Project SRE: Part 2 - Seismic Analysis of a Structure

Student Names: Menno Marissen, Gabriele Mylonopoulos, Bart Slingerland, Sulongge Sulongge, Jerin Thomas

Student ID number:

A	*B*	*C*	*D*	*E*	*F*	*G*
5	3	8	1	8	2	7

INTRODUCTION

In **modules II-III** of the SRE theme, we discussed the dynamic response of SDoF and MDoF systems subjected to ground excitation. Among others, we analysed the linear and non-linear dynamic responses of these systems with various methods applied in seismic design. We also discussed the derivation of linear and non-linear response spectra, the use of theory of SDoF to treat more complex systems (*generalised SDoF systems*), and the pushover method of analysis. We introduced fluid-structure interaction for the design of liquid storage tanks and we discussed soil-structure interaction and its effect on the structural response. In this part of the project, you have the possibility to work on these concepts and to explain the various physical phenomena associated with these effects.

LEARNING OUTCOMES

The learning outcomes of this assignment are that you will be able to:

1. Derive elastic and inelastic response spectra and assess their correctness.
2. Apply the pushover method of analysis.
3. Analyse the dynamic response of a complex structure subjected to seismic excitation.
4. Evaluate and compare the results of different methods of seismic analysis.
5. Consider the dynamic soil-structure interaction (SSI) of a simplified structure.

INSTRUCTIONS & QUESTIONS TO BE ANSWERED

The assignment is divided into three parts. Part A assesses the first learning outcome. Part B assesses learning outcome 2. Part C assesses learning outcomes 3-5.

Note that some of the variables are given in values based on student's ID number.

e.g. Students ID =

A	*B*	*C*	*D*	*E*	*F*	*G*
1	*2*	*3*	*4*	*5*	*6*	*7*

, *G* = *1F0*MPa = *160*MPa

Problem statement

You are part of a specialized team tasked with evaluating the dynamic response of a complex structural system. The structure has already undergone architectural design, and an initial structural analysis has been performed using RFEM. All applied loads must be implemented in accordance with the standards specified in EN 1990 and EN 1998.

The seismological and geotechnical features at the location, which can be used to obtain the design seismic action according to EN1998-1 (Eurocode 8), are given below:

- Geotechnical investigation: $v_{s,30} = 160\ \mathrm{m/s}$ with $N_{SPT} = 10$ (blows per 0.3m), $c_u = 50$ kPa, Poisson ratio $v = 0.33$, and average soil density $\rho = 1800\ \mathrm{kg/m^3}$.
- Seismic hazard analysis in the region:
  - peak ground acceleration of $a_{gR} = 0.33 + 0.0D\ [g]$ with $g = 9.81\ \mathrm{m/s^2}$.
  - maximum (surface) magnitude of the seismic events that are expected to contribute to the hazard is $M_s = 5.0$.
- Importance class: IV.

<u>If some information is missing, please use your engineering judgement to decide on the modelling approach/parameters.</u>

Part A: Derivation of elastic and inelastic site-specific response spectra

1) As part of the team of specialists, you are asked to use five sets of tri-axial ground motions provided and derive the motions in the three principal directions. For each tri-axial set of recordings decompose the two horizontal (as recorded) components in the principal directions and consider the vertical component acting along the third principal direction. Scale the derived horizontal ones such that the peak ground acceleration (PGA) has a value of $PGA_{hor} = a_{gR}\,\gamma_I$. Scale the vertical component according to the instructions in the Python file distributed.

You are asked to:

a) Plot the elastic acceleration response spectra of the scaled signals as defined above along each principal direction for a damping ratio of $\xi = 0.04C$ and for periods between $T = 0.00$ sec and $T = 4.00$ sec[1].

b) Derive the elastic response spectra (ERS) in the three principal directions corresponding to the mean value of the elastic spectra defined in question (1a). Explain the differences between the obtained ERS (mean value) and the ones of EN1998-1 (scaled to the same PGA).[2]

---

[1] For the derivation of the elastic response spectrum, you may use the Newmark beta integration method (Python function created for this module theme which can be found in the folder: 'Python for Project Work SRE_Part 2\ERS', or any other scientifically correct method.

[2] For question 1, you are advised to use the Python file 'ERS.ipynb' located in: 'Python for Project Work SRE_Part 2\ERS\'.

2) The design team is interested in the inelastic response spectra:
   a) Derive the $R_y - \mu - T$ relationship, based on the Newmark and Hall (1982) formulation for ductility equal to $\mu = \max\{1.7D;\ 1.7F\}$. Plot the inelastic acceleration response spectra for the $R_y - \mu - T$ relationship above, using the mean ERS as computed along the two principal horizontal directions in question (1b).

   b) Derive the <u>exact</u> constant ductility inelastic acceleration response spectra[3], for ductility equal to $\mu = \max\{1.7D;\ 1.7F\}$, of the five signals in question (1) when decomposed along the two principal (horizontal) directions, and compute the mean inelastic acceleration response spectra. Compare these with the ones derived by using the simplified $R_y - \mu - T$ relationship of Newmark-Hall (1982). What differences do you observe and why?

Part B: Pushover method of analysis

3) To exploit the non-linearity of the structure, you have been asked to evaluate the seismic capacity by means of a pushover method of analysis. Model the structure in the FE software *RFEM* by employing a non-linear material model (you are free to define the non-linear model of your choice but please make reasonable assumptions). Please answer the following questions:

   a) State what assumptions the pushover method of analysis entails, specifically for the situation in Groningen. Are there any concerns as to the applicability of the pushover method for the induced earthquakes in Groningen?

   b) Check the displacement demand against the displacement capacity of the structure for $a_g = a_{gR}\, \gamma_I\ [g]$ using the pushover method of analysis and EN1998-1 (Eurocode 8). Choose a 'modal' lateral load pattern[4] in your analysis and justify your choice.
      - In the case that the structure does not satisfy the design requirements, specify the acceleration value a_g for which these are met. Describe shortly which measures you would consider to make the structure meet the seismic demand (no need for calculations).
      - In the case that the structure does satisfy the demand, specify for which a_g the structure would fail.

   c) In accordance with EN1998 (see footnote 4), at least both a 'uniform' and a 'modal' lateral load pattern should be applied per principal direction. Provide a reasoned argument as to whether the use of an additional load pattern is necessary (or not) in your case.

[3] You are advised to use the Python code that has been shared for this particular question. The function can be found in the folder 'Python for Project Work SRE_Part 2\IERS'. The Matlab files shared during the classes are also applicable with some minor modifications. Alternatively, you are free to use any other software or code available.

[4] See NEN-EN1998-1:2005 en – 4.3.3.4.2.2

Part C: Seismic analysis of the structure

Please answer the following questions:

4) Perform a dynamic time history analysis with one set of input accelerations derived in question (1) along the principal directions (you are free to choose any set of triaxial motions from the list provided). Determine the maximum displacements and stresses as well as their critical locations. Explain why the chosen locations are considered critical.[5] Conclude as to the seismic capacity of the structure.

5) Perform the response spectrum method of analysis (RSA) using the design spectra provided in Eurocode (EN1998-1) and report the following results:[6]
   a) Plot the first five eigenmodes of the system together with their correspondent eigenfrequencies. Derive the modal participation mass for each of the five modes (in each direction). What do you observe as to the (potential) contribution of the different modes to the final response of the system?
   b) Determine the maximum displacements and stresses at the critical sections. Please substantiate your choice as to the modal combinations rules used for: i) the structural modes; and ii) the different directions of the seismic input motion. Are the critical cross sections the same as the ones found in question 4 above?
   c) How did you choose the upper limit of modes to be considered in the final response of the system? Are the results sensitive to this choice and why? Please substantiate your answer.
   d) Conclude as the seismic capacity of the structure.

6) In your analysis above, the effects of dynamic Soil-Structure Interaction (SSI) were neglected.
   a) Please perform a RSA assuming that the structure is not rigidly connected the ground but is supported by a (thick) rigid foundation block (feel free to define some preliminary dimensions yourself) on top of distributed elastic springs. Feel free to derive the values of the spring stiffness (in the various directions) based on any reasonable assumptions and the given soil information at site and use your solid engineering judgement. Compare the dynamic response to the one of question (5) above and assess any differences.
   b) Please perform a dynamic analysis (in the frequency domain) by considering the horizontal motion in the major principal direction and a generalised SDoF system that represents (adequately) your structure. You may assume a foundation block that is rigid at all support points of your structure (please choose the dimensions of the latter yourself as in (a) above) and that the soil is a homogeneous half-space with the properties defined earlier. Explain any differences you observe compared to the same case analysed without dynamic SSI.

---

[5] Since the current version of RFEM has a bug concerning using the import from .xlsx button, you must directly copy the data from the generated .xlsx into the 'User-Defined' **acceleration** of RFEM.

[6] Since the current version of RFEM has a bug concerning using the import from .xlsx button, you must directly copy the data from the generated .xlsx into the 'User-Defined' **response spectrum** of RFEM.

RESOURCES

For answering the questions in this assignment you are free to use any software you like. However, it is (strongly) advised to use Python and RFEM software together with the provided files distributed in the assignment description folder.

PRODUCTS

The final product of this assignment is a report which you submit individually before the deadline mentioned below. Specifications on the report layout are given below. The report will count for **60%** of the final grade of this part of the module.

ASSESSMENT CRITERIA

The length of the report is irrelevant for your final mark; the grade is based solely on the content of the report. The weight of each part is specified below:

Question Nr.	Points (0-10)	Percentile contribution to final score (%)	Final score per question
1		5%	
2		5%	
3		20%	
4		20%	
5		20%	
6		30%	
Sum		**100%**	
Grade (1-10)			

SUPERVISION AND HELP

Feedback to the assignment will be given in two specific time moments:

- **Feedback on work in progress**: Every week, a student assistant will be available to provide feedback on the progress and help you with the assignment. You can post your questions on the discussion forum in the Brightspace page of the CIEM5220.
- **Feedback on the final report**: you will receive feedback when the report will be checked on Brightspace. You will have the opportunity to discuss your questions after the announcement of the grade.

SUBMISSION FORMAT
Submission deadline June 22, 2025 at 23:59 o'clock. Your report should be submitted electronically as a single pdf-file and the file name should be of the following form: "CIEM5220_SRE_PW2_2025_[Your Names+Your student ID numbers]". Please submit this assignment specification form together with your final report as a single pdf-file **(please compile all documents into a single pdf-file)** directly on Brightspace under the correspondent folder allocated for the specific assignment. Do not forget to fill in your names and Student ID numbers at the front page of this assignment specification form. Clearly indicate the ID number you used for this assignment by filling in the blocks at the front page. When you work out the assignment in groups and you prepare a single report at the end, please upload only ONE assignment (per group) on Brightspace. The solution must be either handwritten or typed (word, LateX, etc.), then converted to PDF-format and submitted as defined above.

Report - Part 1 - ERS Notebook

June 23, 2025

[ ]:
```
import numpy as np
import matplotlib.pyplot as plt
import os
from matplotlib import gridspec
from mpl_toolkits.axes_grid1 import make_axes_locatable
import matplotlib.pyplot as plt
import scipy.io as sio
from scipy.io import loadmat
import random
import pandas as pd
import ERS_functions as ERS
```

1 Section 1

(The first block contains student paramenters, the second block contains the ERS Script. Answers and student code follow)

[2]:
```
#!!!! Read note in docstring below about re-RUNNING This script !!!!

# Defining parameters

student_number = 5381827
student_number = list(str(student_number))
for i in range(len(student_number)):
    student_number[i] = int(student_number[i])
A, B, C, D, E, F, G = student_number


#g = 9.81                      # Gravitational acceleration [m/s^2]
xi = 0.040 + C * 1e-3          # Damping ratio []
a_g_r = (0.33 + D * 1e-2)     # Peak ground acceleration [g]
gamma = 1.4                    # Importance factor [] - Importance class IV␣
↪(assigned) corresponds to gamma=1.4 according to EN1998-1-1 (par 4.2.5)
a_g_ref = a_g_r*gamma          # Horizontal Reference Acceleration (EN1998-1)
a_vg_ref = a_g_ref*0.45        # Vertical Reference Acceleration (EN1998-1) -␣
↪Type 2 Earthquake (M_s = 5.0, assigned parameter)
```

```
# Note: Input accelerations are in [g], the following block converts the signal␣
↪to [m/s^2] and then saves it to the Signal xlsx files.
#       All functions from the ERS module are documented to run in [g] units,␣
↪but data is read from xlsx signal files, therefore we input accelerations in␣
↪[m/s] and output spectra in [m/s] (i.e. no difference for the operation of␣
↪functions)
```

[ ]:
```
'''
(c) TU Delft
This script will analyse all the accelerograms (x,y,z) direction and
construct their ERS based on a numerical calculation. The mean of the
horizontal and vertical acting accelerograms will be calculated as well
and will be saved to a .mat file. Furthermore, the signals will be
made RFEM ready and placed in a folder.

!!!!!!!!!!!!!!!!!!!!!!
Please note that every time
this file runs, it will add the signal data to the existing files in
"Signals", therefore make sure that if you re-run this file you
delete those files.

At some locations you will find '<---- fill in equation', please fill in the
appropriate equation here to continue.
'''

# generate the random number to choose specific files
files = random.sample(range(8),5)
print('Chosen files are: ' + str(files))

## importing acclerograms and looping over them to create the ERS

# Determine scaling value:
gamma = gamma                           # (defined in parameter section)␣
↪<------ fill in equation here
PGA = a_g_r*gamma                       # (defined in parameter section)␣
↪<------ fill in equation here


# initialising loop
N = len(files)
SA_el_hor = np.zeros((201,2*N))
SA_el_ver = np.zeros((201,N))
ag_all = [None] * (3*N)
NPTS_all = np.zeros((N,))
Dt_all = np.zeros((N,))

## loops for reading files and decomposing in principal directions
```

```
jj = 0
for ifile in files:
    NPTS, Dt, ag_time = ERS.fnc_read_induced_signals(ifile)
    for ii in range(3):
        ag_all[3*jj+ii] = ag_time[ii]
    NPTS_all[jj] = NPTS
    Dt_all[jj] = Dt
    jj += 1

for iN in range(N):
    # get the file locations
    hor1 = 3*iN + 1
    hor2 = 3*iN + 2
    ver = 3*iN + 3

    ag_1, ag_2 = ERS.fnc_calc_covariance(ag_all[hor1-1], ag_all[hor2-1],␣
 ↪ag_all[ver-1], Dt_all[iN])
    ag_all[hor1-1] = ag_1 # store 1st principal hor. dir.
    ag_all[hor2-1] = ag_2 # store 2nd principal hor. dir.


# loop for calculation response spectra and saving decomposed signals such that␣
 ↪ZeusNL is able to use them

PGA_init = [None] * (3*N)

for iN in range(1, N+1):
    # get the file locations
    hor1 = 3*(iN-1) + 0
    hor2 = 3*(iN-1) + 1
    ver = 3*(iN-1) + 2
    for ifile in [hor1, hor2, ver]:
        # read the signal
        ag_time = ag_all[ifile]
        NPTS = NPTS_all[iN-1]
        Dt = Dt_all[iN-1]

        # In order to scale down the time history signal to a predefined PGA we
        # have to:
        #   1) Find the initial PGA (absolute value) of the signal named after
        #       "PGA_init" below;
        #   2) We scale the signal such that PGA = PGA_max in case of the first␣
 ↪and second principal hor. dir.
        #       and the vertical one is scaled with the same ratio as the first␣
 ↪principal hor. dir.
```

```python
        PGA_init[ifile] = np.max(np.abs(ag_time))/(100*9.81)    #  (data is in␣
↪cm/s^2) - convert to g

        if ifile != ver:
            ag_time = PGA/PGA_init[ifile] * ag_time / (100*9.81)
        else:
            ag_time = PGA/PGA_init[hor1] * ag_time / (100*9.81)

        ## calculate the elastic response spectrum
        xi = xi                           # (defined in parameter section)␣
↪<------ fill in equation here

        SA_el, Fel  =  ERS.fnc_Elastic_Response_Spectrum(ag_time, Dt, NPTS, xi)

        # determine in which array it should be stored
        if ifile == hor1:
            loc = 2*(iN-1) + 1
            name = 'hor1'
            SA_el_hor[:, loc-1] = SA_el
            ag_hor1 = ag_time * 9.81 # convert back to m/s2 for RFEM
        elif ifile == hor2:
            loc = 2*(iN-1) + 2
            name = 'hor2'
            SA_el_hor[:, loc-1] = SA_el
            ag_hor2 = ag_time * 9.81 # convert back to m/s2 for RFEM
        elif ifile == ver:
            loc = (iN-1) + 1
            name = 'ver'
            SA_el_ver[:, loc-1] = SA_el
            ag_vert = ag_time * 9.81 # convert back to m/s2 for RFEM

        ## Use the earthquake signal data to generate signals for ZeusNL for␣
↪question 4
        t = np.linspace(0, len(ag_time)*Dt, len(ag_time))  # create time vector
        signal = np.vstack((t, ag_time)).T  # create signal matrix

        # save the file
        filename = f"Signal-{iN}-{name}.txt"
        savepath = 'Signals'
        os.makedirs(savepath, exist_ok=True)
        fileID = open(os.path.join(savepath, filename), 'at')
        np.savetxt(fileID, signal, fmt='%.6f')
        fileID.close()  # Close the file.

    # Also save all signals in .xlsx format for RFEM
    df = pd.DataFrame({
        'Time [s]': t,
```

```
        'Horizontal 1 [m/s^2]': ag_hor1,
        'Horizontal 2 [m/s^2]': ag_hor2,
        'Vertical [m/s^2]': ag_vert,
    })

    # Save to .xlsx
    savepath = 'Signals'
    os.makedirs(savepath, exist_ok=True)
    filename = f"Signal-Set-{iN}.xlsx"
    df.to_excel(os.path.join(savepath, filename), index=False)


# create period vector
Tn = np.arange(0, 4.02, 0.02)

SA_el_hor1_mean = np.mean(SA_el_hor[:,0::2], axis=1) # mean value of principal␣
↪hor. spectra
SA_el_hor2_mean = np.mean(SA_el_hor[:,1::2], axis=1) # mean value of hor.␣
↪spectra
SA_el_ver_mean = np.mean(SA_el_ver, axis=1) # same as above but vertical

# Convert SA to m/s²
SA_el_hor1_mean_ms2 = SA_el_hor1_mean * 9.81
SA_el_hor2_mean_ms2 = SA_el_hor2_mean * 9.81
SA_el_ver_mean_ms2  = SA_el_ver_mean  * 9.81

# Frequency vector (avoid divide by zero)
f = np.zeros_like(Tn)
f[1:] = 1 / Tn[1:]  # skip Tn=0

# Save directory
savepath = 'ERS_Means'
os.makedirs(savepath, exist_ok=True)

# Helper to save a mean spectrum
def save_spectrum_to_excel(filename, SA_values):
    df = pd.DataFrame({
        'Tn': Tn,
        'f': f,
        'SA': SA_values
    })
    df.to_excel(os.path.join(savepath, filename), index=False, header=True)

# Save each
save_spectrum_to_excel("SA_el_hor1_mean.xlsx", SA_el_hor1_mean_ms2)
save_spectrum_to_excel("SA_el_hor2_mean.xlsx", SA_el_hor2_mean_ms2)
save_spectrum_to_excel("SA_el_ver_mean.xlsx",  SA_el_ver_mean_ms2)
```

