

Politecnico Di Torino

Department of Civil Engineering

MSc Civil Engineering



Development of Numerical Software for Sheet Pile Wall

Supervisor: Prof. Andrea Dominijanni

MSc Graduation Thesis by:

Waleed Arshad

S264621

Academic year 2021/2021

Abstract

This report presents the methodology and validation of the software program developed to analyze the sheet pile walls using the Soil Structure Interaction method. The report presents the general theory behind the SSI method, and how it can be used to analyze the behavior of sheet pile walls and the anchors. The software is based on some assumptions that it can only consider a single layer of soil with constant section throughout its depth. MATLAB coding was used to develop a program that solves multiple equations developed from SSI method to generate the moment, shear, deflection, and slope diagrams for the sheet pile walls. The results obtained from the software were compared with the existing spreadsheet for the analysis of sheet pile walls using SSI method. The results obtained were identical as that of the spreadsheet. The results were also compared with other software programs developed to analyze sheet pile walls based on other methods. The SPW software based on beam theory was used to compare the results. The difference in the extreme values of moment and shear in the case of sheet piles without anchors was less than 2% which is negligible. However, the difference in case of sheet pile wall with anchors generated slightly different values. The difference was 16.25% in case of minimum shear. This difference can be because of the different approaches used by both the methods to analyze the anchors in the sheet pile walls. The future recommendations include the inculcation of multiple soil layers and varying section of the sheet pile wall in the analysis procedure.

Acknowledgement

First of all, I would like to express my gratitude toward my supervisor Dr. Andrea Dominijanni, for his continuous support and guidance throughout work.

Moreover, I would like to thank my family for the support they have given me to achieve the dream of obtaining the master's degree in Civil Engineering. Lastly, I thank my friends who has supported and guided me during my works

Contents

Abstract.....	2
Acknowledgement	3
List of Figures.....	6
List of Tables	6
1. Introduction.....	8
2. Literature Review.....	10
2.1 Problem Statement	10
2.2 Soil Structure Interaction Method (SSI).....	11
2.3 Assumptions.....	11
3. Methodology.....	12
3.1 Winkler Spring Model.....	12
3.2 Boundary Conditions.....	14
3.2.1 Boundary Conditions at Top.....	15
3.2.2 Boundary Conditions at Bottom	15
3.3 MATLAB coding.....	16
4. Comparison with Excel Sheet Results	19
4.1 Excel Sheet Results	21
4.2 Results from New Software	24
5. Validation with Other Methods	26
5.1 Case 1: No Anchors.....	27
5.1.1 Results from SPW software.....	27
5.1.2 Results from New Software	29

5.2	Case 2: With Anchors	31
5.2.1	Results from SPW Software	32
5.2.2	Results from New Software	33
6.	Conclusion	36
7.	Future Recommendations	38
8.	References.....	39
9.	Annex.....	40

List of Figures

Figure 1: Watertightness of Sheet Piles	9
Figure 2: SSI method schematic diagram	13
Figure 3: User Interface	17
Figure 4: Results' Tab	18
Figure 5: Comparison Problem.....	19
Figure 6: Fictitious points and springs.....	20
Figure 7: Diagrams from Excel Sheet.....	23
Figure 8: Input Parameters for Case 1 Validation	24
Figure 9: Diagrams from New Software.....	25
Figure 10: Input Parameters for SPW Software case 1.....	28
Figure 11: Results from SPW Software Case 1	28
Figure 12: Input Parameter for New Software Case 1	29
Figure 13: Results from New Software Case 1.....	30
Figure 14: Input Data for SPW Software Case 2	32
Figure 15: Results from SPW Software Case 2	33
Figure 16: Input Parameters for New Software Case 2	34
Figure 17: Results from New Software Case 2.....	34

List of Tables

Table 1: Parameters.....	20
Table 2: Results from Excel Sheet.....	21
Table 3: Shear, Moment, Deflection and Slope from Excel.....	22
Table 4: Input Data for Case 1 of Validation.....	27
Table 5: Percentage Difference in Results for Case 1	30

Table 6: Input Parameters for Case 2..... 31

Table 7: Percentage Difference in Results for Case 2 35

1. Introduction

Civil Engineering is related to the design of different structures. A structure is good only if all the elements of it are functioning properly. Even if a small element does not fulfill the serviceability or ultimate limits, it becomes unsafe. Foundation of any structure is also very important in this regard. A structure is as good as its foundation. A rigorous analysis of the soil present at the base is needed to determine the different strength parameters for the soil. These parameters are then used to determine the type of foundation needed for the structure. If the geotechnical investigation of the soil is not up to the mark, no matter how good the structural design is, the structure will not be able to meet the serviceability requirement as the settlement of the soil can cause the structure to deform differentially.

There are numerous incidents where the complete structure collapsed because of the failure of the soil foundation. One of these examples is the Ocean Tower in South Padre Island in USA. The project was a huge inspiration, but after the construction of 31 floors, it was found that there are some cracks in the columns of the garage. The issue was reported that the soil underneath the structure was having different properties. The clay under the garage was stiffer which resulted in comparatively less settlement of the garage, on the other hand, the other elements of the structure settled more. This differential settlement caused the cracks in the columns. The structure was later demolished to avoid any mishaps. So, it is very important that the soil types and properties are determined properly to recommend and design the best suited foundation for the soil.

There are different kind of foundations depending upon the type of soil present at the site. If the soil has strong bearing capacity, and the building is not very large, so spread footing can be used, but in some cases, when either the load coming to the foundation is very large or the soil does not have sufficient bearing capacity, deep foundations are necessary. These foundations have different benefits as they transfer the load to the deep bedrock and provide the resistance to the load due to the friction between the soil along the complete length. There are also some supports which are provided to retain the load acting during the construction, like the sheet piles which are provided to hold the earth pressure while the construction is under process. The loads acting on all these foundations are different, so, the resulting moment and shear diagrams are also very different. This difference can lead to the completely different design processes for all these foundations.

Sheet piles are used mostly to hold the soil back while the construction is being done. This avoids the additional load acting on the structural elements before they gain their strength by setting. These sheets are also used to avoid the water coming to the construction sites where the water table is too high. The excavation can lead to the collection of water in the excavation pit causing huge trouble. So, the sheet piles are also installed to hold the water back. The effectiveness of this watertightness depends on the craftsmanship of the people installing the sheets. If correctly installed, these sheets can be extremely efficient as shown in the following diagram.



Figure 1: Watertightness of Sheet Piles

As explained above, the shear force and bending moment diagrams for the structural elements are very important as they can demand different strength. So, the generation of these diagrams along with the deformation is very important. These values of deformation, moment and shear are then factorized as per the regulations. These values help in the prediction of failure and factor of safety. So, this study is based on the shear force and bending moment diagram of the sheet piles.

2. Literature Review

Sheet piles are used frequently in the construction. These sheets are made of continuously interlocked pile segments which are basically designed to resist the lateral forces (Stanislav. 2006). There are different kind of materials which are used to make these sheet piles. The most common ones are steel, aluminum, wood, and concrete. There are several types of sheet piles. These include anchored sheet piles, cantilevered sheet piles, and cofferdam sheet piles (Kalantari. 2011).

The purpose of sheet piles can be different in different scenarios. In some cases, where there are existing buildings in the surroundings, it is important that the excavation is supported in such a way that the existing buildings are not disturbed (Sobala, 2017). There are some cases when it is necessary to lower the groundwater table inside the excavation area without changing the hydrological properties of the regime beyond the area of the construction (Sobala, 2017), in these cases, sheet piles can be helpful.

The moment of inertia of the sheet piles is very important in controlling the displacement of the sheets, however Young's Modulus causes little to no effect on this parameter. The soil parameters are also slightly irrelevant when it comes to the pressure distribution on the sheet piles (Tang, 2018). The analysis and the determination of the distribution of the forces and the resulting displacements is a very complex process as the constitutive laws of the soils are not so straightforward. So, some special methods can be used for this purpose. One of these methods is Soil Structure Interface (SSI) method (An-Bin Huang).

2.1 Problem Statement

The analysis and design of laterally loaded sheet piles is a complex process due to complex soil constitutive laws. The displacement is critical in these elements. So, it is necessary that the correct shear and bending forces acting on the sheet piles are determined. There are different methods which are used for the determination of these forces. Some of these methods are quite complex and require lengthy calculation process. However, if such a program is developed which could provide the different forces' diagrams for the sheet piles by providing the soil properties, it could simplify the process. There are several other programs available for the determination of these diagrams. These programs are based on different methods, so, the accuracy of these programs is

also dependent on the method being used. However, Soil Structure Interaction (SSI) method has never been used in any of the programs. The use of this method for the development of this program can also provide another reference for the shear force and bending moment diagrams of the sheet piles.

2.2 Soil Structure Interaction Method (SSI)

The loading on the sheet piles is different. The analysis and design of these sheets require utmost care and understanding. A small mistake can cause huge damage and the loss in term of money. Soil structure interaction (SSI) method is one of the methods used to design the laterally loaded flexible structures. There are different assumptions which are made to solve the sheet piles as the beams which are laterally loaded. Similarly, the resistance provided by the sheet is modelled as coming from the series of the springs.

2.3 Assumptions

The analysis of any structure uses some assumptions as the ground conditions are not always ideal. The process used in the development is based on the following assumptions.

- 1) The sheet piles are assumed to be fixed at the base
- 2) The section of the sheet piles is considered constant with a constant moment of inertia over the complete depth of the sheet piles
- 3) A single soil type has been assumed for the analysis. Multiple soil types have been avoided for the analysis
- 4) The soil layer is considered to have a constant thickness over the depth of the investigation
- 5) The soil properties are homogeneous, and they remain constant over the entire depth of the soil
- 6) The effect of ground water table can be inculcated by selecting the depth of ground water table, and the saturated specific weight of the soil below ground water table

3. Methodology

3.1 Winkler Spring Model

Soil Structure Interaction method is based on Winkler spring model. Winkler model is used to model the behavior of the soil using different springs, whose stiffness is considered based on the properties of the soil. The sheet piles can be divided into different sections, where each section can be considered as a beam. The beam has the boundary conditions according to the situation at the site. In our case, the beam can be considered fixed at the bottom. The active earth pressure acting on the sheet piles can be considered as a uniform load acting on the beam. The soil ahead of the sheet piles is considered acting as springs which let us use the Winkler spring model. The resistance depends on the soil properties, and it is also linked directly to the displacement. When the displacement of the sheet piles increase, the resistance provided by the springs also increase. Initially, this method used the soil's resistance as springs, with the extended numerical modelling, the beams can also be considered resting on nonlinear soils. The anchors can also be provided which should also be included in the model. The flexibility of the sheet piles along with the anchors used make it difficult to describe the behavior using the limit equilibrium method.

SSI method is also linked to p-y method as the resistance provided by the sheet piles is also related to the displacement. In the name, p represents the resistance offered by the soil, whereas y represents the displacement occurred at the sheet pile under the lateral load. This method is basically used for the laterally loaded piles but can also be used for sheet piles by considering a series of single piles placed closely in a line. For this method, the piles are considered column-beams which have axial as well as the transversal load. The p-y curve can be nonlinear which suggests that the behavior of the soil is influenced by the displacement of the piles. A typical representation of the p-y curves shown for a pile is given in the following diagram (An-Bin Huang). The Model shows the resistance of the soil as springs with the p-y curve for each spring.

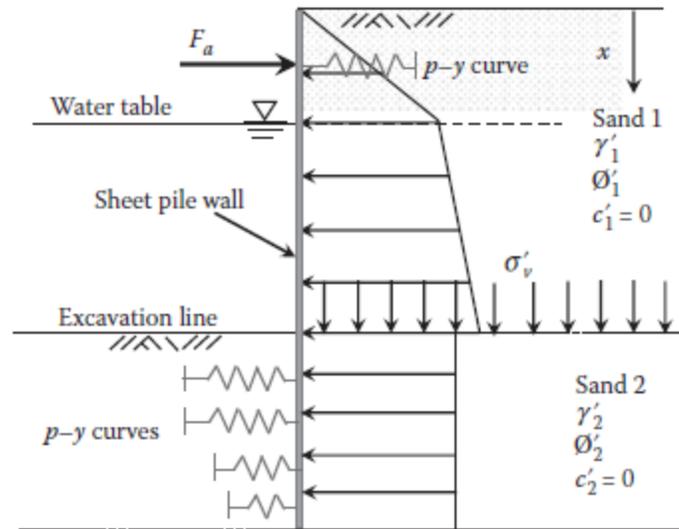


Figure 2: SSI method schematic diagram

The anchors are modelled above the excavation level with p-y model at each location of the anchor. On the other hand, the soil under the excavation level is modelled to provide the passive resistance. A differential equation can be used to solve these kinds of problems.

$$E_p I_p \frac{d^4 y}{dx^4} + P_x \frac{d^2 y}{dx^2} + E_{py} y - w = 0$$

In this equation

E_p is the modulus of elasticity of the pile

I_p is the moment of inertia of the pile

E_{py} is the modulus of the soil

P_x is the axial load

w is the uniform load acting on the beam

In the equation, the soil resistance is represented by the term $E_{py}y$ which shows the direct relationship between the resistance and the displacement y . The uniform load is the active earth

pressure which is assumed to be acting on the beam. The modelling is simplified by the assumption that the active earth pressure acts on both side of the sheet pile below the excavation level. So, in this case, the pressure on each side cancels the other one, and the resulting pressure is by the only soil which is above the excavation level.

