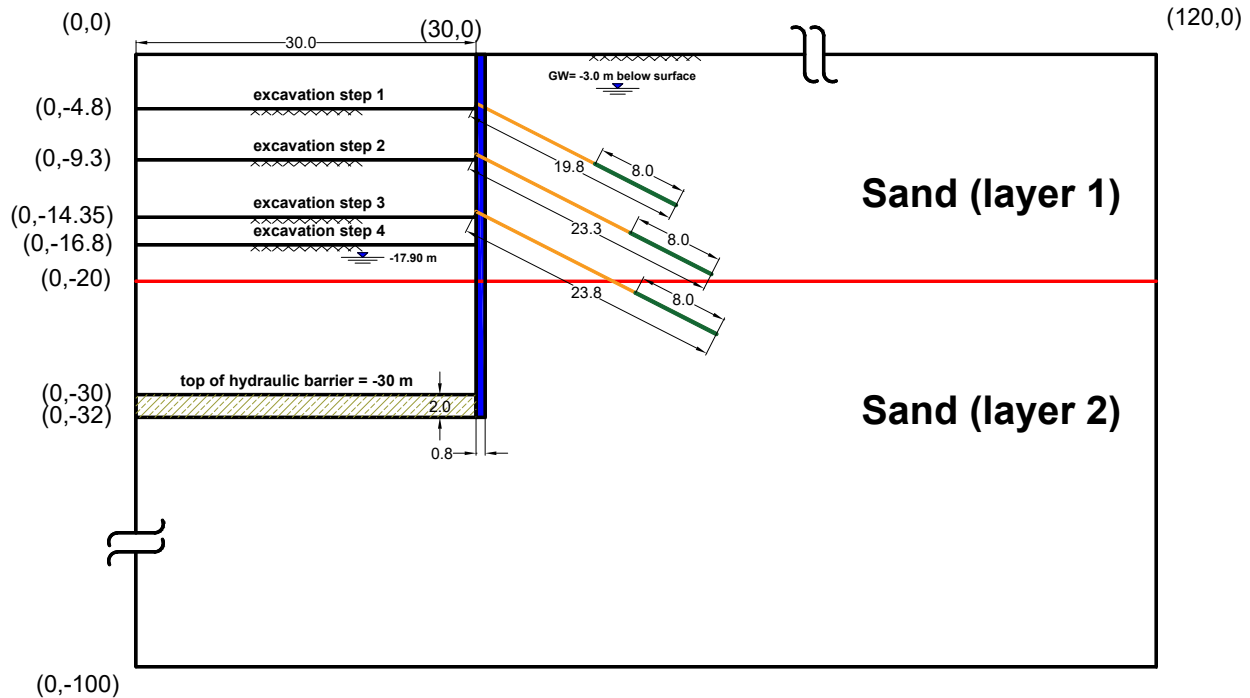


Plastic Hardening Soil Model in Deep Excavation

Created By: [Roozbeh Geraili Mikola, PhD, PE](#)

Email: adonis4geo@outlook.com

Web Page: www.geowizard.org



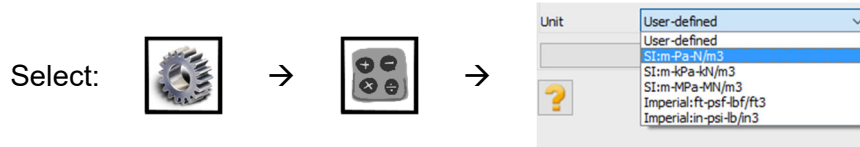
This tutorial demonstrates the importance of modeling excavation problems with the use of the Plastic Hardening (P-Hardening) Soil model. The benchmark exercise described by Benz (2007), Schweiger (2002) as well as Obrzurd and Truty (2018) consisting of a deep excavation problem in Berlin sand is the basis for this tutorial. The problem is also documented in the FLAC (2020) commercial programs. The geometry, basic assumptions and computational steps adopted for this example are taken from the benchmark exercise. The model parameters are adapted from studies performed by Benz and Schweiger. The results for wall deflection and surface settlements are compared to measurements and to ZSoil (Obrzurd and Truty, 2018), Plaxis (Benz, 2007) and FLAC (2020) predictions, respectively. The analysis described by Schweiger (2002) includes nine steps:

- Stage 1 – Initialize stress state, including groundwater table, 3 m below soil surface (given by $\sigma'_v = \rho g d$, $\sigma'_h = K_0 \sigma'_v$).
- Stage 2 – Install diaphragm wall and lower water level to -17.90 m in excavation zone.
- Stage 3 – Excavation step 1 (to level -4.80 m).
- Stage 4 – Install tieback row 1 at level -4.30 m and prestress anchors.
- Stage 5 – Excavation step 2 (to level -9.30 m).
- Stage 6 – Install tieback row 2 at level -8.80 m and prestress anchors.
- Stage 7 – Excavation step 3 (to level -14.35 m).

Stage 8 – Install tieback row 3 at level –13.85 m and prestress anchors.
Stage 9 – Excavation step 4 (to level –16.80 m).

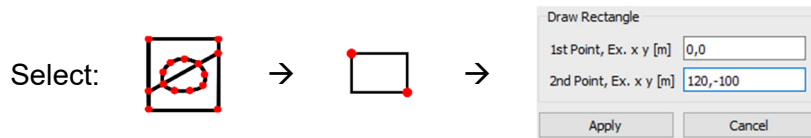
Select Unit

Since we will be using Metric (Pa) units for this tutorial, let’s select “SI: m-Pa-N/m³” from **Calculation Setting** under **Setting** button.



Create Geometry

Select the **Create Geometry** button, then select **Draw Rectangle** and enter the first- and second-points’ coordinates (0,0 and 120,-100) and press **Apply**.



Select **Draw Line** to draw material and boundary lines as listed below:

Type	Start Point	End Point
Line	0,-3	120,-3
	0,-20	120,-20
	0,-32	30,-32
	0,-4.8	30,-4.8

Type	Start Point	End Point
Line	0,-9.3	30,-9.3
	0,-14.35	30,-14.35
	0,-16.8	30,-16.8
	0,-40	120,-40

In this tutorial, the diaphragm wall will be modeled by beam element sandwiched between two interface elements. It worth noting that, in **ADONIS** interface element, can only be applied on the boundary edges (i.e. edge that shared by only one active solid element). Therefore, to prepare the geometry, a new drawing capability named **Crack** introduced to **ADONIS** (i.e. after V2.75) which makes the elements along it to be separated (e.g. new sets of nodes for each side of the crack will be generated). Please note that each side of crack will be treated as open boundary and it is user’s responsibility to add interface element between the edges otherwise model won’t be converged during cycling. To add crack, simply click on **Crack** and specify start and end points and click on **Apply** Button, or press **Enter**.

Type	Start Point	End Point
Crack	30,0	30,-32

The geometry for this example has now been entered as shown in Figure 1.

Note: In discretization and clean-up process **ADONIS** will remove the unconnected segments (dangling segments). So to generate boundary line that delineates the diaphragm wall a new dummy line (from 0,-32 to 30,-32) is added to the end of line to prevent it from being removed.

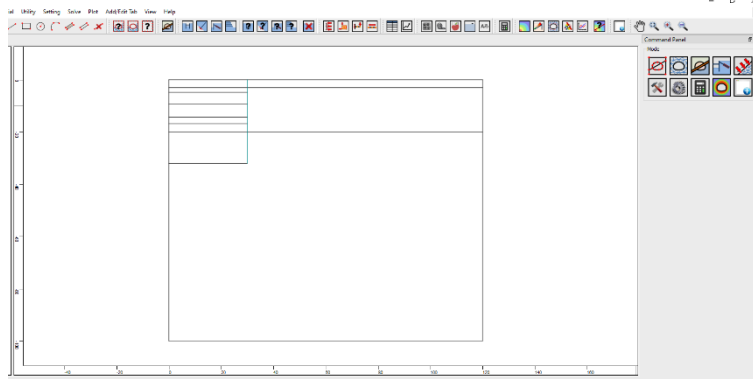
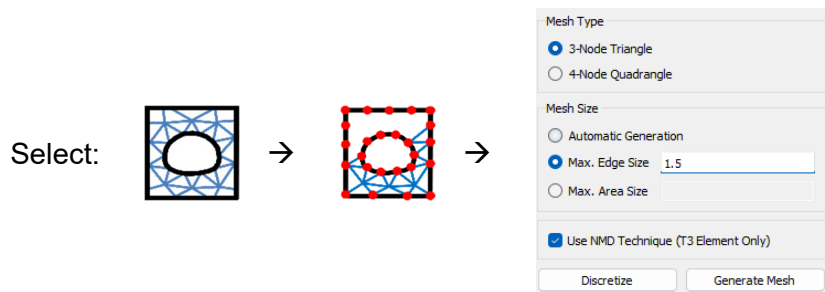


Figure 1- Model geometry.