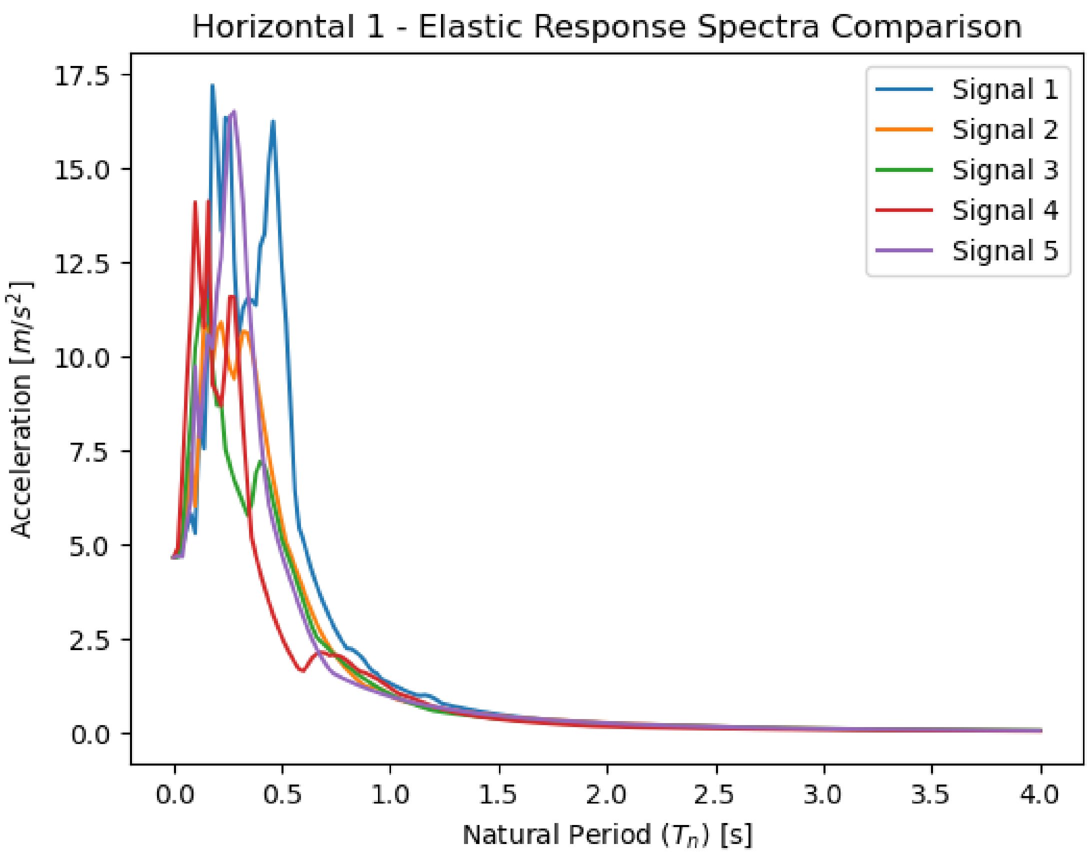
```
```

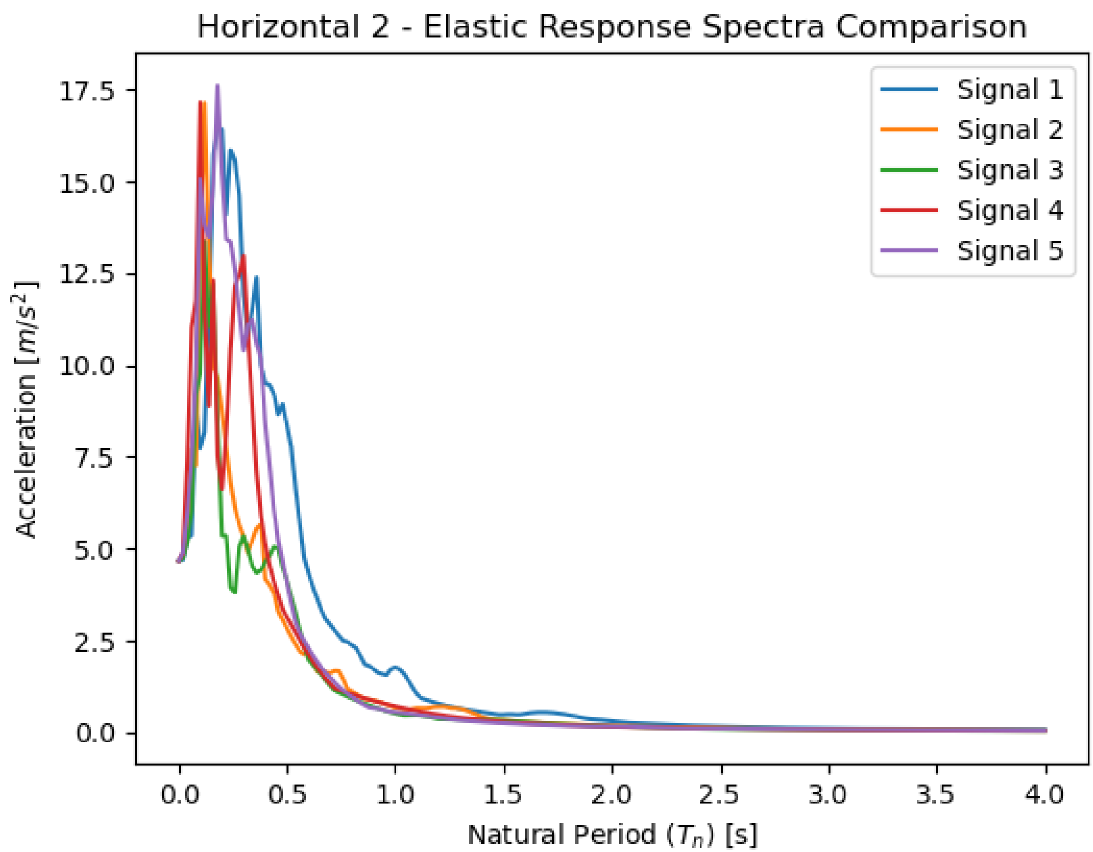
```
Chosen files are: [2, 5, 4, 7, 3]
```

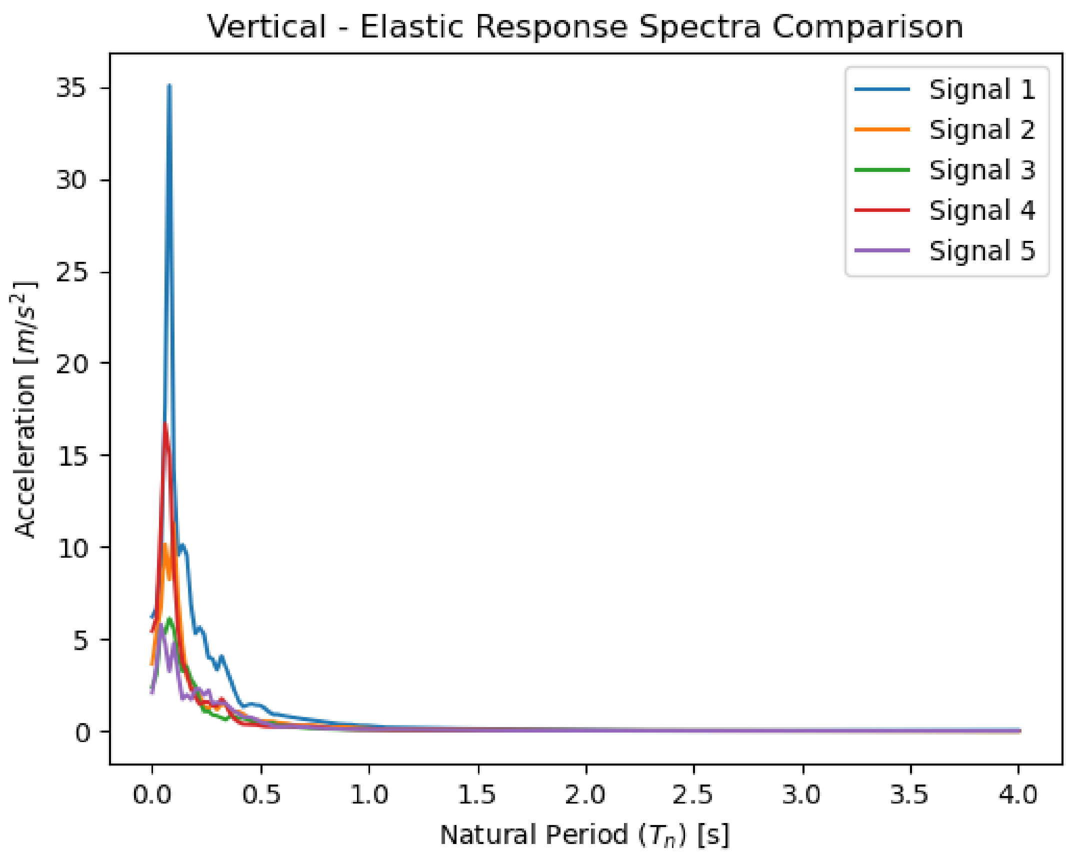
2 Section 1a:

[3]:
```
df_signals = [pd.read_excel(f'Signals/Signal-Set-{i}.xlsx', engine='openpyxl')␣
 ↪for i in range(1,6)]

time_vecs = [i.iloc[:,0] for i in df_signals]      # Time vectors (all equal in␣
 ↪principle) - Matrix of dimension 5 (number of signals) x 3440 (number of␣
 ↪samples)
acc_hor1_vecs = [i.iloc[:,1] for i in df_signals]  # Horizontal-1 acceleration␣
 ↪vectors - Matrix of dimension 5 (number of signals) x 3440 (number of␣
 ↪samples)
acc_hor2_vecs = [i.iloc[:,2] for i in df_signals]  # Horizontal-2 acceleration␣
 ↪vectors - Matrix of dimension 5 (number of signals) x 3440 (number of␣
 ↪samples)
acc_ver_vecs = [i.iloc[:,3] for i in df_signals]   # Vertical acceleration␣
 ↪vectors - Matrix of dimension 5 (number of signals) x 3440 (number of␣
 ↪samples)

SA_el_hor1_vecs = []
for time, acc in zip(time_vecs,acc_hor1_vecs):
    assert len(time)==len(acc)
    Dt = time[1]-time[0]
    NPTS = len(time)
    SA_el, _ = ERS.fnc_Elastic_Response_Spectrum(acc, Dt, NPTS, xi)
    SA_el_hor1_vecs.append (SA_el)

SA_el_hor2_vecs = []
for time, acc in zip(time_vecs,acc_hor2_vecs):
    assert len(time)==len(acc)
    Dt = time[1]-time[0]
    NPTS = len(time)
    SA_el, _ = ERS.fnc_Elastic_Response_Spectrum(acc, Dt, NPTS, xi)
    SA_el_hor2_vecs.append (SA_el)

SA_el_ver_vecs = []
for time, acc in zip(time_vecs,acc_ver_vecs):
    assert len(time)==len(acc)
    Dt = time[1]-time[0]
    NPTS = len(time)
    SA_el, _ = ERS.fnc_Elastic_Response_Spectrum(acc, Dt, NPTS, xi)
    SA_el_ver_vecs.append (SA_el)

# create period vector
```

```python
Tn = np.arange(0, 4.02, 0.02)

# Plots
# Horizontal 1 ERS
for i,SA_el in enumerate(SA_el_hor1_vecs,1):
    plt.plot(Tn, SA_el, label=f"Signal {i}")
plt.title("Horizontal 1 - Elastic Response Spectra Comparison")
plt.xlabel(r"Natural Period ($T_n$) [s]")
plt.ylabel(r"Acceleration [$m/s^2$]")
plt.legend()
plt.show()

# Horizontal 2 ERS
for i,SA_el in enumerate(SA_el_hor2_vecs,1):
    plt.plot(Tn, SA_el, label=f"Signal {i}")
plt.title("Horizontal 2 - Elastic Response Spectra Comparison")
plt.xlabel(r"Natural Period ($T_n$) [s]")
plt.ylabel(r"Acceleration [$m/s^2$]")
plt.legend()
plt.show()

# Vertical ERS
for i,SA_el in enumerate(SA_el_ver_vecs,1):
    plt.plot(Tn, SA_el, label=f"Signal {i}")
plt.title("Vertical - Elastic Response Spectra Comparison")
plt.xlabel(r"Natural Period ($T_n$) [s]")
plt.ylabel(r"Acceleration [$m/s^2$]")
# plt.axvline(0.065,0,1)
plt.legend()
plt.show()
```


Horizontal 1 - Elastic Response Spectra Comparison
Acceleration [m/s^2]
Natural Period (T_n) [s]
Signal 1
Signal 2
Signal 3
Signal 4
Signal 5
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0


Horizontal 2 - Elastic Response Spectra Comparison
Signal 1
Signal 2
Signal 3
Signal 4
Signal 5
Acceleration [m/s^2]
Natural Period (T_n) [s]
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0



[4]:
```
# Check for correct scaling (safetycheck)

print("\nHorizontal 1:")
for i in range(5):
    print(f"Signal {i+1}: {np.max(np.abs(acc_hor1_vecs[i]))}")

print("\nHorizontal 2:")
for i in range(5):
    print(f"Signal {i+1}: {np.max(np.abs(acc_hor2_vecs[i]))}")

print("\nVertical:")
for i in range(5):
    print(f"Signal {i+1}: {np.max(np.abs(acc_ver_vecs[i]))}")
# Memo: The vertical direction is scaled with the same factor as the "first␣
↪principal horizontal direction"
```

```
Horizontal 1:
Signal 1: 4.66956
Signal 2: 4.66956
```

```
Signal 3: 4.669560000000001
Signal 4: 4.66956
Signal 5: 4.66956

Horizontal 2:
Signal 1: 4.66956
Signal 2: 4.66956
Signal 3: 4.669560000000001
Signal 4: 4.669560000000001
Signal 5: 4.669560000000001

Vertical:
Signal 1: 6.219005124426918
Signal 2: 3.66223794974537
Signal 3: 2.406385836522875
Signal 4: 5.430044066354665
Signal 5: 2.0849876926035
```

3 Section 1b:

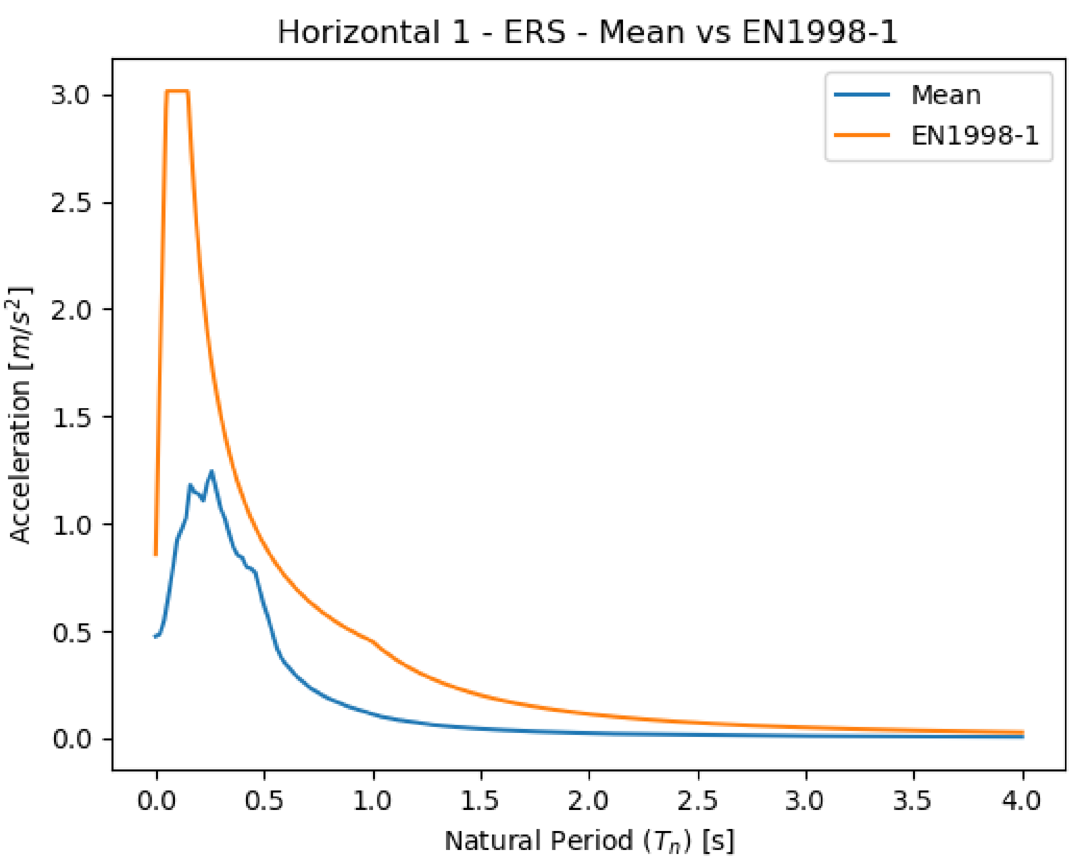
[5]:
```python
# EN1998-1 Spectra

eta = np.sqrt(10/(5+xi))
print(f"Eta {eta:.2f}") #Add eta check
T = np.linspace(0,4,1000)

# Parameters corresponding to Ground Type D for Horizontal Spectrum
S = 1.8
TB = 0.10
TC = 0.30
TD = 1.2

def horizontal_elastic_acc_spectrum(T, eta):
 Se_T_values = []
 for T in T:
   if 0 <= T <= TB:
      Se_T = a_g_ref * S * (1 + (T / TB) * (eta * 2.5 - 1))
      Se_T_values.append(Se_T)
   elif TB < T <= TC:
      Se_T = a_g_ref * S * eta * 2.5
      Se_T_values.append(Se_T)
   elif TC < T <= TD:
      Se_T = a_g_ref * S * eta * 2.5 * (TC / T)
      Se_T_values.append(Se_T)
   elif TD < T <= 4.0:
      Se_T = a_g_ref * S * eta * 2.5 * (TC * TD) / (T ** 2)
      Se_T_values.append(Se_T)
```

