S T K

Scientific ToolKit for OpenSees

REVOLUTIONARY TOOLKIT FOR OPENSEES

USER MANUAL

Contents

INTRODUCTION

The Scientific Toolkit for Opensees (STKO) User Manual contains all the essential and practical information needed for the proper utilization of the program. This manual describes the functions and capabilities of the program from the preprocessing to the postprocessing phases, including procedures for meshing, analysis, and an introduction to customizing the program.

What is STKO?

STKO is a cutting-edge complex data visualization tool for Opensees. By using STKO, a user can create a Tool Command Language (TCL) input file for Opensees, and after processing, read the output file and data through its graphic interface. STKO uses an HDF5 database library and Python-based scripting interface for the manipulation and customization of results.
webpage: www.asdeasoft.net/stko

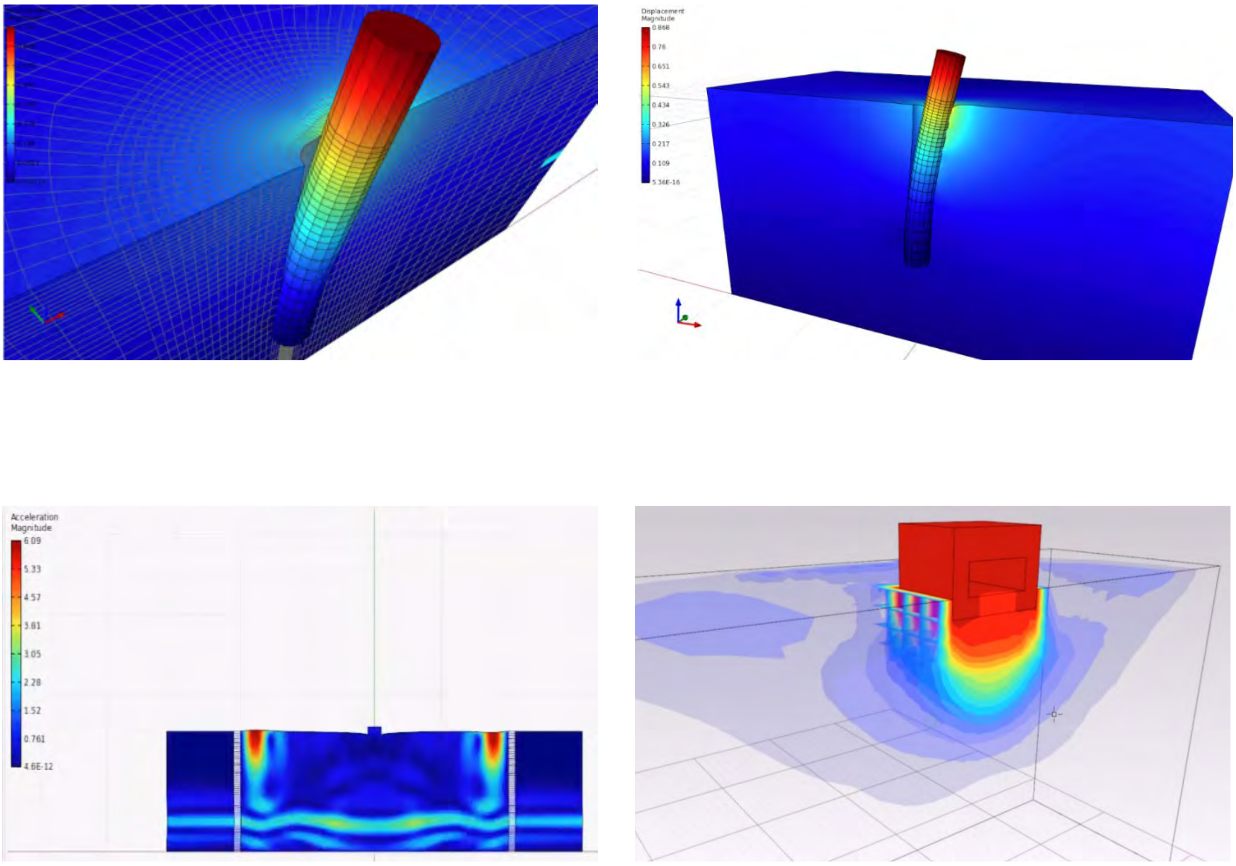


Figure 1. STKO examples

1. GET STARTED - STKO USER INTERFACE

STKO's user-friendly interface is easy to use and enables users to manage models with full control. The program includes two modules: the **Preprocessor** and **the Postprocessor**.

1.1. System Requirements

In order to run correctly, STKO has the following system requirements:

- 64-bit processor architecture
- 4 GB memory
- At least 500 MB disk space available for installation
- Video Card supporting at least OpenGL 3.0 or higher and GLSL (OpenGL Shading Language) 1.20 or higher

1.2. User Interface (How to Navigate)

This section introduces the preprocessing and postprocessing modules. The following examples are from the postprocessing module. It should be noted that some features exist in both modules.

The program interface is divided into 8 sections (*Figure 2)*:

(1) Pre- and Postprocessor interfaces
(2) Main Toolbar
(3) Work Tree window
(4) Terminal window
(5) Editor window
(6) Quick access window
(7) Render view
(8) Python Script API

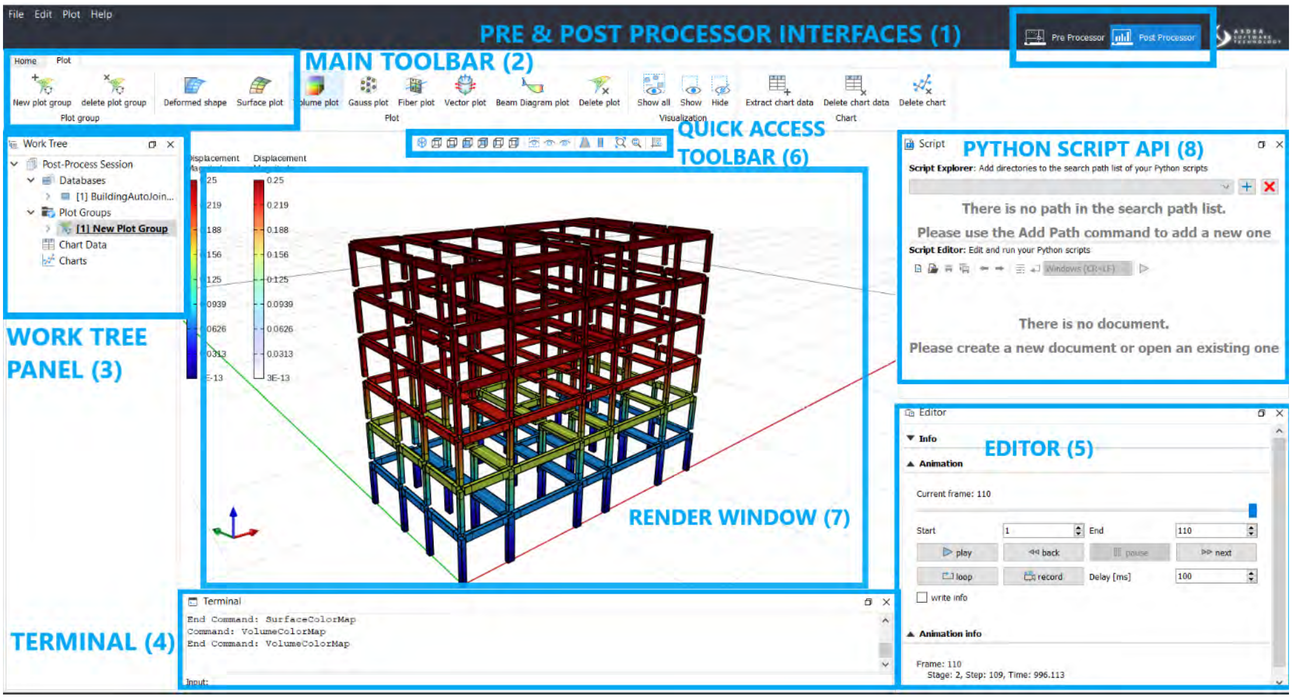


Figure 2. STKO Interface.

(1) Pre-and Postprocessor Interfaces Users can select the desired module by *clicking* on the corresponding button on the upper right-hand side of the toolbar next to the ASDEA software logo.