Mesh

Now let's generate the finite element (FE) mesh. In **ADONIS**, meshing is a simple two-step process. First you must discretize the boundaries, and then the Mesh can be generated. You can also configure various Mesh Setup parameters before generating the mesh.

Select the **Discretize/Mesh** button in command panel or toolbar or the Mesh menu, then select **Mesh Property** button then select the element type. Currently only two types of FE mesh is available 1) 3-Node triangular and 2) 4-Node quadrilateral elements. For this tutorial 3-Node triangular element is selected. Then select the **Discretize/Mesh** button in command panel or toolbar or the Mesh menu, then select **Discretize/Mesh** button. Enter 1.5 as **Max. Edge Size** in order to create finer mesh size. Now press **Discretize** then press **Generate Mesh** buttons to discretize the boundaries and generate FE mesh.



The finite element mesh is generated.

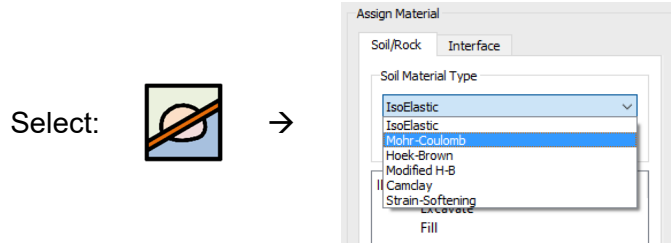
Low order elements tend to exhibit “locking” when undergoing plastic deformation, which leads to inaccurate estimates of loading capacity. ADONIS adopts “Nodal Mixed Discretization (NMD) methodology to overcome this problem. NMD is introduced by Detournay & Dzik (2006) as an improvement of Mixed Discretization (MD) technique. In order to activate NMD, check “Use NMD Technique”, then click on **Apply** Button.

Notes:

1. NMD is only available for T3 element (not Q4).
2. NMD is activated by default.

Assign Initial Soil Properties

To create a new material, select the **Assign Material/Excavate** button from command panel, toolbar or menu, then select **Soil/Rock** tab. From drop down select **Mohr-Coulomb** material type then click on **Add Soil/Rock Material** button.



Now enter the following rock mass properties for the soil layers:

Material ID	Material Name	Density (kg/m ³)	Shear Modulus (Pa)	Bulk Modulus (Pa)	Cohesion (Pa)	Friction Angle (deg.)	Dilation Angle (deg.)
1	Layer1-Dry	1900	7.5x10 ⁷	1.0x10 ⁸	0	35	5
2	Layer1-Wet	2000	7.5x10 ⁷	1.0x10 ⁸	0	35	5
3	Layer2-Wet	2000	1.25x10 ⁸	1.67x10 ⁸	0	38	6

Now we need to assign the created material to the proper region. To do that, select the “Layer1-Dry” material (i.e. created material) from the material list and then use mouse and click on the Sand regions in the main window and click on **Apply** Button, or press **Enter** (Figure 2). Repeat the sequence to create another material for other soil layers.

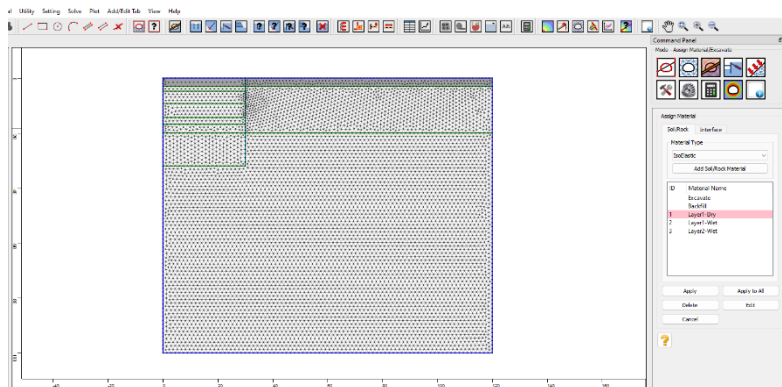
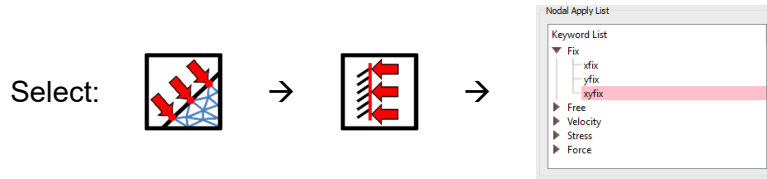


Figure 2- Assign sand soil material.

Boundary Condition

In order to fix (i.e. zero displacement) the external boundary, select **Initial Condition** then select **Apply Boundary Condition** then from the "Keyword List" click on **xyfix** under **Fix** keyword. Then Use the mouse to select the nodes located on bottom of the model. When finished, click on **Apply** Button, or press **Enter**.



Then select **xfix** and use the mouse to select the nodes located on left and right sides of the model. The boundary condition has been applied as shown in Figure 3.

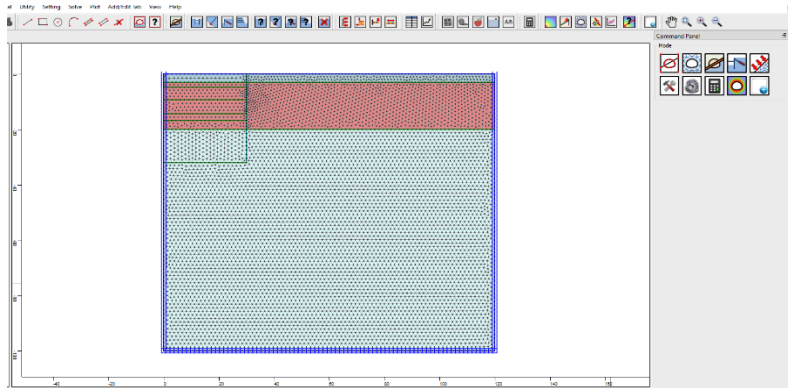


Figure 3- Model configuration after applying boundary condition.

Install Diaphragm Wall

Diaphragm wall is modeled by beam structural elements. The beam element will be sandwiched between two interface elements. To install the diaphragm wall at the right place (i.e. along 30,-32->30,0 crack line) you need to temporarily excavate (i.e. hide) the excavation zone and install the diaphragm wall then backfill it later on. To do so, select the **Assign Material/Excavate** button from command panel, toolbar or menu, then select **Excavate** from the material name list (Figure 4). Please make sure that **Reset Element after Excavation** checkbox is unchecked. This way you can make sure that initialized stress in the elements won't be zeroed out after backfill. Then use mouse and select three regions in the excavation zone and press **Apply**.

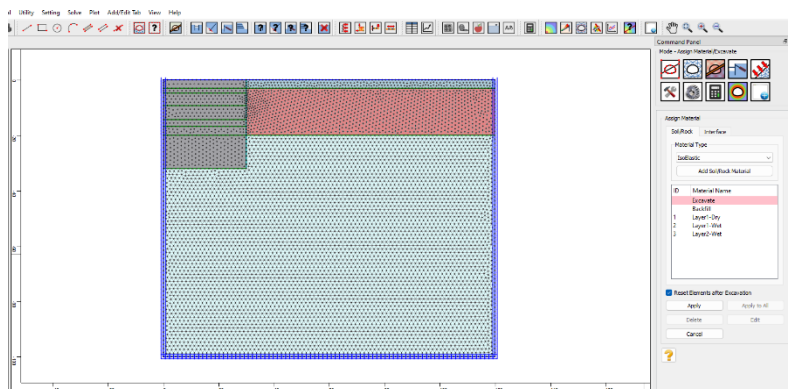
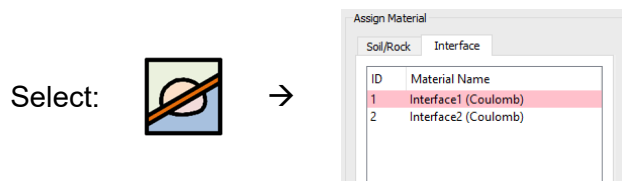


Figure 4- Selected zone to be hidden (temporarily excavated).