The differential equation shown above can be solved using finite difference method. In most of the cases, the sheet piles are only loaded laterally. So, there is no axial load.

$$Px = 0$$

The solution of the equation is then simplified as

$$\begin{aligned} & (R_p)_{m-1} y_{m-2} + \{-2(R_p)_{m-1} - 2(R_p)_m\} y_{m-1} + \{(R_p)_{m-1} + 4(R_p)_m + (R_p)_{m+1} \\ & + (E_{py})_m h^4\} y_m + \{-2(R_p)_m - 2(R_p)_{m+1}\} y_{m+1} + (R_p)_{m+1} y_{m+2} - w_m h^4 \\ & = 0 \end{aligned}$$

Here, m represents the point for which the equation has been generalized. So, y_m is the displacement of the point m, and w_m is the load applied at that generic point. This equation can be written for multiple points to get a set of equations which can be solved to obtain the complete solution.

3.2 Boundary Conditions

The solution of the above equation requires boundary conditions. It is important that the boundary conditions are linked with the displacement as it makes the solution easy. The boundary conditions can be provided in terms of shear, moment, or slope. All these parameters can be linked with the displacement in the following manner.

$$Shear = V = \frac{EId^3y}{dx^3}$$

$$Moment = M = \frac{EId^2y}{dx^2}$$

$$\text{Slope} = S = \frac{EI dy}{dx}$$

The conditions can be defined for both bottom and top of the sheet pile.

3.2.1 Boundary Conditions at Top

At the top of the sheet pile, the moment and shear can be taken as zero for the simplest case. The difference equations can be written for both moment and shear as

$$y_{t-2} - 2y_{t-1} + 2y_{t+1} - y_{t+2} = \frac{2V_t h^3}{(R_p)_t} = 0$$

$$y_{t-1} - 2y_t + y_{t+1} = \frac{2M_t h^2}{(R_p)_t} = 0$$

Here, V_t and M_t are shear and moment, respectively. R_p is the flexural rigidity of the sheet pile. According to the assumptions, this value remains constant across the complete length of the sheet pile.

3.2.2 Boundary Conditions at Bottom

At the bottom of the sheet pile, again the moment and shear will be zero to consider the simple case. The equations can be written as

$$y_{-2} - 2y_{-1} + 2y_1 - y_2 = 0$$

$$y_{-1} - 2y_0 + y_1 = 0$$

The first equation represents the condition of the moment, whereas the second equation represents the condition of the shear at the base of the sheet pile. These equations are also the result of the assumption of zero axial force.

These multiple equations resulting from the boundary conditions can be solved simultaneously to get the different variables involved in these equations in terms of the displacement y_m . The resulting explicit equations after the solution can be written as

$$S_m = \frac{(y_{m-1} - y_{m+1})}{2h}$$

$$V_m = \frac{(R_p)_m (y_{m-2} - 2y_{m-1} + 2y_{m+1} - y_{m+2})}{2h^3} + \frac{P_x (y_{m-1} - y_{m+1})}{h}$$

$$M_m = \frac{(R_p)_m (y_{m-1} - 2y_m + y_{m+1})}{h^2}$$

3.3 MATLAB coding

The complex equations written for multiple points are quite difficult to be solved manually. So, the solution requires either an excel worksheet or some coding language. In case excel worksheet, we have to prepare new worksheet depending on the length of the pile, whereas in MATLAB we can make a software that would work for any length of piles. So for the development of the program, MATLAB has been used. The equations were solved using MATLAB, and eventually, the shear, bending, and deflection diagrams were drawn for the sheet piles.

To draw the diagrams, all the equations specified above were written in matrices form. These equations correspond to n number of points, where each point represents a different shear, moment, and deflection. The equations can be written as

$$AX = B$$

Here,

A is the matrix of all the coefficients that are multiplied with the displacement at that point

X is the matrix containing the deformation of all the point

B is the matrix comprising of the righthand side of all the equations

The modulus of subgrade below the excavation level has also been included in matrix B. In the similar manner, the resistance provided by the anchors is also added in the B matrix.

User Interface

A simple user friendly interface has been added in the program for the ease of providing the input data. The interface looks like as shown in the following diagram

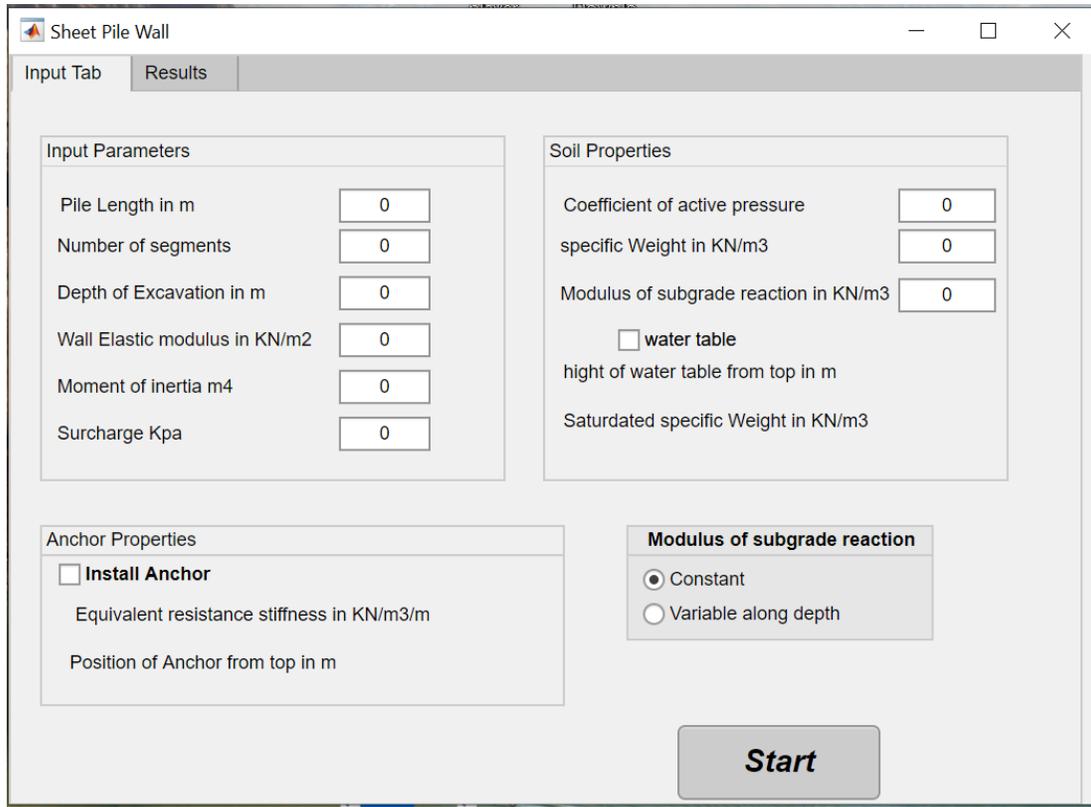


Figure 3: User Interface

User has the option to provide the necessary data for the piles. Different parameters of the piles can be given like the length and the moment of inertia. The effect of geometry of the sheet pile is included in the calculation through the moment of inertia. However, the user has to provide the moment of inertia as the program does not calculate it automatically.

The program also allows the user to install anchors to the sheet piles. If the user chooses, he can analyze the sheet pile without the anchors. By selecting the anchors, it is necessary to provide the depth of the anchor from the top of the sheet pile along with Equivalent resistance stiffness.

Soil properties can also be given by the user. The program only considers one soil type; however, the effect of water table can be included in the calculations. If water table is included into the analysis, the saturated weight of the soil should also be provided. The modulus of subgrade

reaction of the soil is also necessary to be provided as it affects the deformation of the pile. The calculations can be done either by considering a constant modulus of subgrade reaction, or a variable modulus of subgrade reaction along the depth of the sheet pile.

After providing the input data, the user has to click on “START” button, which will calculate the necessary parameters, and take the user to another “RESULTS” tab. This tab has the results of the analysis. The screenshot of this tab is shown in the following diagram.

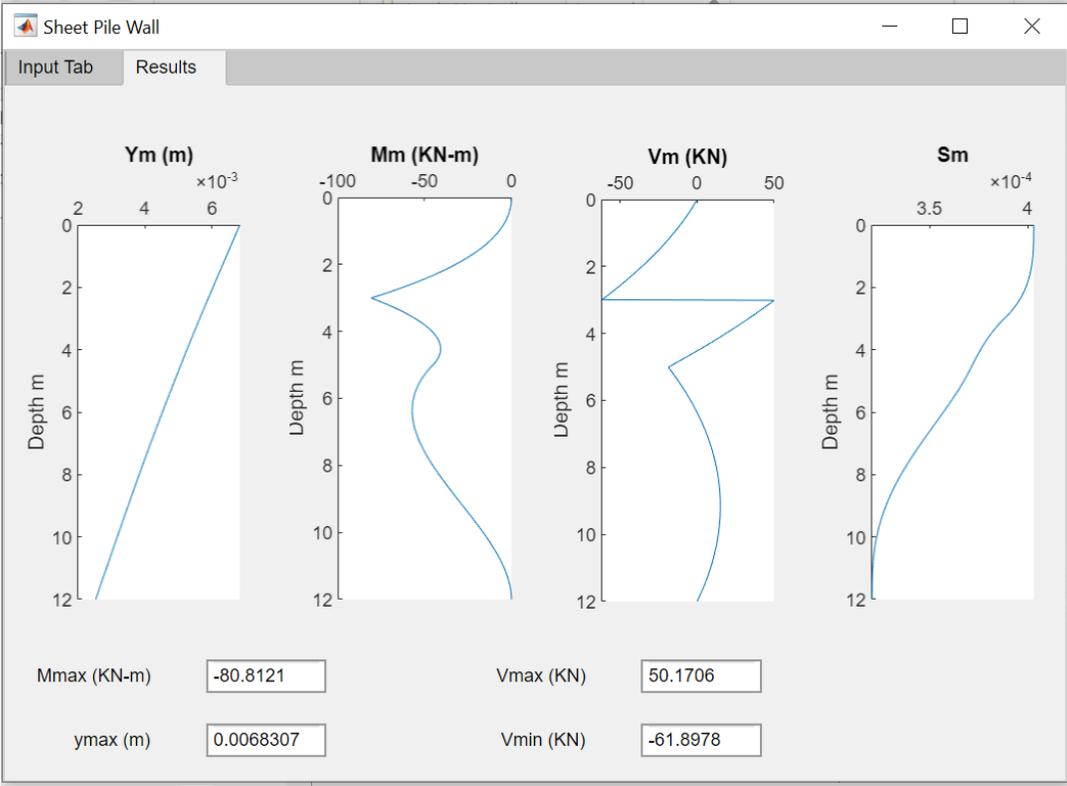


Figure 4: Results' Tab

The tab provides the diagrams for shear force, bending moment, deformation, and slope of the sheet pile. The units for these parameters are KN, KN-m, meters, and radian, respectively. The program also provides the maximum values of all the parameters mentioned above.

4. Comparison with Excel Sheet Results

The results obtained from the built software can be verified by the excel sheet which was developed to solve for the unknown parameters for the pile sheets using the same method. The results are being verified for given soil parameters and the sheet pile wall properties. The results are being summarized in the following paragraphs.

The conditions for the sheet piles are shown in the following diagram.