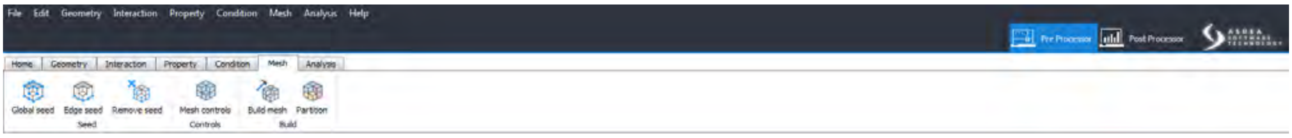


Figure 3. Preprocessor Toolbar

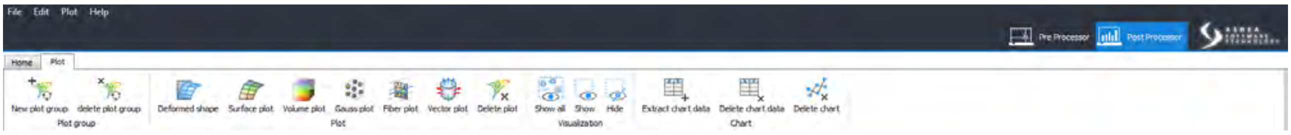


Figure 4. Postprocessor Toolbar

(2) Main Toolbar This Toolbar organizes and displays tools by task.

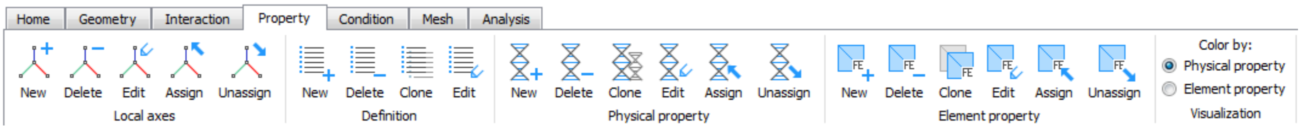


Figure 5. Main Toolbar

(3) Work Tree The Work Tree contains all the hierarchical information about the models. An example of the Preprocessor work tree is shown in Figure 6.

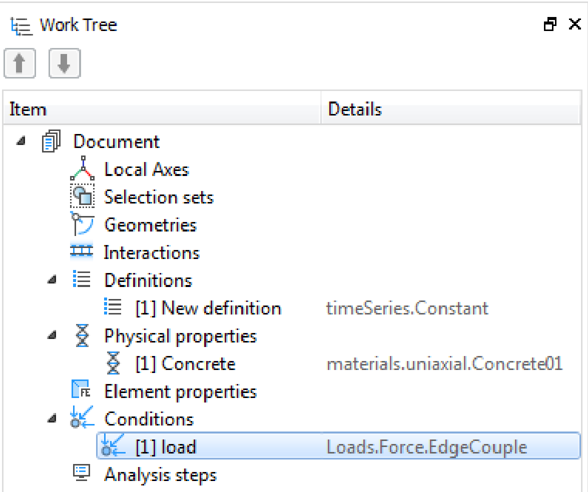


Figure 6. Preprocessor Work Tree

The **Preprocessor Work Tree** uses **Colored Definitions**. These colored definitions help the user track which definitions, physical and element properties, and conditions are Not Used, Used, Referenced, or Used and Referenced. See the figure below:

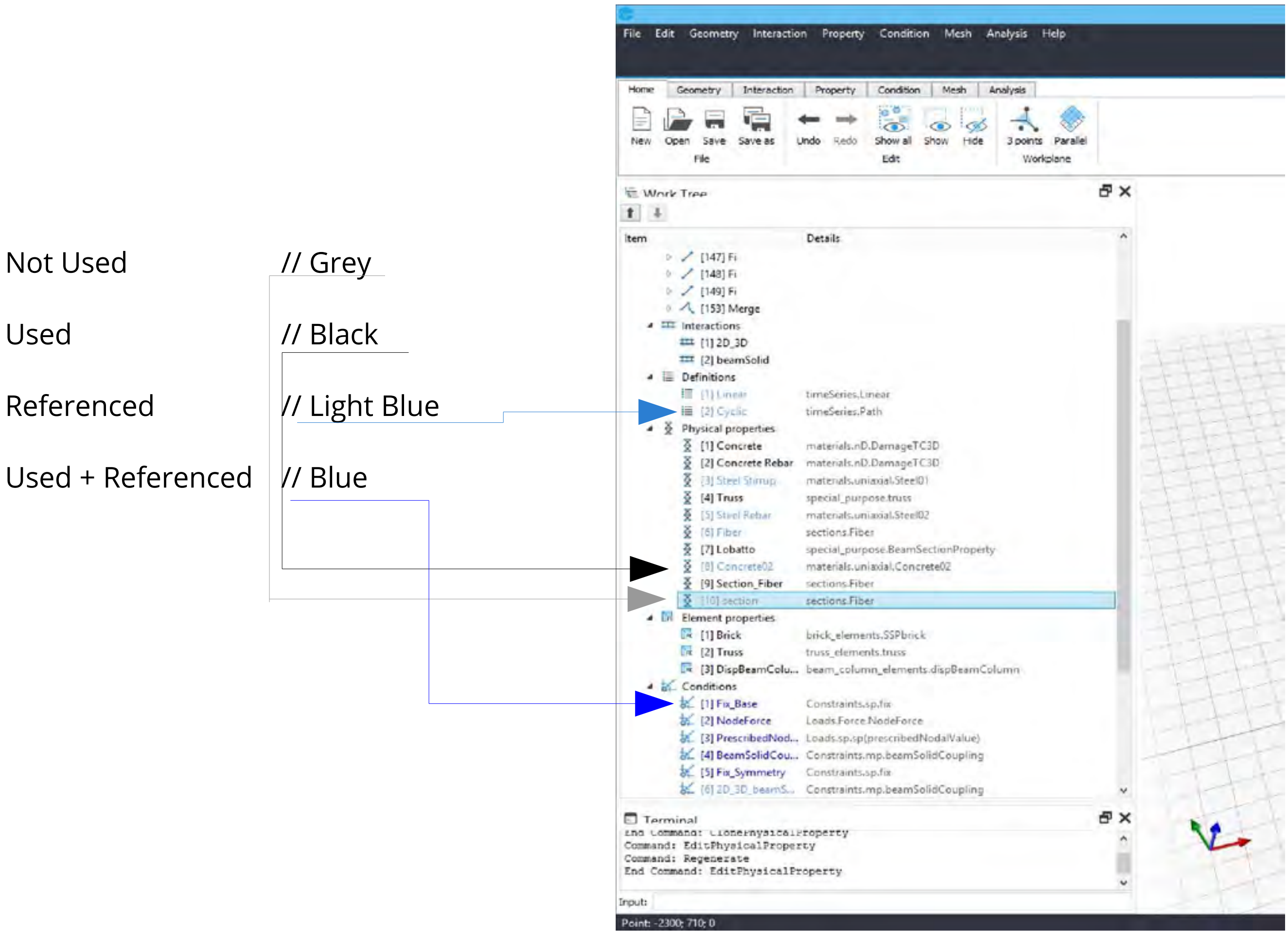


Figure 7. Colored Definitions in the Preprocessor Work Tree

Not Used: Items in the Work Tree (listed above) that are not applied in the analysis or to the geometry will appear grey (properties and elements are applied to the geometries, the definitions in the analysis, and the conditions are applied both to the geometries and the analysis).

Used: The work tree items only applied during the analysis or to the geometry will appear black (properties and elements are applied to geometries, definitions to the analysis, and conditions to both the geometry and the analysis).

Referenced: The work tree items only referring to another property will appear light blue. For example, when a material section->layeredShell is made with materials inside (ex. elasticIsotropic), the materials inside the layeredShell are considered referenced (so light blue).

Used + Referenced: The work tree item is both referenced and used and will appear blue. For example, when a material section->layeredShell is made with materials inside (ex. elasticIsotropic), the internal materials are considered referenced, but, if we apply the same elasticIsotropic material to a geometry, it will become blue as it is both referenced in another material and used on its own.

In the **Preprocessor Work Tree**, the user can **move** previously created items to list them in the desired order regardless of when they were created. The up and down arrows allow the user to move the items along the Work Tree.

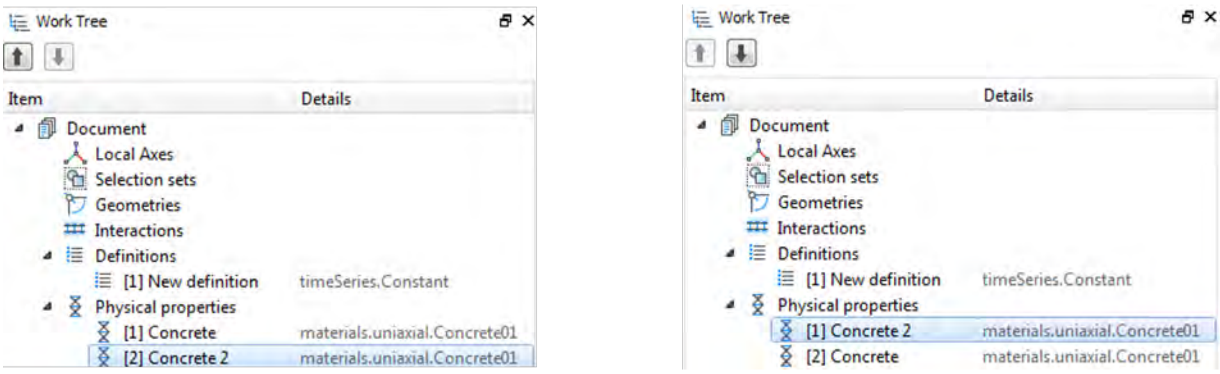


Figure 8. Before (left) and after (right) moving a previously created item

The **Postprocessor Work Tree** organizes your databases, plot groups, chart data, and charts, as shown in the figure below.

