Scour Project

Kyeong Sun Kim

3/12/23

Table of contents

1 Design Parameters

1.1 Design Cases

Case	Note
1	Horizontal load = 1000 kN @ tower end
2	Horizontal load = 300 kN @ suction bucket
3	Horizontal load = 800 kN @ transition piece

1.2 Wind Turbine System

Component	Parameter	Value
Turbine	Nacelle weight	193 Ton
	Lateral load	1000 kN
Tower	Tower weight	auto dead load (gravity)
	Length	70 m
	Diameter	4.5 m
	Thickness	22 mm
	Young's modulus	210 GPa
	Poisson's ratio	0.30
Transition piece (Main)	Length	
	Diameter	4.5 m
	Thickness	35 mm
Transition piece (Bracing)	Length	
	Diameter	1.5 m
	Thickness	40 mm
	Young's modulus	210 GPa
	Poisson's ratio	0.30
Suction bucket	Length	6
	Diameter	12
	Thickness	19 mm
	Spacing (for tripod)	17.2 m
	Young's modulus	210 GPa

	Poisson's ratio	0.30

1.3 Soil

Constitutive model	Parameter	Value
HSSM	Density	17
	Young's modulus	40E3
	Poisson's ratio	0.30
	Shear modulus	120E3
	Strain at 0.7	0.00015
	Cohesion	1
	Friction angle	40
	Dilation angle	10
	Earth pressure coeff. at rest	0.5

1.4 Computation

Control settings	Parameter	Value
Boundary extent	Nx	120
	Ny	120
	Nz	120
Mesh	Type	Fine
	Elements	117,667
	Nodes	175,012
	Element size	0.4
	Element dimension	3.642
	Global size factor	1
	Min element size	0.001

2 Case1

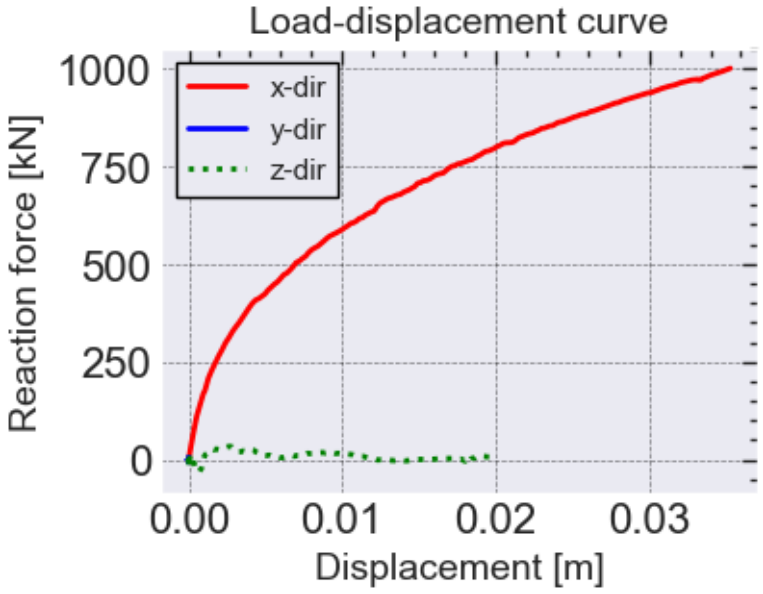
2.1 Load-displacement behavior

```
import os, time, sys, math, random, json, requests, csv, subprocess
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import scienceplots
import multiprocessing as mp
from scipy import stats
from plotnine import *
from plotnine.data import diamonds
from IPython.display import display
import xlsxwriter
import xmltodict
from plxscripting.easy import*
import plxscripting.easy
from collections import defaultdict
from collections import Mapping
from bs4 import BeautifulSoup
from lxml import objectify

##############################################################################
# data to plot
df = pd.read_csv(r'C:\Users\Admin\Desktop\pytools\kyeongsunkim.github.io\saved\saved.csv')
x = df["Ux"].to_numpy()
y = df["Fx"].to_numpy()
x2 = df["Uy"].to_numpy()
y2 = df["Fy"].to_numpy()
x3 = df["Uz"].to_numpy()
y3 = df["Fz"].to_numpy()
##############################################################################
# figure
```

```
plt.rc('font', family='serif')
plt.rc('xtick', labelsize='x-small')
plt.rc('ytick', labelsize='x-small')
plt.style.use(['science', 'notebook', 'grid'])
fig = plt.figure(figsize=(4, 3))
###############################################################################
# plot [1]
ax = fig.add_subplot(1, 1, 1)
#ax.plot(x, y, color='k', ls='solid')
#ax.plot(x2, y2, color='0.20', ls='dashed')
#ax.plot(x3, y3, color='k', ls='dotted')
ax.plot(x, y, color='red', ls='solid')
ax.plot(x2, y2, color='blue', ls='solid')
ax.plot(x3, y3, color='green', ls='dotted')
# labels
fntsz = 14
ax.set_xlabel('Displacement [m]', fontsize=fntsz)
ax.set_ylabel('Reaction force [kN]', fontsize=fntsz)
ax.set_title('Load-displacement curve', fontsize=fntsz)

# legend
ax.legend(['x-dir', 'y-dir', 'z-dir'], loc='upper left', fancybox=False, edgecolor='black'
###############################################################################
plt.show()
```



2.2 Moment-rotation behavior

```
import os, time, sys, math, random, json, requests, csv, subprocess
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import scienceplots
import multiprocessing as mp
from scipy import stats
from plotnine import *
from plotnine.data import diamonds
from IPython.display import display
import xlsxwriter
import xmltodict
from plxscripting.easy import*
import plxscripting.easy
from collections import defaultdict
from collections import Mapping
from bs4 import BeautifulSoup
from lxml import objectify

##############################################################################
# data to plot
df = pd.read_csv(r'C:\Users\Admin\Desktop\pytools\kyeongsunkim.github.io\saved\saved.csv')
x = df["Rotx"].to_numpy()
y = df["Mx"].to_numpy()
x2 = df["Roty"].to_numpy()
y2 = df["My"].to_numpy()
x3 = df["Rotz"].to_numpy()
y3 = df["Mz"].to_numpy()
##############################################################################
# figure
plt.rc('font', family='serif')
plt.rc('xtick', labelsize='x-small')
plt.rc('ytick', labelsize='x-small')
plt.style.use(['science', 'notebook', 'grid'])
fig = plt.figure(figsize=(4, 3))
##############################################################################
# plot [1]
ax = fig.add_subplot(1, 1, 1)
```

```
#ax.plot(x, y, color='k', ls='solid')
#ax.plot(x2, y2, color='0.20', ls='dashed')
#ax.plot(x3, y3, color='k', ls='dotted')
ax.plot(x, y, color='red', ls='solid')
ax.plot(x2, y2, color='blue', ls='solid')
ax.plot(x3, y3, color='green', ls='dotted')
# labels
fntsz = 14
ax.set_xlabel('Rotation [rad]', fontsize=fntsz)
ax.set_ylabel('Reaction moment [kNm]', fontsize=fntsz)
ax.set_title('Moment-rotation curve', fontsize=fntsz)

# legend
ax.legend(['x-dir', 'y-dir', 'z-dir'], loc='upper left', fancybox=False, edgecolor='black'
##############################################################################
plt.show()
```