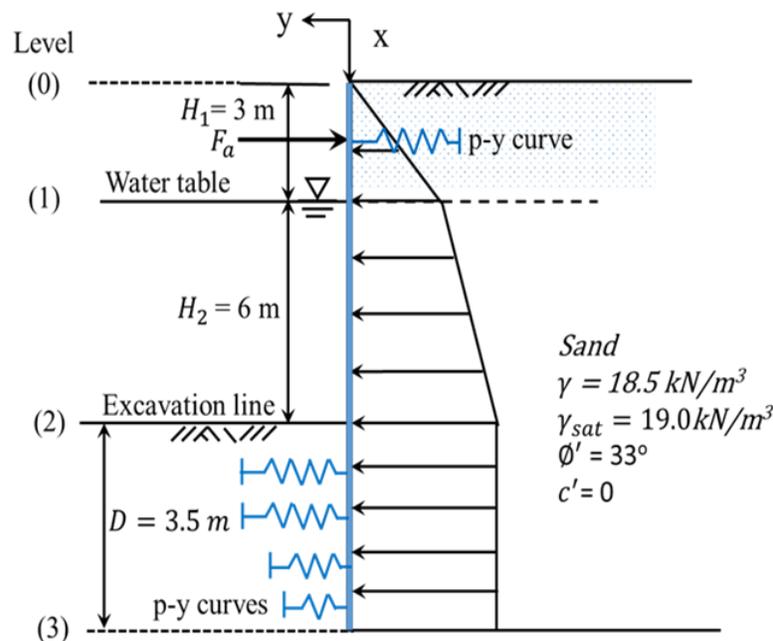


Figure 5: Comparison Problem

The case shows that the total depth of sheet piles is 12.5m. The ground water table is 3m from the top. There is an anchor in the middle of top 3m, the strength of which is being modelled as a spring. There is a depth of 3.5m of soil which is providing the passive resistance. The soil present at the site is sand having a unit weight of 18.5 kN/m^3 , whereas the saturated soil weight becomes 19 kN/m^3 . An important point to notice here is that the excel sheet also has the limitation of soil types. The sheet cannot solve multiple soil types. The resistance of the passive soil is also being modelled as springs.

The resistance provided by the passive earth pressure can also be shown as below.

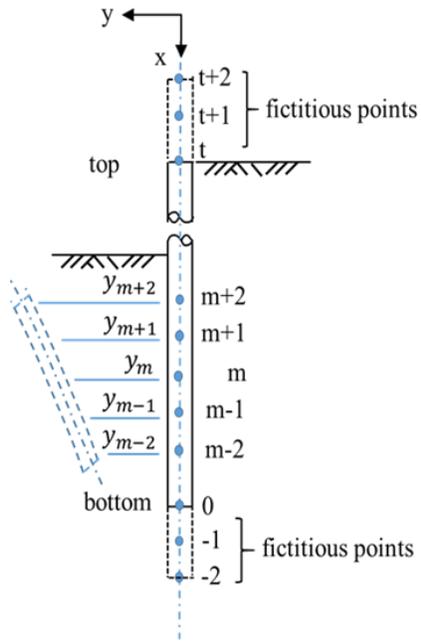


Figure 6: Fictitious points and springs

The piles are resisting the active pressure on one side, whereas on the other side, there is some depth of soil which is providing passive earth pressure. The soil parameters are shown in the following table. The definition of these parameters is given in Figure 5.

Table 1: Parameters

Variable	Value	Unit
g	18.5	kN/m^3
g_{sat}	19	kN/m^3
c'	0	kN/m^2
H	9	m
S_{all}	200	MPa
H_1	3	m
H_2	6	m
g'	9.19	kN/m^3
f'	33	degree
f'	0.576	radian
K_a	0.295	-

The given problem has been solved with the excel sheet and the software. The results for both cases are given here.

4.1 Excel Sheet Results

The results obtained at different depths by dividing the complete depth of the sheet into different segments is given below.

Table 2: Results from Excel Sheet

Depth m	Depth m	Segment	w_m kN/m	E_{pym} kN/m ³ /m	y_m m	am6:am35 Right side	F_a kN/m
	1	27	0	0	-0.0043	0	
	0.5	26	0	0	-0.0033	0	
0	0	25	0	0	-0.0024	0	
0.5	-0.5	24	1.4	0	-0.0014	0.09	
1	-1	23	2.7	0	-0.0005	0.17	
1.5	-1.5	22	4.1	200000	0.0004	0.26	88.788
2	-2	21	5.5	0	0.0014	0.34	
2.5	-2.5	20	6.8	0	0.0023	0.43	
3	-3	19	8.2	0	0.0032	0.51	
3.5	-3.5	18	8.9	0	0.004	0.55	
4	-4	17	9.5	0	0.0048	0.60	
4.5	-4.5	16	10.2	0	0.0055	0.64	
5	-5	15	10.9	0	0.0061	0.68	
5.5	-5.5	14	11.6	0	0.0066	0.72	
6	-6	13	12.2	0	0.007	0.77	
6.5	-6.5	12	12.9	0	0.0073	0.81	
7	-7	11	13.6	0	0.0075	0.85	
7.5	-7.5	10	14.3	0	0.0076	0.89	
8	-8	9	15.0	0	0.0076	0.93	

8.5	-8.5	8	15.6	0	0.0075	0.98	
9	-9	7	16.3	4000	0.0074	1.02	
9.5	-9.5	6	16.3	4000	0.0072	1.02	
10	-10	5	16.3	4000	0.0069	1.02	
10.5	-10.5	4	16.3	4000	0.0067	1.02	
11	-11	3	16.3	4000	0.0064	1.02	
11.5	-11.5	2	16.3	4000	0.0061	1.02	
12	-12	1	16.3	4000	0.0058	1.02	
12.5	-12.5	0	16.3	4000	0.0056	1.02	
	-13	-1	0	0	0.0053	0	
	-13.5	-2	0	0	0.005	0	

These results can be processed to determine the deflection, moment, shear, and the slope of these points. The results are shown in the table below.

Table 3: Shear, Moment, Deflection and Slope from Excel

Depth	ym	Sm	Vm	Mm
m	m x 10⁻³	10⁻³	10⁻³ (MN)	10⁻³ (MN-m)
0	-2.378	-1.881	-6.70E-10	-5.14E-11
0.5	-1.437	-1.881	0.341	-3.89E-10
1	-0.497	-1.881	1.363	3.41E-01
1.5	0.444	-1.883	-19.129	1.36E+00
2	1.386	-1.866	-38.94	-1.88E+01
2.5	2.31	-1.812	-35.872	-3.76E+01
3	3.198	-1.723	-32.123	-5.47E+01
3.5	4.034	-1.604	-27.863	-6.97E+01
4	4.802	-1.457	-23.265	-8.25E+01
4.5	5.491	-1.289	-18.328	-9.30E+01
5	6.091	-1.102	-13.052	-1.01E+02
5.5	6.593	-0.903	-7.438	-1.06E+02

6	6.994	-0.697	-1.485	-1.08E+02
6.5	7.291	-0.49	4.807	-1.08E+02
7	7.484	-0.287	11.437	-1.04E+02
7.5	7.578	-0.095	18.406	-9.61E+01
8	7.579	0.079	25.714	-8.51E+01
8.5	7.499	0.228	33.36	-7.04E+01
9	7.351	0.346	33.994	-5.17E+01
9.5	7.153	0.431	27.645	-3.64E+01
10	6.92	0.489	21.726	-2.41E+01
10.5	6.664	0.526	16.296	-1.46E+01
11	6.394	0.547	11.392	-7.78E+00
11.5	6.117	0.558	7.036	-3.24E+00
12	5.836	0.562	3.237	-7.39E-01
12.5	5.555	0.563	0	9.02E-13

Based on these results, the respective diagrams can be drawn for the sheet piles. These diagrams are as

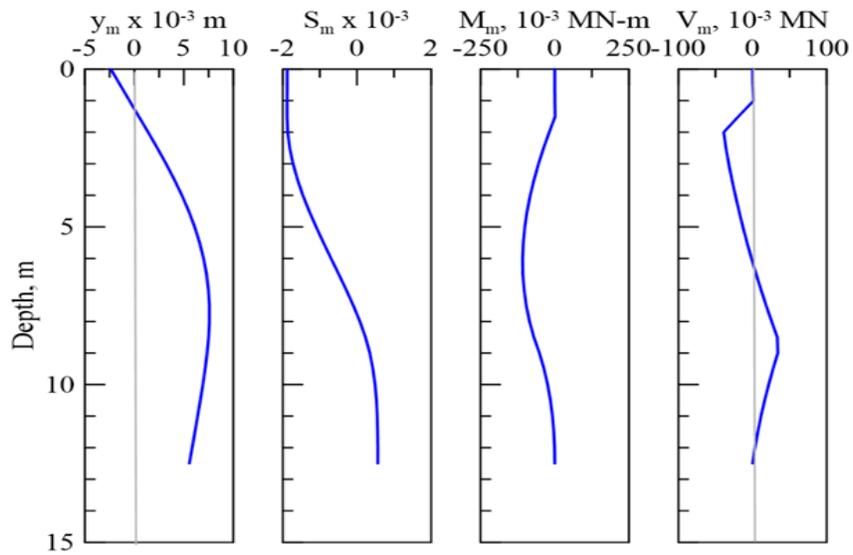
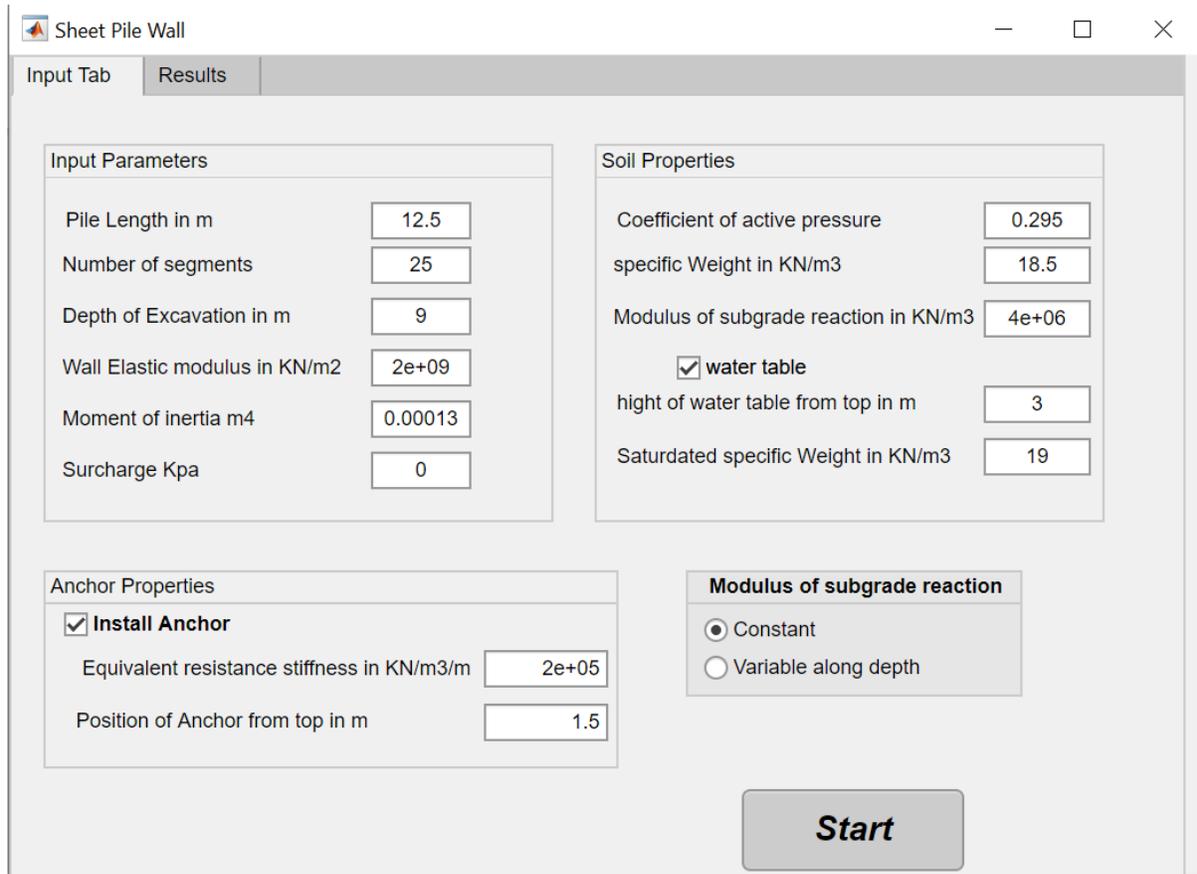


Figure 7: Diagrams from Excel Sheet

4.2 Results from New Software

The same set of sheet piles was also solved using the new software. The input parameters can be provided as shown in the following diagram.