Then select the **Structures** button from command panel, toolbar or menu, then select **Beam** element. Select attach to **Edge with Interface** and after that **Range** from selection mode. Change the interface type to **Both Sides** from the dropdown. Now use mouse to select the range

($xlim=[29.9,30.1]$ and $ylim=-32.1,0.1$) that diaphragm wall should be installed and press **Apply** (Figure 5). Then click on **Beam Element Properties** and assign the diaphragm wall properties as shown in Figure 6. When user selects the “both sided interface”, ADONIS automatically sandwiches the beam elements between two interfaces. The First interface ID is always between beam element and visible solid element and second interface ID will be created between beam element and invisible solid element. Now you need to modify the interface properties. For this select the **Assign Material/Excavate** button from command panel, toolbar or menu, then select **Interface** tab. In the list you should see two interface elements. Select **Interface1 (Coulomb)** and press **Edit** button. Then adjust the following properties:

Material ID	Material Name	jkn (Pa/m)	jks (Pa/m)	Cohesion (Pa)	Friction Angle (deg.)
1	Interface1	1.0×10^9	1.0×10^9	0	25
2	Interface2	1.0×10^9	1.0×10^9	0	25



To backfill the temporarily excavated part, select the **Assign Material/Excavate** button from command panel, toolbar or menu, then select **Backfill** from the material name list. Then use mouse and select three regions that were excavated previously and press **Apply** (Figures 7-8).

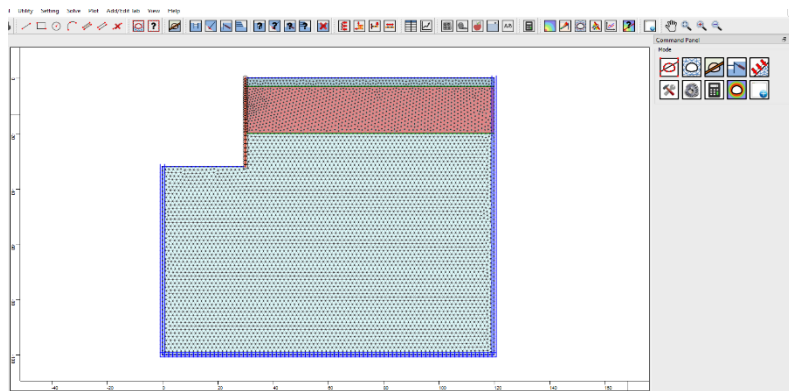


Figure 5- Install beam elements sandwiched between two interfaces.

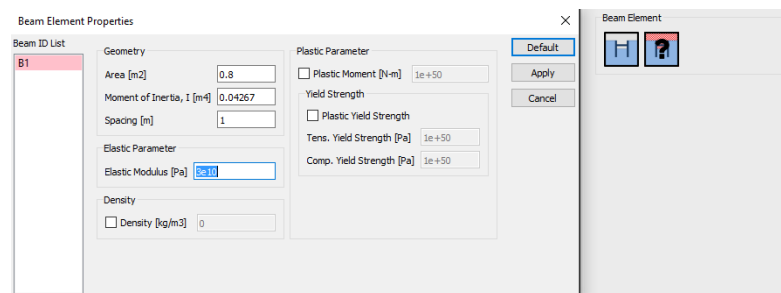


Figure 6- Assign diaphragm wall properties.

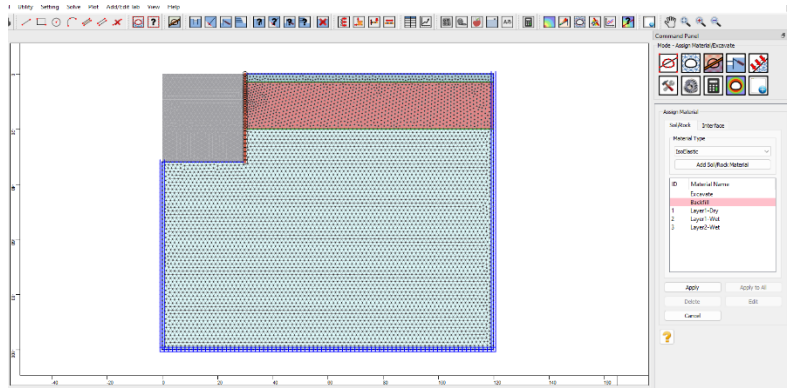


Figure 7- Backfill the excavated zone.

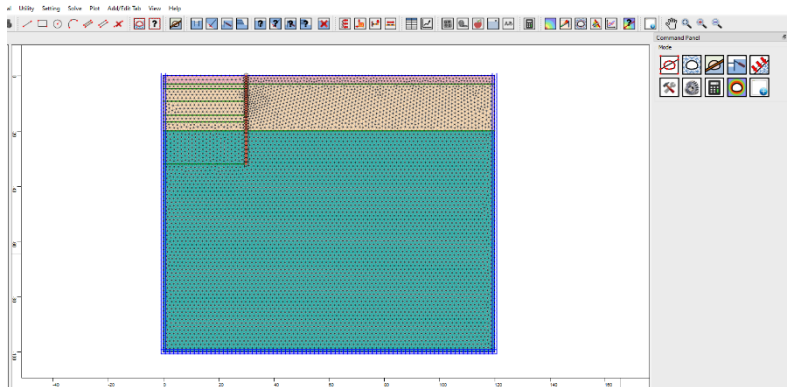
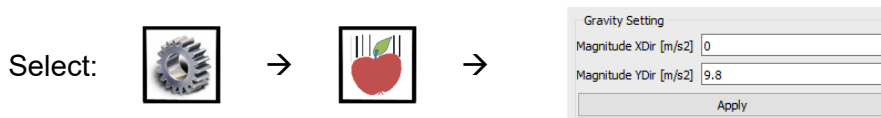


Figure 8- Model configuration after installing the diaphragm wall.

Assign Gravity

Before solving the mode to equilibrium let's specify the gravitational gradient value. For that, select **Setting** then **Gravity Setting** and set the magnitude of gravity in y-direction to 9.8 m/s². Note: Downward gravity is positive.



Initialize the Stress Condition

In order to establish an initial stress equilibrium in the problem the following tasks should be performed:

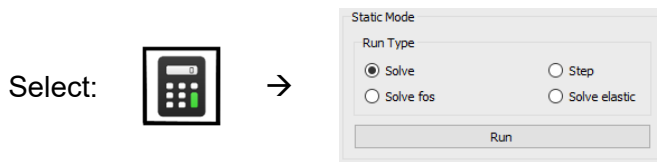
1. The pore pressure should be zero above the water table, and have a gradient of $\rho_w g$ below it.
2. If k_{0x} is the ratio of effective σ_{xx} to effective σ_{yy} , then the total σ_{xx} should be $\sigma_{xx} = k_{0x} (\sigma_{yy} + pp) - pp$.
3. If k_{0z} is the ratio of effective σ_{zz} to effective σ_{yy} , then the total σ_{zz} should be $\sigma_{zz} = k_{0z} (\sigma_{yy} + pp) - pp$.