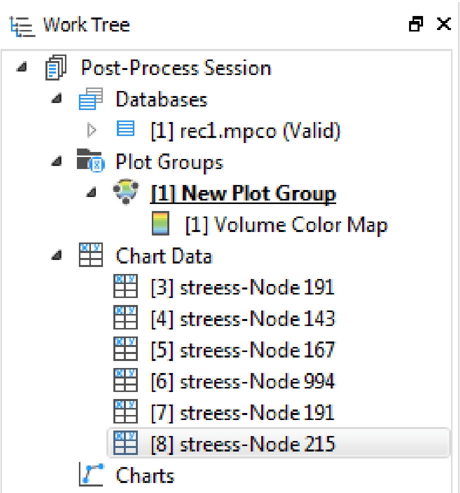


Figure 9. Post-Processor Work Tree

(4) Terminal Window This Window contains two text editors. The first one shows the program command output. The second one shows the **Input** bar, which is used to enter commands, coordinates, and values directly using the keyboard. The Input bar also has an auto compilation feature

Figure 12).

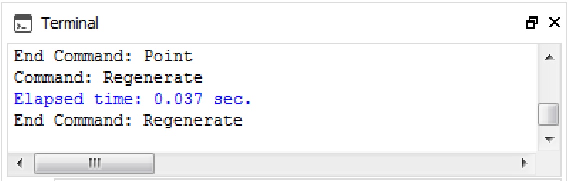


Figure 10. Software output

Figure 11. User Input

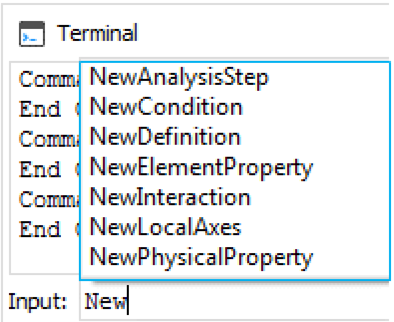


Figure 12. Example of the auto compilation feature

(5) Editor Window This window shows all features related to the object selected in the postprocessor module: *information, surface color map, color map, and visualization.*

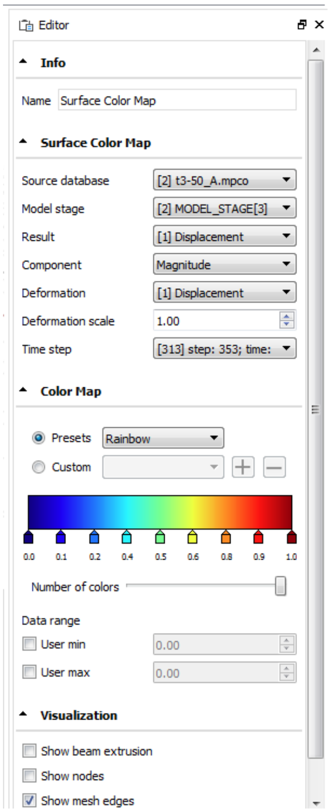


Figure 13. Editor window

(6) Quick access Toolbar This toolbar allows the user to set the views, show/hide geometries based on their physical/element properties, set perspective or parallel, zoom, edit display options and labels, set the grid, show/hide elements and local axes, and select/unselect elements to view.

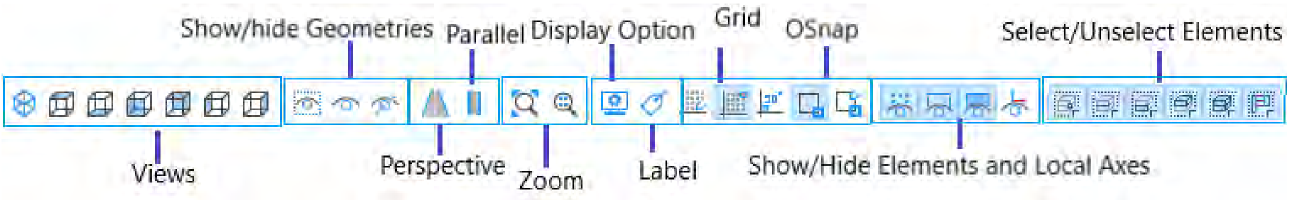


Figure 14. Quick access toolbar

(7) Render window This window shows the model and its features.

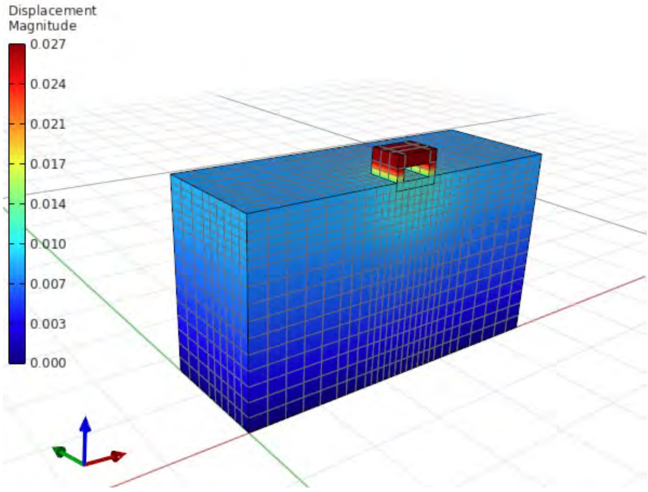


Figure 15. Render window

(8) Python Scripting Interface The interface is available in both the pre and postprocessor to allow users to interact with STKO through Python. The window is divided into two sections: The script explorer and the script editor. The script explorer allows users to search through the scripts that they have added, while the script editor lets users make changes to and then run the scripts they have in their library

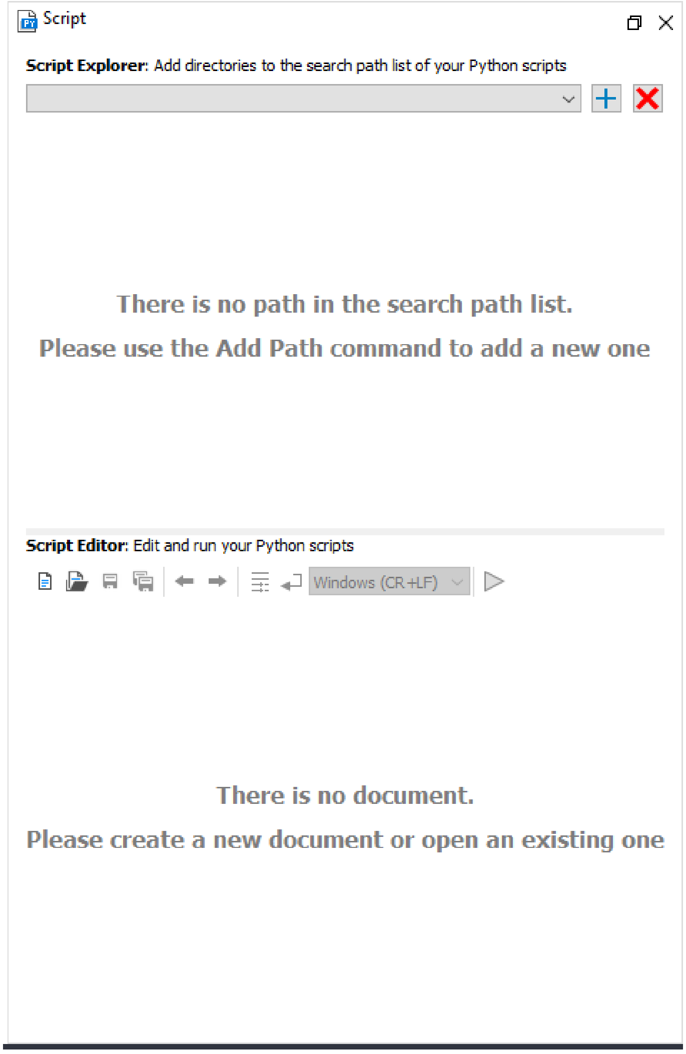


Figure 16. Python Scripting Interface

1.2.1. Handling Selection

The user can select more than one element of the geometry by using the Quick Access toolbar commands above the render window.

Figure 17. Handling Selection Toolbar

From the Quick Access toolbar, users can access the **Display Option Editor,** when set to auto, it shows the active elements. Otherwise, by deactivating auto, the user can select what to view.