The screenshot displays the 'Sheet Pile Wall' software interface with the 'Input Tab' selected. The interface is divided into several sections for parameter input:

- Input Parameters:**
 - Pile Length in m: 12.5
 - Number of segments: 25
 - Depth of Excavation in m: 9
 - Wall Elastic modulus in KN/m²: 2e+09
 - Moment of inertia m⁴: 0.00013
 - Surcharge Kpa: 0
- Soil Properties:**
 - Coefficient of active pressure: 0.295
 - specific Weight in KN/m³: 18.5
 - Modulus of subgrade reaction in KN/m³: 4e+06
 - water table
 - hight of water table from top in m: 3
 - Saturdated specific Weight in KN/m³: 19
- Anchor Properties:**
 - Install Anchor
 - Equivalent resistance stiffness in KN/m³/m: 2e+05
 - Position of Anchor from top in m: 1.5
- Modulus of subgrade reaction:**
 - Constant
 - Variable along depth

A large 'Start' button is located at the bottom center of the interface.

Figure 8: Input Parameters for Case 1 Validation

The software can accommodate the variation of water table. So, in this problem, this variation was considered, and the depth of ground water table was added. The results obtained can be shown in the following diagram.

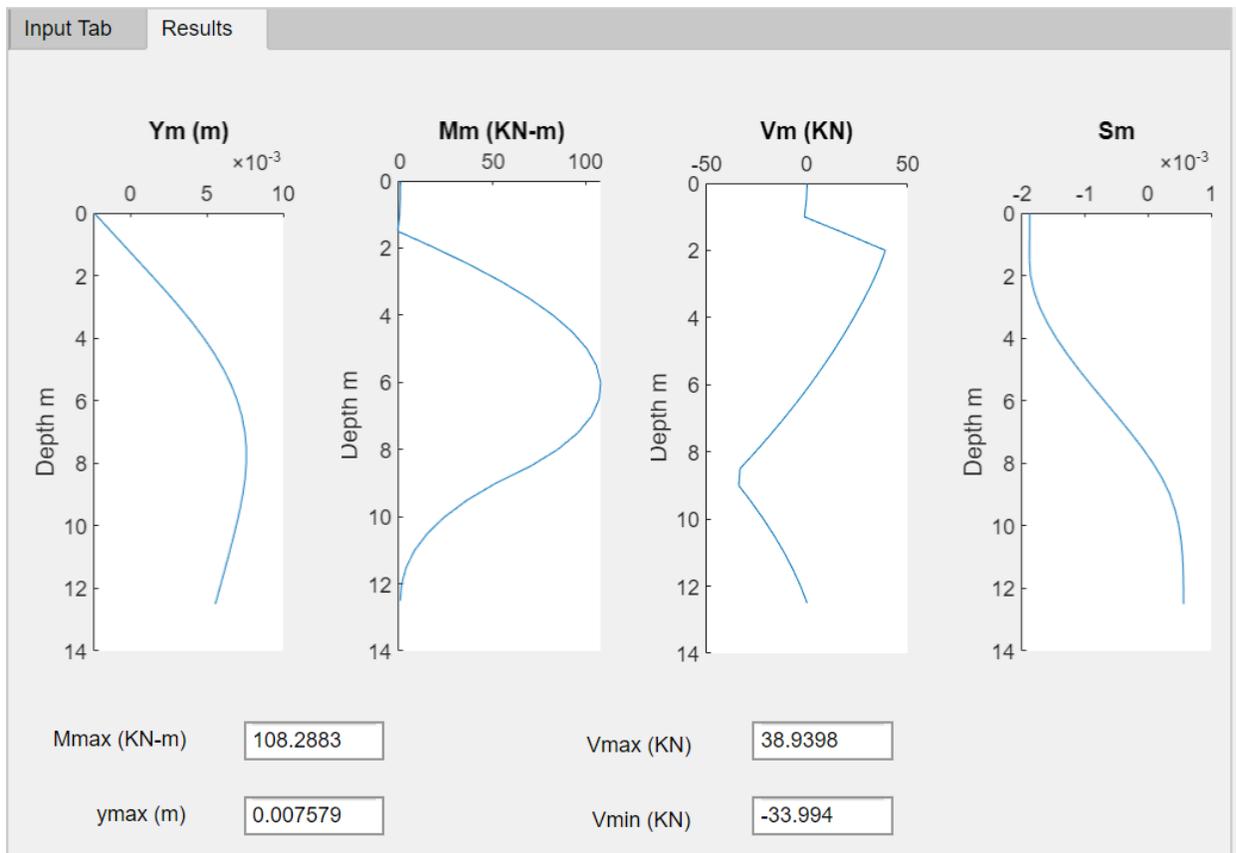


Figure 9: Diagrams from New Software

From the diagrams, it is clear that the results obtained from both the methods are identical. Although the change in the limits of the axis makes the diagrams look a little different, the values obtained are identical. The software also provides the maximum values of all the parameters, whereas the excel sheet does not provide this data separately.

5. Validation with Other Methods

Validation is an important step in the design of any software. The confidence in the software boosts a lot when the results obtained also match with the results of other software or methods. So, the results of the software will be compared with the results of Arnold Verruijt's software SPW. The error in both the results will be determined and compared for the accuracy of the results.

Arnold Verruijt's SPW Software

This software is based on the theory of beams on elastic supports which actually is an extension of the theory of beams. In this software A.Verruijt has accounted the effect of Elasto-plastic behavior of springs. The basic equation which is used in the theory of bending of beams is

$$EI \frac{d^4w}{dx^4} = f$$

Here,

f is the loading on the beam

w is the lateral displacement.

The theory can be extended by considering the supports to be elastic with the addition of the modulus of subgrade reaction, by considering this scenario, the above stated equation can be modified as

$$EI \frac{d^4w}{dx^4} = f - kw$$

Here,

k is the modulus of subgrade reaction.

Numerical models can be used to solve the differential equations to obtain the results of displacement, moment, and shear. In this section of the report, 2 different cases are considered which are solved with the SPW software (existing) which solves the sheet pile walls which the

beam theory, and the results are compared with the new developed software, which considers the SSI Method.

5.1 Case 1: No Anchors

This case involves the sheet pile wall having a length of 6.1m with a depth of excavation of 3m. To simplify the calculations, the moment of inertia for the wall section is provided rather than the dimensions. So, the moment of inertia of the wall is $2.3885 \times 10^{-4} \text{m}^4$. Moreover, the coefficients of active and passive soil pressure are also provided. The input data for the problem is shown in the following table.

Table 4: Input Data for Case 1 of Validation

Parameter	Value	Unit
Length of wall	6.1	m
Depth of Excavation	3	m
Elastic Modulus	2×10^8	kPa
Moment of Inertia	2.3885×10^{-4}	m^4
Modulus of Subgrade Reaction	11612	kN/m^3
Specific Weight	18	kN/m^3
Coefficient of Active Pressure	0.237	-
Coefficient of Passive Pressure	6.51	-

5.1.1 Results from SPW software

The input data in SPW software is provided according to its functionality. The data regarding the width and height of the wall along with the properties of the soil are provided according to the following figure.

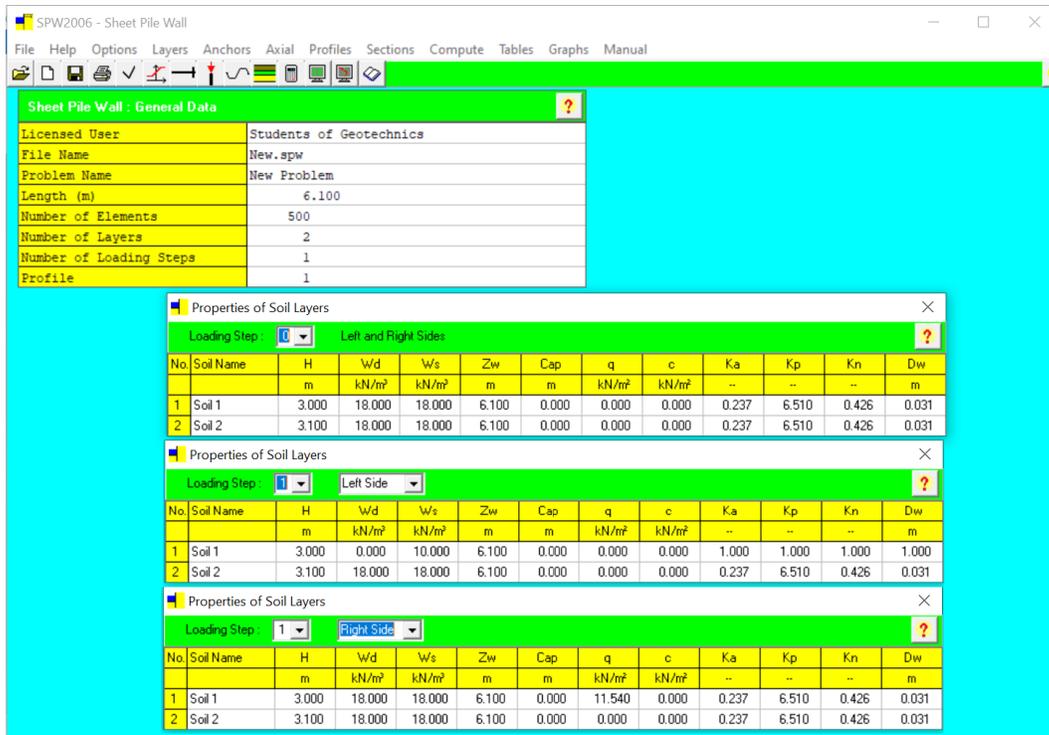


Figure 10: Input Parameters for SPW Software case 1

The results obtained through the software are shown in the following diagram.

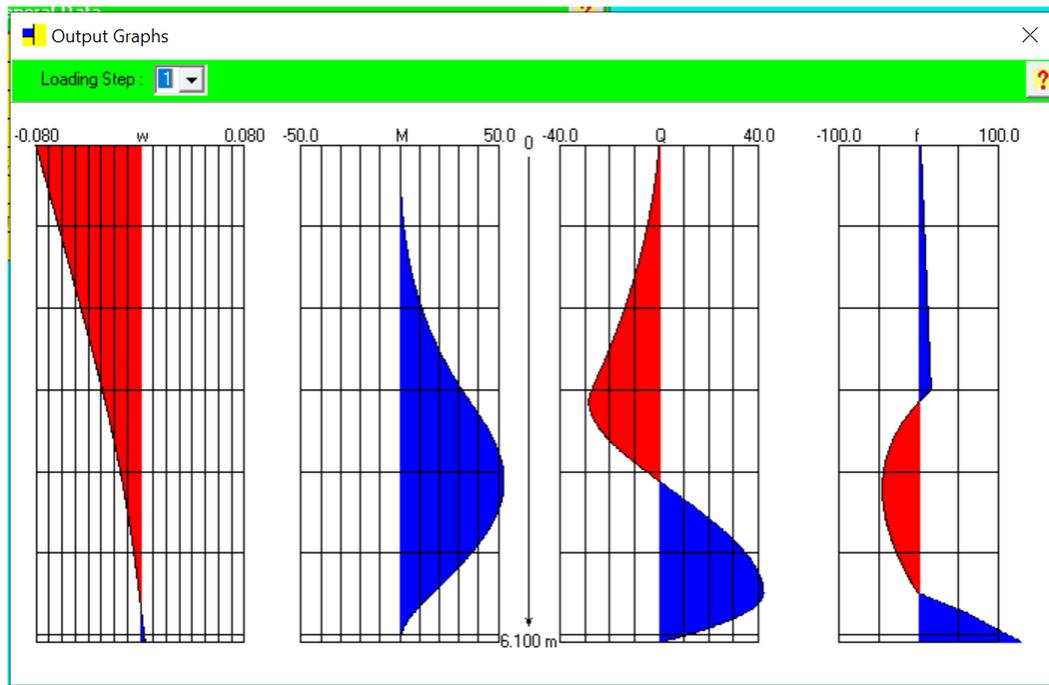


Figure 11: Results from SPW Software Case 1

From the diagram, the maximum load is applied at the top of the wall which gradually decreases to the bottom of the wall. With this load distribution, the maximum moment is at almost 2/3 of the wall's depth, where the shear is zero. The values of the maximum parameters are as

$$M_{max} = 53.03 \text{ MN.m}$$

$$V_{min} = 28.55 \text{ MN}$$

$$V_{max} = 42.35 \text{ MN}$$

5.1.2 Results from New Software

The same input parameters can also be provided in the new software, and the moment and shear diagrams can also be generated. The input parameters are provided as

The screenshot shows the 'Sheet Pile Wall' software interface with the following input parameters:

Parameter	Value
Pile Length in m	6.1
Number of segments	610
Depth of Excavation in m	3
Wall Elastic modulus in KN/m ²	2e+08
Moment of inertia m ⁴	0.000238
Surcharge Kpa	11.54
Coefficient of active pressure	0.237
specific Weight in KN/m ³	18
Modulus of subgrade reaction in KN/m ³	1.161e+0
water table	<input type="checkbox"/>
hight of water table from top in m	
Saturdated specific Weight in KN/m ³	
Install Anchor	<input type="checkbox"/>
Equivalent resistance stiffness in KN/m ³ /m	
Position of Anchor from top in m	
Modulus of subgrade reaction	<input checked="" type="radio"/> Variable along depth
Constant	<input type="radio"/>

A 'Start' button is located at the bottom right of the interface.

