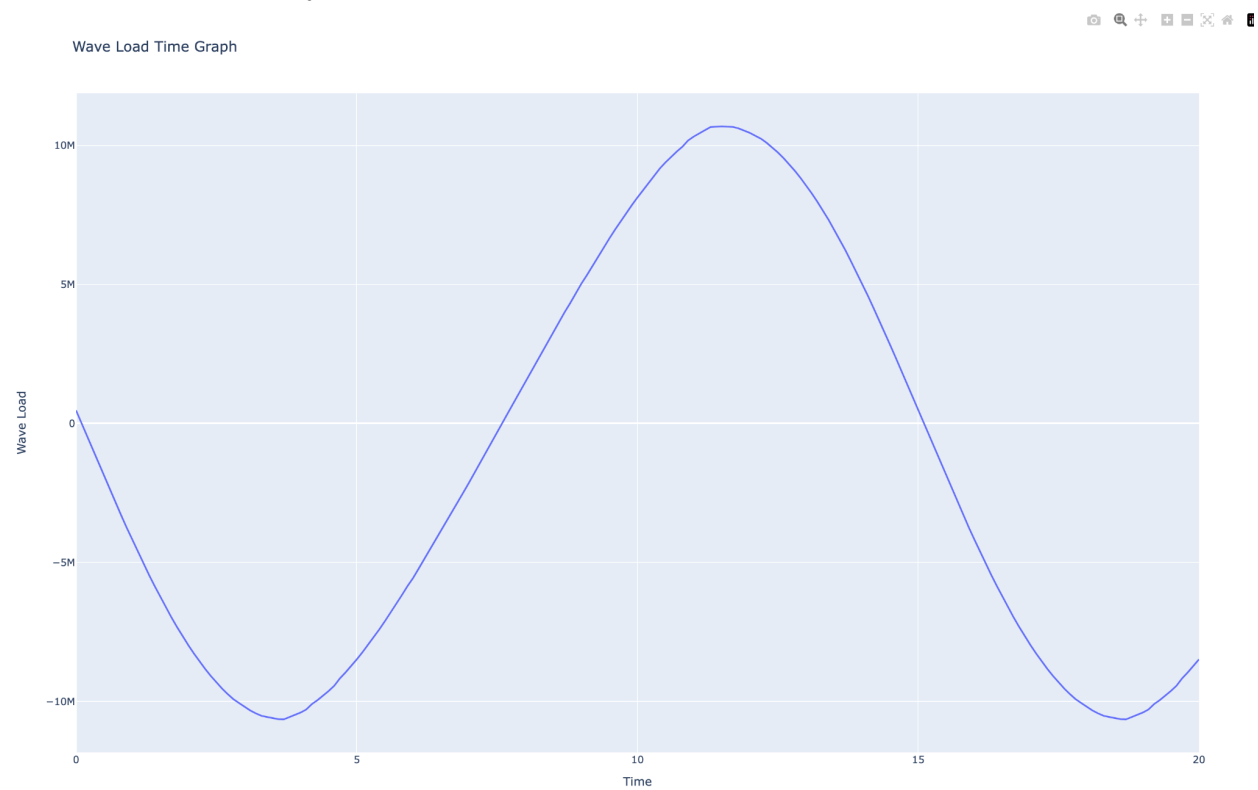


OE4080: Wave Load Assignment (1)

Name: Satyam
Roll. No: NA20B060

Wave Load Time History:



Code:

wave-load.py

```
import numpy as np
import plotly.graph_objs as go
import plotly.offline as pyo

D = 5
h = 90
H = 20
T = 15
g = 9.8
e = 1E-6
Cd = 0.7
Cm = 2
rho = 1025

L0 = 1.56 * T**2
Li = L0 * np.tanh(2 * np.pi * h / L0)
for i in range(1000):
    Lf = L0 * np.tanh(2 * np.pi * h / Li)
    if abs(Li - Lf) < e:
        break
    Li = Lf
L = Lf
print("Calculated Wavelength (L):", L)

z = np.linspace(H * 0.5, -h, 301)
cog = np.zeros(300)
for i in range(len(z) - 1):
    cog[i] = (z[i] + z[i + 1]) * 0.5
fd = Cd * 0.5 * rho * D
fm = Cm * rho * np.pi * D**2 / 4
time = np.linspace(0, 20, 201)
eta = H * 0.5 * np.cos(-2 * np.pi * time / T)
delta_l = cog[0] - cog[1]
u = np.zeros(len(cog))
a = np.zeros(len(cog))
force_seg = np.zeros(len(cog))
F_t = np.zeros(len(time))
k = 2 * np.pi / L

for t in range(len(time)):
    for i in range(len(cog)):
        u_z = (np.pi * H / T) * np.cosh(k * (h + cog[i])) * np.cos(-2
* np.pi * time[t] / T) / np.sinh(k * h)
        a_z = (H / 2) * g * k * np.cosh(k * (h + cog[i])) * np.sin(-2
* np.pi * time[t] / T)
```

```

u0 = np.pi * H * np.cos(-2 * np.pi * time[t] / T) / (T *
np.tanh(k * h))
a0 = (H / 2) * g * k * np.sin(-2 * np.pi * time[t] / T)

if eta[t] >= 0:
    if cog[i] >= 0:
        if cog[i] <= eta[t]:
            u[i] = u0
            a[i] = a0
        else:
            u[i] = 0
            a[i] = 0
    else:
        u[i] = u_z
        a[i] = a_z
else:
    if cog[i] >= eta[t]:
        u[i] = 0
        a[i] = 0
    else:
        u[i] = u_z
        a[i] = a_z
force_seg[i] = fd * u[i] + fm * a[i]

s = np.sum(np.fromiter(((force_seg[j] + force_seg[j + 1]) *
delta_l * 0.5 for j in range(len(force_seg) - 1)), dtype=float))
F_t[t] = s

# Plotting the results using Plotly
trace = go.Scatter(x=time, y=F_t, mode='lines', name='Wave Load')
layout = go.Layout(title='Wave Load Time History',
xaxis=dict(title='Time'), yaxis=dict(title='Wave Load'))
fig = go.Figure(data=[trace], layout=layout)
pyo.plot(fig)

```