Several lines of javascript code has been written (i.e. "[initial_pp_stresses.ajs](#)") to perform all the tasks listed above. In this script the following input parameters must be set:

K0x & K0z (Layer1)	0.43	Dry Density (kg/m3)	1900
K0x & K0z (Layer2)	0.38	Sat. Density (kg/m3)	2000
Ground Water Table	-3.0	Water Density (kg/m3)	1000
ygrav	9.8		

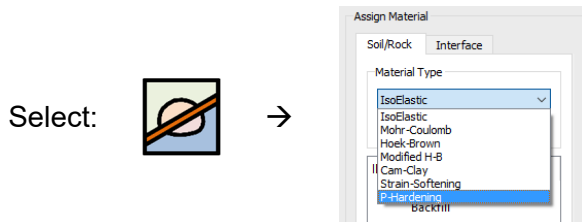
The full script is shown in the **Appendix A**. Now you need to execute the script and for that select **Load Script** from **File** menu and select the “**initial_pp_stresses.ajs**” file. Please note that, compression stresses have a negative sign, in accordance with the general sign convention for internal stresses in **ADONIS**.

Now click on **Solve the Model** button from toolbar, menu or command panel then select **Solve** radio button from the **Run Type** and click on **Run** button. The **ADONIS** Compute engine will proceed in running the analysis. Model should reach equilibrium after several steps.



Assign Plastic Hardening Soil Properties

To create a new material, select the **Assign Material/Excavate** button from command panel, toolbar or menu, then select **Soil/Rock** tab. From drop down select **P-Hardening** material type then click on **Add Soil/Rock Material** button.





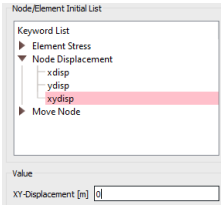
Now enter the rock mass properties for the soil layers based on the parameters provided in Table 1. Material properties for the Berlin sand are adapted from the hardening-soil parameters listed in the FLAC Version 8.10 Manual (2018).



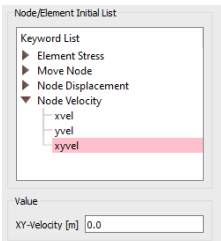
Table 1- Plastic Hardening Properties

Parameter Name	Layer 1-Dry	Layer 1-Wet	Layer 2-Wet
Density (kg/m3)	1900	2000	2000
Friction Angle (deg.)	35	35	38
Dilation Angle (deg.)	5	5	6
E_{50}^{ref} (MPa)	45	45	75
E_{ur}^{ref} (MPa)	180	180	300
E_{oed}^{ref} (MPa)	45	45	75



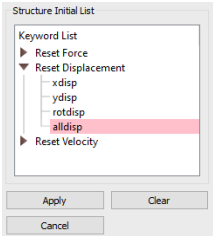
Reference pressure, p_{ref} (MPa)	0.1	0.1	0.1
Poisson's ratio	0.2	0.2	0.2
Cohesion, C (MPa)	0	0	0
Power, m	0.55	0.55	0.55
Failure ratio, R_f	0.9	0.9	0.9
Over consolidation ratio, ocr	1.0	1.0	1.0



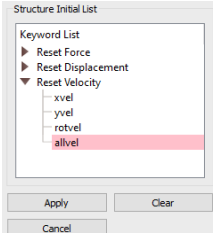
The initial principal effective stresses (σ_1 , σ_2 , σ_3) are also required input for the Plastic Hardening model. The javascript subroutine "initial_principal_stresses.ajs" is used to calculate these values after the initial pre-excavation stress state is established and assigning them to each element. After initialization **Solve** the model to equilibrium then zero out the displacement in the model to get ready for the next stages. For this, select **Initial Condition** then select **Apply Initial Condition** then from the "Keyword List" click on **xydisp** (means both x- and y- displacements) under **Nodal Displacement** keyword and enter 0.0 in XY-Displacement value box then press **Apply**. This way you can reset both x- and y- displacements to zero. Repeat the same for x- and y- velocities as shown below.

Select:  →  → 

Select:  →  → 

Similarly, the displacement and velocity for the structural nodes will be reset:

Select:  →  → 

Select:  →  → 

Dewatering Sequences

To minimize the impact of rapid changes on the model response the dewatering procedure will be performed in four stages at the excavation zone, with the final level at 17.90 m below the ground surface. At each step, a hydrostatic pore pressure distribution as well as total stresses in the elements should be modified at the excavation zone, consistent with the new water table level and the presence of the hydraulic barrier, 30 m below ground level. After each stage of dewatering the model will be cycled to mechanical equilibrium. The javascript function “`modify_pp.ajs`” is used to perform all the tasks listed above. First call the function using the following command:

```
script("call", "filename", "modify_pp.ajs")
```

The for the first stage of dewatering specify the range of the excavation zone up to the 4.8 m depth below surface then solve the model to equilibrium:

```
var x1 = 0.0;
var xu = 30.0;
var y1 = -30.0;
var yu = 0.0;
var gwdepth = -4.8;
modify_pp(x1,xu,y1,yu,gwdepth)
solve()
```

Then repeat the same sequence for 9.3, 14.35 and 17.9 m below the surface:

```
var x1 = 0.0;
var xu = 30.0;
var y1 = -30.0;
var yu = 0.0;
var gwdepth = -9.3;
modify_pp(x1,xu,y1,yu,gwdepth)
solve()
```

```
var x1 = 0.0;
var xu = 30.0;
var y1 = -30.0;
var yu = 0.0;
var gwdepth = -14.35;
modify_pp(x1,xu,y1,yu,gwdepth)
solve()
```

```
var x1 = 0.0;
var xu = 30.0;
var y1 = -30.0;
var yu = 0.0;
var gwdepth = -17.9;
modify_pp(x1,xu,y1,yu,gwdepth)
solve()
```

Contours of pore pressure following dewatering are shown in Figure 9. Displacements induced by dewatering are shown in Figure 10.

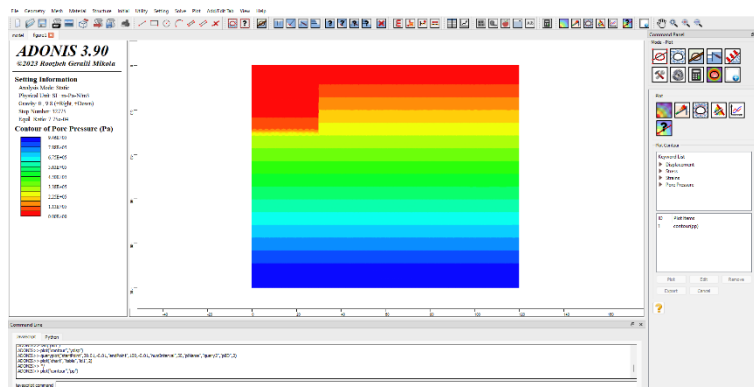


Figure 9- Pore pressure distribution at the end of dewatering.

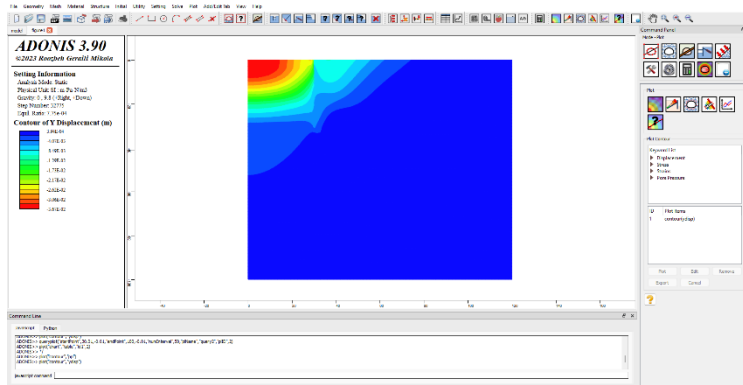


Figure 10- Displacements induced by dewatering.

Perform the Excavation Sequence

Stage 1

First reset the displacement and velocity for the both mesh and structural nodes then select the **Assign Material/Excavate** button from command panel, toolbar or menu, then select **Excavate** from the material name list. Then use mouse and select the top region (from 0 to 4.8 m depth) in the excavation zone and press **Apply**. Now **Solve** the Model until it reaches to equilibrium. Then select the **Structures** button from command panel, toolbar or menu, then select **Tieback** element. Make sure to select **From Struc. Node at Point** from dropdown for the start point (to make sure the start point is attached to existing diaphragm wall structural node) and then enter 10,-1 and 47.6,-13.3 for first and second end points as shown in Figure 11. Make sure the pretension force and grouted length ratio are set to 768 kN and 0.4 (i.e. 40% grouted) respectively. then press Apply. Next click on “**Set Tieback Property**” button in Command Panel and assign the tieback properties (Figure 12) based on the values provided in Table 2.

