A Project Report

On

Off Shore Structures

Submitted by

Sabit Timilsina



Submitted to

Dr. Chao Sun

Professor of Structural Engineering

Department of Civil and Environmental Engineering

May 2025

Project 2 Fixed-leg Offshore Oil/gas Rigs

Fig. 2 shows an offshore oil/gas jacket platform. Water depth d = 50m. Wave height is H = 6m and wave period T = 12s.

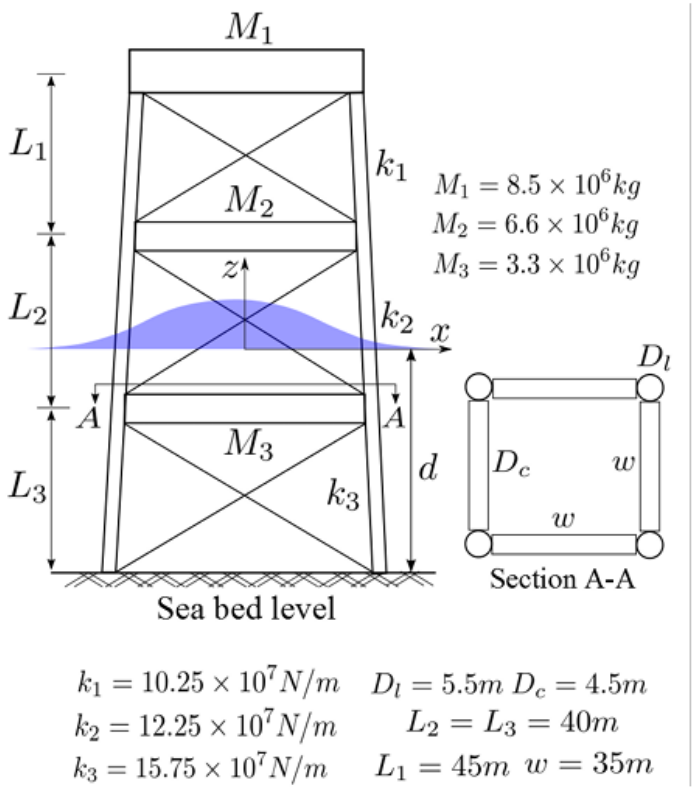


Figure 2 Schematic model of an offshore jacket platform

Part 1 (Calculation of Wind Forces):

First, we need to calculate the wind velocity time series using Kaimal spectrum on mass 1 and 2.

Assume,

Wind velocity at 90 m reference (V_{ref}) = 15 m/s

Density of air (ρ) = 1.225 kg/m^3

Height of M_2 over sea level (Z_2) = $L_2 + L_3 - d$ = 30 m

Height of M_1 over sea level (Z_1) = $L_1 + L_2 + L_3 - d$ = 75 m

Adopt logarithmic wind profile to calculate the mean velocity at M_1 and M_2

$$\bar{v}(z = 30) = V_{ref} \frac{\log(\frac{z}{z_0})}{\log(\frac{H_{ref}}{z_0})}$$ [where z_o is the length of roughness, and its value is 0.003]

= 13.401 m/s

$$\bar{v}(z = 75) = V_{ref} \frac{\log(\frac{z}{z_0})}{\log(\frac{H_{ref}}{z_0})}$$

$$= 14.735 \text{ m/s}$$

The Kaimal Spectrum function for turbulence model is given by:

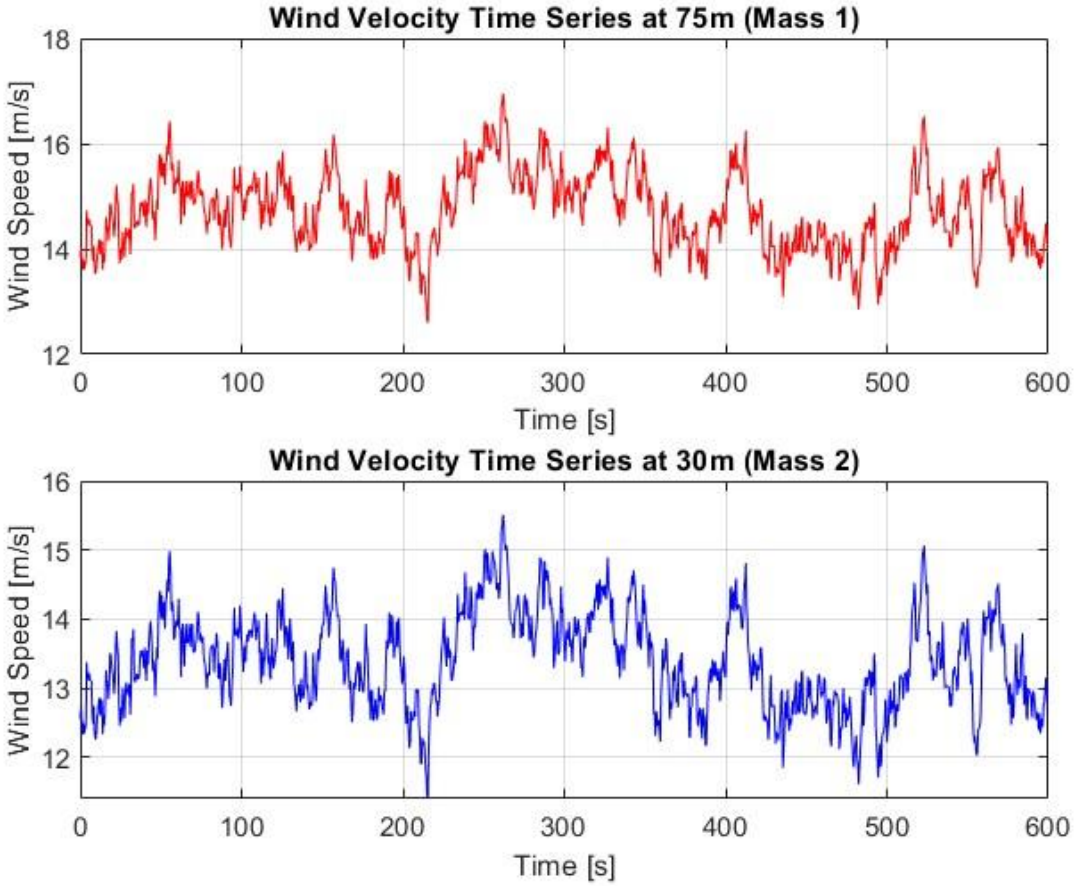
$$S(f) = \frac{4I^2 L}{(1+6f\cdot\frac{L}{u})^{\frac{5}{3}}}$$

where f = cyclic frequency,

L = integral scale parameter which is assumed to be 340.2 m

I = wind turbulence intensity which is assumed to be 10%

The wind velocity time series is given below:



The wind forces on the platform are given by the equation:

$$F = \frac{1}{2}\rho\bar{U}^2 C_d A + \rho\bar{U}\, u(t) C_d A$$

where,

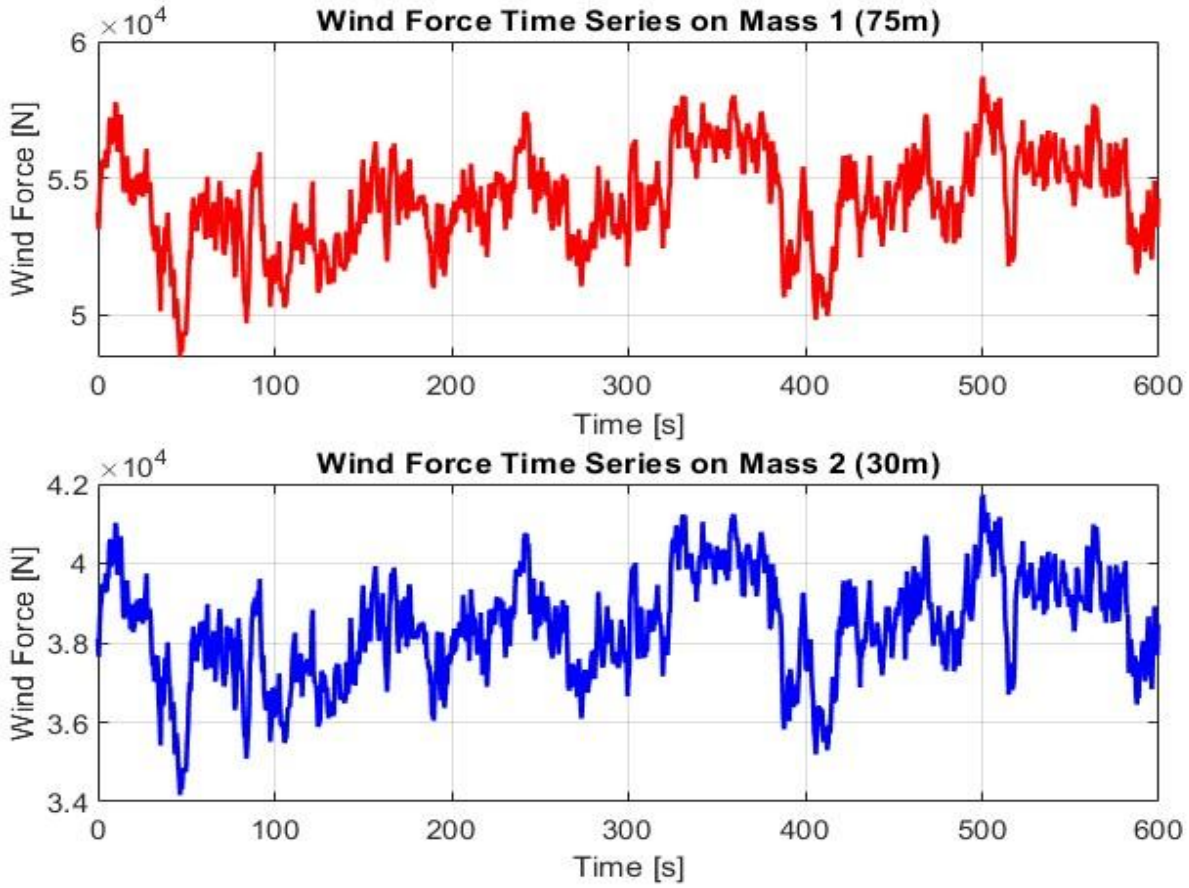
$\bar{U}$ = Mean wind velocity at the platform level

