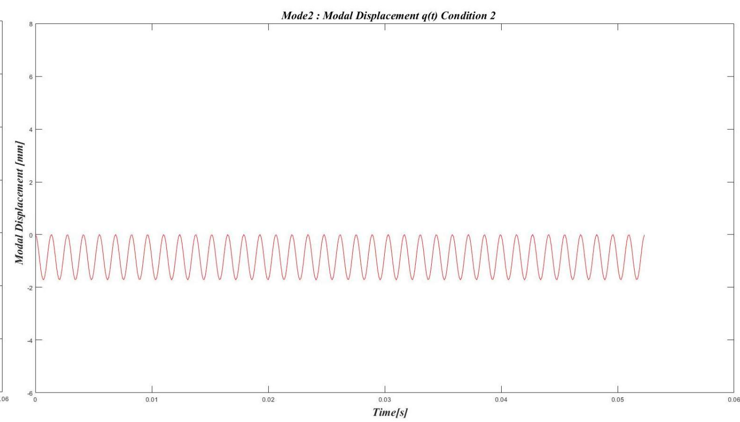
Condition 1

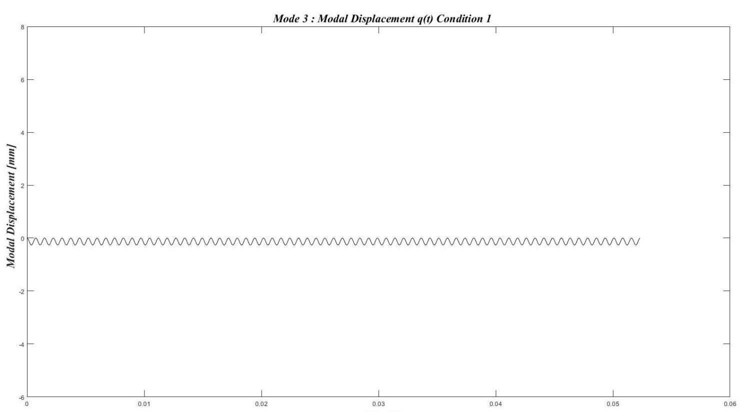
Condition 2

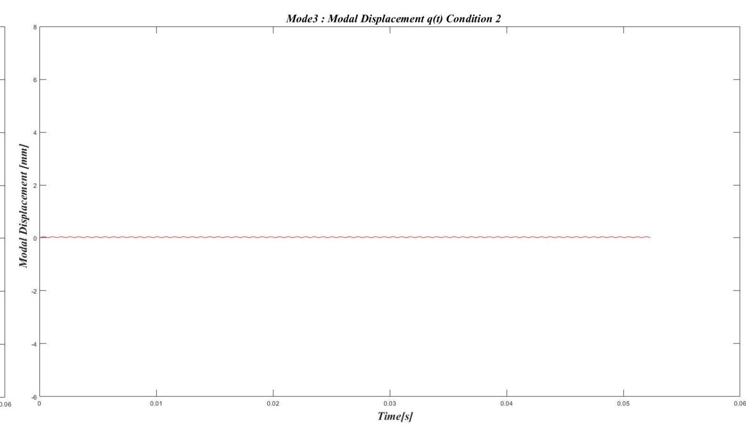


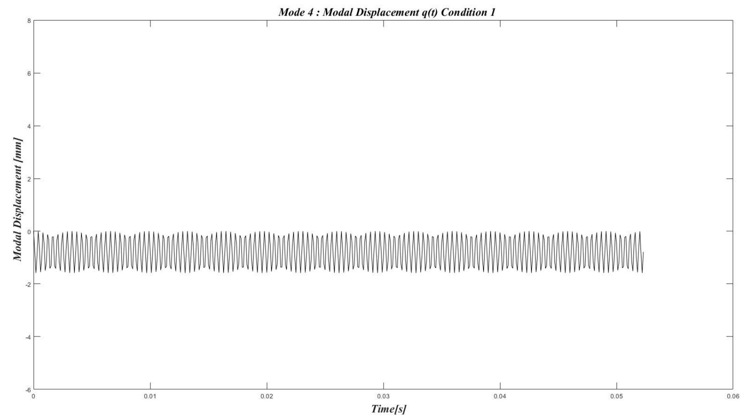


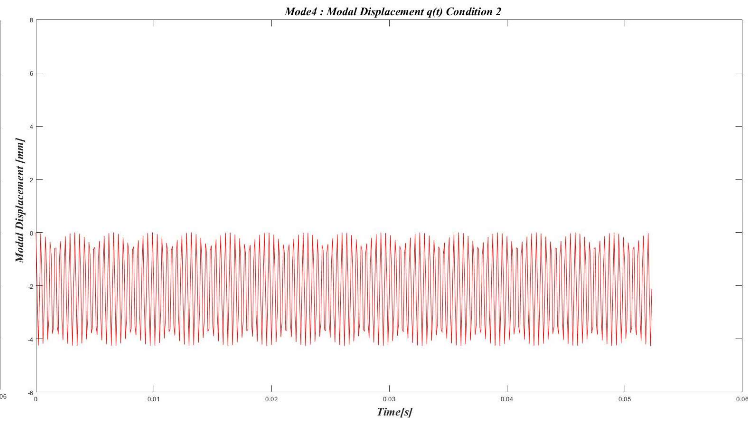


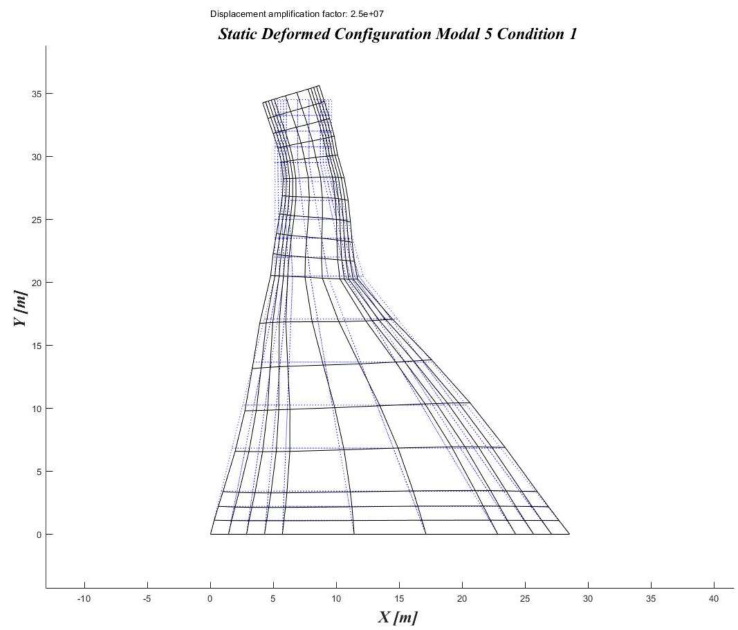


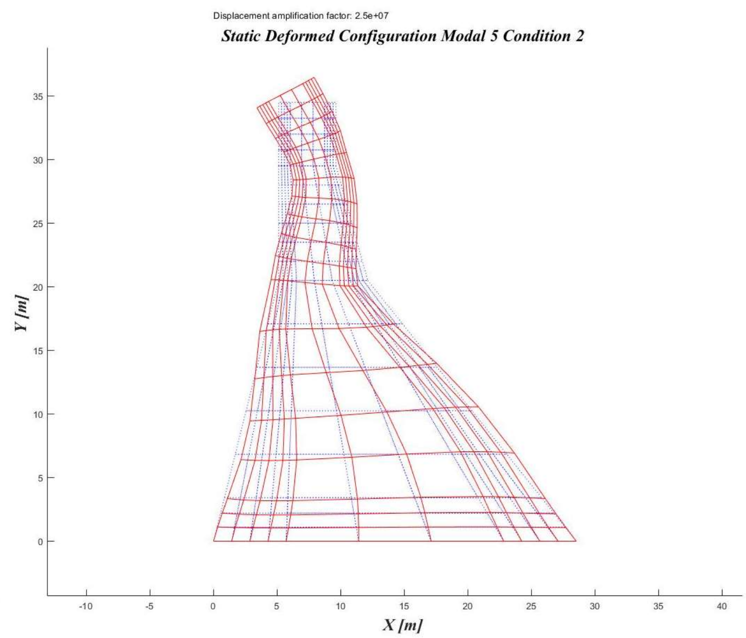












2. Direct Integration

2.1. Calculation of Damping Matrix C

Defining the Global Damping Matrix in terms of understanding coefficients $\boldsymbol{\gamma_b}$:

$$\boldsymbol{C} = \sum_{\boldsymbol{b}} \boldsymbol{\gamma_b M (M^{-1} K)^b} \; \boldsymbol{b = 0, 1, \ldots, N-1}$$

Modal damping C_i for i^{th} mode can be introduced as

$$C_i = \Psi_i^T C \Psi_i = \sum_b \gamma_b \omega_i^{2b} = 2\xi_i \omega_i$$

Assuming a same equivalent damping ratio ξ only for two mode k and r, and assuming $\gamma_0 = \alpha$ and $\gamma_1 = \beta$, the coefficients can be obtained through solving a linear system, such as:

$$2\xi\begin{bmatrix}\omega_k\\ \omega_r\end{bmatrix}=\begin{bmatrix}1 & \omega_k^2\\ 1 & \omega_r^2\end{bmatrix}\begin{bmatrix}\alpha\\ \beta\end{bmatrix}$$

Then, Global Damping Matrix will be $C=\alpha M+\beta K$. For example, in this project, considering $\xi=3\%$ for mode 1 and 3 will result in $\alpha=1.2943$ and $\beta=0.0005$

2.2. Response of Structure using Numerical Integration: Constant acceleration Approach

- Calculation of structural matrices K, M and C

- Calculation of Effective Loading considering the interested excitation at each time t_k, in case of EarthQuake

$$P_{\text{eff}}=-M\,\mathrm{e}\,\ddot{u}_g(t_k)\equiv p_k$$

where $e_{N\times 1}$ is a vector composed of " 1 " (when dof is loaded) and " 0 " (for others), $\ddot{u}_g(t_k)$ is the ground acceleration applied to the structure at time t_k, and p_k is the load vector applied to the dofs at time t_k.

- Considering initial conditions x_0 and $\dot{x}_0$.

- Considering time step $h(=t_{k+1}-t_k)$, with respected to the time step of EQ record, the constant matrices related to the approach can be obtained as

$$A=2C+\frac{4}{h}M\,,\;B=2M\,,\;K^{+}=\frac{2}{h}C+\frac{4}{h^2}M$$

- For each time t_k, stating from $k=0$, calculation of
- Initial acceleration of structure: $\ddot{x}_k=M^{-1}(p_k-C\dot{x}_k-Kx_k)$
- Increment of effective load: $\Delta\hat{p}_k=p_{k+1}-p_k+A\dot{x}_k+B\ddot{x}_0\mathrm{f}$
- Incremental stiffness: $\widehat{K}_k=\text{Tangent Stiffness } K_k+K^{+}$
- Increment of displacement: $\Delta x_k=\widehat{K}_k^{-1}\Delta\hat{p}_k$
- State vectors: $x_{k+1}=x_k+\Delta x_k$ and $\dot{x}_{k+1}=2\frac{\Delta x_k}{h}-\dot{x}_k$

For obtaining overturning moment of the base of the dam with respect to E, the stiffness matrix of each element at the base of the dam is extracted. After extracting the vector of displacement of each element, this vector is multiplied by the corresponding stiffness matrix to derive the reaction forces at the nodes on the ground level constrained to the ground.