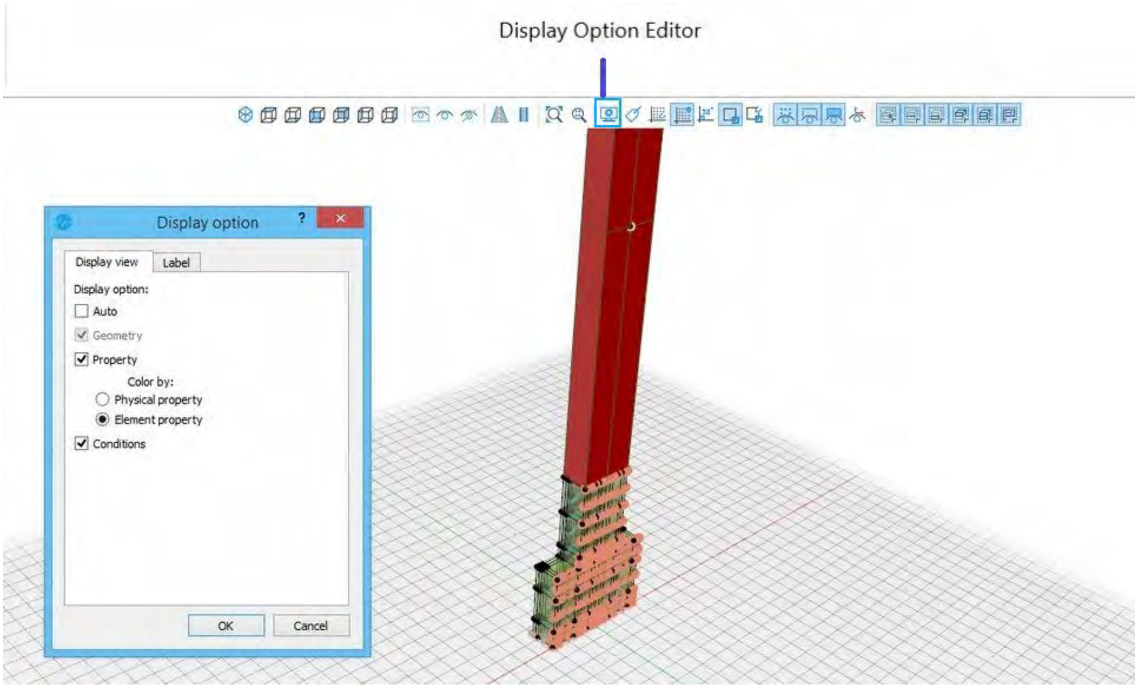


Figure 18. *Display Option Editor*

The **Label Tab** on the **Display Option Editor** allows the user to choose to view labels that contain information about the geometries, or subgeometries, and the properties assigned to them. Using this tab, users can customize what information they wish to view and which geometries/subgeometries.

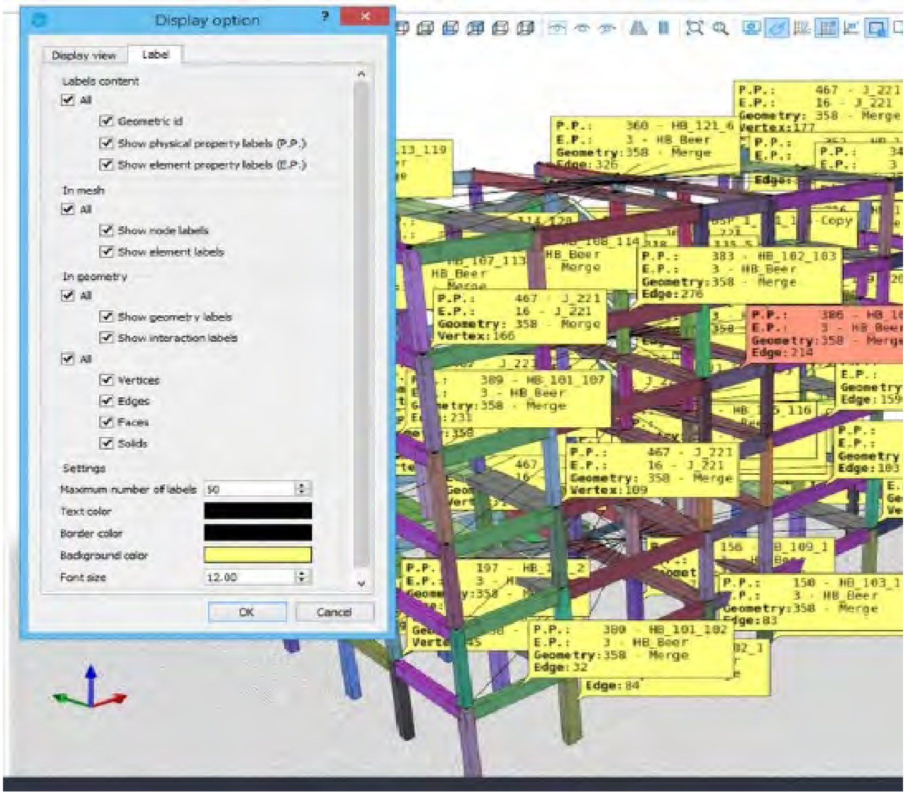


Figure 19. Label Tab on the Display Option Editor

To show or hide labels, simply *Click* Label on the Quick Access Toolbar.

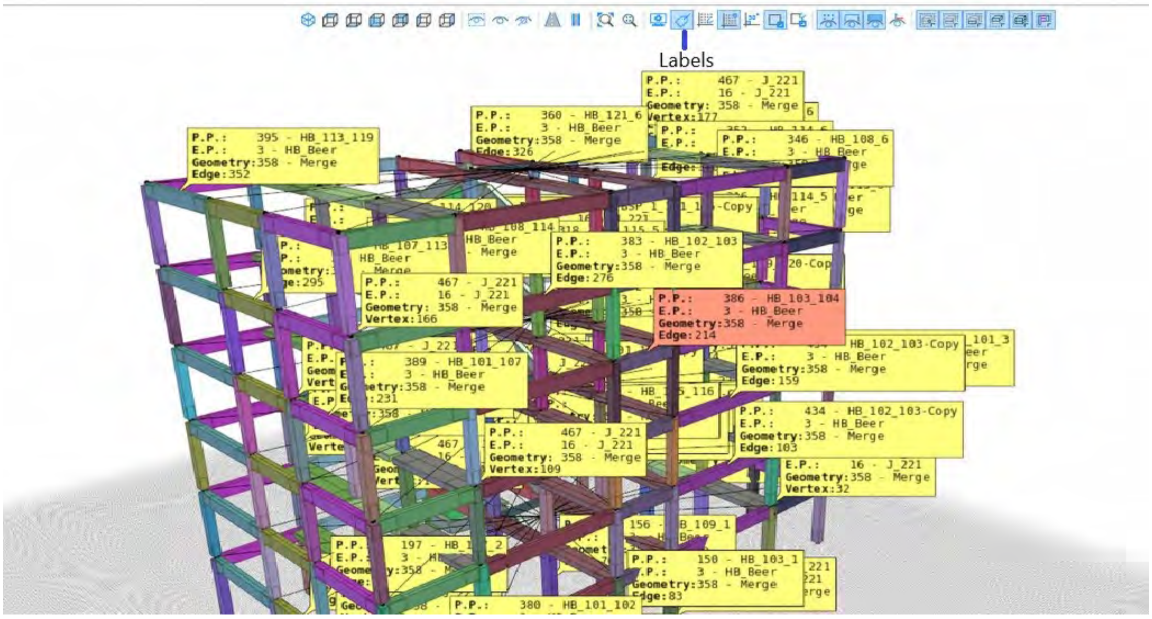


Figure 20. Show Labels using the Quick Access Toolbar

Geometries can be quickly selected in the ***Work Tree,*** where all the model's geometries are listed under **Geometry**. By *click*ing on **Geometry**, a subgroup will appear so that the user may select not only the whole geometry but also individual elements of the geometry.

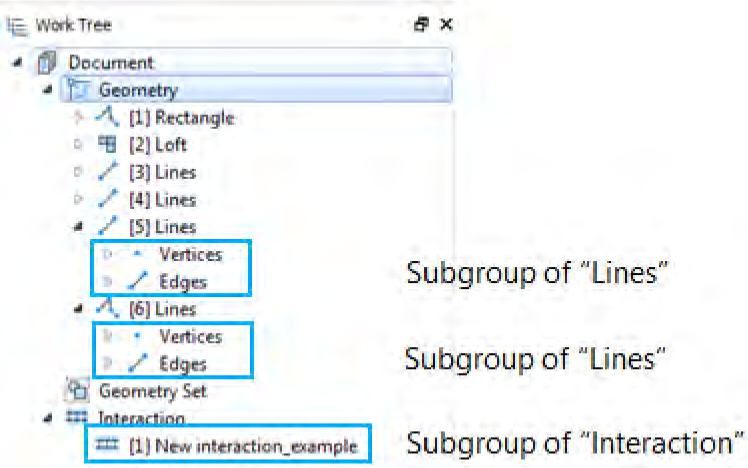


Figure 21. Work Tree

Some commands (such as Loft, CAD commands, etc…) do not permit multiple selections. For this reason, when these commands are used, the selection command in the Quick Access Toolbar will be turned off.