u(t) = fluctuation wind velocity at platform level

C_d = Drag coefficient, which is 0.65

A = Area of the platform

We use top platform area (A_1) = 210 m^2 and middle platform area (A_2) = 180 m^2 and the wind forces were plotted using the MATLAB code and the graph is given below:



Part 2 (Wave profile, water particle velocity and acceleration time series):

The JONSWAP spectrum is given by the equation:

$$S(f) = 0.3125 H_s^2 T_p \left(\frac{f_p}{f}\right)^5 exp\left[-\frac{5}{4}\left(\frac{f}{f_p}\right)^{-4}\right] (1 - 0.287\ ln\gamma)\gamma^{exp\left[-\frac{(\omega-\omega_p)^2}{2\sigma^2\omega_p^2}\right]}$$

Where,

Wave period (T_p) = 12 s

Water depth (d) = 50 m

Significant wave height (H_s) = 6 m

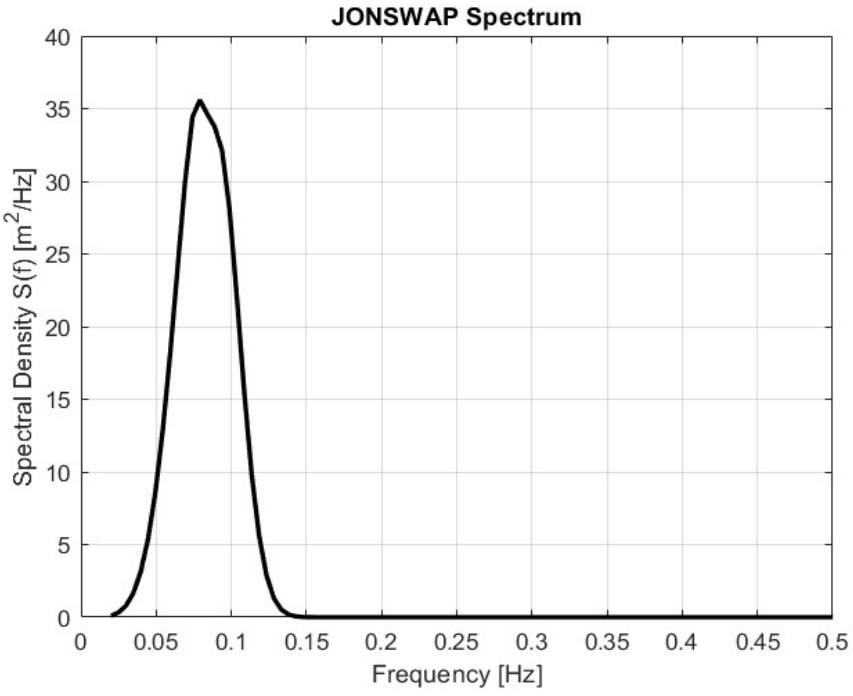
$f_p = 1/T_p$

σ = 0.07 for $f \leq f_p$, σ = 0.09 for $f > f_p$

γ is peakedness parameter which is given by:

$$\gamma = \begin{cases} 5 & T_p/\sqrt{H_s} \leq 3.6 \\ \exp(5.75 - 1.15\ T_p/\sqrt{H_s}) & 3.6 < T_p/\sqrt{H_s} \leq 5.0 \\ 1 & T_p/\sqrt{H_s} < 5.0 \end{cases}$$

The JONSWAP spectrum is shown below:



The wave elevation $\eta(t)$, the fluid particle velocity u and the acceleration a is given by the expressions below:

$$\eta(t) = \sum_{j=1}^{N} A_j \sin(\omega_j t - k_j x + \Phi_j)$$

$$A_j = \sqrt{2S(\omega_j)\Delta\omega}$$

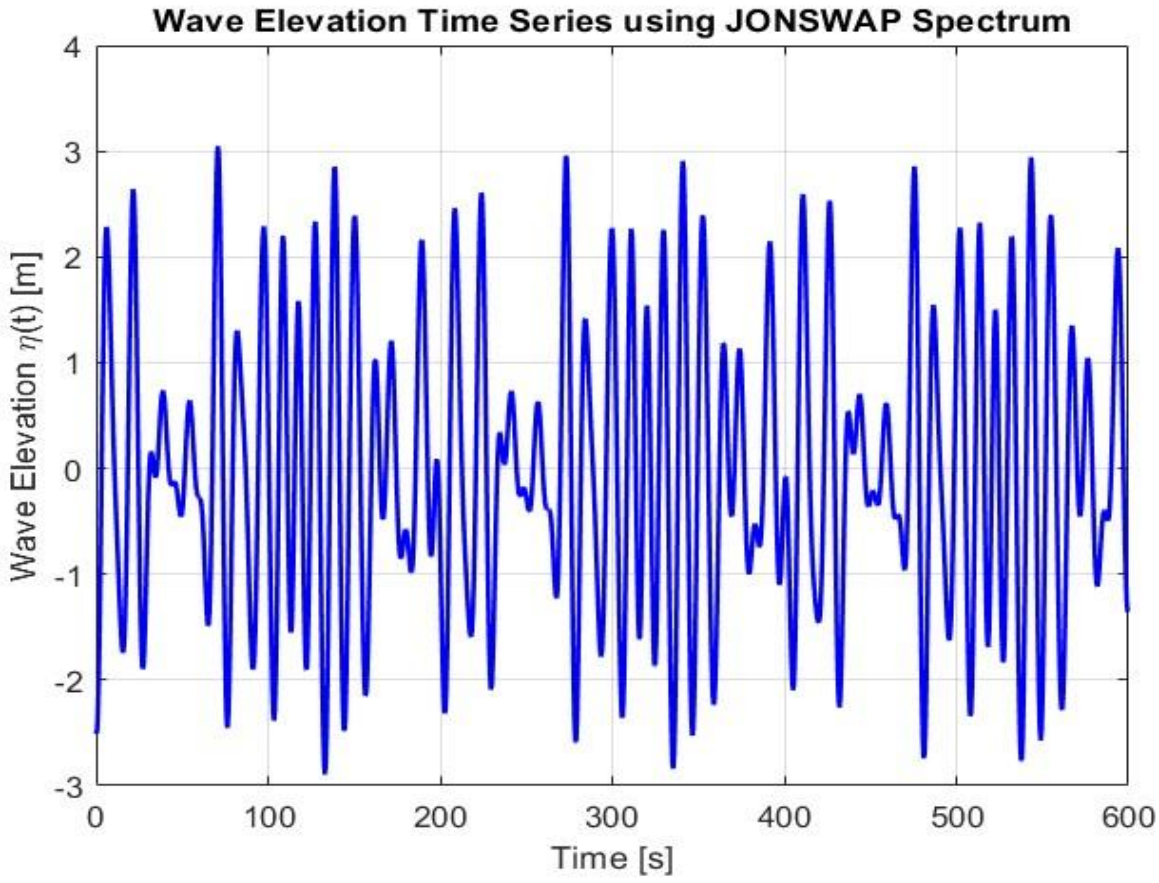
$$u = \sum_{j=1}^{N} \omega_j A_j \frac{\cosh[k(z+d)]}{T_p \sinh(kd)} \sin(\omega_j t - k_j x + \Phi_j)$$