the vertical components of these reaction forces are multiplied by the corresponding arms with respect to node E. and then the resultants are summed. As follows:

```
DOFvert=[2:2:22];
armDOFVert=dXY(12,1) - dXY(1:11,1);

dR=zeros(48,1);
for i=1:length(GAinteg)
```

```
    for j=1:11
       dR(nInc(j,5:end) , 1) = Kmats(:,:,j) * x(    nInc(j,5:end)    ,    i
)   + dR(nInc(j,5:end) , 1) ;
    end

    BendM(i)=dR(DOFvert)'*armDOFVert;
end
```

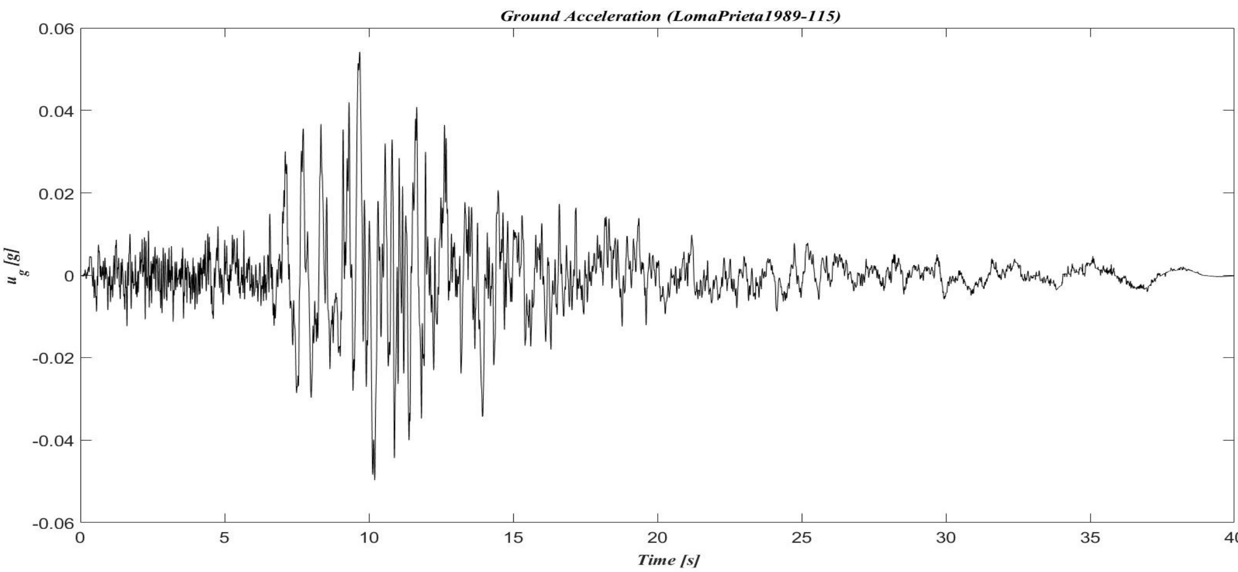


Fig.2.1 Input Ground Acceleration for (LomaPrieta1989-115)