Users can also choose to show or hide geometries based on their physical or element properties. To use this feature, simply *right-click* on the property in the **work tree** and choose **Show** or **Hide**. The geometries the property has been applied to will then either be shown or hidden.

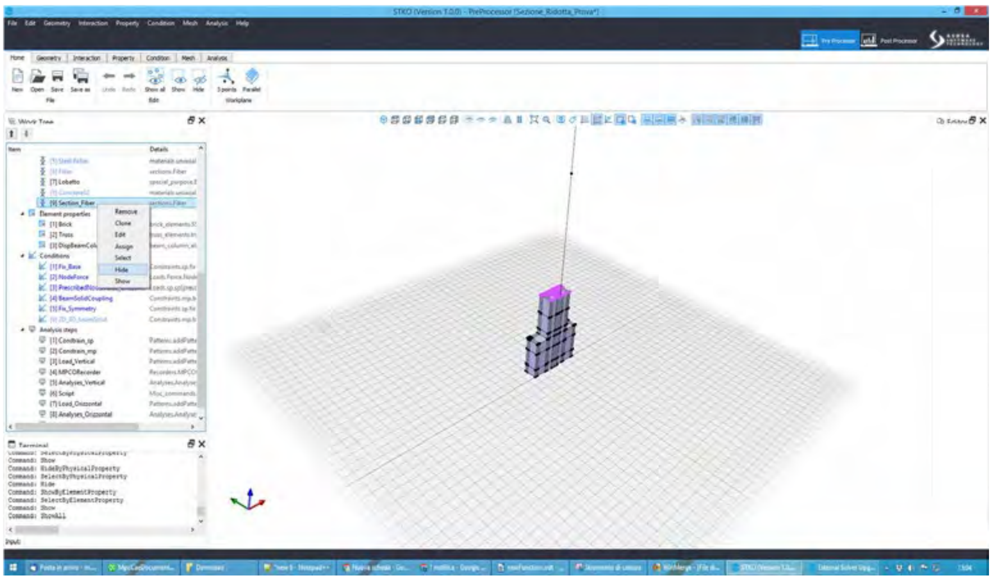


Figure 22. Show/Hide Geometries Based on Physical and Element Properties

1.2.2. Create, Edit, and Save Pre- and Postprocessor Documents

To **Create** a new document, or **Open** an existing one, the user can choose the corresponding commands on the **preprocessor interface**.

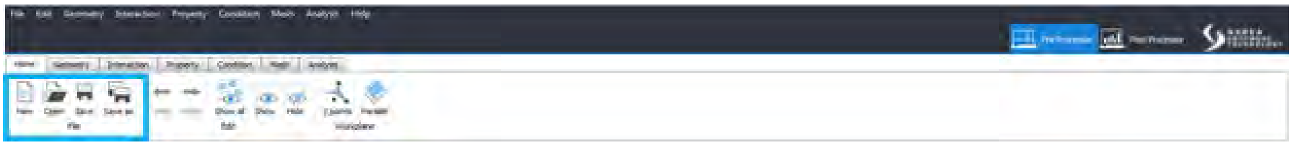


Figure 23. Home section in the Preprocessor main Toolbar

To save the current document, using the main Toolbar select: **Home** > **Save.** *Click* **Save as** to save the current document to a new location. If the current document has never been saved before, the **Save** command will automatically open the **Save as** command for naming the file and choosing the location. The file extension is **.scd** (STKO CAE Document).
The **postprocessor interface** stores data such as loaded output databases, plots, charts, etc. The file extension is **.sped** (STKO Postprocessing Document). Like in the preprocessor interface, to save the current document, use the main Toolbar to select: **Home** > **Save**. Use **Save as** to save the current document to a new location. If the current document has never been saved before, the **Save** command will automatically open the **Save as** command for naming the file and choosing the location.

To **Open** an existing DB (database), **Reload**, or **Close** a DB, *click* on the corresponding commands:

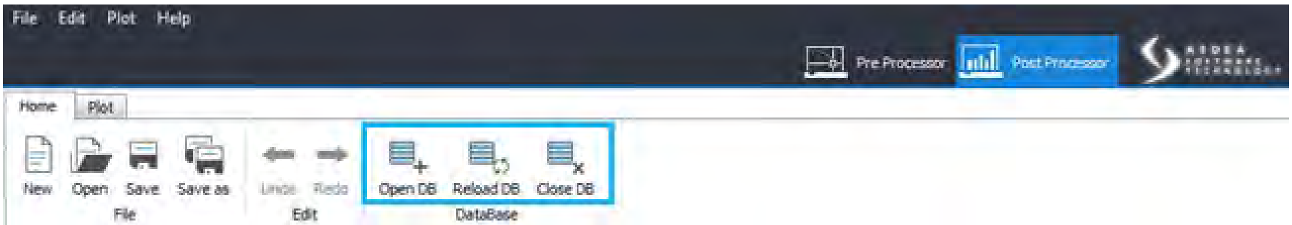


Figure 24. Home section in the Post Processor main Toolbar

The program allows the loading of more than one Database at a time in the **Postprocessor Section**, Plots, Plot Groups, charts, and data. Whenever the user opens a saved postprocessor document, it will include **.mpco** files with Plot Groups and Charts created by the user without losing data.
The release of version 2.0 included added support for AllFiles in the OpenFile Dialog to load the MPCO output database. This means it is now possible to open databases with different extensions, like .h5, the original extension of HDF5 files, the MPCO recorder database type.

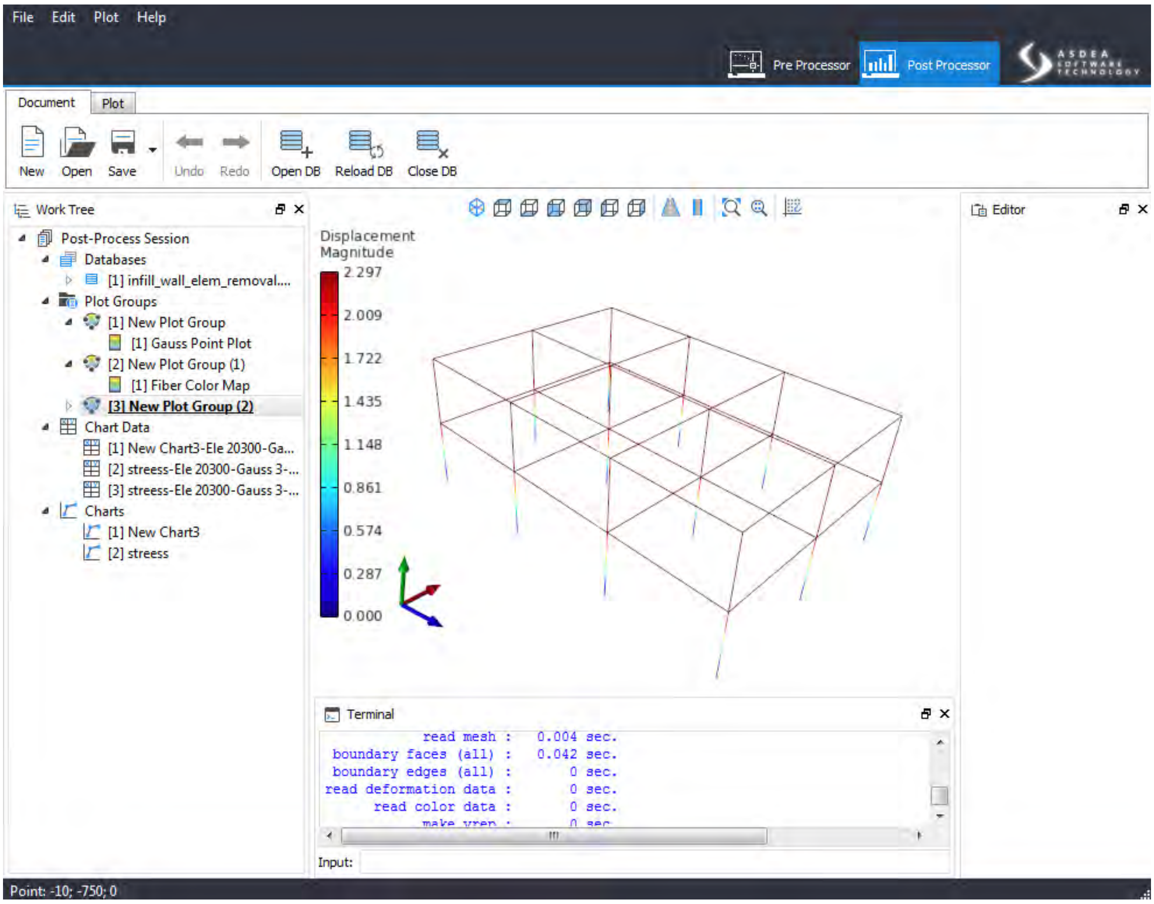


Figure 25. Post-Processor Work Tree of a Structure

The user can choose between different Plot types (Deformed Shape, Surface plot, Volume plot, Beam/Shell Fiber plot, and Gauss Point Plot).