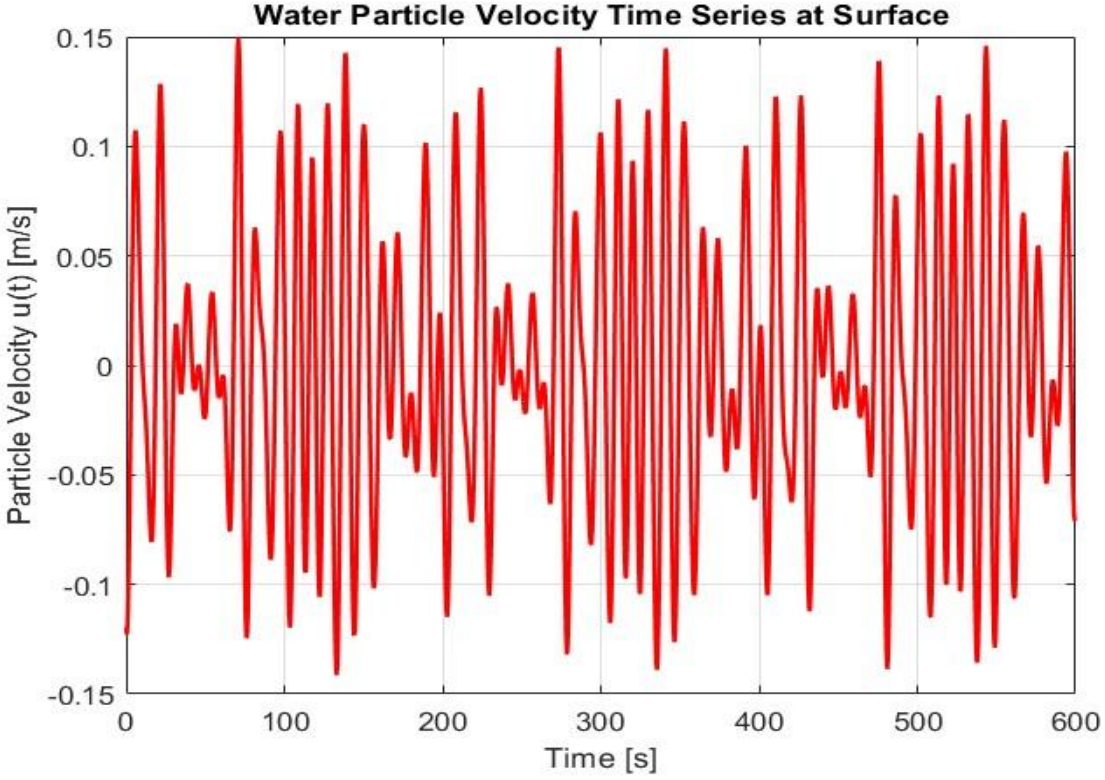
$$a = \sum_{j=1}^{N} \omega_j^2 A_j \frac{\cosh[k(z+d)]}{T_p \sinh(kd)} \cos(\omega_j t - k_j x + \Phi_j)$$

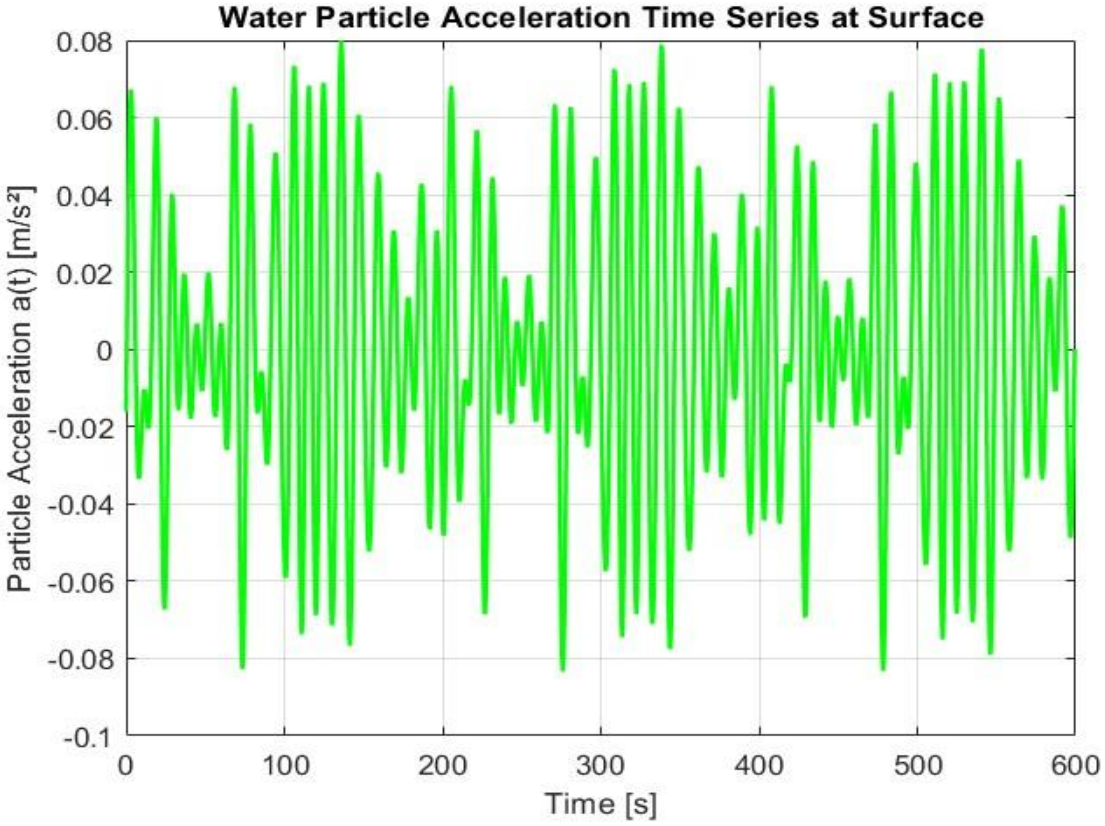
The water depth d, the parameters ω and wave number k are related by the dispersion equation:

$$k\tanh(kd) = \omega^2/g$$

The plot for wave elevation, fluid particle velocity and the acceleration are given below:




Water Particle Velocity Time Series at Surface
Particle Velocity u(t) [m/s]
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
0
100
200
300
400
500
600
Time [s]


Water Particle Acceleration Time Series at Surface
Particle Acceleration a(t) [m/s²]
0.08
0.06
0.04
0.02
0
-0.02
-0.04
-0.06
-0.08
-0.1
0
100
200
300
400
500
600
Time [s]

Part 3 (Calculation of Wave Loading on the Jacket):

Since, D/H < 1.0

So, the horizontal forces on the pile and the cross structure can be calculated using the Morrison equation.

$$F_x = \rho C_M \frac{\pi D^2}{4} a + \frac{1}{2}\pi C_D |u| u$$

Where,

$\rho =$ Density of water; Assumed as 1000 kg/m^3.

C_M = Coefficient of Inertia; Assumed as 2.0 here.

$C_D =$ Coefficient of Drag; Assumed as 1.2 here.

$$acceleration\ (a) = \sum_{j=1}^{N} \omega_j^2 A_j \frac{\cosh\left[k(z+d)\right]}{T_p \sinh\left(kd\right)} \cos(\omega_j t - k_j x + \Phi_j)$$

$$velocity\ (u) = \sum_{j=1}^{N} \omega_j A_j \frac{\cosh\left[k(z+d)\right]}{T_p \sinh\left(kd\right)} \sin(\omega_j t - k_j x + \Phi_j)$$

There is variation of velocity and acceleration of water particles along the depth and longitudinal axis. So, I discretise the vertical piles in 1 m, i.e., from the water surface to the seabed (z = 0 to z = –50 m). At each of these discrete depths, the local water particle velocity and acceleration were computed using above equation, and the corresponding Morison force was calculated per unit length.

For, the pile in the backward direction of flow; similar methodology was done to calculate force while keeping the x = 40.5 m (center-center distance of pile).

The force on the cross-bracing members which is perpendicular to the flow was calculated using Morrison equation, by keeping z = -10 m and x = 0 for 1st cross-bracing, and x = 40.5 m for second cross-bracing.