Table 2- Tieback specifications

Parameters	Row 1	Row 2	Row 3
Depth at wall (m)	4.3	8.8	13.85
Dip (deg.)	27	27	27
Total length (m)	19.8	23.3	23.8
Anchor length (m)	8.0	8.0	8.0
Spacing (m)	2.3	1.35	1.35
Pretension force (kN)	768	945	980
Young's Modulus (GPa)	210	210	210
Cross-sectional area (m ²)	0.0015	0.0015	0.0015
Bond Stiffness, kbond (N/m/m)	1.0x10 ⁸	1.0x10 ⁸	1.0x10 ⁸
Bond Strength, sbond (N/m)	1.0x10 ⁸	1.0x10 ⁸	1.0x10 ⁸

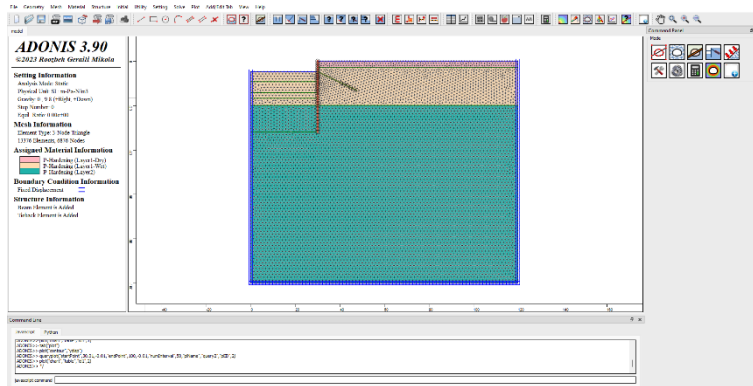


Figure 11- Excavate the top region and install the first tieback.

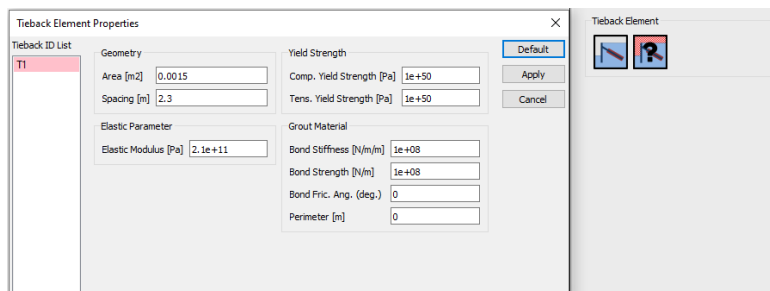


Figure 12- Assign the first tieback properties.

Then select **Solve the Model** button from toolbar, menu or command panel and solve the model until reaching to equilibrium (i.e. select **Solve** and press **Run** button).

Stage 2

Repeat the same sequence and excavate the second part of excavation zone (from 4.8 to 9.3 m depth) **Solve** to equilibrium and install the second tieback and solve to equilibrium. The properties for the second tieback are provided in Table 2. The end points that starts from 30.0,-8.8 to 50.8,-19.4. Please make sure you choose different ID for the second tieback (e.g. ID = 2). Make sure the pretension force and grouted length ratio are set to 945 kN and 0.343 (i.e. 34.3% grouted) respectively.

Stage 3

Again, repeat the sequence and excavate the third part of excavation zone (from 9.3 to 14.35 m depth) and **Solve** to equilibrium then install the third tieback. The properties for the third tieback is provided in Table 2. The end points that starts from 30.0,-13.85 to 51.2,-24.7. Also make sure you choose different ID for the third tieback (e.g. ID = 3). Make sure the pretension force and grouted length ratio are set to 980 kN and 0.336 (i.e. 33.6% grouted) respectively.

Stage 4

For the final stage excavate from 14.35 to 16.80 m depth and **Solve** to equilibrium. The model configuration at final stage is shown in the Figure 13.

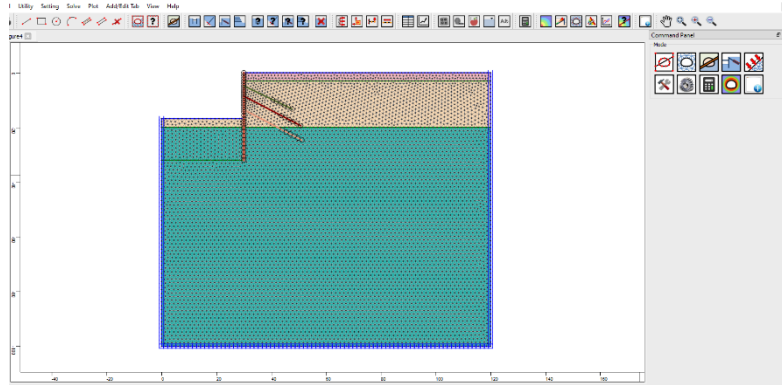


Figure 13- Model configuration at final stage.

Comparisons

After reaching to equilibrium in the final stage, the tools under the **Plot Geometry/Results** in the command panel can be used to plot the results. An extensive set of options is available for plotting the results of the finite element analysis generated by **ADONIS**. These include contouring of the nodal and gauss point variables, strain and stress values, forces in the structural elements and highlighting of plastic zones. Lines or color fills can be selected for all contour plots. To plot horizontal displacement contour, click on **Plot Geometry/Results** button then click on **Plot Contour**, under **Displacement** select **xdisp** and press **Apply** (Figure 14).

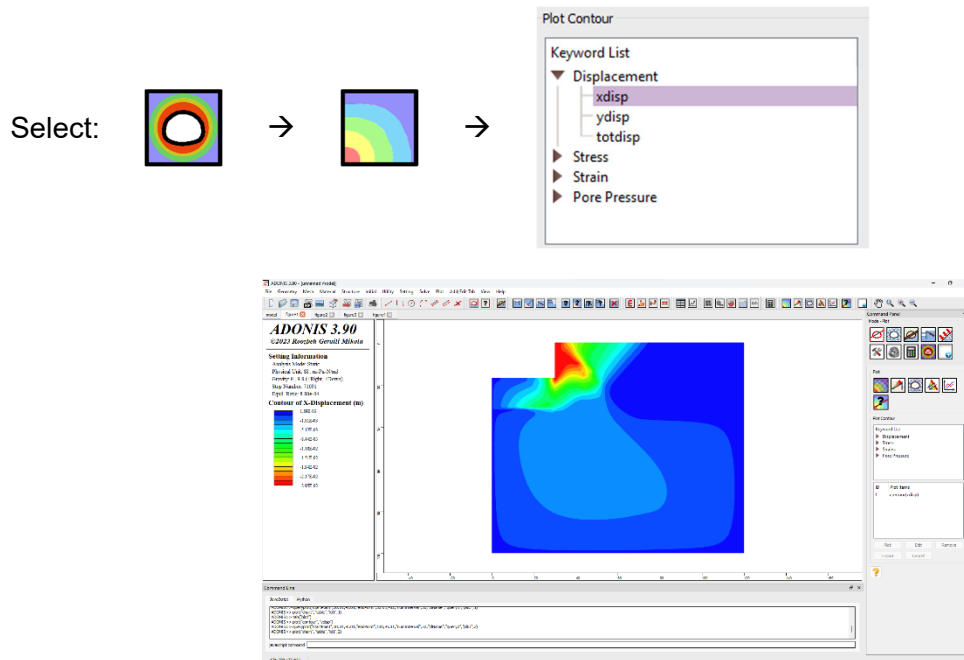
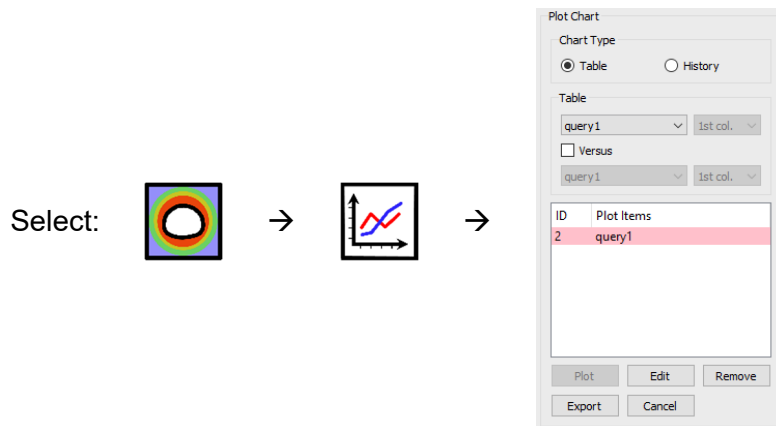


Figure 14- Horizontal displacement contour plot at final stage.