Figure 12: Input Parameter for New Software Case 1

The input parameters are a little different for both the software, as the coding of the software is different, and the methods used to solve the problems are also different. The results obtained from the new software are shown in the following diagram.

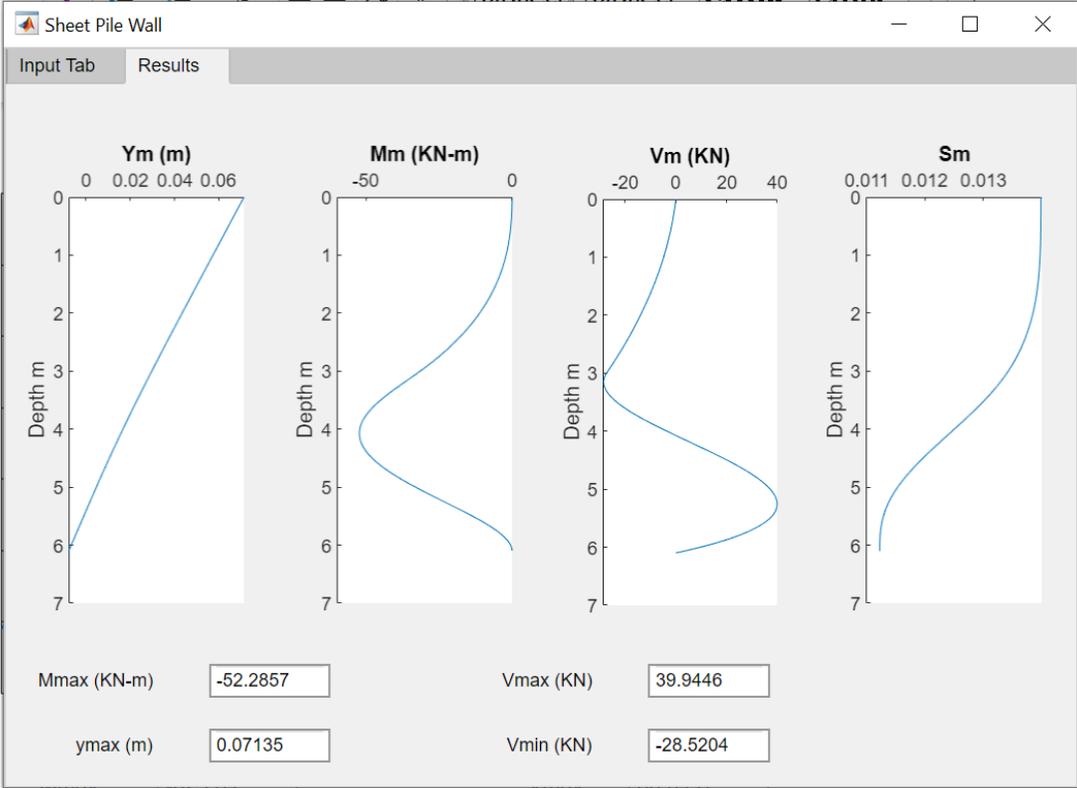


Figure 13: Results from New Software Case 1

The sign conventions used in the software are different, so, the sign of the output parameters are different, but the magnitude is of the same order. The percentage difference in the results obtained is as

Table 5: Percentage Difference in Results for Case 1

Difference in Results			
	Arnold Verruijt’s SPW	Sheet Pile Software	Difference (%)
M_{max} (MN.m)	53.03	-52.28	-1.43
V_{min} (MN)	-28.55	-28.52	0.11
V_{max} (MN)	42.35	39.95	6.01

The results obtained show a very small percentage difference in the values. The difference cannot be termed as an error because the methods used in both the cases are different, so, the difference in the results is because of the formulating equations for both the methods. As stated above, the results obtained from the excel sheet and the software in the section of “Excel Sheet Results” perfectly align.

5.2 Case 2: With Anchors

Another example can be considered with similar parameters with the addition of anchors. The length of the wall is again 6.1m with the excavation depth of 3m. The moment of inertia and the other parameters are same. Some additional parameters are added such as stroke which is a parameter used for SPW software for anchors. Similarly, the equivalent resistance stiffness is also provided as shown in the following table.

Table 6: Input Parameters for Case 2

Parameter	Value	Unit
Length of wall	6.1	m
Depth of Excavation	3	m
Elastic Modulus	2×10^8	
Moment of Inertia	2.3885×10^{-4}	m^4
Modulus of Subgrade Reaction	11612	kN/m^3
Specific Weight	18	kN/m^3
Coefficient of Active Pressure	0.237	-
Coefficient of Passive Pressure	0.237	-
Pre-stresses force	45	kN/m
Fa	300	kN/m
Stroke for Anchor	0.078	m
Equivalent Resistance Stiffness	2000	$MN/m^3/m$

5.2.1 Results from SPW Software

The input data can be provided into the SPW software as shown in the following diagram.

The screenshot displays the SPW2006 - Sheet Pile Wall software interface. The main window shows the 'Sheet Pile Wall : General Data' dialog box with the following input data:

Parameter	Value
Licensed User	al
File Name	wali.spw
Problem Name	New Problem
Length (m)	6.100
Number of Elements	500
Number of Layers	3
Number of Loading Steps	1
Profile	1

The 'Anchors' dialog box shows the following data for Loading Step 1:

No.	Depth (m)	Fx (kN/m)	Fa (kN/m)	Dw (m)
0	0.000	0.000	0.000	1.000
1	1.000	0.000	826.220	-0.215
2	3.500	0.000	0.000	1.000
3	6.100	0.000	0.000	1.000

Three 'Properties of Soil Layers' dialog boxes are shown, each for a different loading step and side:

- Loading Step: 0, Left and Right Sides:**

No.	Soil Name	H (m)	Wd (kN/m²)	Ws (kN/m²)	Zw (m)	Cap (m)	q (kN/m²)	c (kN/m²)	Ka	Kp	Kn	Dw (m)
1	Soil 1	1.000	18.000	18.000	6.100	0.000	0.000	0.000	0.237	6.510	0.426	0.026
2	Soil 2	2.500	18.000	18.000	6.100	0.000	0.000	0.000	0.237	6.510	0.426	0.026
3	Soil 3	2.600	18.000	18.000	6.100	0.000	0.000	0.000	0.237	6.510	0.426	0.026
- Loading Step: 1, Right Side:**

No.	Soil Name	H (m)	Wd (kN/m²)	Ws (kN/m²)	Zw (m)	Cap (m)	q (kN/m²)	c (kN/m²)	Ka	Kp	Kn	Dw (m)
1	Soil 1	1.000	18.000	18.000	6.100	0.000	11.540	0.000	0.237	6.510	0.426	0.026
2	Soil 2	2.500	18.000	18.000	6.100	0.000	0.000	0.000	0.237	6.510	0.426	0.026
3	Soil 3	2.600	18.000	18.000	6.100	0.000	0.000	0.000	0.237	6.510	0.426	0.026
- Loading Step: 1, Left Side:**

No.	Soil Name	H (m)	Wd (kN/m²)	Ws (kN/m²)	Zw (m)	Cap (m)	q (kN/m²)	c (kN/m²)	Ka	Kp	Kn	Dw (m)
1	Soil 1	1.000	0.000	10.000	6.100	0.000	0.000	0.000	1.000	1.000	1.000	1.000
2	Soil 2	2.500	0.000	10.000	6.100	0.000	0.000	0.000	1.000	1.000	1.000	1.000
3	Soil 3	2.600	18.000	18.000	6.100	0.000	0.000	0.000	0.237	6.510	0.426	0.026

Figure 14: Input Data for SPW Software Case 2

The analysis of the sheet pile wall with this software yields the following results.

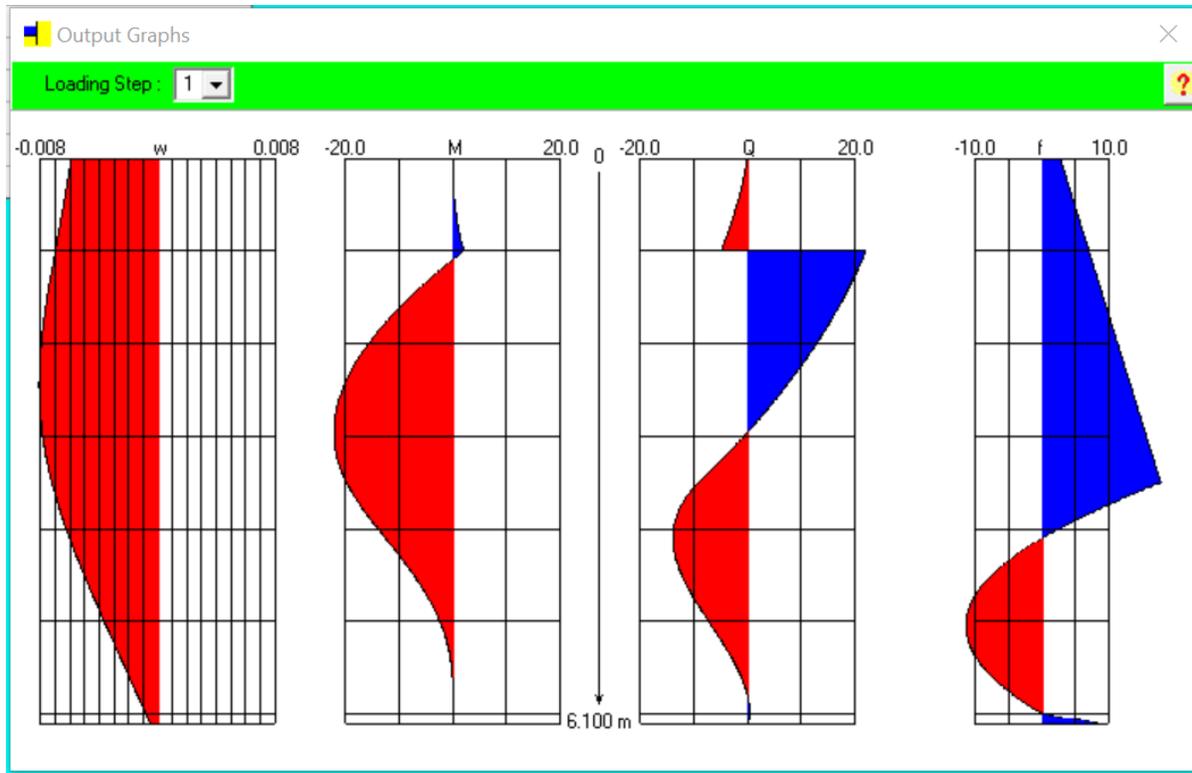


Figure 15: Results from SPW Software Case 2

The extreme values of some of the parameters are shown below.

$$M_{max} = 22.21 \text{ kNm}$$

$$V_{max} = 21.90 \text{ kN}$$

$$V_{min} = -12.96 \text{ kN}$$

These values will be compared with the results obtained from the new software.

5.2.2 Results from New Software

The input parameters can be provided in the new software. To provide the data regarding the anchors, the tab for install anchors can be checked which allows to provide the equivalent resistance stiffness for the anchor, and its position from the top of the sheet pile wall. The tab for input parameters looks like shown in the following diagram.