Right-click **Plot Groups** > **New Plot Group** on the Work Tree**.** *Right-click* on the newly generated **New Plot Group** and select which Plot to show.

Click on the New Plot on the Work Tree and the Editor Panel will open. The Editor Panel allows the user to view all the relevant data i.e.: *Information, Data, Color Map, and Visualization*.

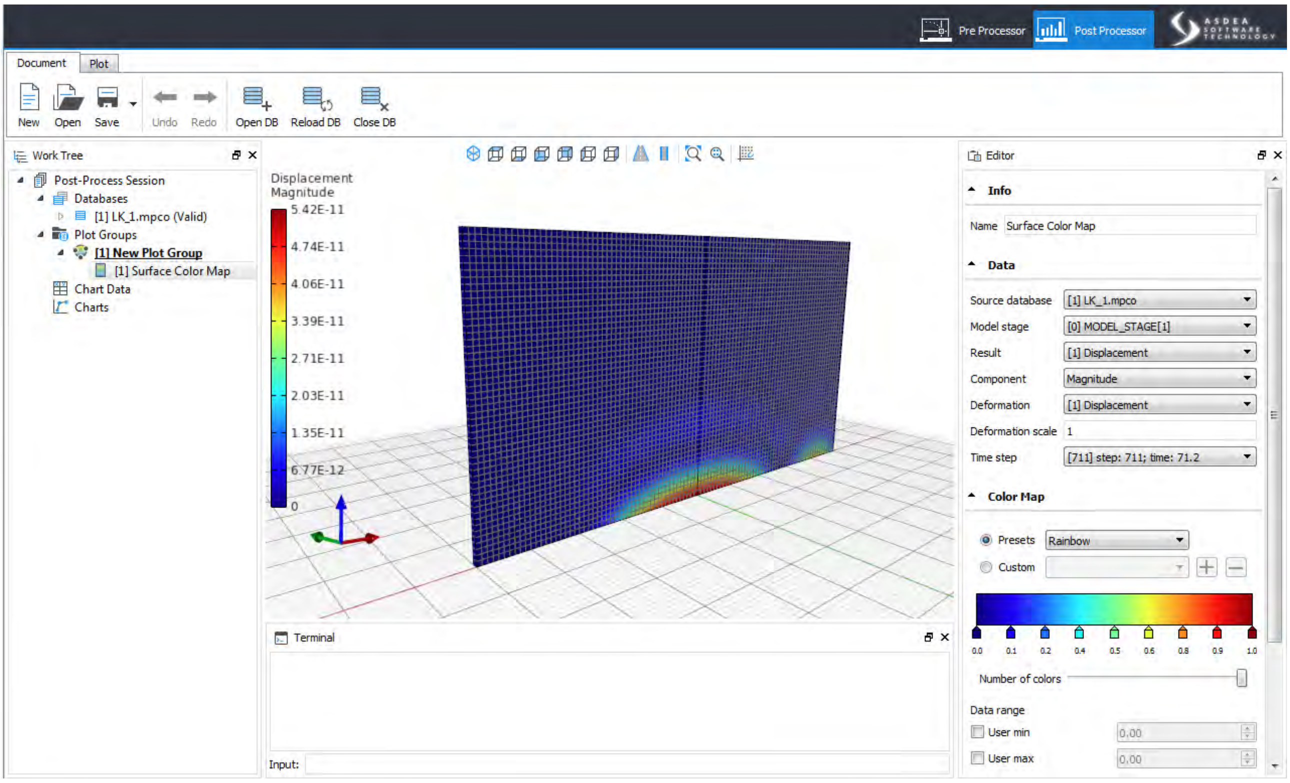


Figure 26. Example of a Surface color map

To **Delete** a **New Plot Group,** or a single **Plot,** *right-click* on the **Plot** (or **Plot Group**) on the Work Tree and select **Delete Plot** (or **Plot Group**) (see § 3-3.2 Postprocessing module).

2. PREPROCESSING MODULE

The preprocessor module creates the Tool Command Language (TCL) input file for OpenSees. From CAD to Mesh

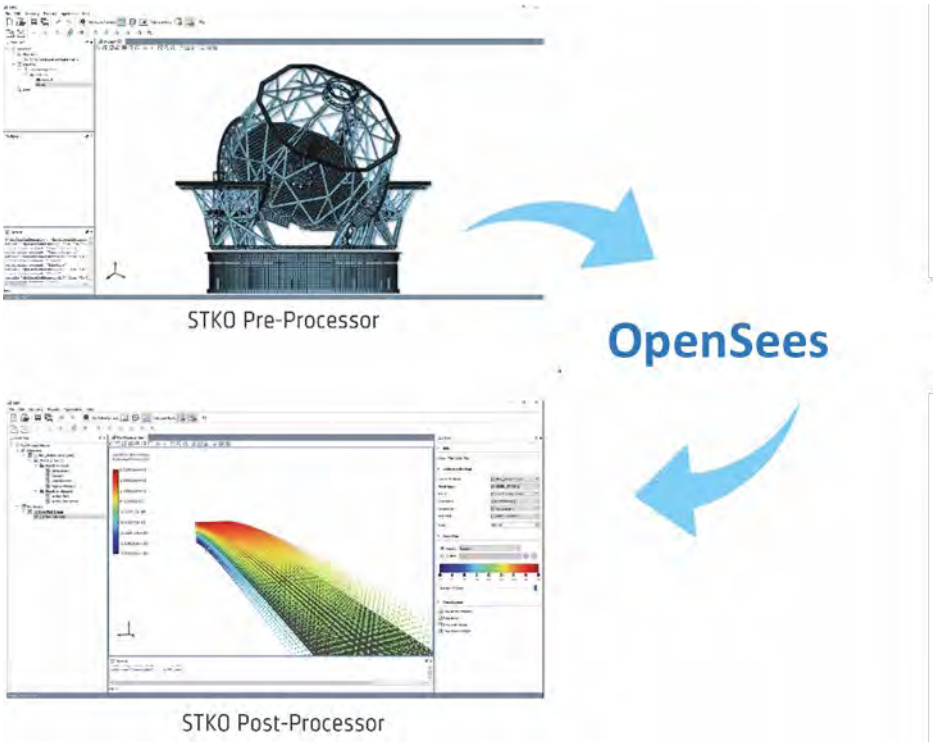


Figure 27. From CAD to mesh

2.1. Geometry

In STKO, the Geometry commands are divided into eight groups: Exchange, Edit, Sets, Points, Curves, Surfaces, Solids, and Boolean. The sections are accessbile in the geometry drop-down menu and on the main Toolbar.

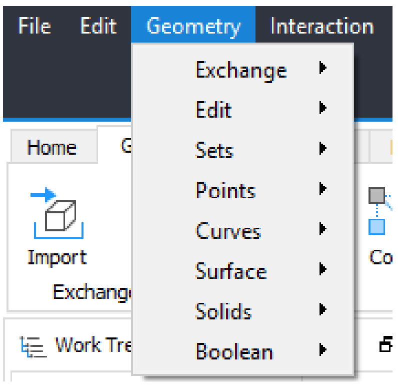


Figure 28. Geometry drop-down menu

Figure 29. Geometry main Toolbar

Viewport and viewport projection

This section explains how to define the design plane. By *clicking* on **view cubes** from the *Quick Access Toolbar*, the user can choose the plane (view) on which they want to operate (above, bottom, front, back, left, or right). Each view can be visualized either in parallel or in perspective projection. In **parallel view**, all gridlines are parallel to each other, and identical objects appear to be the same size, regardless of where they are in space. In **perspective view**, grid lines converge to a vanishing point. This provides the illusion of depth in the viewport. Perspective projection makes objects farther away look smaller. *Click* on **axonometric** cube to have an axonometric projection.

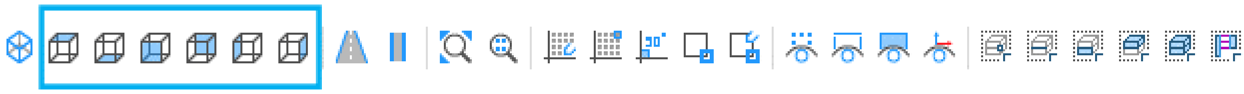


Figure 30. Quick Access Toolbar

Workplane

STKO allows the user to manipulate the construction plane.