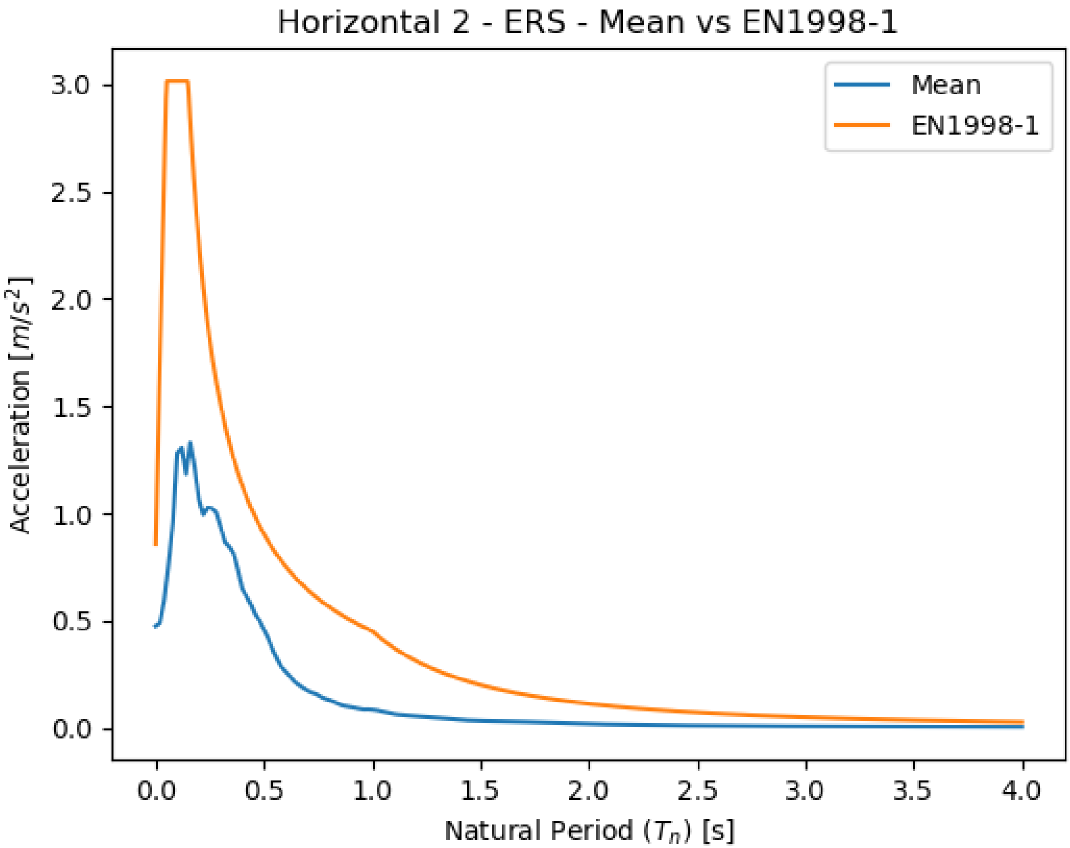
```
    else:
      Se_T = 0
      Se_T_values.append(Se_T)
  return Se_T_values

# Parameters corresponding to Ground Type D for Vertical Spectrum
TB = 0.05
TC = 0.15
TD = 1.0

def vertical_elastic_acceleration_response_spectrum(T, eta):
  Svd_T_values = []
  for T in T:
      if 0 <= T <= TB:
         Svd_T = a_vg_ref * (1 + (T / TB) * (eta * 3.0 - 1))
         Svd_T_values.append(Svd_T)
      elif TB < T <= TC:
         Svd_T = a_vg_ref * eta * 3.0
         Svd_T_values.append(Svd_T)
      elif TC < T <= TD:
         Svd_T = a_vg_ref * eta * 3.0 * (TC / T)
         Svd_T_values.append(Svd_T)
      elif TD < T:
         Svd_T = a_vg_ref * eta * 3.0 * (TC * TD) / (T ** 2)
         Svd_T_values.append(Svd_T)
      else:
         Svd_T = 0
         Svd_T_values.append(Svd_T)
  return Svd_T_values
```

```
Eta 1.41
```

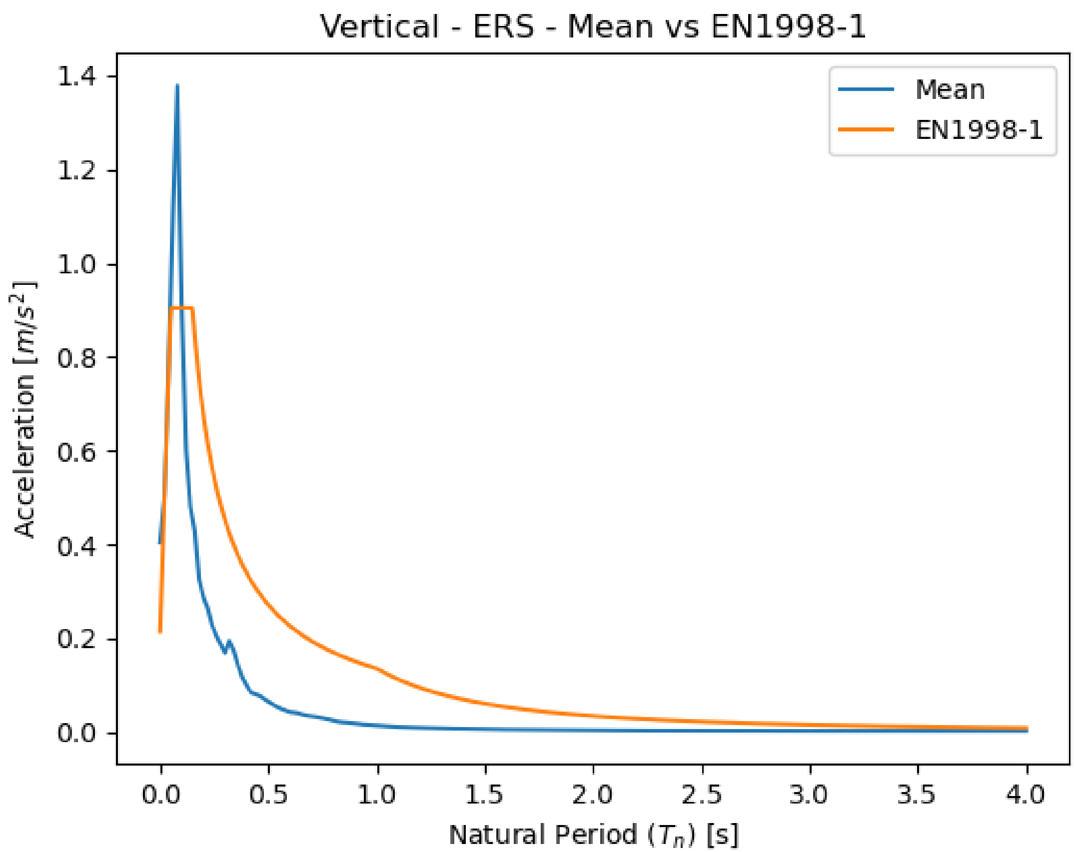
[6]:
```
df_SA_el_hor_1_mean = pd.read_excel('ERS_Means/SA_el_hor1_mean.xlsx',␣
 ↪engine='openpyxl')
df_SA_el_hor_2_mean = pd.read_excel('ERS_Means/SA_el_hor2_mean.xlsx',␣
 ↪engine='openpyxl')
df_SA_el_ver_mean = pd.read_excel('ERS_Means/SA_el_ver_mean.xlsx',␣
 ↪engine='openpyxl')

# Mean accelerations are stored in [m/s^2] in the xlsx files.

plt.plot(df_SA_el_hor_1_mean.iloc[:,0],df_SA_el_hor_1_mean.iloc[:,2]/9.81,␣
 ↪label='Mean')
plt.plot(T,horizontal_elastic_acc_spectrum(T,eta), label='EN1998-1')
plt.title("Horizontal 1 - ERS - Mean vs EN1998-1")
plt.xlabel(r"Natural Period ($T_n$) [s]")
plt.ylabel(r"Acceleration [$m/s^2$]")
```

```
plt.legend()
plt.show()

plt.plot(df_SA_el_hor_2_mean.iloc[:,0],df_SA_el_hor_2_mean.iloc[:,2]/9.81,␣
 ↪label='Mean')
plt.plot(T,horizontal_elastic_acc_spectrum(T,eta), label='EN1998-1')
plt.title("Horizontal 2 - ERS - Mean vs EN1998-1")
plt.xlabel(r"Natural Period ($T_n$) [s]")
plt.ylabel(r"Acceleration [$m/s^2$]")
plt.legend()
plt.show()

plt.plot(df_SA_el_ver_mean.iloc[:,0],df_SA_el_ver_mean.iloc[:,2]/9.81,␣
 ↪label='Mean')
plt.plot(T,vertical_elastic_acceleration_response_spectrum(T,eta),␣
 ↪label='EN1998-1')
plt.title("Vertical - ERS - Mean vs EN1998-1")
plt.xlabel(r"Natural Period ($T_n$) [s]")
plt.ylabel(r"Acceleration [$m/s^2$]")
plt.legend()
plt.show()
```





Horizontal 2 - ERS - Mean vs EN1998-1
Mean
EN1998-1
Acceleration [m/s^2]
Natural Period (T_n) [s]
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0

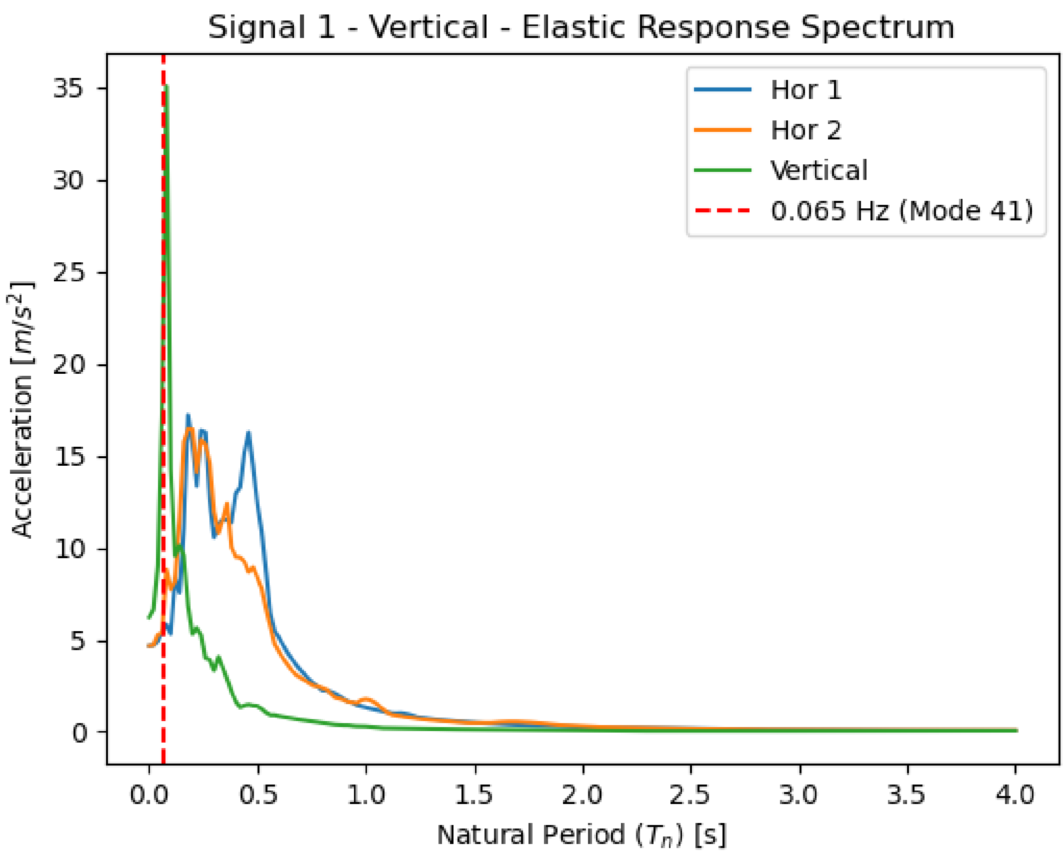


The first difference between the Mean Spectra and the EN1998-1 Spectra is conceptual: the former is the result of a limited amount of seismic recordings, while the latter is an empirical curve derived from probabilistic seismic hazard analysis. As such, the first may be more accurate in describing the seismic behavior of a specific region (assuming all signals are part of a somewhat coherent dataset) while the second provides a safe estimate of excitations based on a limited amount of assumptions (i.e. surface-wave magnitude and a gross ground characterization).

It is apparent how the Mean Spectra show different behavior for each principal direction (since it is based on observations), whereas EN1998-1 prescribes the use of the same curve for both horizontal principal directions. As it should be expected, the code version of the ERS returns higher horizontal excitations compared to the ones derived from the recorded signals. Since the signals hereby examined concern the seismic activity in Groningen (from script documentation), it is not surprising to notice the Mean Spectrum exceed the EN1998-1 Spectrum in the last plot. The vertical acceleration component is in fact frequently dominant in human-induced Earthquakes, whereas it is usually secondary in natural Earthquakes. The curve given by EN1998-1 is therefore not calibrated to capture these specific anthropic phenomena.

3.0.1 Extra Plot for Section 3

[ ]:
```python
# Plot to demonstrate extreme deformations for mode 41 of the structure

# Vertical ERS
plt.plot(Tn, SA_el_hor1_vecs[0], label=f"Hor 1")
plt.plot(Tn, SA_el_hor2_vecs[0], label=f"Hor 2")
plt.plot(Tn, SA_el_ver_vecs[0], label=f"Vertical")
plt.title("Signal 1 - Vertical - Elastic Response Spectrum")
plt.xlabel(r"Natural Period ($T_n$) [s]")
plt.ylabel(r"Acceleration [$m/s^2$]")
plt.axvline(0.065,0,1, color='r', label='0.065 s (Mode 41)', ls="--")
plt.legend()
plt.show()
```



Report - Part 2 - IERS Notebook

June 23, 2025

[3]:
```python
import numpy as np
import matplotlib.pyplot as plt
import os
from matplotlib import gridspec
from mpl_toolkits.axes_grid1 import make_axes_locatable
import matplotlib.pyplot as plt
import scipy.io as sio
from scipy.io import loadmat
import pandas as pd

import IERS_functions as IERS
```

1 Section 2

(The first two blocks contain the IERS Script. Answers and student code follow)

[ ]:
```python
# Defining parameters

student_number = 5381827
student_number = list(str(student_number))
for i in range(len(student_number)):
    student_number[i] = int(student_number[i])
A, B, C, D, E, F, G = student_number



#g = 9.81                    # Gravitational acceleration [m/s^2]
xi = 0.040 + C * 1e-3        # Damping ratio []

# Note: Input accelerations are in [g], the following block converts the signal␣
↪to [m/s^2] and then saves it to the Signal xlsx files.
#       All functions from the IERS module are documented to run in [g] units,␣
↪but data is read from xlsx signal files, therefore we input accelerations in␣
↪[m/s] and output spectra in [m/s] (i.e. no difference for the operation of␣
↪functions)
```

[ ]:
```python
'''
---------------------------------------------------------------------------
```

```
(c) TU Delft
This is the main script that will assist the student on how to answer
question 2 of Part 2 of the assignment

It loads the horizontal component of a ground acceleration signal from
the directory: 'Python for Project Work SRE_Part 2/IERS/Signal'. The folder␣
↪Signal
is initially empty. The student has to put there signals from the previous
question.

Following that, the script calculates the elastic response spectrum, the
indirect inelastic response spectrum (by applying the Newmark-Hall (1982)
Ry - mu - Tn relationship) and the direct inelastic response spectrum.

The solver that is used to calculate the response of the elastic and the
elastoplastic SDoF systems is the Newmark beta integration.
The student is asked to complete the script by filling in the lines
indicated by "<----fill in equation here".

You will have to write your own code for plotting purposes.
----------------------------------------------------------------------------
                         initialisation
----------------------------------------------------------------------------

One can choose for 'file' either 1 or 2, etc. , such that a specific signal
will be analysed.

Note that the student may have to make changes to the code in order to answer␣
↪the various questions.
'''

file = 1                                        # <------ fill in equation␣
↪here

FilePath = 'Signal'
FilePath_full = './' + FilePath + '/'
FilePath_contents = os.listdir(FilePath_full)
FileName = FilePath_full + FilePath_contents[file-1]
fileID = open(FileName, 'r')
info = np.loadtxt(fileID)

NPTS = len(info[:,0])
Dt = info[1,0]
ag_time = info[:,1]


'''
```