ADONIS provides an option to query data along the specified line anywhere in the model to obtain values interpolated from the contour plots. To create query, user can either use mouse and select start and end points of the query line or directly specify the end points. For this case the start and end points are 30.1,-0.1 and 30.1,-32.



After pressing **Apply** button program automatically will generate the chart for specified query line as shown in Figure 15. Now you can select the name of the plot (i.e. "query1") from the Plot Items list and press Export button to export the values into the text file for future post processing:



The same process can be followed to plot the vertical displacement contour as shown in Figure 16. Create a query line along the 30.1,-0.1 and 100,-0.1 line (Figure 17) and also export the vertical displacement values into text file.

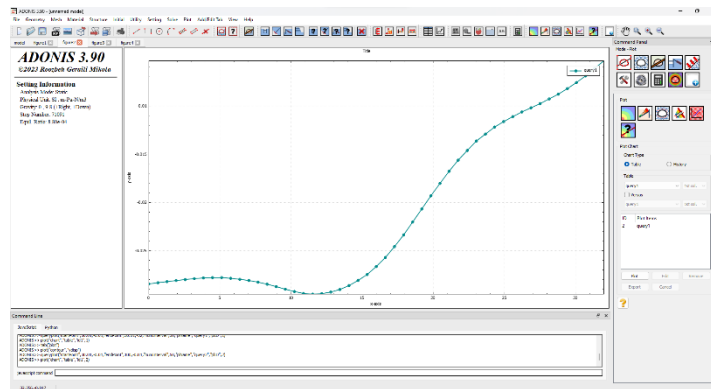


Figure 15- Wall deflection at final stage.

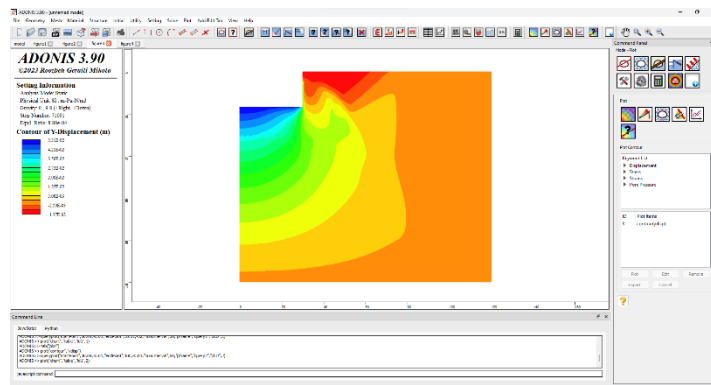


Figure 16- Vertical displacement contour plot at final stage.

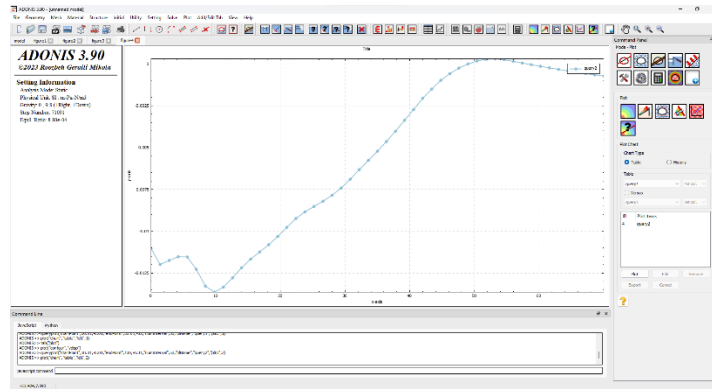


Figure 17- Surface settlement at final stage.

The calculated wall displacements and surface settlement profile after the final excavation step are compared to measurements and to ZSoil (Obrzurd and Truty, 2018), Plaxis (Benz, 2007) and FLAC (2020) predictions in Figures 18 and 19.

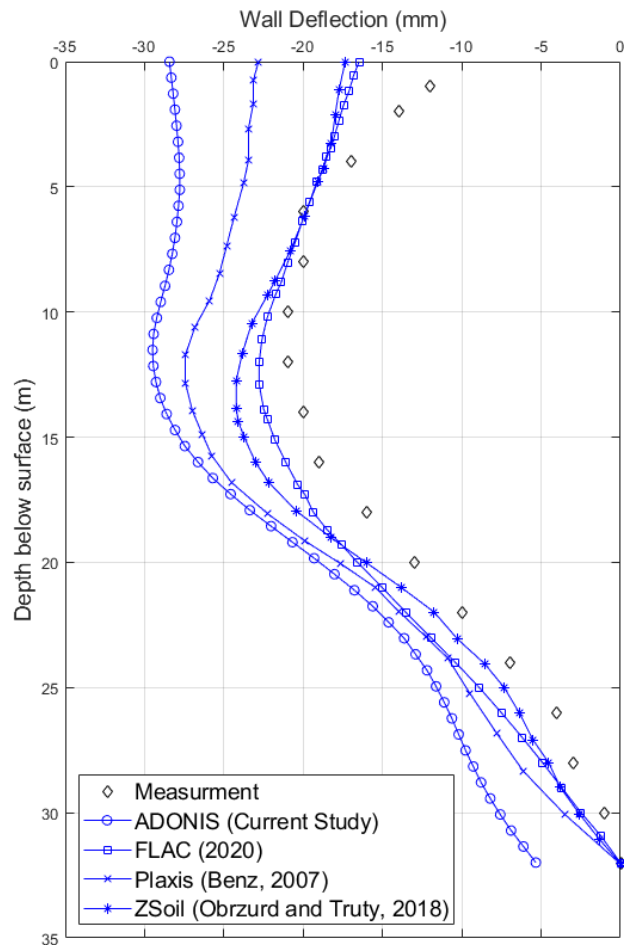


Figure 18- Comparison of wall deflection between measurement and ZSoil, Plaxis, FLAC as well as ADONIS predictions.

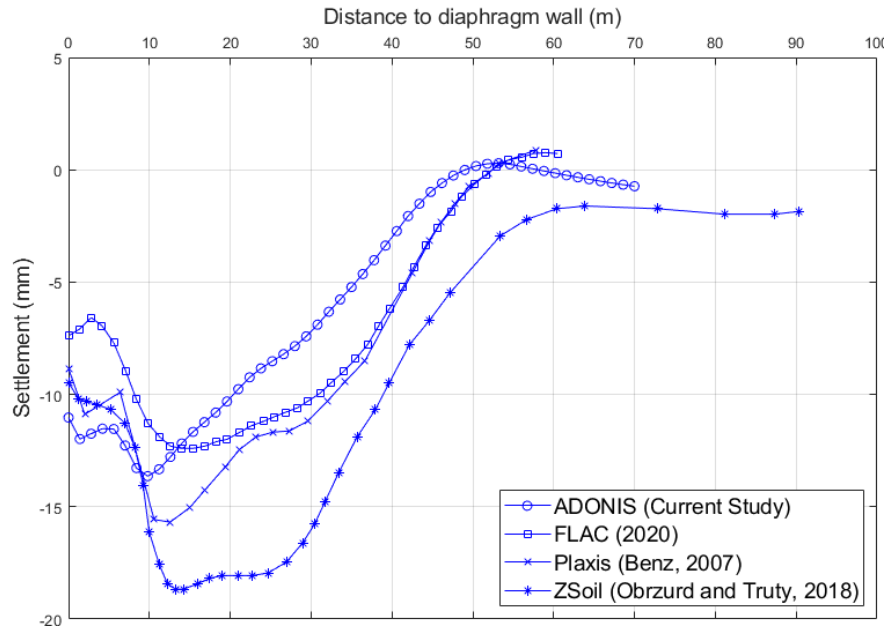


Figure 19- Comparison of surface settlement between measurement and ZSoil, Plaxis, FLAC as well as ADONIS predictions.