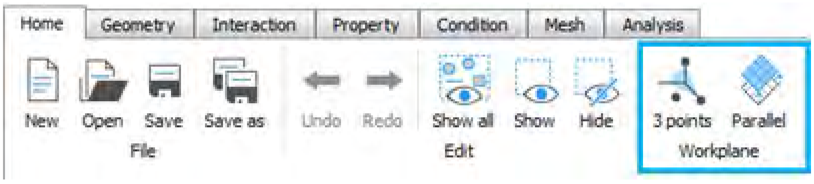


Figure 31. Workplane in the Document main Toolbar

The **Workplane 3 points** command sets the origin and orientation of the construction plane. *Click* **Home** > **3 points** (Workplane).

A new workplane can be created manually by selecting 3 points, or by typing the x, y, and z (optional) coordinates.

Workplane parallel

The **Workplane Parallel** command sets the origin and orientation of the construction plane. *Click* **Home** > **Parallel** (Workplane) to offset the Work Plane.

This command sets the work plane and moves the Cartesian references.
After creating the workplanes, the user can save each created plane, select, and control them directly from the **Work tree** where they are listed.

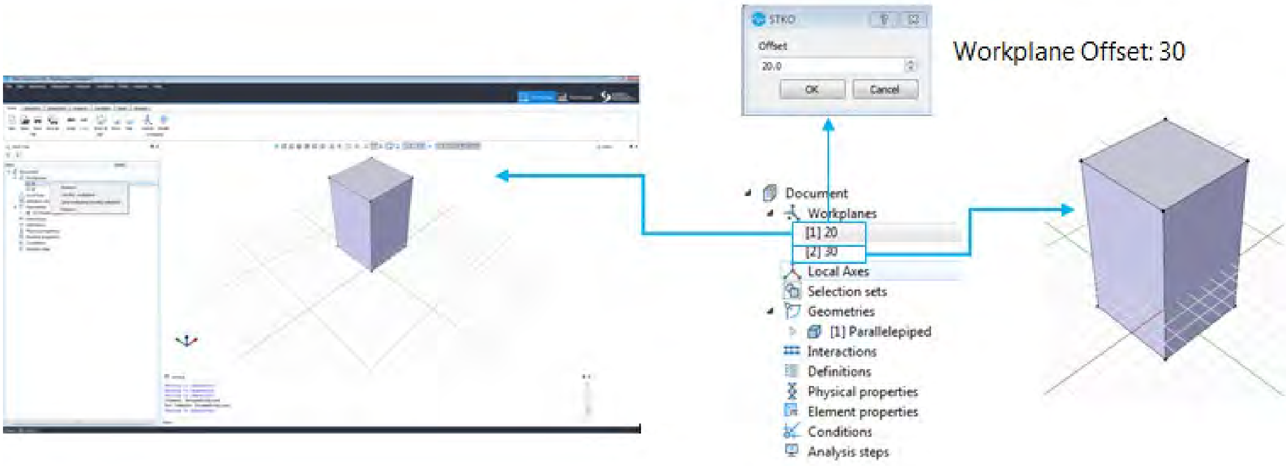


Figure 32. Workplane settings

2.1.1. Geometry – Exchange

The import and export commands present under **Exchange** are a subset of the **Geometry** category. Geometries can be imported from or exported to external files.

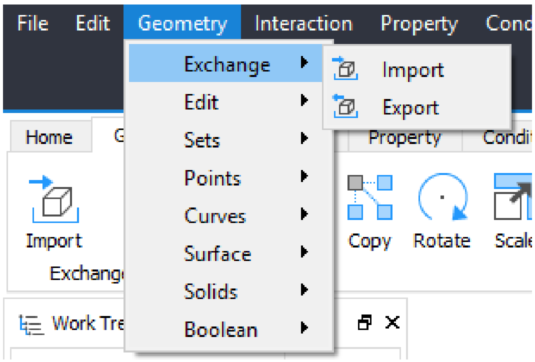


Figure 33. Exchange section in the Geometry drop-down menu

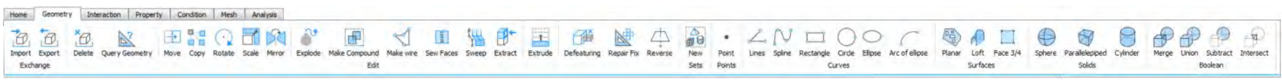
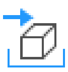


Figure 34. Exchange section in the Geometry main Toolbar



Import

Geometry > Exchange > Import

1. *Click* Import from the Geometry > Exchange menu

2. Input geometries from file

The supported file extensions are **IGES Files** (*.iges, *.igs), **STEP Files** (*.stp *.step), **BREP Files** (*.brep *.rle). A **Regenerate Model** window will appear:

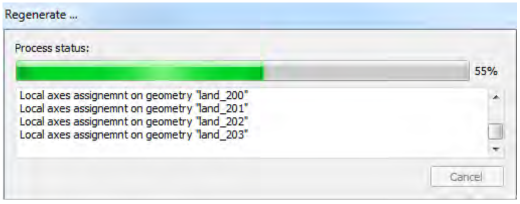


Figure 35. Window of Process status

The terminal bar shows all geometries selected by the user.

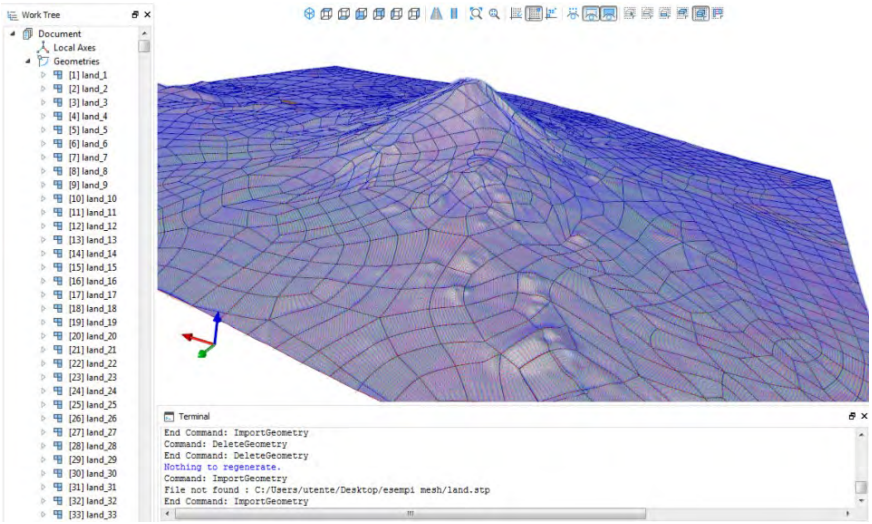


Figure 36. Example of a meshed soil-modeling

Export

Geometry > Exchange > Export

1. *Click* Export from the Geometry > Exchange menu

2. Export geometries to another file

The supported file extensions are **IGES Files** (*.iges, *.igs), **STEP Files** (*.stp *.step), **BREP Files** (*.brep *.rle).

2.1.2. Geometry – Edit

The commands in the **Edit** section are subsets of the **Geometry** category. The **Edit** commands modify the geometries. To execute a command, *Click* the desired command, then *Click* to select the geometry, and *Right-click* to execute the command. Press [*Esc*] on the keyboard or *Right-click* in the render window to abort a command in execution.

Note: All commands can be alternatively made by typing their name into the *Terminal input bar* (which also has an auto compilation feature) and following the instructions given.

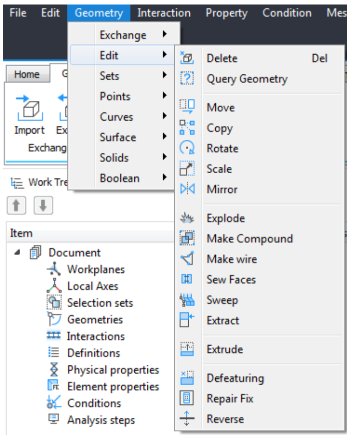


Figure 37. Edit section in the Geometry drop-down menu

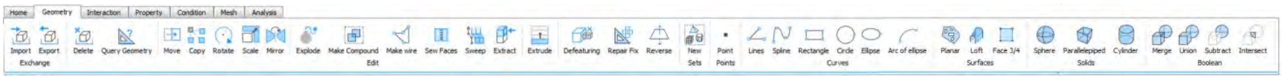


Figure 38. Edit section in the Geometry main Toolbar

Delete

Geometry > Edit > Delete

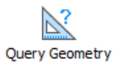
1. *Click* **Delete** from the **Geometry > Edit** menu

2. Select geometry

Click on any part of the geometry you wish to delete to enable the selection; the selected geometries will turn **red**.