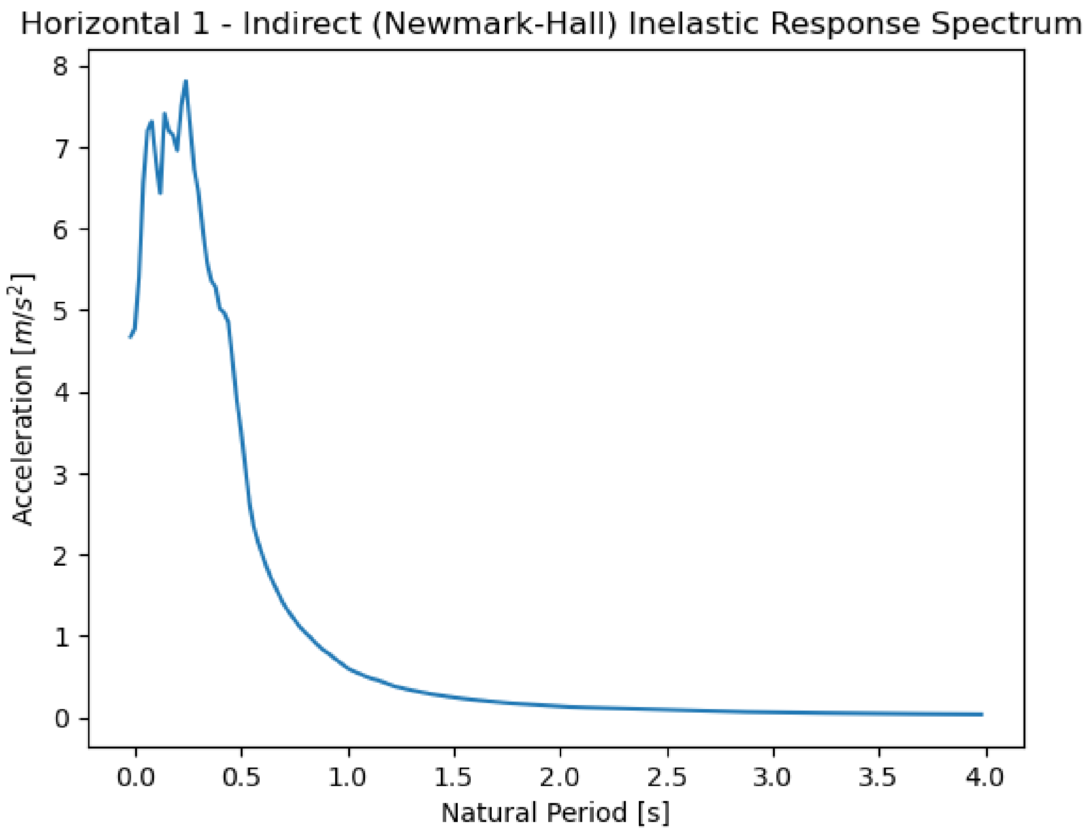
```
Calls the function that calculates the elastic response spectrum.
The student is advised to open the function and have a look at it.
The student has to specify the damping ratio "xi" based on Assignment 3
instructions.
'''
xi = xi                                          # <------ fill in equation here
SA_el,F_el =  IERS.fnc_Elastic_Response_Spectrum(ag_time,Dt,NPTS,xi)

'''
-------------------------------------------------------------------------
                         Question 2
-------------------------------------------------------------------------
Here the student is asked to calculate the constant ductilty inelastic
acceleration spectra in two ways:
    1) indirect way, by applying the Newmark-Hall (1982) Ry - mu - Tn
    relationship on the Elastic acceleration response spectrum
    2) direct way, following the steps of the iterative procedure
First of all the student has to specify the "mu"
'''
mu = np.max([1.7+0.01*D, 1.7+0.01*F])                         # <------ fill in␣
 ↪equation here

'''
-- Indirect inelastic acceleration response spectrum --
In the following lines fill in the
quantities of the branched of Ry that are missing (Newmark-Hall Ry - mu - Tn␣
 ↪relationship (1982))

Please create the function that defines the value of Ry
'''

# initialisation
Tn = np.zeros(201)
Ry = np.zeros(201)

for i in range(201):
    Tn[i] = (i-1) * 0.02
    if Tn[i] <= 0.05:
        Ry[i] = 1                         # <------ fill in equation here
    elif Tn[i] > 0.05 and Tn[i] <= 0.12:
        Ry[i] = 1 + ((np.sqrt(2*mu-1)-1)/(0.12-0.05))*(Tn[i]-0.05)   # linearly␣
 ↪interpolate between 0.05-12s
    elif Tn[i] > 0.12 and Tn[i] <= 0.5:
        Ry[i] = np.sqrt(2*mu-1)
    elif Tn[i] > 0.5 and Tn[i] <= 1:
```

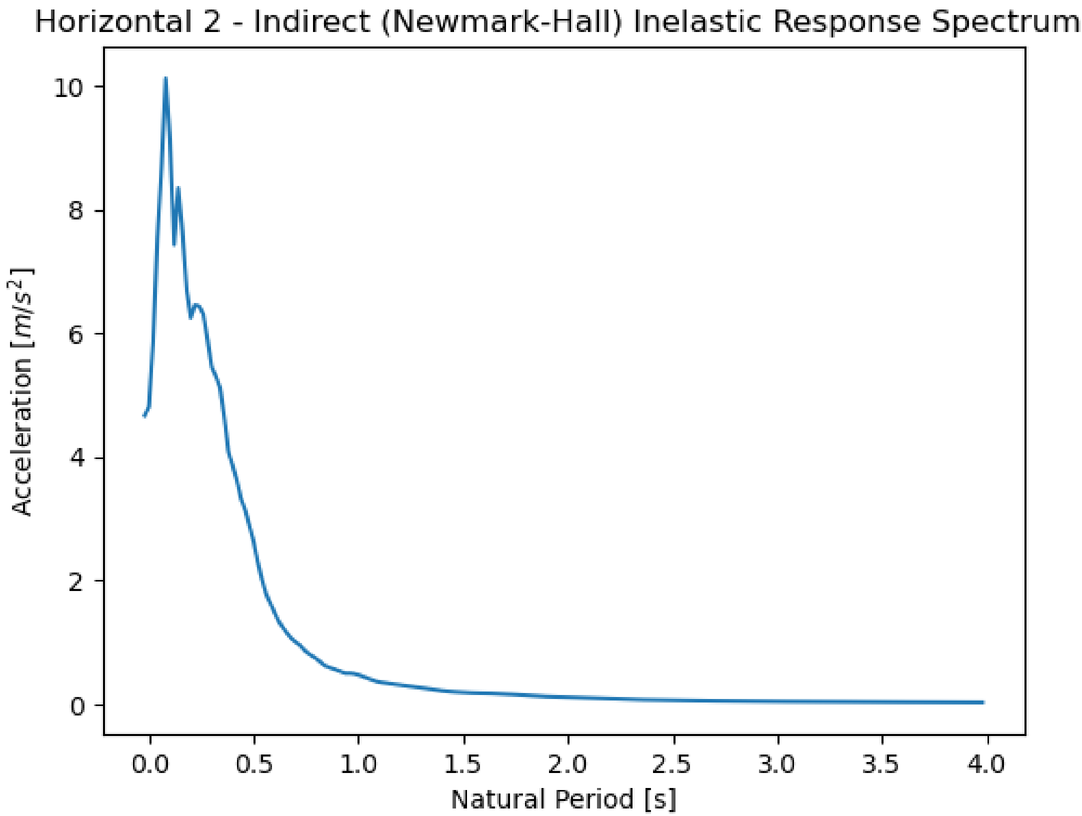
```
        Ry[i] = np.sqrt(2*mu-1) + ((mu - np.sqrt(2*mu-1))/(1-0.5))*(Tn[i]-0.5) ␣
↪ # (a linear interpolation was chosen for this segment as well) <------ fill␣
↪in equation here
    elif Tn[i] > 1:
        Ry[i] = mu                   # <------ fill in equation here

# Calculation of indirect inelastic acceleration response spectrum as a
# function of the elastic response spectrum derived previously.
SA_in_ind = np.zeros(201)

for i in range(201):
    SA_in_ind[i] = SA_el[i]/Ry[i]                # <------ fill in equation␣
↪here

'''
-- Direct inelastic acceleration response spectrum
The following Python function calculates the direct inelastic response spectrum␣
↪for the chosen time history
'''
SA_in_dir =  IERS.fnc_Direct_Inel_Resp_Spec(ag_time,Dt,NPTS,xi,mu,SA_el,F_el);
```

2 Section 2a:

[20]:
```
# Directly Import Mean ERS as previously computed
# Note: Mean accelerations are stored in [m/s^2] in the xlsx files.
df_SA_el_hor1_mean = pd.read_excel('ERS_Means/SA_el_hor1_mean.xlsx',␣
↪engine='openpyxl')
df_SA_el_hor2_mean = pd.read_excel('ERS_Means/SA_el_hor2_mean.xlsx',␣
↪engine='openpyxl')
df_SA_el_ver_mean = pd.read_excel('ERS_Means/SA_el_ver_mean.xlsx',␣
↪engine='openpyxl')

# Create variable from dataframe
SA_el_hor1_mean = df_SA_el_hor1_mean.iloc[:,2]
SA_el_hor2_mean = df_SA_el_hor2_mean.iloc[:,2]

# Compute reduced spectra
SA_in_ind_hor1_mean = np.zeros(201)
SA_in_ind_hor2_mean = np.zeros(201)
for i in range(201):
    SA_in_ind_hor1_mean[i] = SA_el_hor1_mean[i] / Ry[i]
    SA_in_ind_hor2_mean[i] = SA_el_hor2_mean[i] / Ry[i]

plt.plot(Tn,SA_in_ind_hor1_mean)
plt.title("Horizontal 1 - Indirect (Newmark-Hall) Inelastic Response Spectrum ")
plt.xlabel(r'Natural Period [s]')
```

```
plt.ylabel(r'Acceleration [$m/s^2$]')
plt.show()

plt.plot(Tn,SA_in_ind_hor2_mean)
plt.title("Horizontal 2 - Indirect (Newmark-Hall) Inelastic Response Spectrum ")
plt.xlabel(r'Natural Period [s]')
plt.ylabel(r'Acceleration [$m/s^2$]')
plt.show()
```





3 Section 2b:

```
[ ]: # Import signals from xlsx as a vector containing multiple dataframes
df_signals = [pd.read_excel(f'Signal/Signal-Set-{i}.xlsx', engine='openpyxl')␣
 ↪for i in range(1,6)]

# Create vectors containing data from dataframes
time_vecs = [i.iloc[:,0] for i in df_signals]      # Time vectors (all equal in␣
 ↪principle) - Matrix of dimension 5 (number of signals) x 3440 (number of␣
 ↪samples)
acc_hor1_vecs = [i.iloc[:,1] for i in df_signals]  # Horizontal-1 acceleration␣
 ↪vectors - Matrix of dimension 5 (number of signals) x 3440 (number of␣
 ↪samples)
acc_hor2_vecs = [i.iloc[:,2] for i in df_signals]  # Horizontal-2 acceleration␣
 ↪vectors - Matrix of dimension 5 (number of signals) x 3440 (number of␣
 ↪samples)
acc_ver_vecs = [i.iloc[:,3] for i in df_signals]   # Vertical acceleration␣
 ↪vectors - Matrix of dimension 5 (number of signals) x 3440 (number of␣
 ↪samples)
```

```
# Set mu for Inelastic Spectrum calculation
mu = np.max([1.7+0.01*D, 1.7+0.01*F])  #(redefined here for code readability) ␣
 ↪

# Horizontal-1 direction
SA_el_hor1_vecs = []
SA_in_dir_hor1_vecs = []
for time, acc in zip(time_vecs,acc_hor1_vecs):
    assert len(time)==len(acc)
    Dt = time[1]-time[0]
    NPTS = len(time)
    #Compute elastic spectrum
    SA_el, _ = IERS.fnc_Elastic_Response_Spectrum(acc, Dt, NPTS, xi)
    SA_el_hor1_vecs.append (SA_el)
    #Compute inelastic spectrum with direct method
    SA_in_dir = IERS.fnc_Direct_Inel_Resp_Spec(acc,Dt,NPTS,xi,mu,SA_el,F_el);
    SA_in_dir_hor1_vecs.append(SA_in_dir)

SA_in_dir_hor1_mean = np.mean(SA_in_dir_hor1_vecs, axis=0)

# Horizontal-2 direction
SA_el_hor2_vecs = []
SA_in_dir_hor2_vecs = []
for time, acc in zip(time_vecs,acc_hor2_vecs):
    assert len(time)==len(acc)
    Dt = time[1]-time[0]
    NPTS = len(time)
    #Compute elastic spectrum
    SA_el, _ = IERS.fnc_Elastic_Response_Spectrum(acc, Dt, NPTS, xi)
    SA_el_hor2_vecs.append (SA_el)
    #Compute inelastic spectrum with direct method
    SA_in_dir = IERS.fnc_Direct_Inel_Resp_Spec(acc,Dt,NPTS,xi,mu,SA_el,F_el);
    SA_in_dir_hor2_vecs.append(SA_in_dir)

SA_in_dir_hor2_mean = np.mean(SA_in_dir_hor2_vecs, axis=0)

# Vertical direction
SA_el_ver_vecs = []
SA_in_dir_ver_vecs = []
for time, acc in zip(time_vecs,acc_ver_vecs):
    assert len(time)==len(acc)
    Dt = time[1]-time[0]
    NPTS = len(time)
    #Compute elastic spectrum
    SA_el, _ = IERS.fnc_Elastic_Response_Spectrum(acc, Dt, NPTS, xi)
    SA_el_ver_vecs.append (SA_el)
```

```
    #Compute inelastic spectrum with direct method
    SA_in_dir = IERS.fnc_Direct_Inel_Resp_Spec(acc,Dt,NPTS,xi,mu,SA_el,F_el);
    SA_in_dir_ver_vecs.append(SA_in_dir)

SA_in_dir_ver_mean = np.mean(SA_in_dir_ver_vecs, axis=0)
```

[10]:

```
# Save processed data to avoid long running times
import pickle

if not os.path.exists("SA_in_dir_vec_stored.pkl"):
    SA_in_dir_vec_stored = [
        SA_in_dir_hor1_vecs,
        SA_in_dir_hor2_vecs,
        SA_in_dir_ver_vecs
    ]
    pickle.dump(SA_in_dir_vec_stored, open("SA_in_dir_vec_stored.pkl", "wb"))

if not os.path.exists("SA_in_dir_mean_stored.pkl"):
    SA_in_dir_mean_stored = [
        SA_in_dir_hor1_mean,
        SA_in_dir_hor2_mean,
        SA_in_dir_ver_mean
    ]
    pickle.dump(SA_in_dir_mean_stored, open("SA_in_dir_mean_stored.pkl", "wb"))
```

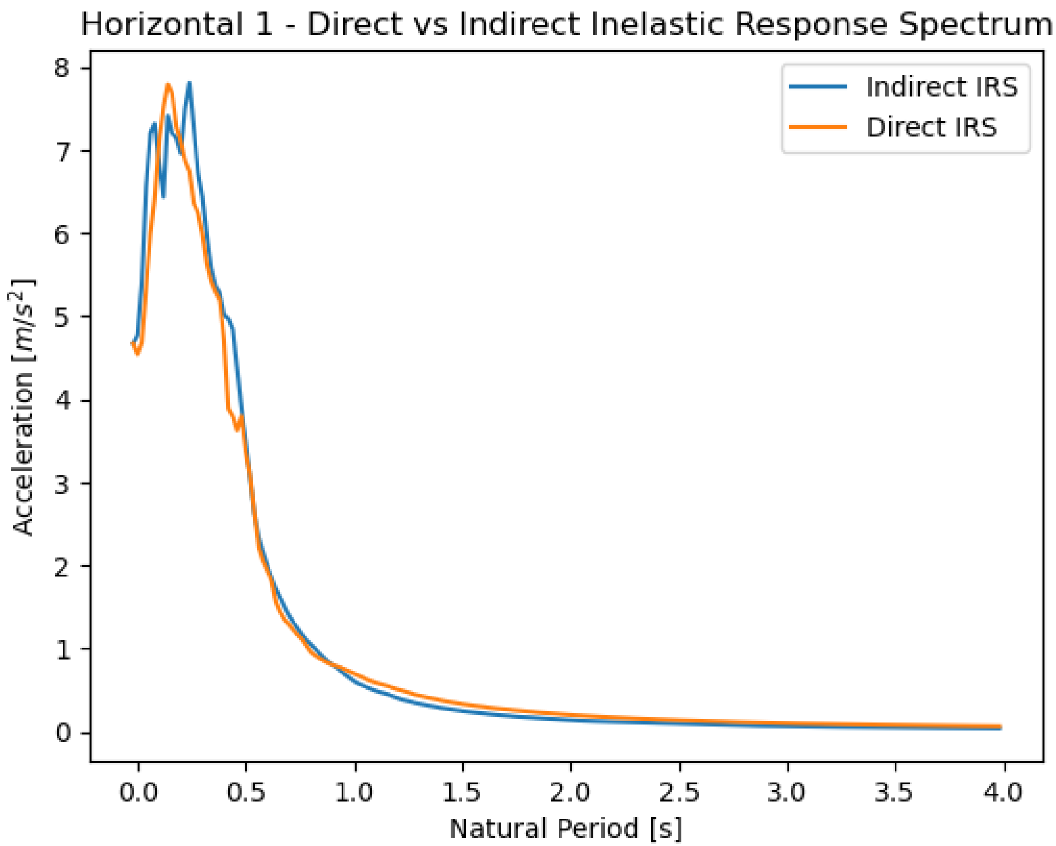
[15]:

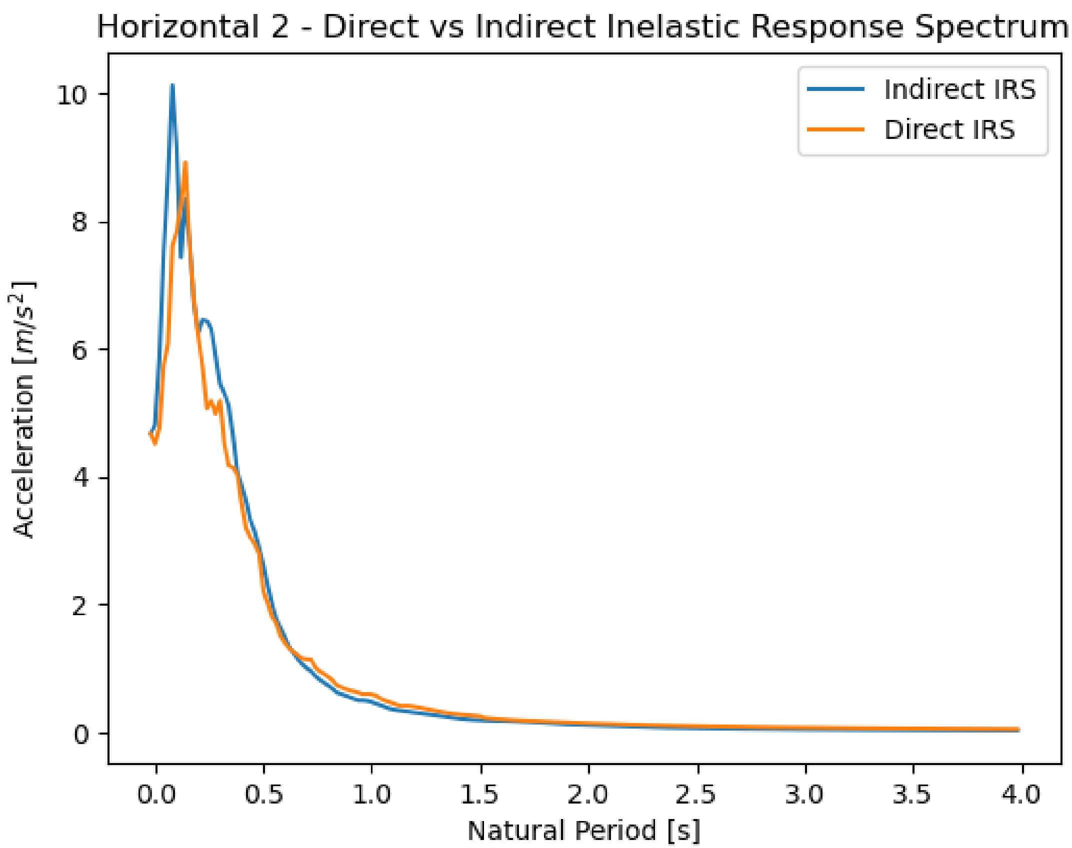
```
# Load processed data from pickle to avoid long running times
# SA_in_dir_vec_store  = pickle.load(open("SA_in_dir_vec_stored.pkl","rb"))
# SA_in_dir_hor1_vecs = SA_in_dir_vec_store[0]
# SA_in_dir_hor2_vecs = SA_in_dir_vec_store[1]
# SA_in_dir_ver_vecs  = SA_in_dir_vec_store[2]

SA_in_dir_mean_store = pickle.load(open("SA_in_dir_mean_stored.pkl","rb"))
SA_in_dir_hor1_mean = SA_in_dir_mean_store[0]
SA_in_dir_hor2_mean = SA_in_dir_mean_store[1]
SA_in_dir_ver_mean  = SA_in_dir_mean_store[2]
```