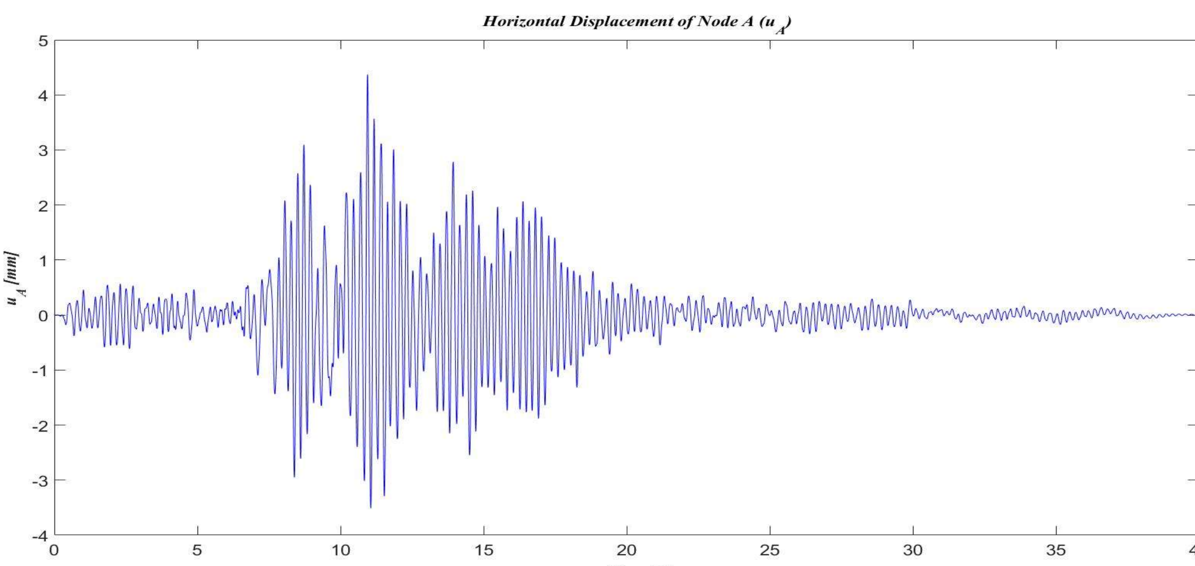


Fig 2.2. Horizontal displacement of point A

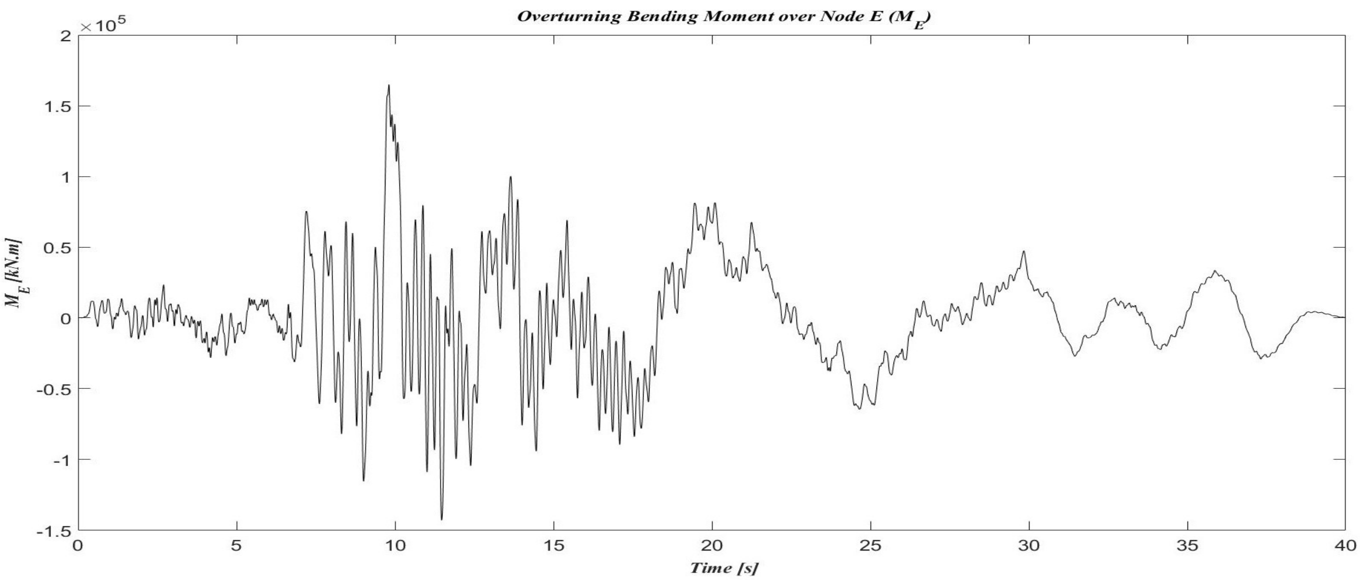


Fig.2.3. overturning moment at the base of the dam, with respect to E