The cross-bracing along the direction of flow were assumed as spherical bodies for simplicity in calculation and calculation was done using the equation below:

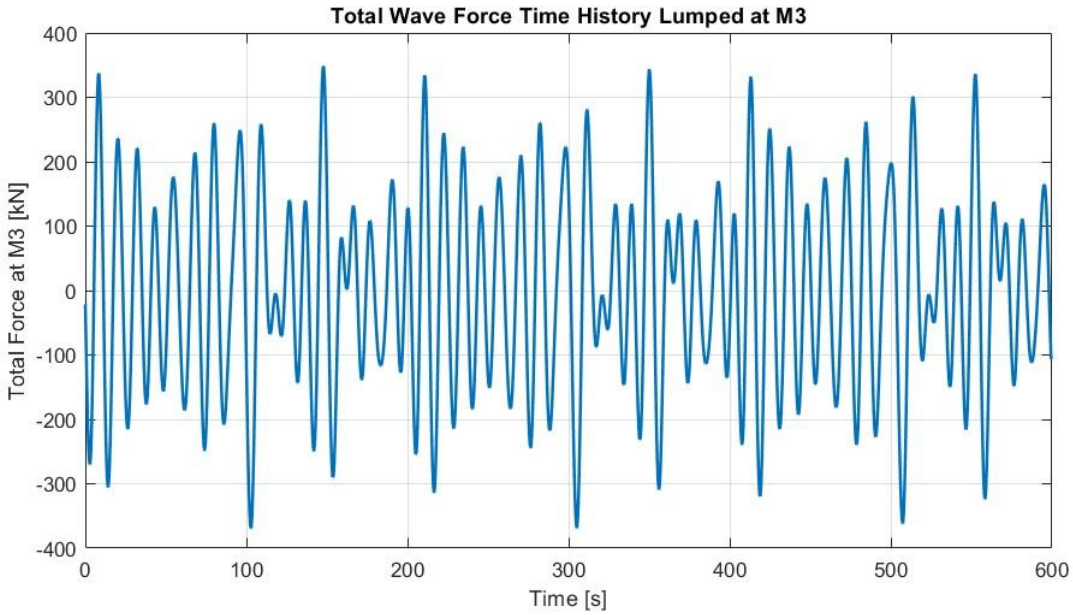
$$F_x = \frac{1}{2}\rho C_D \frac{\pi D^2}{4} |u| u$$

Where:

C_D for the sphere is approximated as 0.45.

D is the equivalent diameter of the sphere (assumed 4.5 m)

The figure below shows the wave forces time-history in the lumped mass (M_3):



Part 4 (Equation of motion of the jacket):

The equation of motion for multi-degree-of-freedom system is given by equation below:

$$M\ddot{u}(t) + C\dot{u}(t) + Ku(t) = F(t)$$

Where:

$\ddot{u}(t)$ = acceleration vector

$\dot{u}(t)$ = velocity vector

$u(t)$ = displacement vector

F(t) = Force vector

Formulation of Mass matrix M:

$$\mathrm{M} = \begin{bmatrix} M_1 & 0 & 0 \\ 0 & M_2 & 0 \\ 0 & 0 & M_3 \end{bmatrix} = \begin{bmatrix} 8.5 & 0 & 0 \\ 0 & 6.6 & 0 \\ 0 & 0 & 3.3 \end{bmatrix} \times 10^6\, kg$$

Stiffness Matrix:

$$\mathrm{K} = \begin{bmatrix} K_1 & -K_1 & 0 \\ -K_1 & K_1 + K_2 & -K_2 \\ 0 & -K_2 & K_2 + K_3 \end{bmatrix} = \begin{bmatrix} 10.25 & -10.25 & 0 \\ -10.25 & 22.5 & -12.25 \\ 0 & -12.25 & 28 \end{bmatrix} \times 10^7\, N/m$$

Finding the natural frequency using the equation:

Det $[\omega^2[M] - [K]] = 0$

On solving the equation:

The natural frequencies are:

$$\omega_1 = 1.882\ rad/s$$

$$\omega_2 = 5.5771\ rad/s$$

$$\omega_3 = 9.8147\ rad/s$$

Now, the modal matrix can be found by the equation:

$\Phi = \omega^2[M] - [K]$

The modal matrix is calculated as:

$$\Phi = \begin{bmatrix} 0.2873 & 0.1860 & 0.0223 \\ 0.2024 & -0.2938 & -0.1556 \\ 0.0924 & -0.2030 & 0.5033 \end{bmatrix} 10^{-3}\ \text{m}$$

The modal mass is calculated as:

$M_n = \Phi^T\ \omega\ \Phi$

So, $M_n = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

The modal stiffness matrix is calculated as:

$K_n = \omega^2 M_n$

So, $K_n = \begin{bmatrix} 3.5652 & 0 & 0 \\ 0 & 31.104 & 0 \\ 0 & 0 & 96.329 \end{bmatrix}$

For the damping matrix, first two damping rations (ξ_1= 0.01 and ξ_2 = 0.02) and let the last damping ratio be ξ_3 = 0.02

$C_n = 2\ \xi * \omega * M_n$

So, $C_n = \begin{bmatrix} 0.03776 & 0 & 0 \\ 0 & 0.22308 & 0 \\ 0 & 0 & 0.39259 \end{bmatrix}$

Hence, the final equation of motion is:

$$\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}\ddot{u}(t) + \begin{bmatrix}0.03776 & 0 & 0\\ 0 & 0.22308 & 0\\ 0 & 0 & 0.39259\end{bmatrix}\dot{u}(t) + \begin{bmatrix}3.5652 & 0 & 0\\ 0 & 31.104 & 0\\ 0 & 0 & 96.329\end{bmatrix}u(t) = \mathrm{F(t)}$$

Part 5 (Dynamic Responses of the system: displacements, shear forces at each story, and overturning moment at the base):

For this part, the jacket is modeled as a 3-DOF lumped mass system.

The equation of motion is as follows:

$$M\ddot{u}(t) + C\dot{u}(t) + Ku(t) = F(t)$$

Where:

M = Modal Mass matrix

C = Modal Damping matrix

K = Modal stiffness matrix

U(t) = displacement vector

F(t) = External force vector (wind + wave)

The force time series acting on the structure was obtained from CSV data based on above parts. These forces were projected into the modal space using normalized eigenvectors.

The system response was solved in modal coordinates using Newmark-beta method with parameters β = 1/4 and γ=1/2.

The governing equation for modal acceleration $\ddot{q}(t),\ velocity\ \dot{q}(t),\ and\ displacement\ q(t)$ were recursively used foe each time step.

The displacement equation is:

$$u_i(t) = \sum_{n=1}^{3} \Phi_{i,n} q_n(t)$$

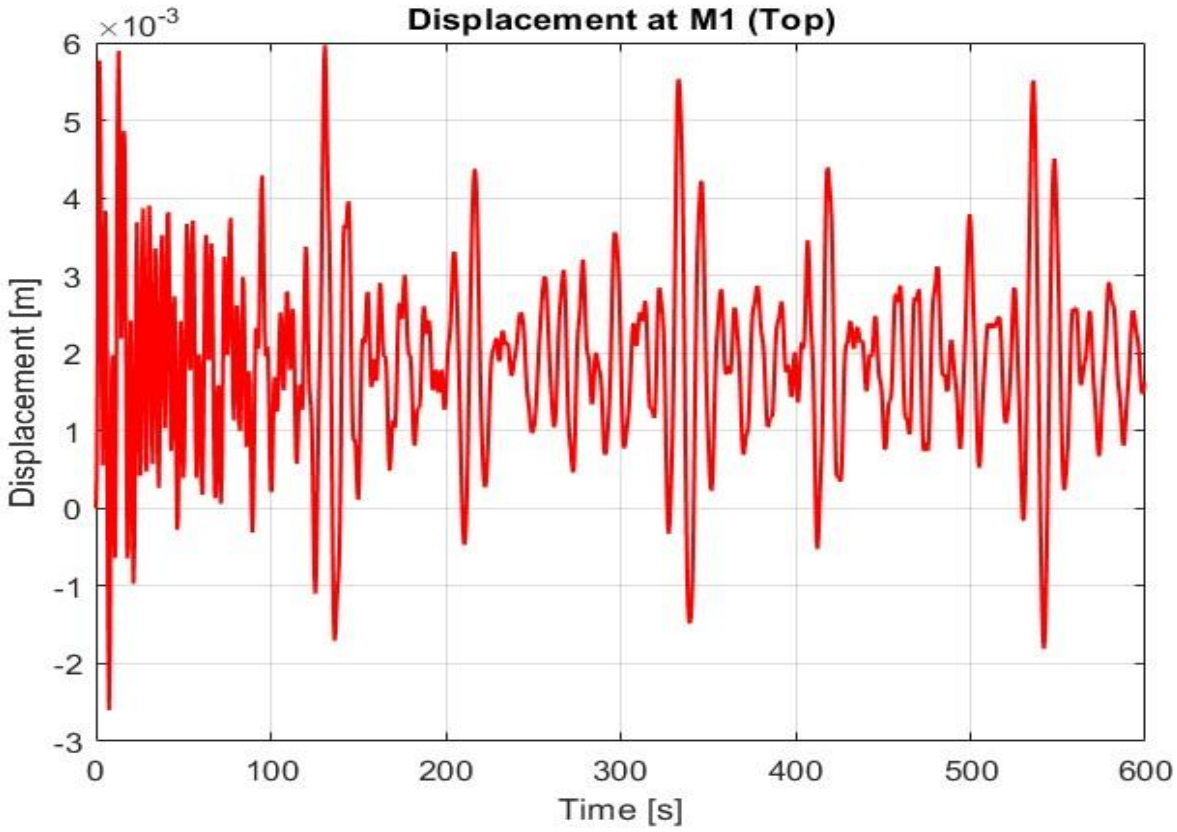
The shear forces are computed as follows:

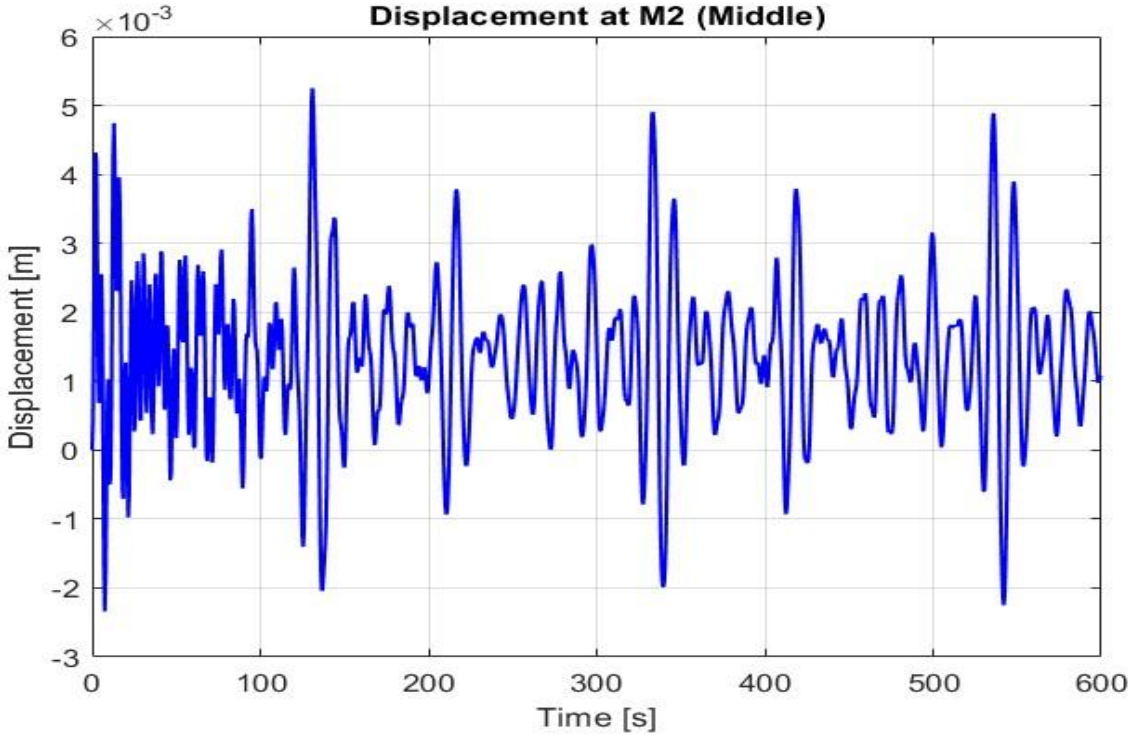
$$V_{base}(t) = -EI \sum_{n=1}^{3} \Phi_n'''(0) q_n(t)$$

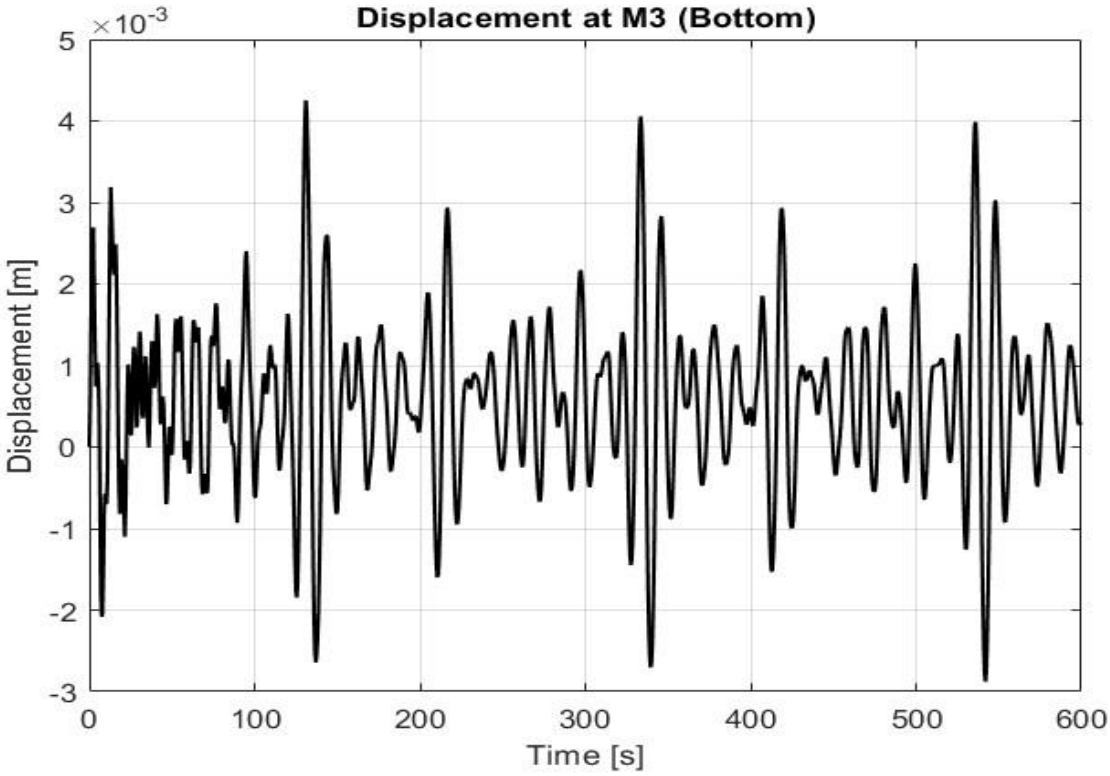
The overturning moments are computed as follows:

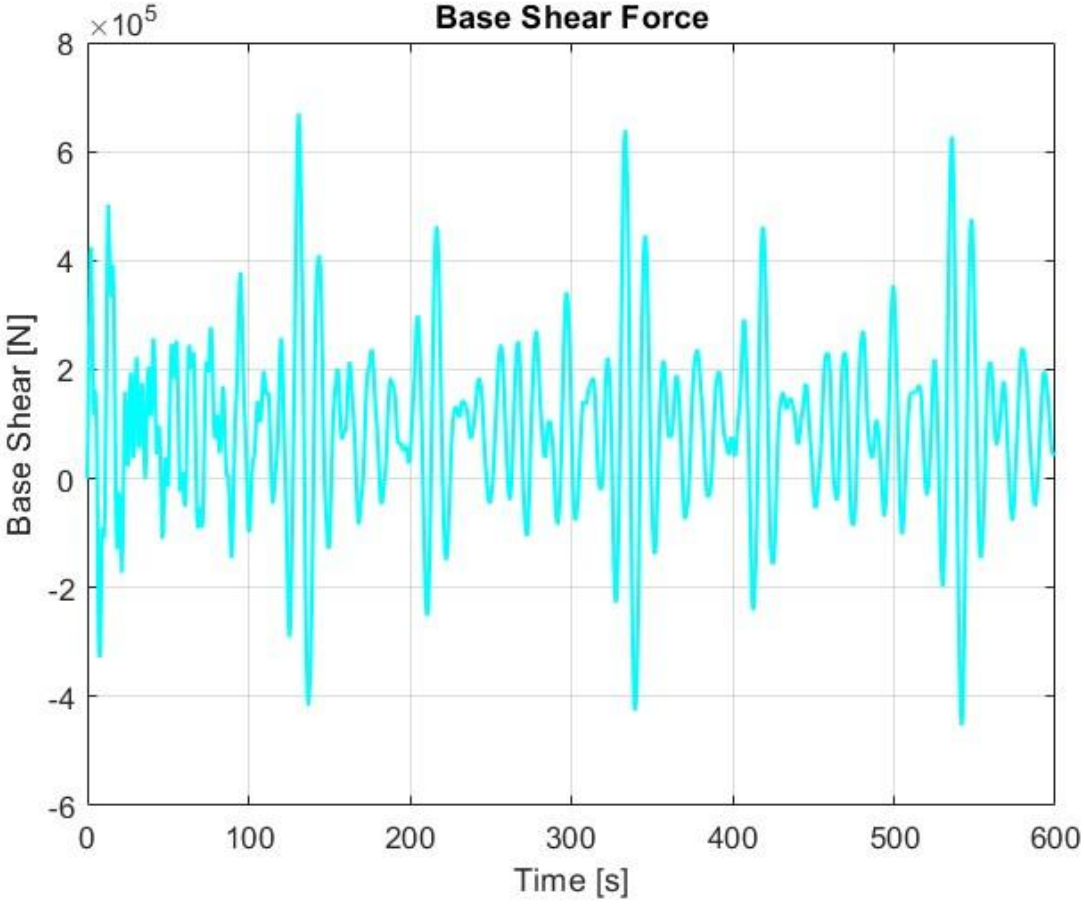
$$M_{base}(t) = -EI \sum_{n=1}^{3} \Phi_n''(0) q_n(t)$$