[21]:

```
# Plots

# Horizontal-1
plt.plot(Tn,SA_in_ind_hor1_mean, label='Indirect IRS')
plt.plot(Tn,SA_in_dir_hor1_mean, label='Direct IRS')
plt.title('Horizontal 1 - Direct vs Indirect Inelastic Response Spectrum')
plt.xlabel('Natural Period [s]')
plt.ylabel(r'Acceleration [$m/s^2$]')
plt.legend()
plt.show()
```

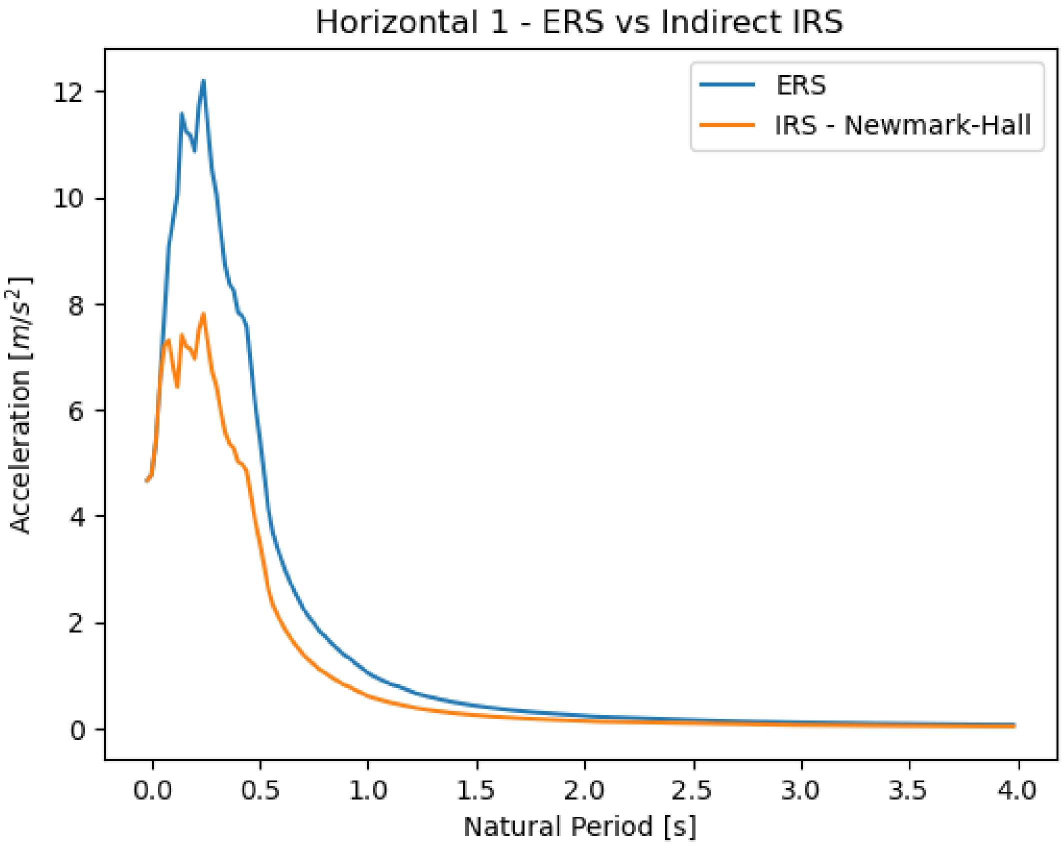
```python
# Horizontal-2
plt.plot(Tn,SA_in_ind_hor2_mean, label='Indirect IRS')
plt.plot(Tn,SA_in_dir_hor2_mean, label='Direct IRS')
plt.title('Horizontal 2 - Direct vs Indirect Inelastic Response Spectrum')
plt.xlabel('Natural Period [s]')
plt.ylabel(r'Acceleration [$m/s^2$]')
plt.legend()
plt.show()
```





The two approaches lead to comparable results: as expected, the accelerations similar in magnitude but the curves features differ slightly. The indirect method scales the Elastic Response Spectrum according to the piece-wise Newmark-Hall simplified relation for R_y, while the direct method iteratively computes the exact R_y producing an exact constant ductility spectrum. While the overall shape is similar, the two spectra show different (local) troughs and peaks. This is due to R_y being a function of the natural period of the structure for both approaches. It is worth noting that the reduced spectra show a (marginally) different profile compared to each other but also compared to the input ERS (which would be the case if the scaling was linear).

[ ]:
```
# EXTRA - Comparison between ERS-IRS for Indirect Method

# Horizontal-1
plt.plot(Tn,SA_el_hor1_mean, label='ERS')
plt.plot(Tn,SA_in_ind_hor1_mean, label='IRS - Newmark-Hall')
plt.title('Horizontal 1 - ERS vs Indirect IRS')
plt.xlabel('Natural Period [s]')
plt.ylabel(r'Acceleration [$m/s^2$]')
plt.legend()
plt.show()
```



Horizontal 1 - ERS vs Indirect IRS
ERS
IRS - Newmark-Hall
Acceleration [m/s^2]
Natural Period [s]
0
2
4
6
8
10
12
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0

In [78]:
```
import numpy as np
import matplotlib.pyplot as plt
import os
from matplotlib import gridspec
from mpl_toolkits.axes_grid1 import make_axes_locatable
import matplotlib.pyplot as plt
import scipy.io as sio
from scipy.io import loadmat
import random
import pandas as pd
```

In [79]:
```
# Defining parameters

student_number = 5381827
student_number = list(str(student_number))
for i in range(len(student_number)):
    student_number[i] = int(student_number[i])
A, B, C, D, E, F, G = student_number



g = 9.81                        # Gravitational acceleration [m/s^2]
xi = 0.040 + C * 1e-3           # Damping ratio []
PGA = (0.33 + D * 1e-2)         # Peak ground acceleration [g]
gamma = 1.4                     # Importance factor [] - Importance class IV (assigned)
a_g_ref = PGA*gamma             # Horizontal Reference acceleration [g] - Type 2 Earthq
a_vg_ref = a_g_ref*0.45         # Vertical Reference acceleration [g] - Type 2 Earthqua

print(f"Damping ratio: {xi:.3}")
print(f"Peak Ground Acceleration (excluding importance factor): {PGA*g} m/s^2")
print(f"Horizontal Reference Acceleration for EN1998-1: {a_g_ref*g} m/s^2")
print(f"Vertical Reference Acceleration for EN1998-1: {a_vg_ref*g} m/s^2")
```

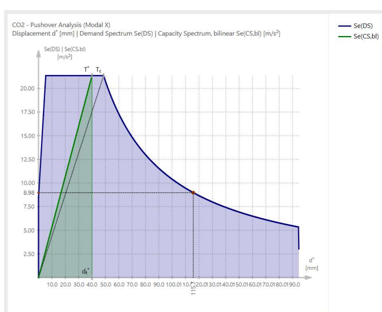
```
Damping ratio: 0.048
Peak Ground Acceleration (excluding importance factor): 3.3354000000000004 m/s^2
Horizontal Reference Acceleration for EN1998-1: 4.66956 m/s^2
Vertical Reference Acceleration for EN1998-1: 2.101302 m/s^2
```

Section 3.a

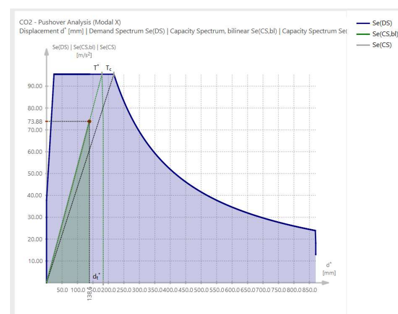
The pushover method of analysis is based on the imposition of a displacement pattern based either on uniform or modal deformed state. The standard operational procedure given by Eurocode 1998-1 begins with the derivation of the elastic spectrum based on a limited number of parameters that aim at predicting both Earthquake excitation and soil behavior. The first potential inconsistency when dealing with the Groningen seismic activity is that vertical accelerations are dominant in artificially induced Earthquakes, whereas the code assumes lateral excitations to be the critical design case (as can be seen in the coefficient table at 3.2.2.3 of EN1998-1). This particular feature is also relevant for the appropriate choice of mode to be used in the application of the modal displacement pattern. The pushover method relies on the ability of the single selected mode to describe the structural response to the excitation. In the case of the structure of interest the most critical loading condition is expected to be caused by lateral excitations, well represented by mode 1 and 2. These modes are however not able to represent properly the vertical component which is expected to be dominant in an Earthquake from the Groningen region. More generally, such considerations are especially crucial for structures whose stiffness relies on self-weight (i.e. a masonry structure with a disconnected roof). Another feature that may be relevant depending on the structure under analysis is that artificially induced earthquakes are characterized by high-energy release at higher frequencies, which may be a determining factor for the appropriate selection of modal pattern and is particularly important for short-period structures.

Section 3.b

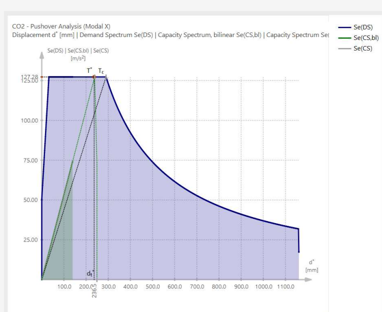
Capacity vs Demand- Iterative procedure



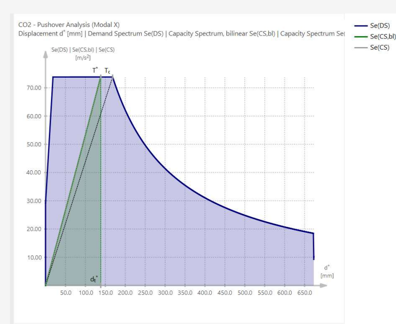
Capacity vs Demand at given agR



Capacity vs Demand at agR=10

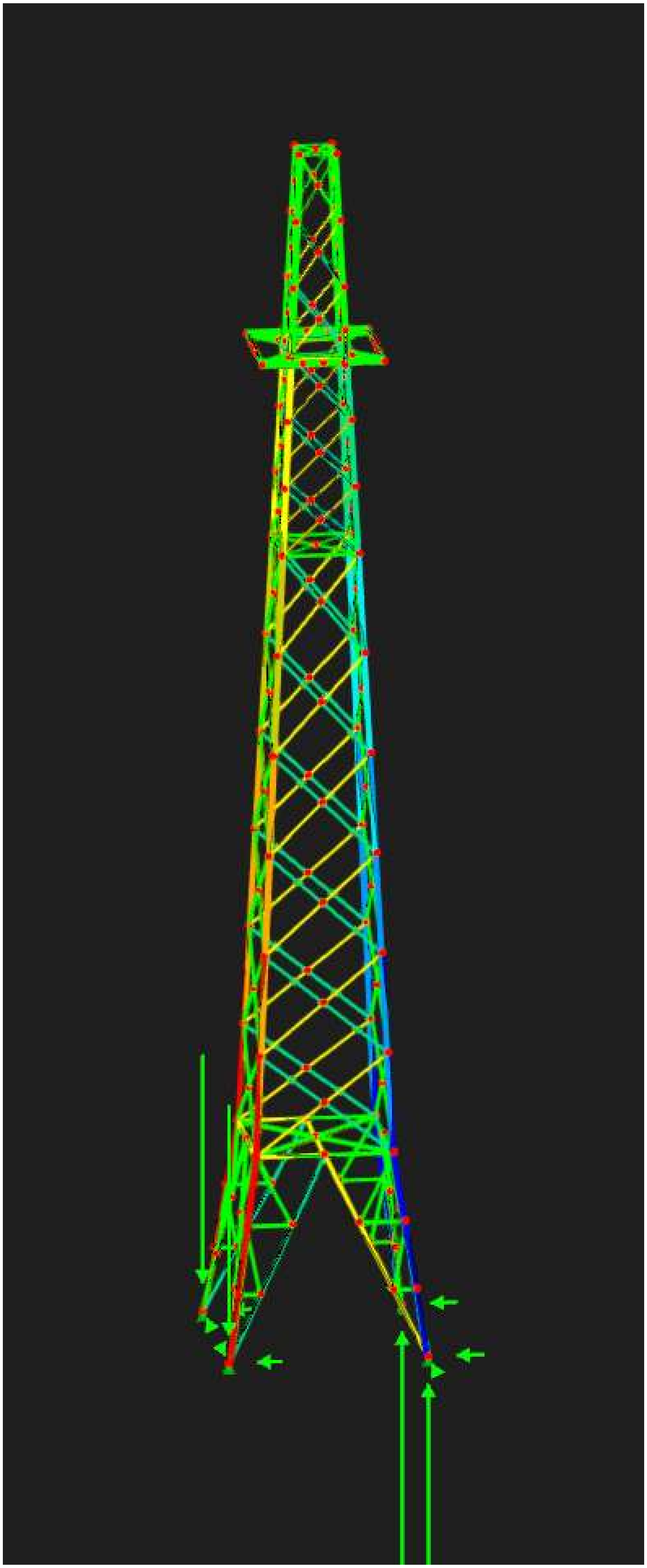


Capacity vs Demand at agR=15



Capacity vs Demand at agR=11.6

Our structure is able to withstand the acceleration imposed by EN1998-1 standard within its elastic regime. According to the simulation no plastic response is developed within failure.So different iterations were done by adjusting the seismic demand to higher values by adjusting the PGA. Upon further research, the conclusion that a more refined model would be necessary in order to develop a plastic phase in the capacity spectrum was reached. I would in fact be necessary to include a model of the connections in order to correctly reproduce structural collapse, whereas the current model can deform indefinitely. This was the interpretation given to the following diagram, where Von Mises equivalent stresses are plotted.



Section 3.c

According to EN1998-1-2005 section 4.3.3.2.1 a single mode approximation is a viable option when the period of the chosen mode of vibration satisfies the following conditions:

$$T_n < \begin{cases} 4 * T_C \\ 2s \end{cases}$$

In our design scenario $T_C = 0.3s$ and $4 * T_C = 1.2s$. Mode 2 was chosen for the analysis in X direction and respects the above mentioned condition with $T_2 = 0.27s$. Mode 1 was chosen for the analysis in Y direction with $T_1 = 0.27s$. No further load combinations are therefore deemed necessary. On a more qualitative level, Mode 1 and 2 present a modal shape that maximize displacement at the top of the structure in the main directions. The corresponding load patterns are considered an appropriate choice in maximizing the overturning moment, which is expected to be the most critical action for a cantilever beam structure. Finally, the relatively high Effective Modal Mass Factor of Mode 1 and 2 set at 0.44 (see table in Section 5 Part A) supports such decision, especially since a single modal shape needs to be selected in this analysis.

Section 4 - Time History Analysis

A Time History Analysis was carried out as part of the seismic assessment of the structure. Signal 1 was chosen from previously processed data and the model was subjected to the derived accelerations in the three principal directions.
As part of this analysis a limitation of the RFEM software was encountered: the Stress-Analysis module did not support nonlinear material behavior for members. As a consequence the analysis was carried out in both bilinear and linear regimes, but stresses were computed for linear regime. It is therefore implicitly assumed that the linear behavior of the structure is sufficient to identify critical stresses location.