Sheet Pile Wall

Input Tab Results

Input Parameters

Pile Length in m

Number of segments

Depth of Excavation in m

Wall Elastic modulus in KN/m²

Moment of inertia m⁴

Surcharge Kpa

Soil Properties

Coefficient of active pressure

specific Weight in KN/m³

Modulus of subgrade reaction in KN/m³

water table

hight of water table from top in m

Saturdated specific Weight in KN/m³

Anchor Properties

Install Anchor

Equivalent resistance stiffness in KN/m³/m

Position of Anchor from top in m

Modulus of subgrade reaction

Constant

Variable along depth

Start

Figure 16: Input Parameters for New Software Case 2

The results obtained from the new software are shown in the following diagram.

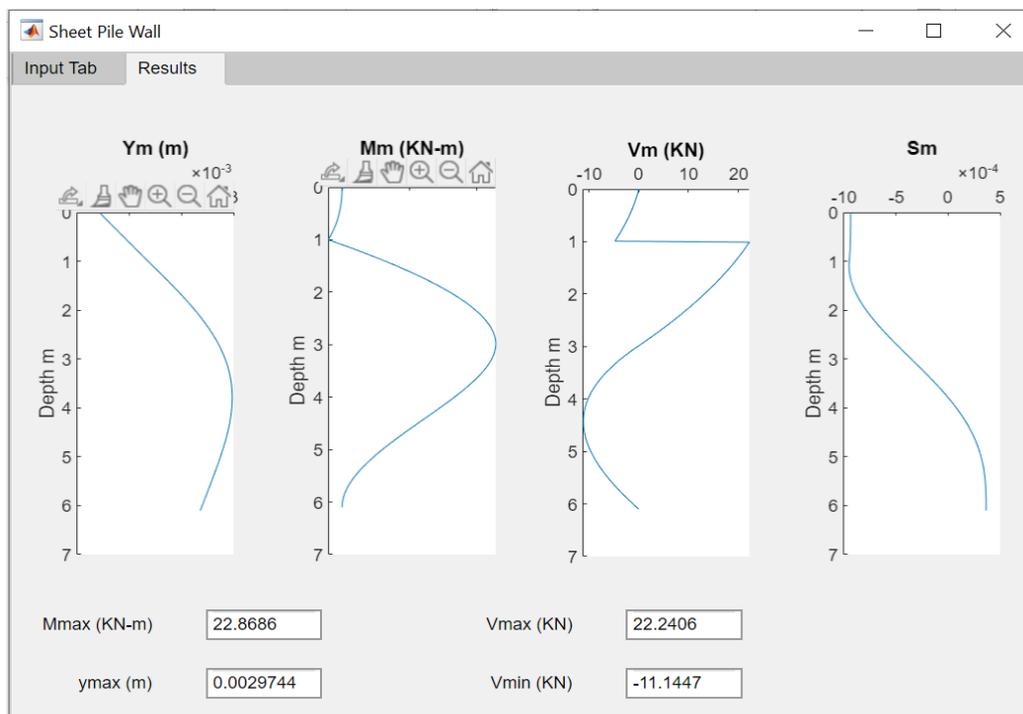


Figure 17: Results from New Software Case 2

The difference of sign convention shifts the diagram in the opposite direction in both the cases. The difference in the values can be summarized in the following table.

Table 7: Percentage Difference in Results for Case 2

Difference in Results			
	Arnold Verruijt's SPW	Sheet Pile Software	Difference (%)
M_{\max} (KNm)	22.21	22.87	2.90
V_{\min} (KN)	-12.96	-11.15	16.25
V_{\max} (kKN)	21.90	22.24	1.54

The moment and maximum shear shows a negligible difference in the results obtained from both the methods. The difference is quite significant when the minimum shear is considered, this is because the values of shear are smaller as compared to the maximum moment and maximum shear. So, the difference is just 1.8KN, but because of being smaller values, this difference becomes significant in percentage.

6. Conclusion

The above report summarizes the process and methodology involved in the development of the software. The software is based on the Soil Structure Interaction method which deals with the behavior of the soil when it interacts with any structural element. There are several methods to analyze and design the sheet pile walls such as limit equilibrium method and finite element method. There are also different programs based on these methods, but there is no program based on soil structure interaction method. The development of this program was based on soil structure interaction method some assumptions such as the soil type, soil layer thickness, and soil properties were assumed to be constant over the complete depth of the wall. Moreover, the section of the wall was considered also constant. MATLAB coding was used to generate an app that solves the multiple equations generated from the SSI method for a given sheet pile. The program also considered the anchoring of the sheet pile walls according to the conventions of SSI method. The results obtained from the software were first compared with the same method using the Excel spreadsheet available for the calculation, and then, with the other methods.

The Excel spreadsheet results showed that the results obtained from the software are identical to those of the spreadsheet. The error was in the magnitude of 10^{-11} , which is negligible. The results were also compared with the SPW software which is based on the beam theory on the elastic foundation. The results obtained from these two methods cannot be identical because the methods used in the software are not the same. So, a slight difference in the numbers is expected because of these different methods.

For the case where the sheet pile did not have anchors, the maximum difference was found to be 6.01% which occurred in the maximum shear force. For minimum shear and maximum moment, the difference was very low. So, this difference can be attributed to the use of different methods which have different assumptions for the modelling of the sheet pile walls. For the second case, the anchors were also provided to compare the results. In this case, the maximum difference was 16.25% which occurred for minimum shear. For maximum shear and moment, the difference was again negligible. This difference occurs because of the handling of anchors in different ways for both the methods. For SSI method, the anchors are considered as springs having equivalent resistance, whereas for the SPW software, there is a separate parameter “Stroke” which considers

the effect of the anchors. So, due to different assumptions and numerical methods, some difference is expected between the values. So, the results obtained are satisfactory, and ensures the safe use of the software for the future analytical purposes.

7. Future Recommendations

There are several aspects of the program that can be improved. The software has the limitation of using only one soil type. This limitation can be removed in future where the software will be able to solve the sheet pile walls having several soil types. Similarly, the properties are kept constant for the soil throughout the depth. This can also be dealt. The section of the sheet pile wall is kept constant without any changes. There can be some scenarios where the constant section throughout the depth might not be optimal economically, so, the sheet pile wall with a varying section can also be considered for future work.

8. References

- 1) Tang, L., Cong, S., Xing, W., Ling, X., Geng, L., Nie, Z., & Gan, F. (2018). Finite element analysis of lateral earth pressure on sheet pile walls. *Engineering Geology*, 244, 146-158.
- 2) Huang, A. B., & Yu, H. S. (2017). *Foundation engineering analysis and design*. CRC Press.
- 3) Verruijt, A. (1995). *Computational geomechanics* (Vol. 7). Springer Science & Business Media.
- 4) M.Hetenyi, Beams on elastic foundation, University of Michigan Press, Ann Arbor, 1946
- 5) Eskandari, L., & Kalantari, B. (2011). Basic Types of Sheet Pile Walls and Their Application in the Construction Industry--a Review. *Electronic Journal of Geotechnical Engineering*, 16.
- 6) Sobala, D., & Rybak, J. (2017, October). Steel Sheet Piles--Applications and Elementary Design Issues. In *IOP Conference Series: Materials Science and Engineering* (Vol. 245, No. 2, p. 022072). IOP Publishing.
- 7) King, G. (1995). Analysis of cantilever sheet-pile walls in cohesion less soil, *Journal of Geotechnical Engineering*, vol. 121, no. 9.
- 8) Day, R.A. (1999). Net pressure analysis of cantilever sheet pile walls, *Geotechnique*, vol. 49, No. 2, pp. 231-45
- 9) Day, R. A. & Potts, D. M. (1993). Modelling sheet pile retaining walls. *Computers Geotechnics* 15, 125-143.
- 10) Bowles, J.E. (1996). *Foundation analysis and design*, McGraw-Hill, New York

9. Annex

```
10. classdef app2 < matlab.apps.AppBase
11.
12. % Properties that correspond to app components
13. properties (Access = public)
14. ShetPileWall matlab.ui.Figure
15. TabGroup matlab.ui.container.TabGroup
16. InputTab matlab.ui.container.Tab
17. AnchorPropertiesPanel matlab.ui.container.Panel
18. InstallAnchorCheckBox matlab.ui.control.CheckBox
19. EquivalentresistancestiffnessinKNm3mEditFieldLabel matlab.ui.control.Label
20. EquivalentresistancestiffnessinKNm3m matlab.ui.control.NumericEditField
21. PositionofAnchorfromtopinmEditFieldLabel matlab.ui.control.Label
22. PositionofAnchorfromtopinm matlab.ui.control.NumericEditField
23. InputParametersPanel matlab.ui.container.Panel
24. PileLengththinmEditFieldLabel matlab.ui.control.Label
25. PileLengththinm matlab.ui.control.NumericEditField
26. DepthofExcavationinmEditFieldLabel matlab.ui.control.Label
27. DepthofExcavationinm matlab.ui.control.NumericEditField
28. NumberofsegmentsLabel matlab.ui.control.Label
29. numberofsegments matlab.ui.control.NumericEditField
30. WallElasticmodulusinKNm2EditFieldLabel matlab.ui.control.Label
31. WallElasticmodulusinKNm2 matlab.ui.control.NumericEditField
32. Momentofinertiam4EditFieldLabel matlab.ui.control.Label
33. Momentofinertiam4 matlab.ui.control.NumericEditField
34. SurchargeKpaEditFieldLabel matlab.ui.control.Label
35. SurchargeKpa matlab.ui.control.NumericEditField
36. SoilPropertiesPanel matlab.ui.container.Panel
37. CoefficientofactivepressureLabel matlab.ui.control.Label
38. FrictionAngleinDegrees matlab.ui.control.NumericEditField
39. specificWeightinKNm3EditFieldLabel matlab.ui.control.Label
40. specificWeightinKNm3 matlab.ui.control.NumericEditField
41. watertableCheckBox matlab.ui.control.CheckBox
42. hightofwatertablefromtopinmEditFieldLabel matlab.ui.control.Label
43. hightofwatertablefromtopinm matlab.ui.control.NumericEditField
44. SaturdatedspecificWeightinKNm3EditFieldLabel matlab.ui.control.Label
45. SaturdatedspecificWeightinKNm3 matlab.ui.control.NumericEditField
46. ModulusofsubgradereactioninKNm3EditFieldLabel matlab.ui.control.Label
47. ModulusofsubgradereactioninKNm3 matlab.ui.control.NumericEditField
48. watertableCheckBox_2 matlab.ui.control.CheckBox
49. ModulusofsubgradereactionButtonGroup matlab.ui.container.ButtonGroup
50. ConstantButton matlab.ui.control.RadioButton
51. VariablealongdepthButton matlab.ui.control.RadioButton
52. StartButton matlab.ui.control.Button
53. ResultsTab matlab.ui.container.Tab
54. Sm1 matlab.ui.control.UIAxes
55. Vm1 matlab.ui.control.UIAxes
56. Ym1 matlab.ui.control.UIAxes
```

```

57.MmaxKNmEditFieldLabel matlab.ui.control.Label
58.MmaxKNmEditField matlab.ui.control.EditField
59.ymaxmEditFieldLabel matlab.ui.control.Label
60.ymaxmEditField matlab.ui.control.EditField
61.VmaxKNEditFieldLabel matlab.ui.control.Label
62.VmaxKNEditField matlab.ui.control.EditField
63.VminKNEditFieldLabel matlab.ui.control.Label
64.VminKNEditField matlab.ui.control.EditField
65.Mm1 matlab.ui.control.UIAxes
66.end
67.
68.% Callbacks that handle component events
69.methods (Access = private)
70.
71.% Button pushed function: StartButton
72.function StartButtonPushed(app, event)
73.L= app.PileLengthinm.Value;
74.seg= app.numberofsegments.Value;
75.h0= app.DepthofExcavationinm.Value;
76.Ep= app.WallElasticmodulusinKNm2.Value;
77.Ip= app.Momentofinertiam4.Value;
78.qd= app.SurchargeKpa.Value;
79.ka=app.FrictionAngleinDegrees.Value;
80.g =app.specificWeightinKNm3.Value;
81.Epy=app.ModulusofsubgradereactioninKNm3.Value;
82.wt= app.hightofwatertablefromtopinm.Value;
83.gsat=app.SaturdatedspecificWeightinKNm3.Value;
84.anchor=app.EquivalentresistancestiffnessinKNm3m.Value;
85.anchor_h=app.PositionofAnchorfromtopinm.Value;
86.if (app.ConstantButton.Value)
87.modulus = 0;
88.else
89.modulus =1;
90.end
91.%code
92.
93.h =L/seg; %segment size m
94.Rp =Ep*Ip ; %KN-m2
95.if gsat==0
96.gprime=g;
97.else
98.gprime = gsat-9.81;
99.end
100.
101.
102.     nr= (seg)+1;
103.     nc= (seg)+5;
104.
105.     depth=zeros(nc,1);
106.     wm = zeros(nc, 1);