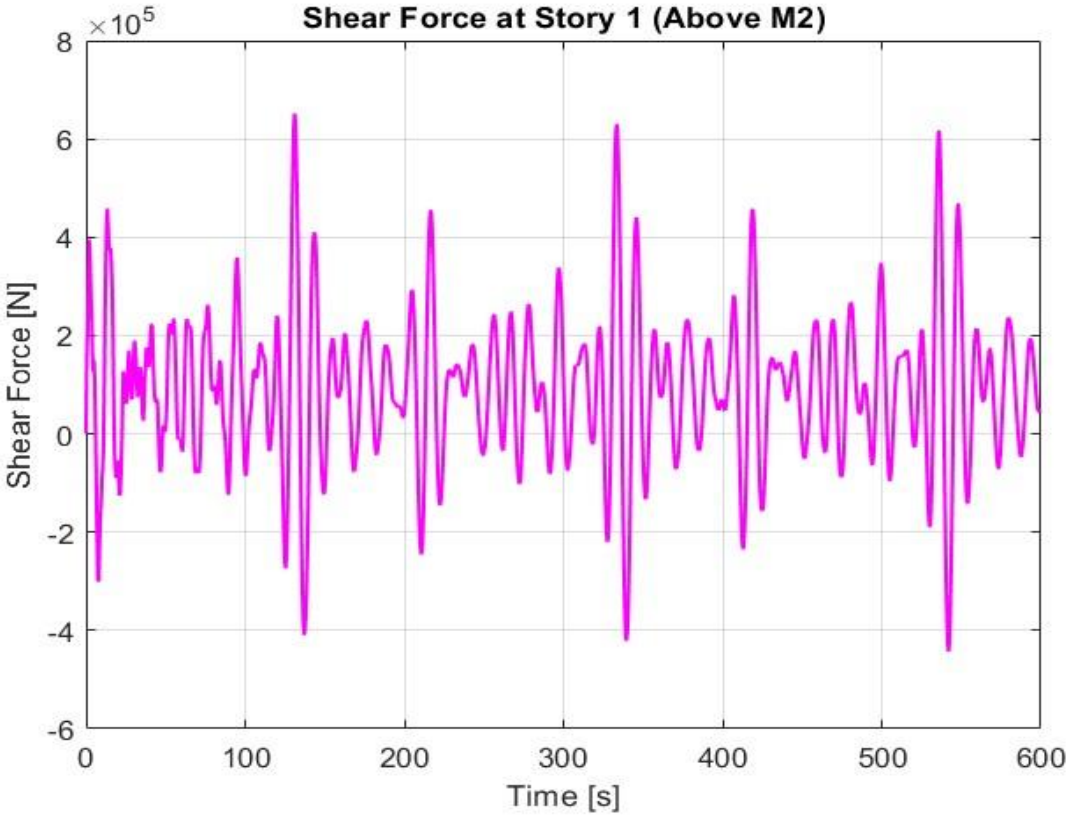
The generated graphs are shown below:

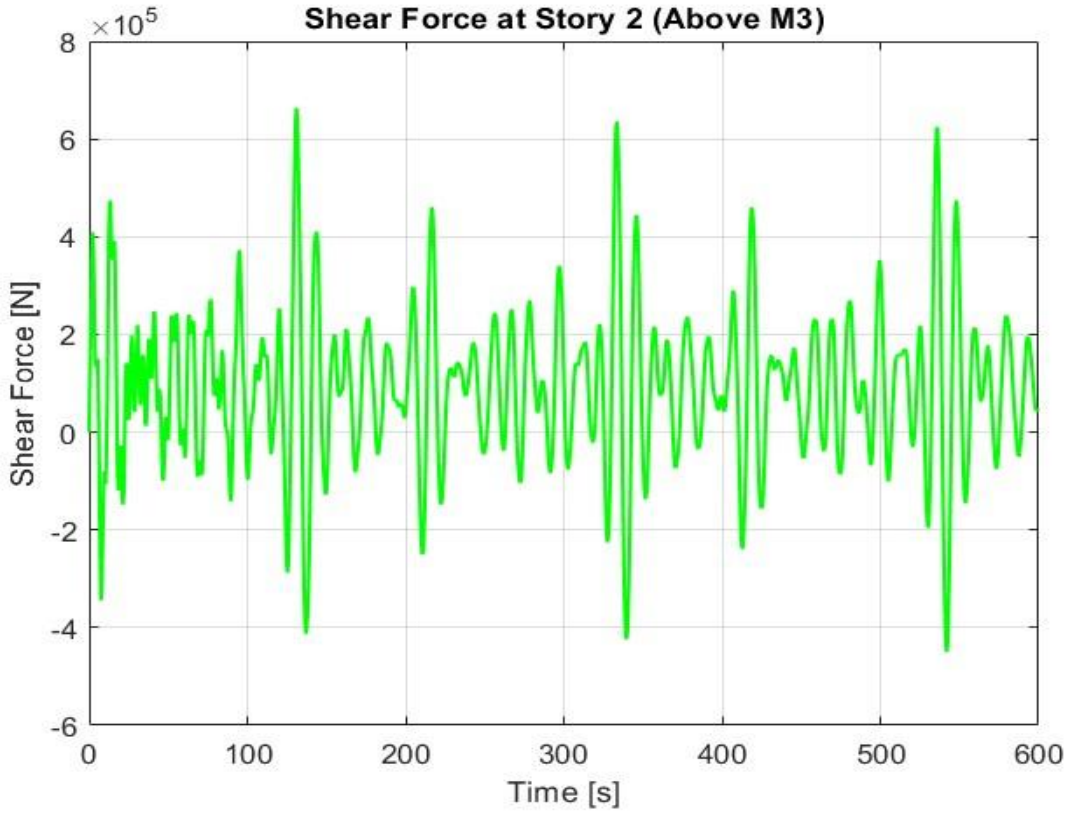


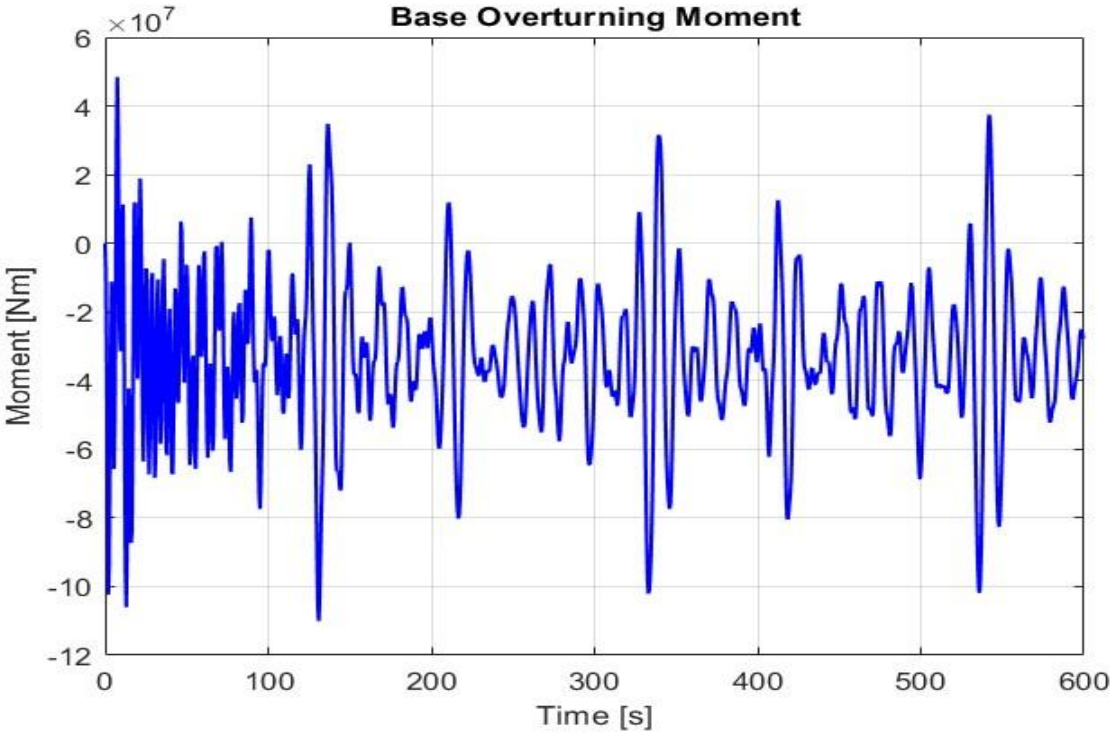

Displacement at M2 (Middle)
×10^-3
6
5
4
3
2
1
0
-1
-2
-3
Displacement [m]
0
100
200
300
400
500
600
Time [s]


Displacement at M3 (Bottom)
×10^-3
5
4
3
2
1
0
-1
-2
-3
Displacement [m]
0
100
200
300
400
500
600
Time [s]


Base Shear Force
×10^5
8
6
4
2
0
-2
-4
-6
Base Shear [N]
0
100
200
300
400
500
600
Time [s]


Shear Force at Story 1 (Above M2)
×10^5
8
6
4
2
0
-2
-4
-6
Shear Force [N]
0
100
200
300
400
500
600
Time [s]