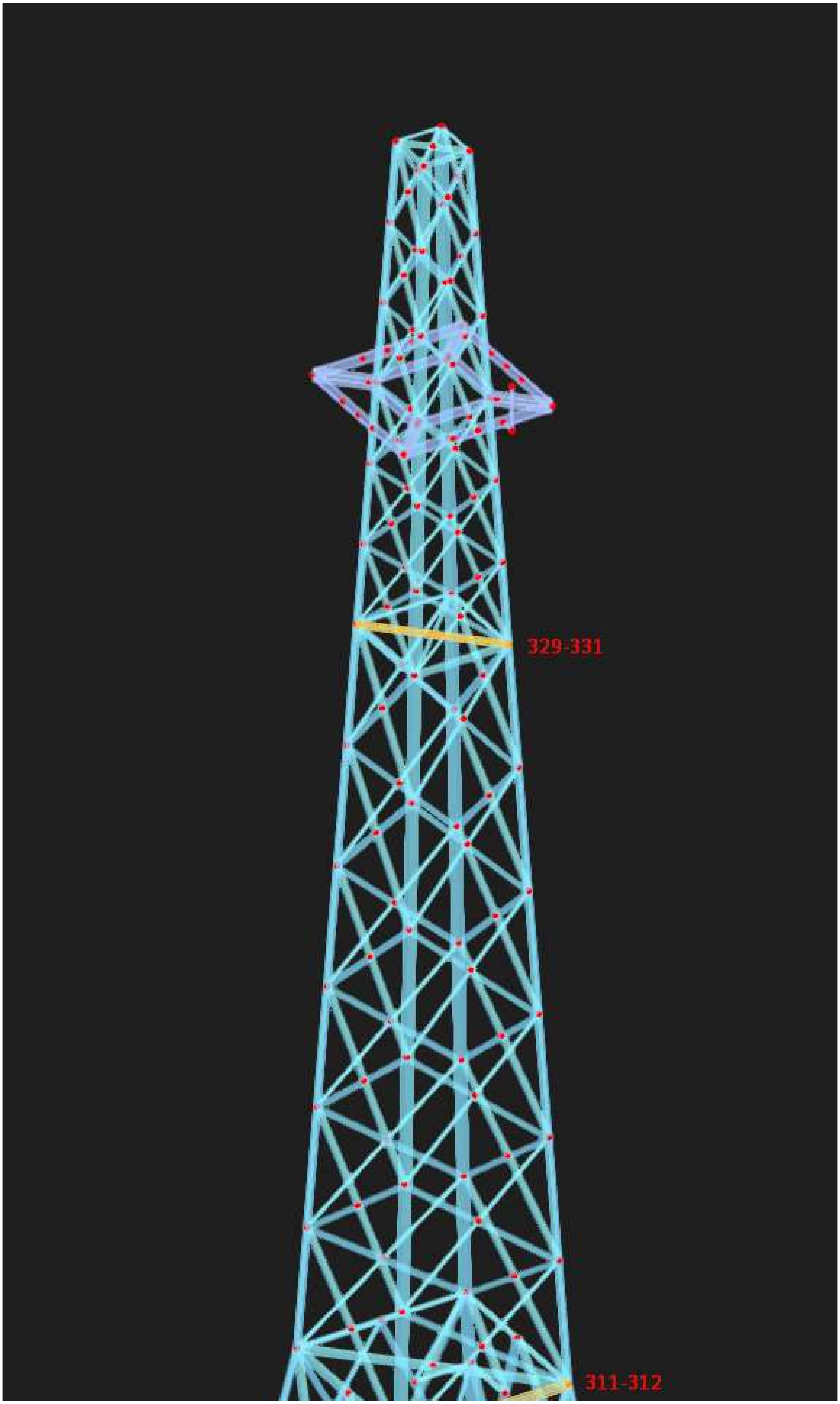
The comparison of **maximum displacements** yielded the following results:

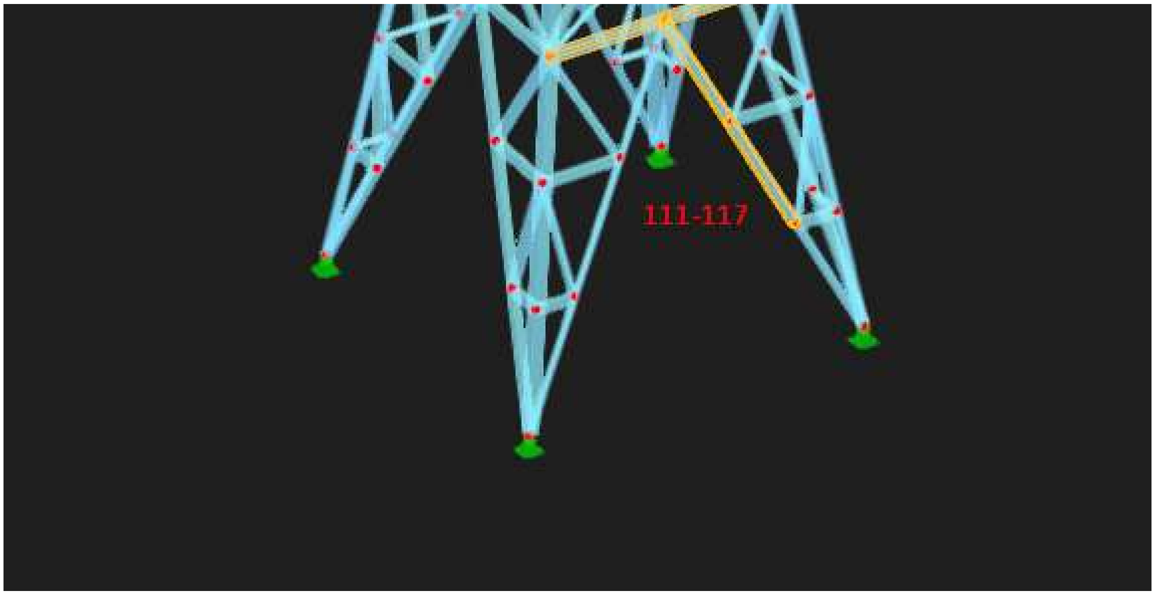
DOF	Member	Material Bilinear	Material Linear	Description
X	313	-83.5 mm	-83.3 mm	Top of the tower
Y	365	87.0 mm	86.3 mm	Antenna on mounting bracket
Z	329	-19.3 mm	17.9 mm	Horizontal Bracing (2nd order)
ϕ_x	365	41.7 mrad	41.0 mrad	Antenna on mounting bracket

DOF	Member	Material Bilinear	Material Linear	Description
ϕ_y	329	-18.4 mrad	17.5 mrad	Horizontal Bracing (2nd order)
ϕ_z	119	-70.3 mrad	-70.6 mrad	Strut in basal truss (lowest)

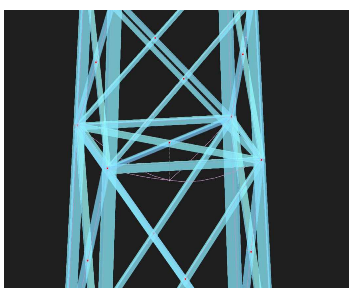
High-stress locations (most relevant):

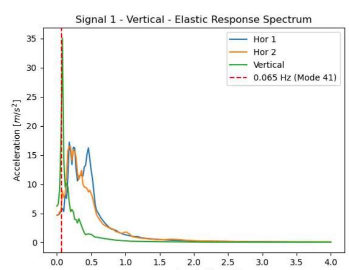
Member No.	Location x [m]	Stress Existing [N/mm2]	Stress Limit [N/mm2]	Stress Ratio η [-]
111	1.874	200.340	235.000	0.853
117	0.000	201.047	235.000	0.856
311	1.707	211.905	235.000	0.902
312	0.000	215.914	235.000	0.919
329	0.000	258.318	235.000	1.099
331	1.354	258.399	235.000	1.100


329-331
311-312



Displayed stresses were computed with Von Mises equivalent stresses criterion. Comparing stresses with yielding stress values for the chosen material shows that most critical excitation is located at the interface between Members 329-331. At such locations the computed stresses due to the ground acceleration provided by the timeseries (Signal 1) input in the model exceed the limit stress of the material causing failure of the structure. This finding is consistent with Spectral and Modal analysis results. Mode 41, which excites precisely members 329-331 as shown in the following figure, has in fact a period of $0.065s$ which closely matches the peak frequency of the vertical component of the Elastic Response Spectrum computed for the chosen signal.

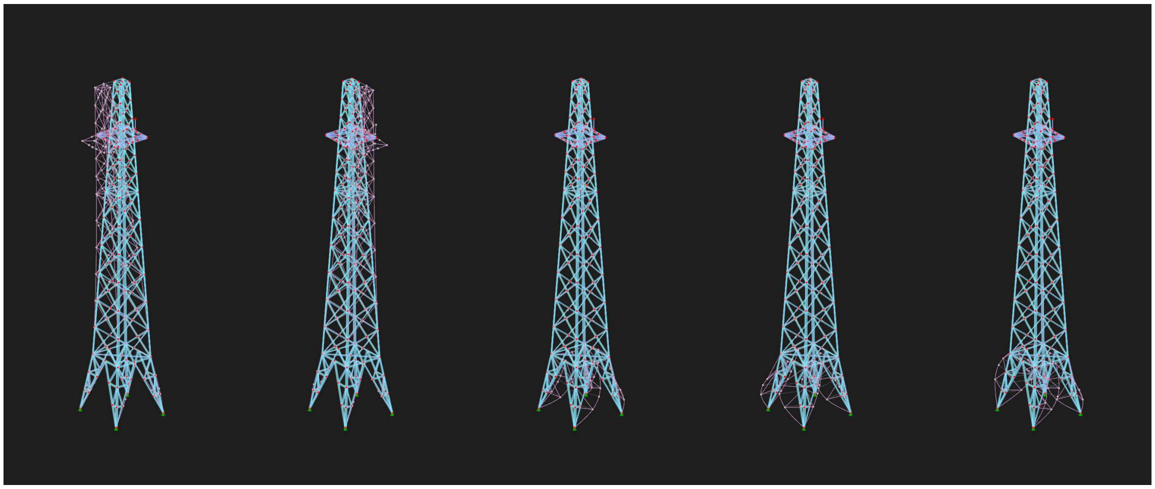




In conclusion, the results of the Time Series Analyisis would hint towards a vulnerability of the structure to vertical accelerations and therefore inadequacy of the design to withstand seismic load. It is however worth pointing out that the peak stress observed in members 329-331 is the result of a resonance between the spectral component located at 15.38 Hz in Signal 1 and mode 41 of the structure. Given that Signal 1 shows an unusually high peak when compared to the other signals that were taken into consideration, this result may require further examination of the validity and/or statistical relevance of Signal 1 and the design may not need further refinement. If the signal were to be considered relevant, the possible options to mitigate the problem would include the use of stronger profiles for members 329-331 and/or damping devices that would target displacements at the intersection of member 329-331. In both cases the analysis should be repeated as both options may in principle affect the dynamic behavior of the structure, potentially relocating the critical section.

Section 5 - Response Spectrum Analysis

Part A



Mode Nr	NF	Modal Mass	EMM X	EMM Y	EMM Z	EMM ϕ_x	EMM ϕ_y	EMM ϕ_z	Factor X	Factor Y
1	3.667	1683.0	89.4	3848.9	0.0	364106.00	7104.35	0.00	0.010	0.440
2	3.668	1682.2	3849.0	89.4	0.0	7102.31	364118.00	0.01	0.440	0.010
3	5.839	449.9	428.6	217.7	0.0	28487.50	41881.80	0.00	0.049	0.025
4	5.839	450.0	217.7	428.5	0.0	41887.40	28488.40	0.00	0.025	0.049
5	5.855	544.5	0.0	0.0	0.0	0.00	0.00	0.00	0.000	0.000
28	12.026	0	0	0	0.4	0.16	10369	0	0	0
55	18.897	0	0.5	0.4	18.92	0.03	10792.4	0	0	0
84	25.518	0.2	0	3684.2	0.61	16.97	13.18	0	0	0.421

Inspecting the results of the modal analysis for 500 modes, the most relevant modes are mode 1 and 2 with a modal mass factor of 0.44 in for translational y and x direction respectively. These modes are also relevant to describe the global bending behavior of the truss structure with a modal mass factor of 0.51 in rotational x-direction and y-direction respectively. The similarity in properties between the two modes is to be expected as a result of the highly symmetric geometry of the structure. Mode 3 and 4 concern deformations of the basal supports of the truss, however their contribution is secondarily relevant to describe the dynamic behavior of the structure and may be arguably excluded from the analysis. In fact, EN1998-1 recommends a threshold of 0.05 for the participation ratio in the selection of modes as part of the truncation process, however 0.049 is hereby regarded as a relatively high contribution for a single mode. Mode 5 is interpreted as a numerical artifact and therefore irrelevant. The first 5 modes are highly inadequate to describe the deformation of the structure in vertical and torsional directions (translation and rotation about z). The most relevant mode for the former direction would be mode 84 (with a mass participation ratio (z) of 0.421) while modes 28 and 55 are the main contributors for the latter direction (counter-clockwise and clockwise torsion) with mass participation ratio of 0.339 and 0.349 respectively.

Part B

As mentioned in relation to Time History Analysis, due to RFEM features, maximum displacements were computed for both linear and nonlinear material, while stress-analysis was carried out for linear material only.

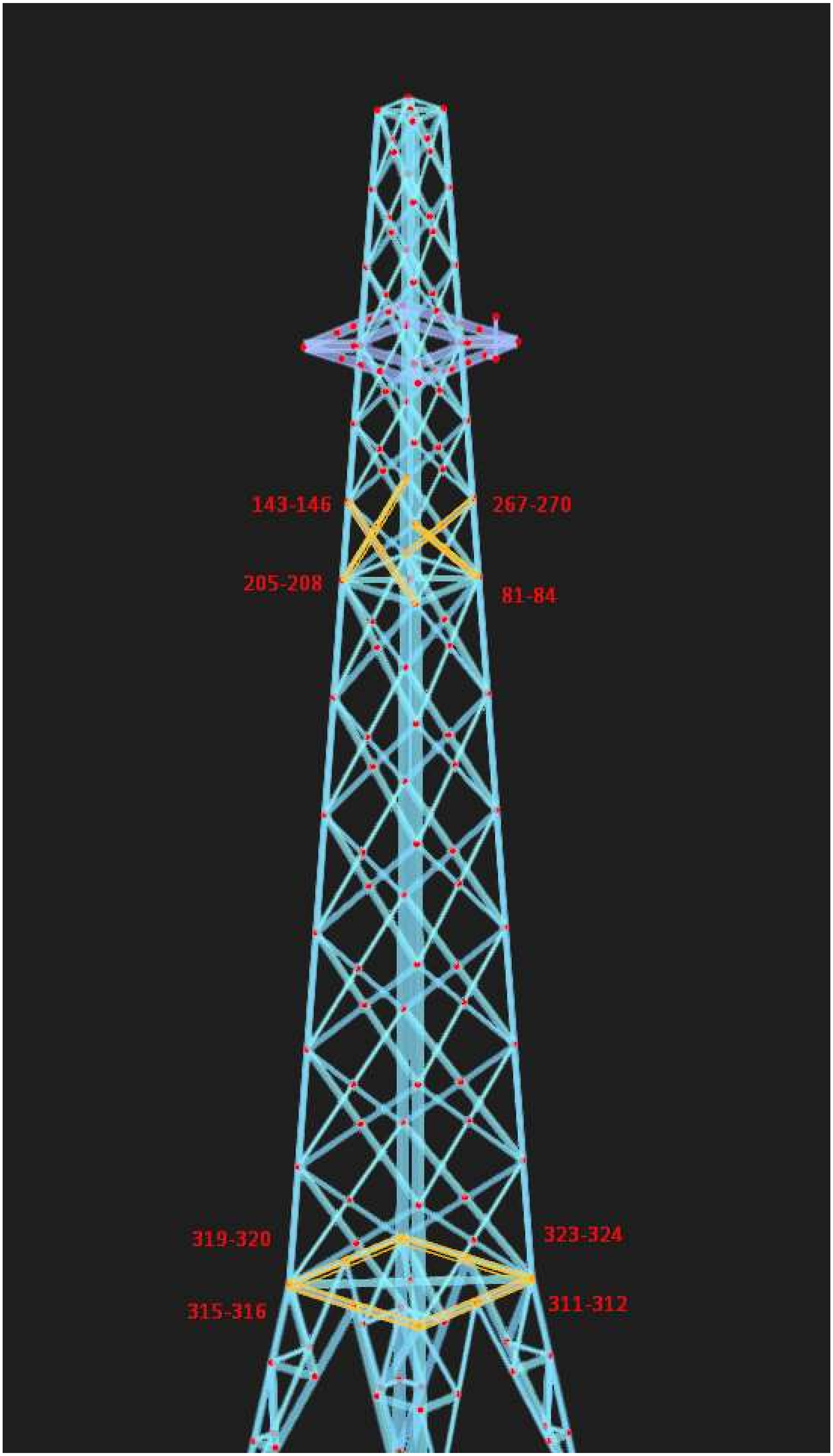
CQC method was preferred to the SRSS rule to combine structural modes. This choice was based on the robustness of CQC when dealing with modes of vibration close in frequency. Although the case at hand did not specifically require such attention CQC was the preferred choice as more reliable and advanced modelling option. Moreover a damping ratio of 5% was taken as a realistic assumption for all modes, setting the domain within the range of applicability of the CQC method. Given the EN1998-1 spectra wide band range CQC was also deemed more appropriate to the specific design situation. SRSS was in turn selected to combined different directional seismic components. This method was preferred to the absolute sum method, which was deemed over-conservative, and to the scaled sum method, which was discarded for its questionable approach to modelling the real physical relation among seismic components from different directions.

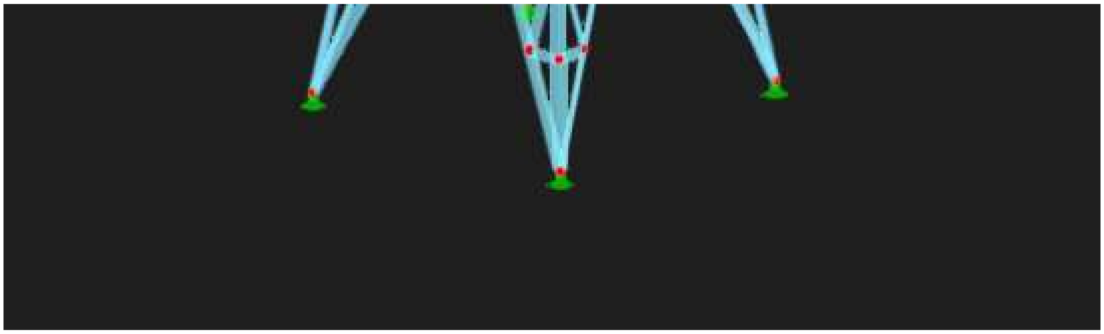
DOF	Member	Material Bilinear	Material Linear	Description
X	321	61.3 mm	61.3 mm	Top of tower
Y	309	61.3 mm	61.3 mm	Top of tower
Z	333	11.5 mm	11.5 mm	Horizontal Bracing (1st order)
ϕ_x	239	12.0 mrad	12.0 mrad	Strut in basal truss (2nd from bottom)
ϕ_y	177	12.0 mrad	12.0 mrad	Strut in basal truss (2nd from bottom)
ϕ_z	181	36.5 mrad	36.5 mrad	Strut in basal truss (lowest)

High-stress locations (most relevant):

Member No.	Location x [m]	Stress Existing [N/mm2]	Stress Limit [N/mm2]	Stress Ratio η [-]
81	1.183	117.558	235.000	0.500
84	0.000	119.666	235.000	0.509

Member No.	Location x [m]	Stress Existing [N/mm2]	Stress Limit [N/mm2]	Stress Ratio η [-]
143	1.183	117.466	235.000	0.500
146	0.000	119.577	235.000	0.509
205	1.183	117.314	235.000	0.499
208	0.000	119.421	235.000	0.508
267	1.183	117.533	235.000	0.500
270	0.000	119.641	235.000	0.509
311	1.707	126.505	235.000	0.538
312	0.000	126.471	235.000	0.538
315	1.707	126.560	235.000	0.539
316	0.000	126.524	235.000	0.538
319	1.707	126.516	235.000	0.538
320	0.000	126.471	235.000	0.538
323	1.707	126.512	235.000	0.538
324	0.000	126.444	235.000	0.538


143-146
267-270
205-208
81-84
319-320
323-324
315-316
311-312



As shown by the previous figure, the elements that present the highest stress ratio can be collected in two sets. As it could be expected there is only partial correspondence between the critical cross sections obtained from Response Spectrum Analysis and the Time History Analysis. It is first interesting to point out that the RSA results are completely symmetric whereas the THA are not. This is consistent with the spectra in the x and y direction being equal as prescribed by EN1998-1 in the RSA, as opposed to the recorded accelerogram considered for the THA. Even if the input spectra differ significantly, elements 311-312 appear as critical sections in both analysis. This is not surprising as those members are at the interface between the basal part and the upper part of the structure, which is where the load is transferred from the main truss structure to the support trusses. However it is apparent how the RSA produces significantly lower stress-ratios. Finally, the set of elements located in the upper part of the structure (81,84,143,146,205,208,267,270) do not find correspondence in the THA results. Relatively to the RSA results alone, this is explained by a higher relevance of horizontal excitations in the RSA. In relation to the THA, the dominance of vertical accelerations make these members irrelevant compared to others.