```

```

107.     k1 = zeros(nc, 1);
108.
109.     for i=4:1:nc-2
110.     depth(i,1)= depth(i-1,1) + h;
111.     end
112.
113.
114.     for i=4:1:nc-2
115.     if depth(i,1)<h0
116.     if depth(i,1) <= wt
117.     wm(i,1) = ka*g*depth(i,1) +qd*ka;
118.     else
119.     wm(i,1) = ka*g*wt + ka*gprime*(depth(i,1)-wt) +qd*ka;
120.     end
121.     else
122.     wm(i,1) = ka*g*wt + ka*gprime*(h0-wt) + qd*ka;
123.     if modulus ==0
124.     k1(i,1) = Epy;
125.     else
126.     k1(i,1) = Epy * ((depth(i,1)-h0)/(L-h0));
127.     end
128.     end
129.     end
130.
131.     wm= wm*h^4 /Rp;
132.
133.
134.     k1((anchor_h/h+3) ,1) = anchor;
135.
136.     k1= k1*h^4 /Rp;
137.
138.     %1st two rows for boundary condition at top pile
139.     r1=[-1 2 0 -2 1]; %eq7.53
140.     r2=[0 1 -2 1 0]; %eq7.54
141.
142.     c1 = zeros(1, (nc-5));
143.     c2 = zeros(1, (nc-5));
144.
145.     r1 =[r1 c1];
146.     r2 =[r2 c2];
147.
148.     %last 2 rows for boundary condition at bottom pile
149.     rend1 = [0 1 -2 1 0]; %eq7.50
150.     rend2 = [-1 2 0 -2 1]; %eq7.52
151.
152.     cend1 = zeros(1, (nc-5));
153.     cend2 = zeros(1, (nc-5));
154.
155.     rend1 = [cend1 rend1];
156.     rend2 = [cend2 rend2];

```

```

157.
158.
159.
160.   %A* y =B
161.   A = zeros(nr,nc);
162.
163.
164.
165.   %FEM equation eq7.48
166.   for i = 1:1:nr
167.       const1 = 1; %y(m-2)
168.       const2 = -4 ; %y(m-1)
169.       const3 = 6 +k1(i+2,1); %y(m)
170.       const4 = -4 ; %y(m+1)
171.       const5 = 1; %y(m+2)
172.
173.       A(i,i) =const1;
174.       A(i,i+1) =const2;
175.       A(i,i+2) =const3;
176.       A(i,i+3) =const4;
177.       A(i,i+4) =const5;
178.   end
179.
180.   A = [r1;
181.       r2;
182.       A;
183.       rend1;
184.       rend2;];
185.   ym = A\wm; %deflection matric
186.   %fa=anchor* ym((anchor_h/h+3) ,1);
187.   Sm = zeros(nr,1);
188.   Vm = zeros(nr,1);
189.   Mm = zeros(nr,1);
190.   for i = 1:1:nr
191.       Sm(i,1)= ( ym(i+1) -ym(i+3) ) /(2*h);
192.       Vm(i,1)= -(Rp* (-ym(i) + 2*ym(i+1) -2* ym(i+3) +ym(i+4)))/ (2*h^3);
193.       Mm(i,1)= -(Rp *(ym(i+1) - 2 *ym(i+2) + ym(i+3)))/ (h^2);
194.   end
195.
196.   r = zeros(nr,1);
197.   j=1;
198.   for i=0:h:(L)
199.       r(j,1)=i;
200.       j=j+1;
201.   end
202.   ymnew=zeros(nr,1);
203.   for i=3:1:(nc-2)
204.       ymnew(i-2,1)= ym(i,1);
205.   end
206.   [~,x] = max(abs(ymnew));

```

```

207.     app.ymaxmEditField.Value=num2str(ymnew(x));
208.     [~,x] = max(Vm);
209.     app.VmaxKNEditField.Value=num2str(Vm(x));
210.     [~,x] = min(Vm);
211.     app.VminKNEditField.Value=num2str(Vm(x));
212.     [~,x] = max(abs(Mm));
213.     app.MmaxKNmEditField.Value= num2str(Mm(x));
214.     plot(app.Ym1,ymnew,r)
215.     plot(app.Sm1,Sm,r)
216.     plot(app.Vm1,Vm,r)
217.     plot(app.Mm1,Mm,r)
218.     app.TabGroup.SelectedTab = app.ResultsTab;
219.     end
220.
221.     % Value changed function: InstallAnchorCheckBox
222.     function InstallAnchorCheckBoxValueChanged(app, event)
223.     value = app.InstallAnchorCheckBox.Value;
224.     switch value
225.     case 0
226.     app.EquivalentresistancestiffnessinKNm3m.Visible= 'off';
227.     app.PositionofAnchorfromtopinm.Visible = 'off';
228.     case 1
229.     app.EquivalentresistancestiffnessinKNm3m.Visible= 'on';
230.     app.PositionofAnchorfromtopinm.Visible = 'on';
231.     end
232.     end
233.
234.     % Value changed function: watertableCheckBox_2
235.     function watertableCheckBox_2ValueChanged(app, event)
236.     value = app.watertableCheckBox_2.Value;
237.     switch value
238.     case 0
239.     app.hightofwatertablefromtopinm.Visible= 'off';
240.     app.SaturdatedspecificWeightinKNm3.Visible = 'off';
241.     case 1
242.     app.hightofwatertablefromtopinm.Visible= 'on';
243.     app.SaturdatedspecificWeightinKNm3.Visible = 'on';
244.     end
245.     end
246.     end
247.
248.     % Component initialization
249.     methods (Access = private)
250.
251.     % Create UIFigure and components
252.     function createComponents(app)
253.
254.     % Create ShetPileWall and hide until all components are created
255.     app.ShetPileWall = uifigure('Visible', 'off');
256.     app.ShetPileWall.Position = [100 100 697 480];

```

```

257.     app.ShetPileWall.Name = 'Sheet Pile Wall';
258.
259.     % Create TabGroup
260.     app.TabGroup = uitabgroup(app.ShetPileWall);
261.     app.TabGroup.Position = [1 1 697 480];
262.
263.     % Create InputTab
264.     app.InputTab = uitab(app.TabGroup);
265.     app.InputTab.Title = 'Input Tab';
266.
267.     % Create AnchorPropertiesPanel
268.     app.AnchorPropertiesPanel = uipanel(app.InputTab);
269.     app.AnchorPropertiesPanel.Title = 'Anchor Properties';
270.     app.AnchorPropertiesPanel.Position = [20 63 336 116];
271.
272.     % Create InstallAnchorCheckBox
273.     app.InstallAnchorCheckBox = uicheckbox(app.AnchorPropertiesPanel);
274.     app.InstallAnchorCheckBox.ValueChangedFcn = createCallbackFcn(app,
        @InstallAnchorCheckBoxValueChanged, true);
275.     app.InstallAnchorCheckBox.Text = 'Install Anchor';
276.     app.InstallAnchorCheckBox.FontWeight = 'bold';
277.     app.InstallAnchorCheckBox.Position = [12 73 102 22];
278.
279.     % Create EquivalentresistancestiffnessinKNm3mEditFieldLabel
280.     app.EquivalentresistancestiffnessinKNm3mEditFieldLabel =
        uilabel(app.AnchorPropertiesPanel);
281.     app.EquivalentresistancestiffnessinKNm3mEditFieldLabel.HorizontalAli
        gnment = 'right';
282.     app.EquivalentresistancestiffnessinKNm3mEditFieldLabel.Position =
        [19 47 231 22];
283.     app.EquivalentresistancestiffnessinKNm3mEditFieldLabel.Text =
        'Equivalent resistance stiffness in KN/m3/m';
284.
285.     % Create EquivalentresistancestiffnessinKNm3m
286.     app.EquivalentresistancestiffnessinKNm3m =
        uieditfield(app.AnchorPropertiesPanel, 'numeric');
287.     app.EquivalentresistancestiffnessinKNm3m.Visible = 'off';
288.     app.EquivalentresistancestiffnessinKNm3m.Position = [257 47 73 22];
289.
290.     % Create PositionofAnchorfromtopinmEditFieldLabel
291.     app.PositionofAnchorfromtopinmEditFieldLabel =
        uilabel(app.AnchorPropertiesPanel);
292.     app.PositionofAnchorfromtopinmEditFieldLabel.Position = [19 16 231
        22];
293.     app.PositionofAnchorfromtopinmEditFieldLabel.Text = 'Position of
        Anchor from top in m';
294.
295.     % Create PositionofAnchorfromtopinm
296.     app.PositionofAnchorfromtopinm =
        uieditfield(app.AnchorPropertiesPanel, 'numeric');

```

```

297.     app.PositionofAnchorfromtopinm.Visible = 'off';
298.     app.PositionofAnchorfromtopinm.Position = [257 16 73 22];
299.
300.     % Create InputParametersPanel
301.     app.InputParametersPanel = uipanel(app.InputTab);
302.     app.InputParametersPanel.Title = 'Input Parameters';
303.     app.InputParametersPanel.Position = [20 207 298 221];
304.
305.     % Create PileLengthinmEditFieldLabel
306.     app.PileLengthinmEditFieldLabel = uilabel(app.InputParametersPanel);
307.     app.PileLengthinmEditFieldLabel.Position = [13 165 92 22];
308.     app.PileLengthinmEditFieldLabel.Text = 'Pile Length in m';
309.
310.     % Create PileLengthinm
311.     app.PileLengthinm = uieditfield(app.InputParametersPanel,
    'numeric');
312.     app.PileLengthinm.HorizontalAlignment = 'center';
313.     app.PileLengthinm.Position = [191 165 59 22];
314.
315.     % Create DepthofExcavationinmEditFieldLabel
316.     app.DepthofExcavationinmEditFieldLabel =
    uilabel(app.InputParametersPanel);
317.     app.DepthofExcavationinmEditFieldLabel.Position = [11 109 139 22];
318.     app.DepthofExcavationinmEditFieldLabel.Text = 'Depth of Excavation
    in m';
319.
320.     % Create DepthofExcavationinm
321.     app.DepthofExcavationinm = uieditfield(app.InputParametersPanel,
    'numeric');
322.     app.DepthofExcavationinm.HorizontalAlignment = 'center';
323.     app.DepthofExcavationinm.Position = [191 109 59 22];
324.
325.     % Create NumberofsegmentsLabel
326.     app.NumberofsegmentsLabel = uilabel(app.InputParametersPanel);
327.     app.NumberofsegmentsLabel.Position = [11 139 117 22];
328.     app.NumberofsegmentsLabel.Text = 'Number of segments';
329.
330.     % Create numberofsegments
331.     app.numberofsegments = uieditfield(app.InputParametersPanel,
    'numeric');
332.     app.numberofsegments.HorizontalAlignment = 'center';
333.     app.numberofsegments.Position = [191 139 59 22];
334.
335.     % Create WallElasticmodulusinKNm2EditFieldLabel
336.     app.WallElasticmodulusinKNm2EditFieldLabel =
    uilabel(app.InputParametersPanel);
337.     app.WallElasticmodulusinKNm2EditFieldLabel.Position = [11 79 168
    22];
338.     app.WallElasticmodulusinKNm2EditFieldLabel.Text = 'Wall Elastic
    modulus in KN/m2';