Shear Force at Story 2 (Above M3)
×10^5
8
6
4
2
0
-2
-4
-6
Shear Force [N]
0
100
200
300
400
500
600
Time [s]


Base Overturning Moment
×10^7
6
4
2
0
-2
-4
-6
-8
-10
-12
Moment [Nm]
0
100
200
300
400
500
600
Time [s]

Codes for the project:

1) Calculation of wind velocity and forces:

```
%% Wind Velocity and Wind Force Time Series for Mass 1 and Mass 2
% Based strictly on Kaimal Spectrum and provided force formula

clear; clc; close all

%% Step 1: Parameters

% Reference parameters
Vref = 15;            % Reference wind speed at 90 m [m/s]
Href = 90;            % Reference height [m]
z0 = 0.003;           % Roughness length [m]

% Heights for masses
z1 = 75;              % Height for Mass 1 [m]
z2 = 30;              % Height for Mass 2 [m]

% Mean velocities calculated using logarithmic profile
v_mean_75 = 14.735;   % [m/s]
v_mean_30 = 13.401;   % [m/s]

% Air density and drag coefficient
rho_air = 1.225;      % [kg/m^3]
Cd = 0.65;            % Drag coefficient

% Wind pressure areas
A1 = 210;             % Area for Mass 1 [m^2]
A2 = 180;             % Area for Mass 2 [m^2]

% Turbulence intensity
I = 0.10;             % 10%

% Integral length scale
L = 340.2;            % [m]

% Time settings
T_total = 599.5;      % Total simulation time [s]
dt = 0.5;             % Time step [s]
t = (0:dt:T_total)';  % Time vector
n = length(t);        % Number of points

%% Step 2: Frequency Domain Setup

df = 1/T_total;           % frequency resolution
f = (0:n/2)' * df;        % positive frequencies
f(1) = 1e-6;              % avoid division by zero

%% Step 3: Kaimal Spectrum using given formula

Su_75 = (4 * I^2 * L) ./ ( (1 + 6 * f * L ./ v_mean_75).^(5/3) );  % Mass 1 (75m)
Su_30 = (4 * I^2 * L) ./ ( (1 + 6 * f * L ./ v_mean_30).^(5/3) );  % Mass 2 (30m)
```

```
%% Step 4: Generate Turbulent Wind Time Series

% Random phase angles
phi = 2*pi*rand(length(f),1);

% Fourier amplitudes
A_75 = sqrt(2*Su_75*df);
A_30 = sqrt(2*Su_30*df);

% Build full symmetric Fourier spectrum
U_hat_75 = [A_75 .* exp(1i*phi); conj(flipud(A_75(2:end-1) .* exp(1i*phi(2:end-
1))))];
U_hat_30 = [A_30 .* exp(1i*phi); conj(flipud(A_30(2:end-1) .* exp(1i*phi(2:end-
1))))];

% Inverse FFT to get fluctuating wind speed
u_fluct_75 = real(ifft(U_hat_75))*length(U_hat_75);
u_fluct_30 = real(ifft(U_hat_30))*length(U_hat_30);

%% Step 5: Add Mean Wind Speed

u_total_75 = v_mean_75 + u_fluct_75;
u_total_30 = v_mean_30 + u_fluct_30;

%% Step 6: Calculate Wind Forces (based on provided formula)

% Force on Mass 1
F1 = 0.5 * rho_air * (v_mean_75^2) * Cd * A1 + rho_air * Cd * A1 * v_mean_75 *
u_total_75;

% Force on Mass 2
F2 = 0.5 * rho_air * (v_mean_30^2) * Cd * A2 + rho_air * Cd * A2 * v_mean_30 *
u_total_30;

%% Step 7: Plot Wind Velocity and Wind Force Time Series

figure;
subplot(2,1,1)
plot(t, u_total_75, 'r', 'LineWidth', 1.5)
xlabel('Time [s]')
ylabel('Wind Speed [m/s]')
title('Wind Velocity Time Series at 75m (Mass 1)')
grid on

subplot(2,1,2)
plot(t, u_total_30, 'b', 'LineWidth', 1.5)
xlabel('Time [s]')
ylabel('Wind Speed [m/s]')
title('Wind Velocity Time Series at 30m (Mass 2)')
grid on

figure;
subplot(2,1,1)
plot(t, F1, 'r', 'LineWidth', 1.5)
xlabel('Time [s]')
```

```
ylabel('Wind Force [N]')
title('Wind Force Time Series on Mass 1 (75m)')
grid on

subplot(2,1,2)
plot(t, F2, 'b', 'LineWidth', 1.5)
xlabel('Time [s]')
ylabel('Wind Force [N]')
title('Wind Force Time Series on Mass 2 (30m)')
grid on

%% Step 8: Export Wind Forces to CSV File

% Combine time and force data into a table
wind_force_table = table(t, F1, F2, ...
    'VariableNames', {'Time_s', 'Force_Mass1_N', 'Force_Mass2_N'});

% Write to CSV
writetable(wind_force_table, 'wind_force_time_series.csv');

disp('Wind force time series saved to "wind_force_time_series.csv".');
```

2) JONSWAP spectrum, wave displacement, velocity, acceleration and wave forces:

```
%% Wave Force Computation: 4 Piles + 4 Braces (2 Cylinders, 2 Spheres)
clear; clc; close all

%% Step 1: Input Parameters
Tp = 12; d = 50; Hs = 6; fp = 1/Tp; g = 9.81;
rho = 1025; D_pile = 5.5; D_brace = 4.5; CM = 2.0; CD = 1.2;
CD_sphere = 0.45; % Drag coefficient for spherical brace approximation
x_front = 0; x_back = 40.5;

%% Step 2: Frequency Setup
N = 200;
f = linspace(0.02, 1.0, N)'; df = f(2)-f(1); omega = 2*pi*f;
omega_p = 2*pi*fp;

%% Step 3: JONSWAP Spectrum
Tp_Hs = Tp / sqrt(Hs);
if Tp_Hs <= 3.6
    gamma = 5;
elseif Tp_Hs <= 5.0
    gamma = exp(5.75 - 1.15 * Tp_Hs);
else
    gamma = 1;
end

sigma = 0.07 * ones(size(f));
sigma(f > fp) = 0.09;

S_f = 0.3125 * Hs^2 * Tp .* (f/fp).^5 ...
```

```
    .* exp(-5/4 * (f./fp).^4) ...
    .* (1 - 0.287 * log(gamma)) .* gamma.^exp( -( (omega - omega_p).^2 ) ./
(2*(sigma.^2)*(omega_p^2)) );

%% Step 4: Wave Number Calculation
k = zeros(size(omega));
for i = 1:N
    k_i = omega(i)^2 / g;
    for j = 1:100
        f1 = k_i * tanh(k_i*d) - omega(i)^2/g;
        df1 = tanh(k_i*d) + k_i*d*(1 - tanh(k_i*d)^2);
        k_i = k_i - f1/df1;
    end
    k(i) = k_i;
end

%% Step 5: Time Vector
T_total = 599.5; dt = 0.5; t = (0:dt:T_total)'; nt = length(t);

%% Step 6: Random Phase and Amplitudes
phi = 2*pi*rand(N,1); Aj = sqrt(2*S_f*df);

%% Step 7: u(t,z,x) and a(t,z,x) for Piles
z_vec = 0:-1:-50; Nz = length(z_vec);
u_pile_front = zeros(nt,Nz); a_pile_front = zeros(nt,Nz);
u_pile_back  = zeros(nt,Nz); a_pile_back  = zeros(nt,Nz);

for zi = 1:Nz
    z = z_vec(zi);
    u_f = zeros(nt,1); a_f = zeros(nt,1);
    u_b = zeros(nt,1); a_b = zeros(nt,1);

    for i = 1:N
        omega_i = omega(i); k_i = k(i); Aj_i = Aj(i); phi_i = phi(i);
        cosh_fac = cosh(k_i*(z + d)) / sinh(k_i*d);

        u_f = u_f + omega_i * Aj_i * cosh_fac / Tp .* sin(omega_i*t - k_i*x_front +
phi_i);
        a_f = a_f + omega_i^2 * Aj_i * cosh_fac / Tp .* cos(omega_i*t - k_i*x_front +
phi_i);

        u_b = u_b + omega_i * Aj_i * cosh_fac / Tp .* sin(omega_i*t - k_i*x_back +
phi_i);
        a_b = a_b + omega_i^2 * Aj_i * cosh_fac / Tp .* cos(omega_i*t - k_i*x_back +
phi_i);
    end

    u_pile_front(:,zi) = u_f; a_pile_front(:,zi) = a_f;
    u_pile_back(:,zi)  = u_b; a_pile_back(:,zi)  = a_b;
end

%% Step 8: Forces on Piles (4 total)
Fx_front = zeros(nt,1); Fx_back = zeros(nt,1);

for zi = 1:Nz
```

```
    u_f = u_pile_front(:,zi); a_f = a_pile_front(:,zi);
    u_b = u_pile_back(:,zi);  a_b = a_pile_back(:,zi);

    Fx_f = rho * CM * (pi*D_pile^2/4) .* a_f + 0.5 * rho * CD * pi * D_pile .*
abs(u_f).*u_f;
    Fx_b = rho * CM * (pi*D_pile^2/4) .* a_b + 0.5 * rho * CD * pi * D_pile .*
abs(u_b).*u_b;

    Fx_front = Fx_front + Fx_f;
    Fx_back  = Fx_back  + Fx_b;
end

Fx_piles_total = 2 * (Fx_front + Fx_back);

%% Step 9A: Forces on 2 Cylindrical Braces at z = -10
z_brace_cyl = -10;
u_bf = zeros(nt,1); a_bf = zeros(nt,1);
u_bb = zeros(nt,1); a_bb = zeros(nt,1);

for i = 1:N
    omega_i = omega(i); k_i = k(i); Aj_i = Aj(i); phi_i = phi(i);
    cosh_fac = cosh(k_i*(z_brace_cyl + d)) / sinh(k_i*d);

    u_bf = u_bf + omega_i * Aj_i * cosh_fac / Tp .* sin(omega_i*t - k_i*x_front +
phi_i);
    a_bf = a_bf + omega_i^2 * Aj_i * cosh_fac / Tp .* cos(omega_i*t - k_i*x_front +
phi_i);

    u_bb = u_bb + omega_i * Aj_i * cosh_fac / Tp .* sin(omega_i*t - k_i*x_back +
phi_i);
    a_bb = a_bb + omega_i^2 * Aj_i * cosh_fac / Tp .* cos(omega_i*t - k_i*x_back +
phi_i);
end

F_bf = rho * CM * (pi*D_brace^2/4) .* a_bf + 0.5 * rho * CD * pi * D_brace .*
abs(u_bf).*u_bf;
F_bb = rho * CM * (pi*D_brace^2/4) .* a_bb + 0.5 * rho * CD * pi * D_brace .*
abs(u_bb).*u_bb;

Fx_cyl_braces_total = F_bf + F_bb;

%% Step 9B: Forces on 2 Spherical Braces at z = -20 (drag-only)
z_brace_sph = -20;
u_bf_sph = zeros(nt,1);
u_bb_sph = zeros(nt,1);

for i = 1:N
    omega_i = omega(i); k_i = k(i); Aj_i = Aj(i); phi_i = phi(i);
    cosh_fac = cosh(k_i*(z_brace_sph + d)) / sinh(k_i*d);

    u_bf_sph = u_bf_sph + omega_i * Aj_i * cosh_fac / Tp .* sin(omega_i*t -
k_i*x_front + phi_i);
    u_bb_sph = u_bb_sph + omega_i * Aj_i * cosh_fac / Tp .* sin(omega_i*t -
k_i*x_back + phi_i);
end
```

```
A_sphere = pi * D_brace^2 / 4;
F_bf_sph = 0.5 * rho * CD_sphere * A_sphere .* abs(u_bf_sph) .* u_bf_sph;
F_bb_sph = 0.5 * rho * CD_sphere * A_sphere .* abs(u_bb_sph) .* u_bb_sph;

Fx_sph_braces_total = F_bf_sph + F_bb_sph;

%% Step 10: Total Force (Lumped at M3)
Fx_total = Fx_piles_total + Fx_cyl_braces_total + Fx_sph_braces_total;

%% Step 11: Plot
figure;
plot(t, Fx_total/1e3, 'LineWidth', 1.5)
xlabel('Time [s]')
ylabel('Total Force at M3 [kN]')
title('Total Wave Force at Lumped Mass M3')
grid on
```