Part C

An incremental number of modes was considered in order to obtain a 90% cumulative sum of modal participation factors, as recommended by EN1998-1.

N Modes	X	Y	Z	ϕ_x	ϕ_y	ϕ_z
10	52.44 %	52.40 %	0.28 %	61.85 %	61.86 %	5.69 %
50	72.97 %	73.01 %	2.26 %	70.04 %	70.02 %	43.17 %
100	96.24 %	96.31 %	46.66 %	92.35 %	92.24 %	83.58 %
500	98.01 %	98.01 %	79.69 %	96.39 %	96.40 %	93.43 %

The Response Spectrum Analysis was carried out with a 500 mode approximation. It is worth noting that the summation of mass ratios in the Z direction does not meet the recommended value. The main reason behind such truncation decision was ensuring computational manageability of the model. From a physical point of view, this modelling choice should not impact significantly the results of the analysis, since the (cantilever) beam-like behavior of the structure should guarantee high resistance in the axial direction. Moreover, the dominant excitations imposed by EN1998-1 are horizontally directed, which further supports the implemented assumption, however this vulnerability should not go overlooked when vertical accelerations are relevant, as shown by the results of Time History Analysis where mode 41 proved to be determining.

Elaborating further on the sensitivity of the results to the number of modes used in the analysis, it can be observed by comparing the previous and the following table that no difference is encountered in the numerical results between 50, 100 and 500 (disclaimer: Member 115 is a symmetric counterpart of Member 239). A small numerical difference is instead recorded between 10 and 50 modes approximation.

N Modes	Max Disp. X	Max Disp. Y	Max Disp. Z	Max Disp. ϕ_x	Max Disp. ϕ_y	Max Disp. ϕ_z
10	61.4 mm Member No. 313	70.1 mm Member No. 365	11.5 mm Member No. 365	25.5 mrad Member No. 365	11.9 mrad Member No. 301	36.4 mrad Member No. 119
50	61.3 mm Member No. 313	61.3 mm Member No. 309	11.5 mm Member No. 333	12.0 mrad Member No. 115	12.0 mrad Member No. 177	36.5 mrad Member No. 181
100	61.3 mm Member No. 313	61.3 mm Member No. 309	11.5 mm Member No. 333	12.0 mrad Member No. 239	12.0 mrad Member No. 177	36.5 mrad Member No. 181
500	61.3 mm Member No. 313	61.3 mm Member No. 309	11.5 mm Member No. 333	12.0 mrad Member No. 239	12.0 mrad Member No. 177	36.5 mrad Member No. 181

Part D

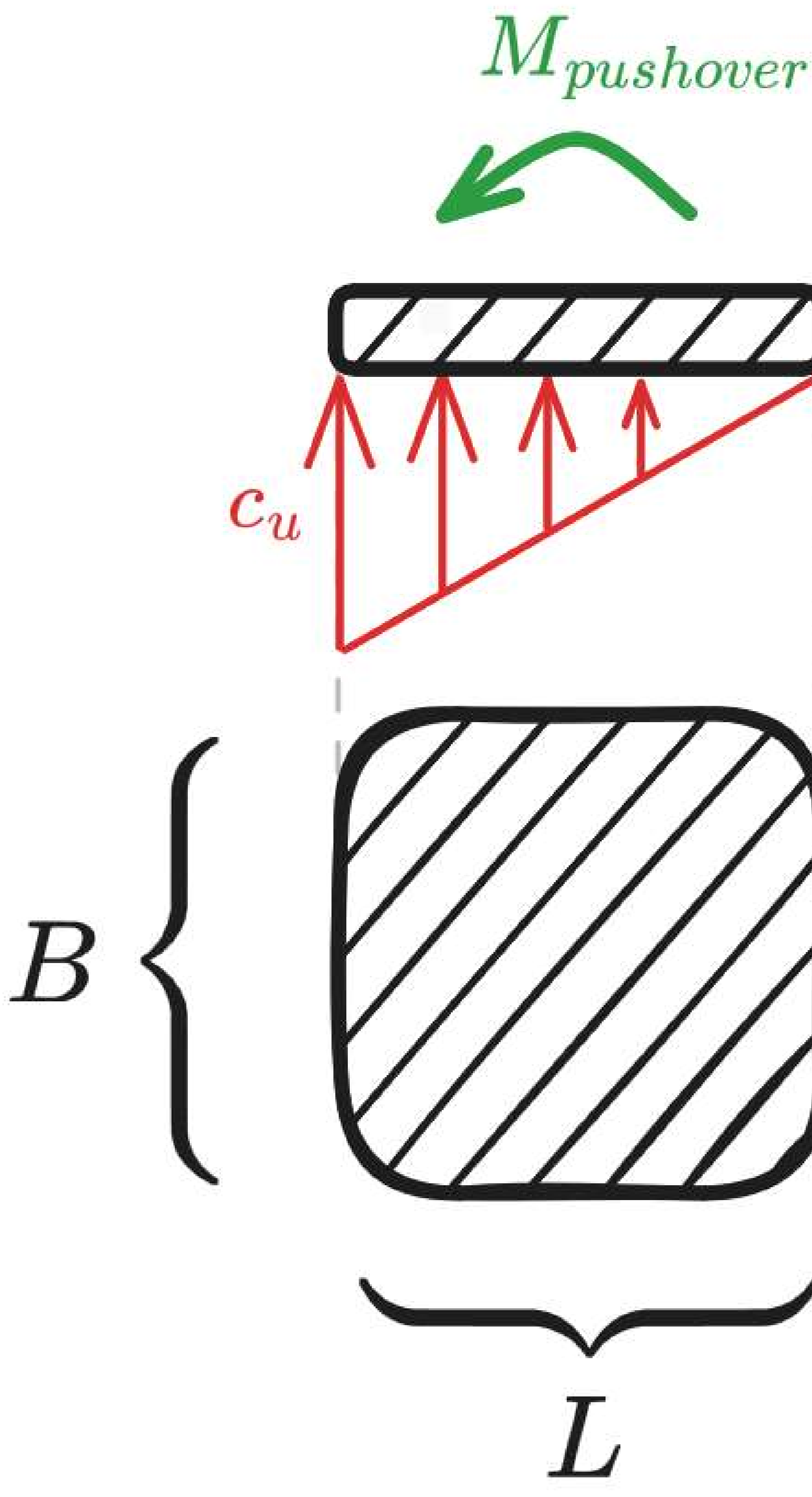
The structure is able to withstand the standardized seismic loads prescribed by EN1998-1. The computed stress ratios are well within unity check for all members. Additionally, no significant resonance was found.

Section 6 - Soil Structure Interaction

A preliminary design of the foundation was carried out as a first step. In order to match the double symmetry of the structure a square slab was the preferred choice. The dimensions were derived based on the maximum overturning moment developed during the pushover analysis, corresponding to 2811 kNm. Within the scope of approximation, a linear stress distribution was assumed along the length (maximum soil stress il reached on the left side and no lift was considered on the right side) and such distribution was assumed to be constant along width. An arbitrary depth of 0.3 m was consider a valid first guess based on a similar example in the study material.

Equilibrium: $M_{pushover} = (\frac{c_u L}{2} \cdot L) \cdot \frac{L}{6}$

Dimensions: $L = B = \sqrt[3]{\frac{12 * M_{pushover}}{c_u}} = 8.77m$



Part A - Frequency Dependent Soil Stiffness Matrix

In [82]:
```
# Soil Properties (indipendent of Student Number)
nu = 0.33       #[-] Poisson Ratio
c_u = 50e3      #[Pa] Undrained Shear Strength
v_s = 160       #[m/s] Velocity of shear waves in soil
rho = 1800      #[kg/m^3] Soil Density
beta = 0.05     #[-] Soil hysteretic (material) damping (arbitrarily chosen)

# Foundation dimensions (rectangular slab shallow foundation)
side_x = side_y = 8.77 #[m] As a first implementation the slab is as large as the b
depth = 0.3 #[m] height of foundation

# Structure
## Modal parameters from RFEM model (mode 2)
omega = 23.047 #[rad/s] Frequency of interest (Mode 2 - Main mode in X direction)
```

The frequency dependent dynamic stiffness matrix was derived according to the model proposed for a surface foundation on a homogenous half space as per Gazetas, G. (1991).

- Citation: Gazetas, G. 1991. Formulas and charts for impedances of surface and embedded foundations. Journal of Geotechnical Engineering, 117(9), 1363–1381. https://doi.org/10.1061/(asce)0733-9410(1991)117:9(1363)

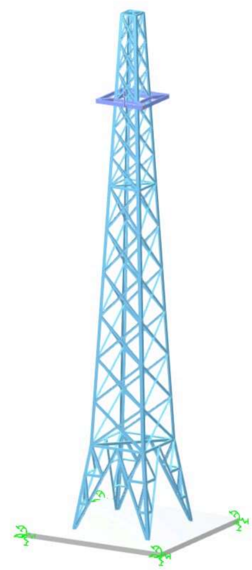
In [83]:
```
L = side_x/2                                   #[m] Half-length between center of th
B = side_y/2                                   #[m] Half-width between center of the
A_b = side_x*side_y                            #[m] Foundation surface area
I_bx = side_x*side_y**3/12                     #[m^4] Area moment of inertia about x
I_by = side_x**3*side_y/12                     #[m^4] Area moment of inertia about y
I_bz = side_x*side_y*(side_x**2+side_y**2)/12 #[m^4] Area moment of inertia about z
G = c_u                                        #[Pa] Shear modulus
v_la = 3.4*v_s/(np.pi*(1-nu))                  #[m/s] Lymer's analog wave velocity (
chi = A_b/(4*L**2)                             #[-] Parameter from cited paper'"Part

print("\nINPUT RECAP:")
print(f"L: {L:.3e} m")
print(f"B: {B:.3e} m")
print(f"A_b: {A_b:.3e} m")
print(f"I_bx: {I_bx:.3e} m^4")
print(f"I_by: {I_by:.3e} m^4")
print(f"I_bz: {I_bz:.3e} m^4")
print(f"G: {G:.3e} Pa")
print(f"v_la: {v_la:.3e} m/s")
print(f"chi: {chi:.3e} -")

#Static Stiffness
K_z = (2*G*L/(1-nu))*(0.73+1.54*chi**0.75)
K_y = (2*G*L/(2-nu))*(2+2.50*chi**0.85)
K_x = K_y - (0.2/(0.75-nu))*G*L*(1-(B/L))
K_rx = (G/(1-nu))*I_bx**0.75*(L/B)**0.25*(2.4+0.5*(B/L))
K_ry = (3*G/(1-nu))*I_by**0.75*(L/B)**0.15
K_t = 3.5*G*I_bz**0.75*(B/L)**0.4*(I_bz/B**4)**0.2
```

```
# Dynamic Stiffness
a_0 = omega * B / v_s
print(f"a_0: {a_0:.3e} ")

## Table values
k_z_tab = 0.92          #(table Fig.2(a))
k_y_tab = 0.94          #(table Fig.2(b))
k_x_tab = 1             #(constant)
k_rx_tab = 1-0.20*a_0   #(Table 1)
k_ry_tab = 1-0.26*a_0   #(Table 1)
k_t_tab = 1-0.14*a_0    #(Table 1)

## Coefficients
K_z_dyn = K_z*k_z_tab
K_y_dyn = K_y*k_y_tab
K_x_dyn = K_x*k_x_tab
K_rx_dyn = K_rx*k_rx_tab
K_ry_dyn = K_ry*k_ry_tab
K_t_dyn = K_t*k_t_tab

print("\nDynamic Stiffness Coefficients:")
print(f"K_z_dyn: {K_z_dyn:.3e} N/m")
print(f"K_y_dyn: {K_y_dyn:.3e} N/m")
print(f"K_x_dyn: {K_x_dyn:.3e} N/m")
print(f"K_rx_dyn: {K_rx_dyn:.3e} N/m")
print(f"K_ry_dyn: {K_ry_dyn:.3e} N/m")
print(f"K_t_dyn: {K_t_dyn:.3e} N/m")

#  Radiation Dashpot Coefficients
## Table values
c_z_tab = 0.93 #(table Fig.2(c))
c_y_tab = 0.87 #(table Fig.2(d))
c_rx_tab = 0.18 #(table Fig.2(e))
c_ry_tab = 0.18 #(table Fig.2(f))
c_t_tab = 0.15 #(table Fig.2(g))

## Coefficients
C_z = rho*v_la*A_b*c_z_tab
C_y = rho*v_s*A_b*c_y_tab
C_x = rho*v_s*A_b
C_rx = rho*v_la*I_bx*c_rx_tab
C_ry = rho*v_la*I_by*c_ry_tab
C_t = rho*v_s*I_bz*c_t_tab

## Total Damping (inclusive of materia damping)
C_z_tot = C_z+2*K_z_dyn*beta/omega
C_y_tot = C_y+2*K_y_dyn*beta/omega
C_x_tot = C_x+2*K_x_dyn*beta/omega
C_rx_tot = C_rx+2*K_rx_dyn*beta/omega
C_ry_tot = C_ry+2*K_ry_dyn*beta/omega
C_t_tot = C_t+2*K_t_dyn*beta/omega
print("\nTotal Damping Coefficients")
print(f"C_z_tot: {C_z_tot:.3e} N/(m*s)")
print(f"C_y_tot: {C_y_tot:.3e} N/(m*s)")
print(f"C_x_tot: {C_x_tot:.3e} N/(m*s)")
print(f"C_rx_tot: {C_rx_tot:.3e} N/(m*s)")
```

```
print(f"C_ry_tot: {C_ry_tot:.3e} N/(m*s)")
print(f"C_t_tot: {C_t_tot:.3e} N/(m*s)")

# Dynamic Stiffness Matrix (inclusive of damping)
K_dyn_vec_6dof = [ K_x_dyn, K_y_dyn, K_z_dyn, K_rx_dyn, K_ry_dyn, K_t_dyn ]
C_tot_vec_6dof = [ C_x_tot, C_y_tot, C_z_tot, C_rx_tot, C_ry_tot, C_t_tot ]
K_tilde_6dof = np.diag([k+1j*omega*c for k,c in zip(K_dyn_vec_6dof,C_tot_vec_6dof)]
print("\nDynamic Stiffness Matrix (inclusive of damping):")
print(K_tilde_6dof)
```

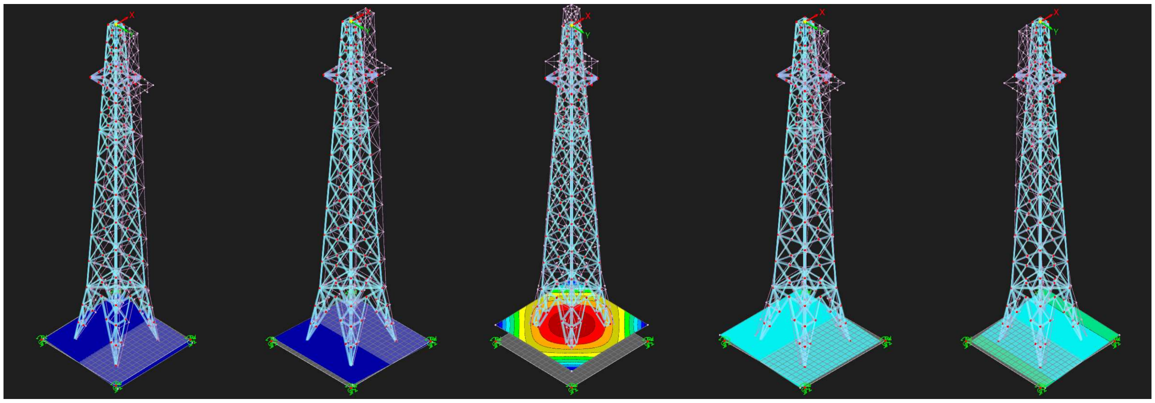
```
INPUT RECAP:
L: 4.385e+00 m
B: 4.385e+00 m
A_b: 7.691e+01 m
I_bx: 4.930e+02 m^4
I_by: 4.930e+02 m^4
I_bz: 9.859e+02 m^4
G: 5.000e+04 Pa
v_la: 2.584e+02 m/s
chi: 1.000e+00 -
a_0: 6.316e-01

Dynamic Stiffness Coefficients:
K_z_dyn: 1.367e+06 N/m
K_y_dyn: 1.111e+06 N/m
K_x_dyn: 1.182e+06 N/m
K_rx_dyn: 1.978e+07 N/m
K_ry_dyn: 1.958e+07 N/m
K_t_dyn: 3.415e+07 N/m

Total Damping Coefficients
C_z_tot: 3.328e+07 N/(m*s)
C_y_tot: 1.928e+07 N/(m*s)
C_x_tot: 2.216e+07 N/(m*s)
C_rx_tot: 4.137e+07 N/(m*s)
C_ry_tot: 4.136e+07 N/(m*s)
C_t_tot: 4.274e+07 N/(m*s)

Dynamic Stiffness Matrix (inclusive of damping):
[[ 1181586.82634731+5.10630301e+08j        0.        +0.00000000e+00j
         0.        +0.00000000e+00j        0.        +0.00000000e+00j
         0.        +0.00000000e+00j        0.        +0.00000000e+00j]
 [       0.        +0.00000000e+00j  1110691.61676647+4.44256633e+08j
         0.        +0.00000000e+00j        0.        +0.00000000e+00j
         0.        +0.00000000e+00j        0.        +0.00000000e+00j]
 [       0.        +0.00000000e+00j        0.        +0.00000000e+00j
   1366811.04477612+7.67044678e+08j        0.        +0.00000000e+00j
         0.        +0.00000000e+00j        0.        +0.00000000e+00j]
 [       0.        +0.00000000e+00j        0.        +0.00000000e+00j
         0.        +0.00000000e+00j 19781320.21990046+9.53351004e+08j
         0.        +0.00000000e+00j        0.        +0.00000000e+00j]
 [       0.        +0.00000000e+00j        0.        +0.00000000e+00j
         0.        +0.00000000e+00j        0.        +0.00000000e+00j
  19575778.96330155+9.53330450e+08j        0.        +0.00000000e+00j]
 [       0.        +0.00000000e+00j        0.        +0.00000000e+00j
         0.        +0.00000000e+00j        0.        +0.00000000e+00j
         0.        +0.00000000e+00j 34151351.57732812+9.85039369e+08j]]
```

The foundation-soil interaction was simulated using four discrete nodal supports positioned at the corners of the foundation . The soil spring constants, derived from Gazetas tables were distributed among these four corner nodes to accurately capture the load transfer mechanism and soil response under various loading conditions.The damping dashpots were neglected as a conservative assumption.