References

- Schweiger, H. F. "Results from numerical benchmark exercises in geotechnics," in *5th European Conference Numerical Methods in Geotechnical Engineering*, Vol. 1, pp. 305-314. Numge, Paris (2002).
- Benz T. "Small-Strain Stiffness of Soils and its Numerical Consequences", PhD thesis, Universität Stuttgart (2007).
- Obrzud R.F., Truty A., 2018. "The Hardening Soil model – a practical guidebook", ZSoil.PC 100701 report, Zace Services, Switzerland.
- Itasca Consulting Group (2020). "Fast Lagrangian Analysis of Continua (FLAC)", Minneapolis.

Main Script

The commands for this tutorial are listed below:

```
// create new/fresh model
newmodel()

// set unit
set("unit","stress-pa")

// create geometry
rect("startPoint",0,0,"endPoint",120,-100)
line("startPoint",0,-3,"endPoint",120,-3)
line("startPoint",0,-20,"endPoint",120,-20)
line("startPoint",0,-32,"endPoint",30,-32)
line("startPoint",0,-4.8,"endPoint",30,-4.8)
line("startPoint",0,-9.3,"endPoint",30,-9.3)
line("startPoint",0,-14.35,"endPoint",30,-14.35)
line("startPoint",0,-16.8,"endPoint",30,-16.8)
crack("startPoint",30,0,"endPoint",30,-32)

// discretize
discretize("maxedge",1.5)

// create finer segments in some areas
segment("id",24,"numedge",3)
segment("id",26,27,"numedge",6)
segment("id",2,"numedge",75)
segment("id",16,"numedge",75)

// generate mesh
gmsht("maxedge",1.5,"elementype","T3","useNMD","on")

// create Mohr-Coulomb materials
material("create","Mohr-Coulomb","matid",1,"matname","Layer1-
Dry","density",1900,"shear",7.5e+07,"bulk",1e+08,"coh",0,"fric",35,"dil",5)
material("create","Mohr-Coulomb","matid",2,"matname","Layer1-
Wet","density",2000,"shear",7.5e+07,"bulk",1e+08,"coh",0,"fric",35,"dil",5)
material("create","Mohr-Coulomb","matid",3,"matname","Layer2-
Wet","density",2000,"shear",1.25e+08,"bulk",1.67e+08,"coh",0,"fric",38,"dil",6)

// assign Mohr-Coulomb materials
material("assign","matid",1,"region",16,-1)
material("assign","matid",1,"region",65,-1)
material("assign","matid",2,"region",16,-4)
material("assign","matid",2,"region",16,-7)
material("assign","matid",2,"region",17,-12)
material("assign","matid",2,"region",16,-15)
material("assign","matid",2,"region",16,-18)
material("assign","matid",2,"region",60,-12.5)
material("assign","matid",3,"region",20,-27)
material("assign","matid",3,"region",50,-33)

// assign boundary condition
applybc("xfix","xlim",-0.1,0.1,"ylim",-100.1,0.1)
applybc("xfix","xlim",119.9,120.1,"ylim",-100.1,0.1)
applybc("xyfix","xlim",-0.1,120.1,"ylim",-100.1,-99.9)

// hide the excavation zone to be able to install the diaphragm wall
excavate("region",16.507,-1.40485,"reset","off")
excavate("region",15.4534,-4.03894,"reset","off")
excavate("region",15.1021,-6.67304,"reset","off")
excavate("region",15.2777,-11.2388,"reset","off")
excavate("region",14.2241,-15.2777,"reset","off")
excavate("region",14.2241,-18.0874,"reset","off")
excavate("region",15.2777,-26.1653,"reset","off")

// install the diaphragm wall
structure("drawliner","beamid",1,"iftype","bothSides","ifid1",1,"ifid2",2,"xlim",29.9,30.1,"ylim",
,-32.1,0.1)

// assign beam element properties
structure("material","beamid",1,"area",0.8,"I",0.04267,"ymod",3.0E10,"dens",0)

// assign interfaces properties
imaterial("assign","Coulomb","ifid",1,"matname","Interface1","jkn",1e9,"jks",1e9,"friction",25)
imaterial("assign","Coulomb","ifid",2,"matname","Interface2","jkn",1e9,"jks",1e9,"friction",25)

// backfill the temporarily excavated zones
```

```

backfill("region",16.507,-1.40485,"reset","off")
backfill("region",15.4534,-4.03894,"reset","off")
backfill("region",15.1021,-6.67304,"reset","off")
backfill("region",15.2777,-11.2388,"reset","off")
backfill("region",14.2241,-15.2777,"reset","off")
backfill("region",14.2241,-18.0874,"reset","off")
backfill("region",15.2777,-26.1653,"reset","off")

// apply gravity force
set("gravity",0,9.8)

// modify pore-pressure and initialize horizontal stresses based on k0 values
script("call","filename","initial_pp_stresses.ajs")

// solve to equilibrium
solve()

// create P-Hardening soil models
material("create","P-Hardening","matid",5,"matname","Layer1-
Dry","density",1900,"E50_ref",4.5E7,"Eur_ref",1.8E8,"p_ref",100000.0,"m",0.55,"ocr",1.0,"Eoed_ref",
4.5E7,"cohesion",0,"friction",35,"dilation",5);
material("create","P-Hardening","matid",6,"matname","Layer1-
Wet","density",2000,"E50_ref",4.5E7,"Eur_ref",1.8E8,"p_ref",100000.0,"m",0.55,"ocr",1.0,"Eoed_ref",
4.5E7,"cohesion",0,"friction",35,"dilation",5);
material("create","P-
Hardening","matid",7,"matname","Layer2","density",2000,"E50_ref",7.5E7,"Eur_ref",3.0E8,"p_ref",10
0000.0,"m",0.55,"ocr",1.0,"Eoed_ref",7.5E7,"cohesion",0,"friction",38,"dilation",6);

// assign materials
material("assign","matid",5,"region",16,-1)
material("assign","matid",5,"region",65,-1)
material("assign","matid",6,"region",16,-4)
material("assign","matid",6,"region",16,-7)
material("assign","matid",6,"region",17,-12)
material("assign","matid",6,"region",16,-15)
material("assign","matid",6,"region",16,-18)
material("assign","matid",6,"region",60,-12.5)
material("assign","matid",7,"region",20,-27)
material("assign","matid",7,"region",50,-33)

// call subroutine to assign effective principal stresses to each element
// these parameters are required for P-Hardening model
script("call","filename","initial_principal_stresses.ajs")

// solve to equilibrium
solve()

// reset the displacement and velocity
initial("xydisp",0)
initial("xyvel",0,0)
structure("initial","reset","alldisp")
structure("initial","reset","allvel")

// call the script that contains function for
// modifying the pore-pressure and totla stresses
// during dewatering
script("call","filename","modify_pp.ajs")

// dewater from 0 to 4.8 m depth below surface
var x1 = 0.0;
var xu = 30.0;
var y1 = -30.0;
var yu = 0.0;
var gwdepth = -4.8;
modify_pp(x1,xu,y1,yu,gwdepth)
solve()

// dewater from 4.8 to 9.3 m depth below surface
var x1 = 0.0;
var xu = 30.0;
var y1 = -30.0;
var yu = 0.0;
var gwdepth = -9.3;
modify_pp(x1,xu,y1,yu,gwdepth)
solve()

// dewater from 0 to 14.35 m depth below surface
var x1 = 0.0;
var xu = 30.0;
var y1 = -30.0;
var yu = 0.0;