3) Structural Response:

```
%% MODAL NEWMARK METHOD (Displacement + Shear + Overturning Moment)
clc; clear; close all;

%% Step 1: System Properties
% Masses (kg)
M1 = 8.5e6; M2 = 6.6e6; M3 = 3.3e6;
M = diag([M1, M2, M3]);

% Stiffnesses (N/m)
k1 = 10.25e7; k2 = 12.25e7; k3 = 15.75e7;
K = [k1, -k1, 0;
     -k1, k1+k2, -k2;
      0, -k2, k2+k3];

% Modal analysis
[V, D] = eig(K, M);
omega_n = sqrt(diag(D));
[omega_n, idx] = sort(omega_n);
V = V(:, idx);

% Normalize mode shapes
modal_mass = zeros(3,1);
for i = 1:3
    phi = V(:,i);
    V(:,i) = phi / sqrt(phi' * M * phi);
    modal_mass(i) = V(:,i)' * M * V(:,i);
end

modal_stiffness = (omega_n.^2) .* modal_mass;
zeta = [0.01; 0.02; 0.02];
modal_damping = 2 .* zeta .* omega_n .* modal_mass;

%% Step 2: Load Force Time Series from CSV
```

```
F_data = readmatrix('force_time_series.csv');  % format: time, F1, F2, F3
t = F_data(:,1); dt = t(2) - t(1); Nt = length(t);
F_total = F_data(:,2:4)';  % 3×Nt: [F1; F2; F3]

% Project forces to modal space
f_modal = V' * F_total;

%% Step 3: Newmark-beta Integration (Modal Coordinates)
beta = 0.25; gamma = 0.5;
q = zeros(3, Nt); qd = zeros(3, Nt); qdd = zeros(3, Nt);
a0 = 1 / (beta * dt^2); a1 = gamma / (beta * dt);
a2 = 1 / (beta * dt); a3 = 1 / (2*beta) - 1;
a4 = gamma / beta - 1; a5 = dt * (gamma / (2 * beta) - 1);

for n = 1:3
    m_r = modal_mass(n); c_r = modal_damping(n); k_r = modal_stiffness(n);
    for i = 2:Nt
        k_eff = k_r + a0*m_r + a1*c_r;
        A1 = a0*q(n,i-1) + a2*qd(n,i-1) + a3*qdd(n,i-1);
        A2 = a1*q(n,i-1) + a4*qd(n,i-1) + a5*qdd(n,i-1);
        f_eff = f_modal(n,i) + m_r*A1 + c_r*A2;
        q(n,i) = f_eff / k_eff;
        qdd(n,i) = a0*(q(n,i) - q(n,i-1)) - a2*qd(n,i-1) - a3*qdd(n,i-1);
        qd(n,i) = qd(n,i-1) + dt*((1-gamma)*qdd(n,i-1) + gamma*qdd(n,i));
    end
end

%% Step 4: Reconstruct Displacements and Accelerations at Each DOF
u1 = V(1,:) * q;  % M1 displacement
u2 = V(2,:) * q;  % M2 displacement
u3 = V(3,:) * q;  % M3 displacement

a1 = V(1,:) * qdd;  % M1 acceleration
a2 = V(2,:) * qdd;  % M2 acceleration
a3 = V(3,:) * qdd;  % M3 acceleration

%% Step 5: Shear Force Computation (Dynamic Equilibrium)
% Base shear
V_base = k3 * u3;

% Story 2 (above M3)
V23 = V_base + M3 * a3;

% Story 1 (above M2)
V12 = V23 + M2 * a2;

%% Step 6: Overturning Moment at Base (using curvature)
EI = 1e9;  % assumed flexural rigidity
phi_dd = [0.005, 0.002, 0.001];  % assumed curvature of mode shapes at base
M_base = -EI * (phi_dd * q);  % base overturning moment

%% Step 7: Plot Displacements
figure;
plot(t, u1, 'r', 'LineWidth', 1.3);
xlabel('Time [s]'); ylabel('Displacement [m]');
```

```
title('Displacement at M1 (Top)'); grid on;

figure;
plot(t, u2, 'b', 'LineWidth', 1.3);
xlabel('Time [s]'); ylabel('Displacement [m]');
title('Displacement at M2 (Middle)'); grid on;

figure;
plot(t, u3, 'k', 'LineWidth', 1.3);
xlabel('Time [s]'); ylabel('Displacement [m]');
title('Displacement at M3 (Bottom)'); grid on;

%% Step 8: Plot Shear Forces
figure;
plot(t, V12, 'm', 'LineWidth', 1.3);
xlabel('Time [s]'); ylabel('Shear Force [N]');
title('Shear Force at Story 1 (Above M2)'); grid on;

figure;
plot(t, V23, 'g', 'LineWidth', 1.3);
xlabel('Time [s]'); ylabel('Shear Force [N]');
title('Shear Force at Story 2 (Above M3)'); grid on;

figure;
plot(t, V_base, 'c', 'LineWidth', 1.3);
xlabel('Time [s]'); ylabel('Base Shear [N]');
title('Base Shear Force'); grid on;

%% Step 9: Plot Overturning Moment
figure;
plot(t, M_base, 'b', 'LineWidth', 1.3);
xlabel('Time [s]'); ylabel('Moment [Nm]');
title('Base Overturning Moment'); grid on;
```