Response spectrum analysis was conducted to evaluate the structural behavior under dynamic loading conditions. The summary of displacements obtained from this analysis is presented below.

Description	Value	Unit	Notes
Maximum displacement in X-direction	653.5 mm	mm	Member No. 313, x: 0.457 m
Maximum displacement in Y-direction	664.1 mm	mm	Member No. 309, x: 0.457 m
Maximum displacement in Z-direction	229.5 mm	mm	FE node No. 177: (-4.387, 4.387, 30.000 m)
Maximum vectorial displacement	951.7 mm	mm	Member No. 1, x: 0.000 m
Maximum rotation about X-axis	24.6 mrad	mrad	Member No. 365, x: 0.112 m
Maximum rotation about Y-axis	23.7 mrad	mrad	Member No. 150, x: 1.713 m
Maximum rotation about Z-axis	4.5 mrad	mrad	Member No. 119, x: 0.569 m



Implementing SSI yields period lengthening that shifts the tower's natural frequencies to lower values.The distinct modes observed in each analysis case highlight how foundation and soil properties significantly alter the load transfer mechanisms from the superstructure to the ground resulting in non-uniform stress distributions that would not be captured in a fixed-base analysis. The first two modes are mostly translational whereas we can see third mode capturing significant vertical deformations. Mode shapes and modal participation factors become redistributed, with translational and torsional modes coupling more strongly, thereby affecting the accuracy of modal combination methods in RSA. The RSA results with the elastic foundation springs demonstrate a significant increase in maximum structural displacements compared to the fixed-support analysis presented earlier.Additionally, since the damping contribution of soil is neglected, this could further contribute to the higher observed responses.The higher displacements observed in the tabulated results reflect the more realistic structural behavior under seismic loading.These insights underscore the importance of incorporating SSI in RSA for telecommunication towers to capture realistic dynamic behavior and avoid under- or over-design based on rigid support assumptions.

Part B

As previously mentioned, mode 2 presents an Effective Modal Mass factor of 0.45, it was therefore considered the most appropriate to describe the system in a Single Degree of Freedom simplified system. The mass and moment of inertia of the SDoF system correspond to the effective modal mass in the translational x direction and rotational effective mass about the y-axis of the MDoF model. Both parameters can be extracted from the finite element model assembled in RFEM. The modal height can be computed as $h^* = \frac{\Gamma_{ry}}{\Gamma_t}$. A modal damping ratio of $\zeta^* = 0.048$ is assumed, therefore $c^* = 2\zeta^* m^* \omega$. Finally the modal stiffness can be derived as $k^* = \omega^2 M^*$

A summary of the properties used in the conversion from MDoF to SDoF is presented in the following table:

Parameter	Value	Origin
Modal mass	$1683\ kg$	RFEM
Effective Modal Mass (x) > m	$3940\ kg$	RFEM
Effective Modal Mass (r-y) > J	$371117\ kg \cdot m^2$	RFEM
Γ_t	2575	RFEM
Γ_{ry}	24995	RFEM
h^*	$9.705\ m$	Computed
c^*	$3724\ kg \cdot rad / s$	Computed
k^*	$2092787\ kg \cdot rad^2 / s^2$	Computed

Taking into account the presence of the soil the SDOF system can therefore be represented as follows.

SSI_system

Where:

$u^r(t)$ relative displacement between top mass and foundation

$u^t_g(t) = u_g(t) + u^I_g(t)$ with: $u^t_g(t)$ total ground displacement, $u_g(t)$ free-field ground displacement, $u^I_g(t)$ ground displacement deriving from soil-structure interaction

$\theta^t_g(t) = \backslash\text{cancel}\theta_g(t) + \theta^I_g(t)$ with $\theta^t_g(t)$ total rocking motion of the ground, $\theta_g(t)$ free-field rocking motion (neglected), $\theta^I_g(t)$ rocking motion due to soil-structure interaction

The system can then be divided in two subsystems in order to analyze the interaction between soil and structure.

SSI_subsystems

The translational equilibrium equation for the top mass can be formulated as:

$$m\ddot{u}^r(t) + c\dot{u}^r(t) + ku^r(t) + m(\ddot{u}_g(t) + \ddot{u}^I_g(t) + h\ddot{\theta}^I_g(t)) = 0$$

The translational equilibrium at soil-structure interface can be formulated as:

$$m(\ddot{u}_g(t) + \ddot{u}_g^I(t) + \ddot{u^r}(t) + h\ddot{\theta}_g^I(t)) + m_f(\ddot{u}_g(t) + \ddot{u}_g^l(t)) = V_b(t)$$

The rotational equilibrium at the soil-structure interface can be formulated as:

$$mh(\ddot{u}_g(t) + \ddot{u}_g^l(t) + \ddot{u^r}(t)) + (mh^2 + J + J_f)\ddot{\theta}_g^l(t) = M_b(t)$$

Using a Fourier Transform the system can be converted to the frequency domain as follows:

$$(-\omega^2 m + i\omega c + k)\tilde{u^r}(\omega) - \omega^2 m\tilde{u}_g^I(\omega) - \omega^2 mh\tilde{\theta}_g^I(\omega) = -m\tilde{a}_g(\omega)$$

$$-\omega^2 m\tilde{u^r}(\omega) - \omega^2(m + m_f)\tilde{u}_g^I(\omega) - \omega^2 mh\tilde{\theta}_g^I(\omega) - \tilde{V}_b(\omega) = -(m + m_f)\tilde{a}_g(\omega)$$

$$-\omega^2 mh\tilde{u^r}(\omega) - \omega^2 mh\tilde{u}_g^I(\omega) - \omega^2(mh^2 + J + J_f)\tilde{\theta}_g^I(\omega) - \tilde{M}_b(\omega) = -mh\tilde{a}_g(\omega)$$

Where: $\tilde{a}_g = \tilde{\ddot{u}}_g$

Moreover $\tilde{V}_b(\omega)^{SS-I} = -\tilde{V}_b(\omega)^{SS-II}$ and $\tilde{M}_b(\omega)^{SS-I} = -\tilde{M}_b(\omega)^{SS-II}$

$\tilde{V}_b(\omega)$ and $\tilde{M}_b(\omega)$ can therefore be derived from the soil stiffness matrix as:

$$\tilde{V}_b(\omega) = -\tilde{k}_{xx}(\omega)\tilde{u}_g^I(\omega) - \cancel{\tilde{k}_{x\theta}(\omega)\tilde{\theta}_g^I(\omega)} - i\omega\tilde{c}_{xx}(\omega)\tilde{u}_g^I(\omega) - \cancel{i\omega\tilde{c}_{x\theta}(\omega)\tilde{\theta}_g^I(\omega)}$$

$$\tilde{M}_b(\omega) = -\cancel{\tilde{k}_{\theta x}(\omega)\tilde{u}_g^I(\omega)} - \tilde{k}_{\theta\theta}(\omega)\tilde{\theta}_g^I(\omega) - \cancel{i\omega\tilde{c}_{\theta x}(\omega)\tilde{u}_g^I(\omega)} - i\omega\tilde{c}_{\theta\theta}(\omega)\tilde{\theta}_g^I(\omega)$$

Soil stiffness coefficients were derived following the methodology presented in Gazetas, G. (1991). Formulas and charts for impedances of surface and embedded foundations. Journal of Geotechnical Engineering, 117(9), 1363–1381. (see part A). It is worth underlying that cross-coupling terms (translation-rotation interaction) can be neglected when dealing with shallow foundations.

After the final substitution, the system can therefore be rewritten in the following form:

$$(-\omega^2 m + i\omega c + k)\tilde{u^r}(\omega) - \omega^2 m\tilde{u}_g^I(\omega) - \omega^2 mh\tilde{\theta}_g^I(\omega) = -m\tilde{a}_g(\omega)$$

$$-\omega^2 m\tilde{u^r}(\omega) + (\tilde{k}_{xx}(\omega) + i\omega\tilde{c}_{xx}(\omega) - \omega^2(m + m_f))\tilde{u}_g^I(\omega) - \omega^2 mh\tilde{\theta}_g^I(\omega) = -(m + m_f)\tilde{a}_g($$

$$-\omega^2 mh\tilde{u^r}(\omega) - \omega^2 mh\tilde{u}_g^I(\omega) + (\tilde{k}_{\theta\theta}(\omega) + i\omega\tilde{c}_{\theta\theta}(\omega) - \omega^2(mh^2 + J + J_f))\tilde{\theta}_g^I(\omega) = -mh\tilde{a}_g($$

The algebraic system of equations can be solved for the three unknowns: $\tilde{u^r}(\omega)$, $\tilde{u}_g^I(\omega)$, $\tilde{\theta}_g^I(\omega)$

By adopting an analogous approach the system without soil structure interaction could be described by the following equations:

Equation of motion: $m\ddot{u^r}(t) + c\dot{u^r}(t) + ku^r(t) = -ma_g - \cancel{mh\theta_g}$

Equation of motion in frequency domain $(-\omega^2 m + i\omega c + k)\tilde{u^r}(\omega) = -m\tilde{a}_g$

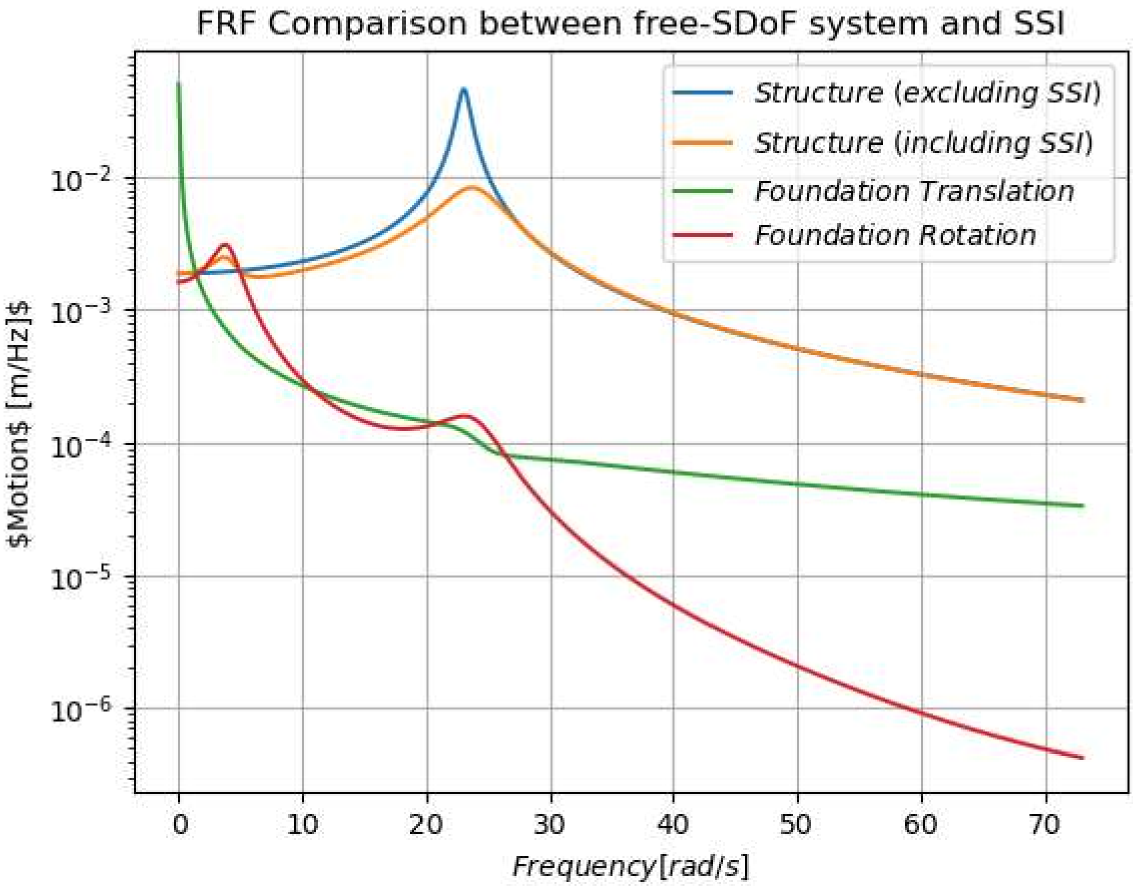
In [86]:
```
# SDOF Model
m_modal = 1683     #[kg] modal mass
M_x = 3940         #[kg] effective translational modal mass in the x direction
J_y = 371117       #[kg*m^2] effective rotational modal mass about the y-axis (mome
gamma_t = 2575.4   #[-] translational participation factor in the x direction
gamma_ry = 24995.1 #[-] rotation participation factor about the y-axis
zeta_star = 0.048  #[-] modal damping ratio (mode 2, arbitrarily assumed)
omega_s = 23.044   #[rad/s] natural frequency of the SDoF model (mode 2)

h_star = gamma_ry/gamma_t
c_star = 2*zeta_star*m_modal*omega_s
k_star = omega_s**2*M_x

rho_f = 2400 #[kg/m^3] Density of concrete (arbitrarily chosen)
m_f = rho_f*side_x*side_y*depth #[kg] Mass of foundation
J_y_f = m_f*(side_x**2+depth**2)/12 #[kg*m^2] Moment of Inertia of foundation about

print("Parameter Recap:")
print(f"h_star: {h_star:.3f} m")
print(f"c_star: {c_star:.3f} kg*rad/s")
print(f"k_star: {k_star:.3f} kg*rad^2/s^2")

omega_table_max = 2*v_s/B
omega_domain = np.linspace(0.001, omega_table_max,10000) #Based on curves applicabi

# Fitting of curves in tables
## Only c_ry
points = [[0, 0.25, 0.5, 0.65, 1, 1.5, 2],[0, 0.05, 0.125, 0.2, 0.3, 0.45, 0.55]]
c_ry_curve = np.poly1d(np.polyfit(points[0], points[1], 5))

# Polynomial graph as sanity check
# plt.plot(omega_domain, c_ry_curve(omega_domain*B/v_s))
# plt.show()

frf_ur_tilde = []
frf_ug_tilde = []
frf_thetag_tilde = []

for counter,omega in enumerate(omega_domain):

    # Frequency Dependent Coefficients for Dynamic Stiffness and Damping of Soil
    # (same as previous code-cell but repeated inside the loop only for relevant di
    a_0 = omega*B/v_s

    K_y_static = (2*G*L/(2-nu))*(2+2.50*chi**0.85) #y coefficient is required to co
    K_x_static = K_y_static - (0.2/(0.75-nu))*G*L*(1-(B/L))
    K_x_dyn = K_x_static*1 #Formula from table
    K_ry_static = (3*G/(1-nu))*I_by**0.75*(L/B)**0.15
    K_ry_dyn = (1-0.26*a_0) * K_ry_static

    C_x = rho*v_s*A_b
    C_x_tot = C_x + 2*K_x_dyn*beta/omega
```

```
    C_ry = rho*v_la*I_by*c_ry_curve(a_0)
    C_ry_tot = C_ry + 2*K_ry_dyn*beta/omega

    # Acceleration
    a_g_tilde = 1 #[(m/s^2)/Hz] Acceleration in the frequency domain (taken as unit

    K_tilde_system = np.zeros((3,3), dtype=complex)
    K_tilde_system[0, 0] = -omega**2*M_x + 1j*omega*c_star + k_star
    K_tilde_system[1, 1] = K_x_dyn + 1j*omega*C_x_tot - omega**2*(M_x+m_f)
    K_tilde_system[2, 2] = K_ry_dyn + 1j*omega*C_ry_tot - omega**2*(M_x*h_star**2+J
    K_tilde_system[0, 1] = K_tilde_system[1, 0] = -omega**2*M_x
    K_tilde_system[0, 2] = K_tilde_system[2, 0] = -omega**2*M_x*h_star
    K_tilde_system[1, 2] = K_tilde_system[2, 1] = - omega**2*M_x*h_star

    F_tilde = np.array([-M_x*a_g_tilde, -(M_x+m_f)*a_g_tilde, -M_x*h_star*a_g_tilde

    u_r_tilde_omega , ug_tilde_omega , u_thetag_tilde_omega  = np.linalg.solve(K_ti
    frf_ur_tilde.append(u_r_tilde_omega)
    frf_ug_tilde.append(ug_tilde_omega)
    frf_thetag_tilde.append(u_thetag_tilde_omega)


# No-interaction SDOF response

frf_ur_tilde_no_SSI = [-M_x*a_g_tilde/(-omega**2*M_x+1j*c_star*omega+k_star) for om

plt.semilogy(omega_domain, np.abs(frf_ur_tilde_no_SSI), label=r"$Structure\ (exclud
plt.semilogy(omega_domain, np.abs(frf_ur_tilde), label=r"$Structure\ (including\ SS
plt.semilogy(omega_domain, np.abs(frf_ug_tilde), label=r"$Foundation\ Translation$"
plt.semilogy(omega_domain, np.abs(frf_thetag_tilde), label=r"$Foundation\ Rotation$
plt.title("FRF Comparison between free-SDoF system and SSI")
plt.xlabel("$Frequency [rad/s]$")
plt.ylabel(r"$Motion$ [m/Hz]$")
plt.grid()
# plt.xlim(0, 10)
plt.legend()
```

```
Parameter Recap:
h_star: 9.705 m
c_star: 3723.173 kg*rad/s
k_star: 2092242.188 kg*rad^2/s^2
```

Out[86]:

```
<matplotlib.legend.Legend at 0x1dd016fd1d0>
```



It is apparent how the introduction of the SSI has a significant impact on the dynamic response of the system. This is not only due to the soil behavior but also to the introduction of the foundation mass and the related degrees of freedom. The FRF of the structure inclusive of the interaction model shows a shifted and milder peak in the natural frequency corresponding to the vibration of the top mass. From the standpoint of the physical accuracy of the model, an important addition compared to the SDoF system without SSI is the energy dissipation due to soil material damping and radiation damping. Moreover a second peak appears in the structure response. Such peak is located at approximately 3 Hz and corresponds to the natural frequency of the rotational degree of freedom of the foundation.