```

```

var gwdepth = -14.35;
modify_pp(xl,xu,y1,yu,gwdepth)
solve()

// dewater from 0 to 17.9 m depth below surface
var xl = 0.0;
var xu = 30.0;
var y1 = -30.0;
var yu = 0.0;
var gwdepth = -17.9;
modify_pp(xl,xu,y1,yu,gwdepth)
solve()

// reset the displacement and velocity to zero
initial("xydisp",0)
initial("xyvel",0.0)
structure("initial","reset","alldisp")
structure("initial","reset","allvel")

// excavate 0-3 m below surface
excavate("region",15,-2.0)

// solve to equilibrium
solve()

// excavate 3-4.8 m below surface
excavate("region",15,-4.0)

// solve to equilibrium
solve()

// install the first tieback (start node will be attached to existing diaphragm wall node)
structure("drawtieback","tieid",1,"fromstrucnodeatpoint",30.000,-4.4,"topoint",47.642,-
13.289,"pretens",768000,"grouted",0.4,"segnum",5)
structure("material","tieid",1,"area",0.0015,"ymod",2.1e+11,"kbond",1e8,"sbond",1e8,"spacing",2.3
)

// solve to equilibrium
solve()

// excavate 4.8-9.3 m below surface
excavate("region",15,-8.0)

// solve to equilibrium
solve()

// install the second tieback (start node will be attached to existing diaphragm wall node)
structure("drawtieback","tieid",2,"fromstrucnodeatpoint",30.000,-8.8,"topoint",50.76,-
19.378,"pretens",945000,"grouted",0.343,"segnum",5)
structure("material","tieid",2,"area",0.0015,"ymod",2.1e+11,"kbond",1e8,"sbond",1e8,"spacing",1.3
5)

// solve to equilibrium
solve()

// excavate 9.3-14.35 m below surface
excavate("region",15,-12.5)

// solve to equilibrium
solve()

// install the third tieback (start node will be attached to existing diaphragm wall node)
structure("drawtieback","tieid",3,"fromstrucnodeatpoint",30.000,-13.85,"topoint",51.2,-
24.655,"pretens",980000,"grouted",0.336,"segnum",5)
structure("material","tieid",3,"area",0.0015,"ymod",2.1e+11,"kbond",1e8,"sbond",1e8,"spacing",1.3
5)

// solve to equilibrium
solve()

// excavate 14.35-16.8 m below surface
excavate("region",15,-15)

// solve to equilibrium
solve()

// plot the horizontal and vertical displacement and extract data
plot("contour","xdisp")
queryplot("startPoint",30.01,-0.01,"endPoint",30.01,-
32,"numInterval",50,"plName","query1","plID",1)
plot("chart","table","id1",1)

```

```
tab("plot")
plot("contour", "ydisp")
queryplot("startPoint", 30.01, -0.01, "endPoint", 100, -
0.01, "numInterval", 50, "plName", "query2", "plID", 2)
plot("chart", "table", "id1", 2)
```

Appendix A

The script in "initial_pp_stresses.ajs" file is listed below:

```
/**
***** loop through all elements and initialize pp and stresses
*****
*/
gwy = -3.0;
dens_water = 1000.0;
ygrav = 9.8;
dens_dry = 1900.0;
dens_wet = 2000.0;
k0x1=0.43
k0z1=0.43
k0x2=0.38
k0z2=0.38

elist = getelem("allid"); // get list of elements
for (i = 0; i < elist.length; i++) {
  var eid = elist[i];
  var ngp = getelem("numgauss", eid); // number of gauss point
  for (j = 0; j < ngp; j++) {
    // get current gauss pont position
    var gp_pos = getelem("gausspointpos", eid, j+1);
    var xp = gp_pos[0];
    var yp = gp_pos[1];

    // calculate pore pressure and total vertical stress
    var depth;
    var syy_val = 0;
    var pp_val = 0;
    if(yp < gwy) {
```

```

    depth = gwy - yp;
    pp_val = depth * dens_water * ygrav;
    syy_val = (gwy * dens_dry - depth * dens_wet) * ygrav;
  } else {
    syy_val = (yp * dens_dry * ygrav);
  }

  // calculate the horizontal totla stresses
  var k0x, k0z;
  if(yp > -20) {
    k0x = k0x1;
    k0z = k0z1;
  } else {
    k0x = k0x2;
    k0z = k0z2;
  }

  sxx_val = k0x * (syy_val + pp_val) - pp_val;
  szz_val = k0z * (syy_val + pp_val) - pp_val;

  // assing pore pressure and stresses to current gauss point
  setelem("pp",eid,j+1,pp_val);
  setelem("stress",eid,j+1,1,sxx_val);
  setelem("stress",eid,j+1,2,syy_val);
  setelem("stress",eid,j+1,4,szz_val);
}
}
}

```

Appendix B

The script in “initial_principal_stresses.ajs” file is listed below:

```

//*****
//*** loop through all elements and calculate sig1, sig2 and sig3
//*** then assign the calculated values into each element
//*****
elist = getelem("allid"); // get list of elements
for (i = 0; i < elist.length; i++) {
  var eid = elist[i];
  var ngp = getelem("numgauss", eid); // number of gauss point
  for (j = 0; j < ngp; j++) {
    // get stresses at current gauss point
    var stress_array = getelem("stress", eid, j+1);
    var sxx_val = stress_array[0];
    var syy_val = stress_array[1];
    var sxy_val = stress_array[2];
    var szz_val = stress_array[3];
    // get pore pressure at current gauss point
    var pp_val = getelem("pp", eid, j+1);

    // calculate the principal stresses
    var sdif = sxx_val - syy_val;
    var s0 = 0.5 * (sxx_val + syy_val);
    var st = 4.0 * sxy_val*sxy_val;
    var rad = 0.5 * Math.sqrt(sdif*sdif + st);
    var si = s0 - rad;
    var sii = s0 + rad;

    var smin, smed, smax;
    if (szz_val > sii) {
      // ---- szz is minor p.s. ----
      smax = si;
    }
  }
}

```

```

    smed = sii;
    smin = szz_val;
  } else if (szz_val < si) {
    // ---- szz is major p.s. ----
    smax = szz_val;
    smed = si;
    smin = sii;
  } else {
    // ---- szz is intermediate ---
    smax = si;
    smed = szz_val;
    smin = sii;
  }

  // effective principal stress
  var sig1_val = smax + pp_val;
  var sig2_val = smed + pp_val;
  var sig3_val = smin + pp_val;

  // assign the calculated effective principal stresses into current gauss point
  // please note that this task is perform for initialization of the P-Hardeing soil model
  setelem("prop", "sig1", eid, j+1, sig1_val);
  setelem("prop", "sig2", eid, j+1, sig2_val);
  setelem("prop", "sig3", eid, j+1, sig3_val);
}
}

```

Appendix C

The script in “[modify_pp.ajs](#)” file is listed below:

```

//*****//
//*** function to modify the pore pressure and total stresses
//*** during dewatering process
//*** Inputs
//*** x1,xu : lower and upper bound along x-axis
//*** y1,yu : lower and upper bound along y-axis
//*** gwdepth : depth of water table
//*****//
function modify_pp(x1,xu,y1,yu,gwdepth) {
  elist = getelem("allid"); // get list of elements
  for (i = 0; i < elist.length; i++) {
    var eid = elist[i];
    var ngp = getelem("numgauss", eid); // number of gauss point
    for (j = 0; j < ngp; j++) {
      // get current gauss point position
      var gp_pos = getelem("gausspointpos", eid, j+1);
      var xp = gp_pos[0];
      var yp = gp_pos[1];

      // if gauss point in the range
      if(xp > x1 && xp < xu && yp > y1 && yp < yu) {

        // get previous pore pressure value
        var old_pp_val = getelem("pp", eid, j+1);
        var new_pp_val = 0;

        // get stresses at current gauss point
        var stress_array = getelem("stress", eid, j+1);
        var sxx_val = stress_array[0];
        var syy_val = stress_array[1];
        var szz_val = stress_array[3];
      }
    }
  }
}

```

```
// calculate new pore pressure value
if(y_p < gwdepth) {
    var depth = (gwdepth-y_p);
    new_pp_val = depth * 1000 * 9.8;
}
// assign the new pore pressure value
setelem("pp",eid,j+1,new_pp_val);

// modify total stress and make sure effective stress remain unchanged
sxx_val = (sxx_val + old_pp_val) - new_pp_val;
syy_val = (syy_val + old_pp_val) - new_pp_val;
szz_val = (szz_val + old_pp_val) - new_pp_val;

// assign new total stresses
setelem("stress",eid,j+1,1,sxx_val);
setelem("stress",eid,j+1,2,syy_val);
setelem("stress",eid,j+1,4,szz_val);
}
}
}
```