```

```

339.
340.     % Create WallElasticmodulusinKNm2
341.     app.WallElasticmodulusinKNm2 = uieditfield(app.InputParametersPanel,
'numeric');
342.     app.WallElasticmodulusinKNm2.HorizontalAlignment = 'center';
343.     app.WallElasticmodulusinKNm2.Position = [191 79 59 22];
344.
345.     % Create Momentofinertiam4EditFieldLabel
346.     app.Momentofinertiam4EditFieldLabel =
    uilabel(app.InputParametersPanel);
347.     app.Momentofinertiam4EditFieldLabel.Position = [11 49 118 22];
348.     app.Momentofinertiam4EditFieldLabel.Text = 'Moment of inertia m4';
349.
350.     % Create Momentofinertiam4
351.     app.Momentofinertiam4 = uieditfield(app.InputParametersPanel,
'numeric');
352.     app.Momentofinertiam4.HorizontalAlignment = 'center';
353.     app.Momentofinertiam4.Position = [191 49 59 22];
354.
355.     % Create SurchargeKpaEditFieldLabel
356.     app.SurchargeKpaEditFieldLabel = uilabel(app.InputParametersPanel);
357.     app.SurchargeKpaEditFieldLabel.Position = [11 19 86 22];
358.     app.SurchargeKpaEditFieldLabel.Text = 'Surcharge Kpa';
359.
360.     % Create SurchargeKpa
361.     app.SurchargeKpa = uieditfield(app.InputParametersPanel, 'numeric');
362.     app.SurchargeKpa.HorizontalAlignment = 'center';
363.     app.SurchargeKpa.Position = [191 19 59 22];
364.
365.     % Create SoilPropertiesPanel
366.     app.SoilPropertiesPanel = uipanel(app.InputTab);
367.     app.SoilPropertiesPanel.Title = 'Soil Properties';
368.     app.SoilPropertiesPanel.Position = [342 207 298 221];
369.
370.     % Create CoefficientofactivepressureLabel
371.     app.CoefficientofactivepressureLabel =
    uilabel(app.SoilPropertiesPanel);
372.     app.CoefficientofactivepressureLabel.Position = [13 165 160 22];
373.     app.CoefficientofactivepressureLabel.Text = 'Coefficient of active
    pressure';
374.
375.     % Create FrictionAngleinDegrees
376.     app.FrictionAngleinDegrees = uieditfield(app.SoilPropertiesPanel,
'numeric');
377.     app.FrictionAngleinDegrees.HorizontalAlignment = 'center';
378.     app.FrictionAngleinDegrees.Position = [227 165 63 22];
379.
380.     % Create specificWeightinKNm3EditFieldLabel
381.     app.specificWeightinKNm3EditFieldLabel =
    uilabel(app.SoilPropertiesPanel);

```

```

382.     app.specificWeightinKNm3EditFieldLabel.Position = [11 139 138 22];
383.     app.specificWeightinKNm3EditFieldLabel.Text = 'specific Weight in
        KN/m3';
384.
385.     % Create specificWeightinKNm3
386.     app.specificWeightinKNm3 = uieditfield(app.SoilPropertiesPanel,
        'numeric');
387.     app.specificWeightinKNm3.HorizontalAlignment = 'center';
388.     app.specificWeightinKNm3.Position = [227 139 63 22];
389.
390.     % Create watertableCheckBox
391.     app.watertableCheckBox = uicheckbox(app.SoilPropertiesPanel);
392.     app.watertableCheckBox.Text = 'water table';
393.     app.watertableCheckBox.Position = [48 79 81 22];
394.
395.     % Create hightofwatertablefromtopinmEditFieldLabel
396.     app.hightofwatertablefromtopinmEditFieldLabel =
        uilabel(app.SoilPropertiesPanel);
397.     app.hightofwatertablefromtopinmEditFieldLabel.Position = [13 58 180
        22];
398.     app.hightofwatertablefromtopinmEditFieldLabel.Text = 'hight of water
        table from top in m';
399.
400.     % Create hightofwatertablefromtopinm
401.     app.hightofwatertablefromtopinm =
        uieditfield(app.SoilPropertiesPanel, 'numeric');
402.     app.hightofwatertablefromtopinm.HorizontalAlignment = 'center';
403.     app.hightofwatertablefromtopinm.Visible = 'off';
404.     app.hightofwatertablefromtopinm.Position = [227 58 63 22];
405.
406.     % Create SaturdatedspecificWeightinKNm3EditFieldLabel
407.     app.SaturdatedspecificWeightinKNm3EditFieldLabel =
        uilabel(app.SoilPropertiesPanel);
408.     app.SaturdatedspecificWeightinKNm3EditFieldLabel.Position = [13 27
        201 22];
409.     app.SaturdatedspecificWeightinKNm3EditFieldLabel.Text = 'Saturdated
        specific Weight in KN/m3';
410.
411.     % Create SaturdatedspecificWeightinKNm3
412.     app.SaturdatedspecificWeightinKNm3 =
        uieditfield(app.SoilPropertiesPanel, 'numeric');
413.     app.SaturdatedspecificWeightinKNm3.HorizontalAlignment = 'center';
414.     app.SaturdatedspecificWeightinKNm3.Visible = 'off';
415.     app.SaturdatedspecificWeightinKNm3.Position = [227 27 63 22];
416.
417.     % Create ModulusofsubgradereactioninKNm3EditFieldLabel
418.     app.ModulusofsubgradereactioninKNm3EditFieldLabel =
        uilabel(app.SoilPropertiesPanel);
419.     app.ModulusofsubgradereactioninKNm3EditFieldLabel.Position = [11 108
        217 22];

```

```

420.     app.ModulusofsubgradereactioninKNm3EditFieldLabel.Text = 'Modulus of
        subgrade reaction in KN/m3';
421.
422.     % Create ModulusofsubgradereactioninKNm3
423.     app.ModulusofsubgradereactioninKNm3 =
        uieditfield(app.SoilPropertiesPanel, 'numeric');
424.     app.ModulusofsubgradereactioninKNm3.HorizontalAlignment = 'center';
425.     app.ModulusofsubgradereactioninKNm3.Position = [227 108 63 22];
426.
427.     % Create watertableCheckBox_2
428.     app.watertableCheckBox_2 = uicheckbox(app.SoilPropertiesPanel);
429.     app.watertableCheckBox_2.ValueChangedFcn = createCallbackFcn(app,
        @watertableCheckBox_2ValueChanged, true);
430.     app.watertableCheckBox_2.Text = 'water table';
431.     app.watertableCheckBox_2.Position = [48 79 81 22];
432.
433.     % Create ModulusofsubgradereactionButtonGroup
434.     app.ModulusofsubgradereactionButtonGroup =
        uibuttongroup(app.InputTab);
435.     app.ModulusofsubgradereactionButtonGroup.TitlePosition =
        'centertop';
436.     app.ModulusofsubgradereactionButtonGroup.Title = 'Modulus of
        subgrade reaction';
437.     app.ModulusofsubgradereactionButtonGroup.BackgroundColor = [0.902
        0.902 0.902];
438.     app.ModulusofsubgradereactionButtonGroup.FontWeight = 'bold';
439.     app.ModulusofsubgradereactionButtonGroup.Position = [395 105 197
        74];
440.
441.     % Create ConstantButton
442.     app.ConstantButton =
        uiradiobutton(app.ModulusofsubgradereactionButtonGroup);
443.     app.ConstantButton.Text = 'Constant';
444.     app.ConstantButton.Position = [11 28 70 22];
445.     app.ConstantButton.Value = true;
446.
447.     % Create VariablealongdepthButton
448.     app.VariablealongdepthButton =
        uiradiobutton(app.ModulusofsubgradereactionButtonGroup);
449.     app.VariablealongdepthButton.Text = 'Variable along depth';
450.     app.VariablealongdepthButton.Position = [11 6 131 22];
451.
452.     % Create StartButton
453.     app.StartButton = uibutton(app.InputTab, 'push');
454.     app.StartButton.ButtonPushedFcn = createCallbackFcn(app,
        @StartButtonPushed, true);
455.     app.StartButton.BackgroundColor = [0.8 0.8 0.8];
456.     app.StartButton.FontSize = 20;
457.     app.StartButton.FontWeight = 'bold';
458.     app.StartButton.FontAngle = 'italic';

```

```

459.     app.StartButton.Position = [428 3 130 48];
460.     app.StartButton.Text = 'Start';
461.
462.     % Create ResultsTab
463.     app.ResultsTab = uitab(app.TabGroup);
464.     app.ResultsTab.Title = 'Results';
465.
466.     % Create Sm1
467.     app.Sm1 = uiaxes(app.ResultsTab);
468.     title(app.Sm1, 'Sm')
469.     xlabel(app.Sm1, '')
470.     ylabel(app.Sm1, 'Depth m')
471.     app.Sm1.PlotBoxAspectRatio = [1 2.32704402515723 1];
472.     app.Sm1.YDir = 'reverse';
473.     app.Sm1.XAxisLocation = 'top';
474.     app.Sm1.Position = [534 112 151 310];
475.
476.     % Create Vm1
477.     app.Vm1 = uiaxes(app.ResultsTab);
478.     title(app.Vm1, 'Vm (KN)')
479.     xlabel(app.Vm1, '')
480.     ylabel(app.Vm1, 'Depth m')
481.     app.Vm1.PlotBoxAspectRatio = [1 2.32704402515723 1];
482.     app.Vm1.YDir = 'reverse';
483.     app.Vm1.XAxisLocation = 'top';
484.     app.Vm1.Position = [361 111 151 310];
485.
486.     % Create Ym1
487.     app.Ym1 = uiaxes(app.ResultsTab);
488.     title(app.Ym1, 'Ym (m)')
489.     xlabel(app.Ym1, '')
490.     ylabel(app.Ym1, 'Depth m')
491.     app.Ym1.PlotBoxAspectRatio = [1 2.32704402515723 1];
492.     app.Ym1.YDir = 'reverse';
493.     app.Ym1.XAxisLocation = 'top';
494.     app.Ym1.Position = [14 112 151 310];
495.
496.     % Create MmaxKNmEditFieldLabel
497.     app.MmaxKNmEditFieldLabel = uilabel(app.ResultsTab);
498.     app.MmaxKNmEditFieldLabel.HorizontalAlignment = 'right';
499.     app.MmaxKNmEditFieldLabel.Position = [17 58 80 22];
500.     app.MmaxKNmEditFieldLabel.Text = 'Mmax (KN-m)';
501.
502.     % Create MmaxKNmEditField
503.     app.MmaxKNmEditField = uieditfield(app.ResultsTab, 'text');
504.     app.MmaxKNmEditField.Position = [133 58 79 22];
505.
506.     % Create ymaxmEditFieldLabel
507.     app.ymaxmEditFieldLabel = uilabel(app.ResultsTab);
508.     app.ymaxmEditFieldLabel.HorizontalAlignment = 'right';

```

```

509.     app.ymaxmEditFieldLabel.Position = [42 16 55 22];
510.     app.ymaxmEditFieldLabel.Text = 'ymax (m)';
511.
512.     % Create ymaxmEditField
513.     app.ymaxmEditField = ueditfield(app.ResultsTab, 'text');
514.     app.ymaxmEditField.Position = [133 16 79 22];
515.
516.     % Create VmaxKNEditFieldLabel
517.     app.VmaxKNEditFieldLabel = uilabel(app.ResultsTab);
518.     app.VmaxKNEditFieldLabel.HorizontalAlignment = 'right';
519.     app.VmaxKNEditFieldLabel.Position = [318 58 64 22];
520.     app.VmaxKNEditFieldLabel.Text = 'Vmax (KN)';
521.
522.     % Create VmaxKNEditField
523.     app.VmaxKNEditField = ueditfield(app.ResultsTab, 'text');
524.     app.VmaxKNEditField.Position = [418 58 79 22];
525.
526.         % Create VminKNEditFieldLabel
527.         app.VminKNEditFieldLabel = uilabel(app.ResultsTab);
528.         app.VminKNEditFieldLabel.HorizontalAlignment = 'right';
529.         app.VminKNEditFieldLabel.Position = [321 16 61 22];
530.         app.VminKNEditFieldLabel.Text = 'Vmin (KN)';
531.
532.         % Create VminKNEditField
533.         app.VminKNEditField = ueditfield(app.ResultsTab,
'text');
534.         app.VminKNEditField.Position = [418 16 79 22];
535.
536.         % Create Mm1
537.         app.Mm1 = uiaxes(app.ResultsTab);
538.         title(app.Mm1, 'Mm (KN-m)')
539.         xlabel(app.Mm1, '')
540.         ylabel(app.Mm1, 'Depth m')
541.         app.Mm1.PlotBoxAspectRatio = [1 2.32704402515723 1];
542.         app.Mm1.YDir = 'reverse';
543.         app.Mm1.XAxisLocation = 'top';
544.         app.Mm1.Position = [188 112 151 310];
545.
546.         % Show the figure after all components are created
547.         app.ShetPileWall.Visible = 'on';
548.     end
549. end
550.
551.     % App creation and deletion
552.     methods (Access = public)
553.
554.         % Construct app
555.         function app = app2
556.
557.             % Create UIFigure and components

```

```
558.         createComponents(app)
559.
560.         % Register the app with App Designer
561.         registerApp(app, app.ShetPileWall)
562.
563.         if nargin == 0
564.             clear app
565.         end
566.     end
567.
568.     % Code that executes before app deletion
569.     function delete(app)
570.
571.         % Delete UIFigure when app is deleted
572.         delete(app.ShetPileWall)
573.     end
574. end
575. end
```