3. Execute the command

Press *[Enter]* on the keyboard or *Right-click* to execute the command and delete the geometry.



Query Geometry

Geometry > Edit > Query

1. *Click* Query Geometry from the Geometry > Edit dropdown menu.

An editing window will appear. On the **selection** tab (Figure 38), the user can select the geometries to query by hierarchy, mass property, or location. It is also possible to query **properties** such as length, area, volume, or the face, edge, or solid centers of mass, or coordinates to points.

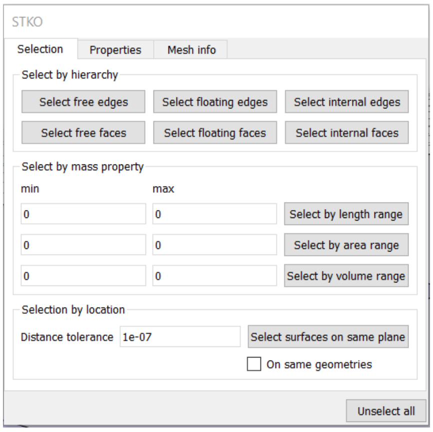


Figure 39. Query properties window_Selection Tab

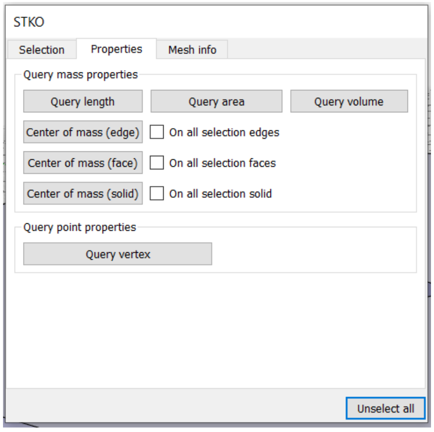


Figure 40.Query properties window_Properties Tab

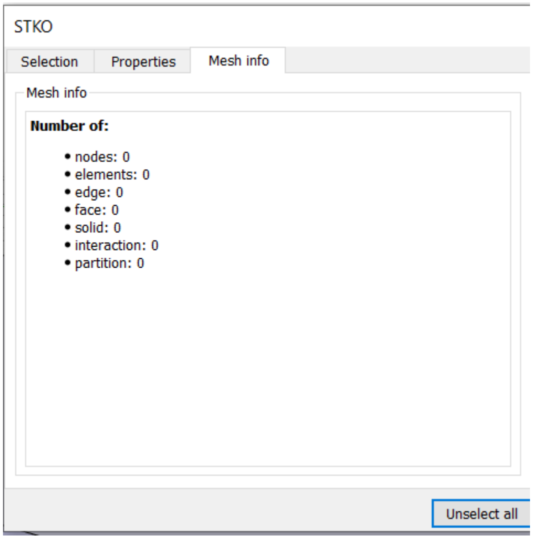


Figure 41. Mesh Info window_Mesh Info Tab

2. Select the geometries

The user can select the geometries using the mouse or the selection window; the selected geometries will turn **red**.

3. Select properties

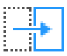
After selecting the geometries, the user can switch to the **properties** tab and see the lengths, areas, volumes, and other information.

4. Mesh Info

If the selected geometries have been meshed, the **Mesh Info tab** allows the user to view all the information contained in the mesh.

5. Terminate the command

Press *[Esc]* on the keyboard to terminate the command and close the Query window.

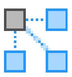


Move

Geometry > Edit > Move

This command moves geometries to another position in space. The action is completed by selecting **two points** (the **first** and the **second points of translation)**. The user can specify, using the mouse or the keyboard, the **first** and **second points** of the **distance**, defining the translating line, to **move** the objects. The *Terminal Input* allows the user to enter the **X, Y, and Z coordinates**, i.e., (100 -100 0), and after the first point, the **distance value**, i.e., (100.5). The Z coordinate is optional, and by default, it is set to 0.

1. *Click* **Move** from the **Geometry > Edit** menu
2. Select the geometries
Click on any point belonging to the geometries, the selected geometries will turn **red.** *Right-click* to enable the selection.
3. Specify the first point
Click anywhere on the work plane, or on a point belonging to the geometry, or **type coordinate X, Y, or Z** to assign a reference point.
4. Specify the second point
Click on the work plane, or **type coordinate X, Y, or Z**, or **type** the **distance value** to define the **move**. The selected geometry will be moved immediately upon the selection of the second point.
5. Abort the command
The command can be aborted by pressing *[Esc]* on the keyboard or *Right-clicking* before the second point has been selected.



Copy

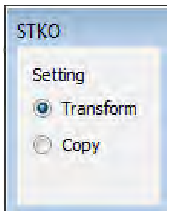
Geometry > Edit > Copy

This command copies geometries to another position in space, preserving their features. The action is completed using **two points** (the **first point** and the **endpoint of translation)**. The user can specify, using the mouse or keyboard, the **first** point (the reference point) and the **second point** (the distance and direction), to define the translating line, to **copy** the objects. The *Terminal Input* allows the user to enter the **X, Y, and Z coordinates**, i.e., (100 -100 0), and after the first point, the **distance value**, i.e., (100.5). The Z coordinate is optional, and by default, it is set to 0. After generating the first copy, the command will remain active to allow for the creation of other copies of the same geometry. The action can be **terminated** by pressing *[Enter]* on the keyboard or by *Right-clicking.*

1. *Click* **Copy** from the **Geometry > Edit** menu
2. Select the geometries
Click on any point belonging to the geometries, the selected geometries will turn **red.** *Right-click* to enable the selection.
3. Specify the first point
Click anywhere on the work plane, or on a point belonging to the geometry, or **type coordinate X, Y, or Z** to assign a reference point.

4. Specify the second point
Click on the work plane, or **type coordinate X, Y, or Z**, or **type** the **distance value** to define the position of the copy.
5. Specify the next point
If you want to generate more copies of the geometry, continue to *Click* on the work plane, or **type coordinate X, Y, or Z**, or **type** the **distance value** to define the position of the copy.
6. Terminate the Command
Press *[Esc]* on the keyboard or *Right-click* to terminate the command**.**

Rotate
Geometry > Edit > Rotate
This command rotates or alternatively generates a rotated copy of the selected geometries. This command works around a point as the center of rotation on a 2D plane and on an angle in relation to the normal axis of the plane. The action is completed using **three points**. The user can specify, using the mouse or keyboard, the **first point** defining the **center of rotation**; **the second point**, as a temporary **reference** line; and the **third point**, as a reference to define the **value** of the **rotation angle**. The textual dialogue interface of the *Terminal* lets the user define the rotation of the geometries. The *Terminal* allows the user to enter the **X, Y, and Z coordinates**, i.e., 100 -100 0, for the three points after selection. The Z coordinate is optional, and by default, it is set to 0. After the insertion of the second point, the *Terminal* will allow the user to enter the **distance value**, i.e., 50, to build the reference line, and the **angle value**, i.e., 90, to define the **rotation angle**.
If the user selects the **Copy** command, after generating the first rotated copy, the command remains active in order to allow for the insertion of more copies. The command can be terminated by pressing the *[Esc]* key or by *Right-clicking*. When the command has been executed, the geometry will remain selected, and the user can decide to delete or leave the object.

1. *Click* **Rotate** from the **Geometry > Edit** menu


After *click*ing Rotate, a setting window will appear. The user can either choose to **Transform** the selected object or to create a transformed **Copy** of it.

2. Select the geometries

Click on any point belonging to the geometries, the selected geometries will turn **red.** *Right-click* to enable the selection.

3. Specify the first point, the center of rotation

Click anywhere on the work plane, or on a point belonging to the geometry, or **type coordinate X, Y, or Z** to assign a reference point for the center of rotation.

4. Specify the second point, the temporary reference line

Click on the work plane, or **type coordinate X, Y, or Z,** or **type** the **distance value** to define the temporary reference line.

5. Specify the third point, the **rotation** angle (degrees), or the reference point

Click on the work plane, or **type coordinate X, Y, or Z,** or **type** the **angle value** to define the rotation angle. If the user selected Transform in step one, the selected geometry will rotate to the new position, and the command will terminate. Instead, if the user selected **Copy**, the original geometry will remain in its position and a copy will appear in the new rotated position; the original geometry will remain selected.

6. Specify the next point (Copy only)

If the user selected **Copy** in step one, the geometry will remain selected after step 5. The user may continue to generate more rotated copies of the geometry, by *Clicking* on the work plane, or by **type coordinate X, Y, or Z,** or **type** the **incremental angle value** to define the position of the rotated copy.

7. Terminate the command (Copy only)

The command can be terminated by pressing *[Enter]* or by *Right-clicking*.

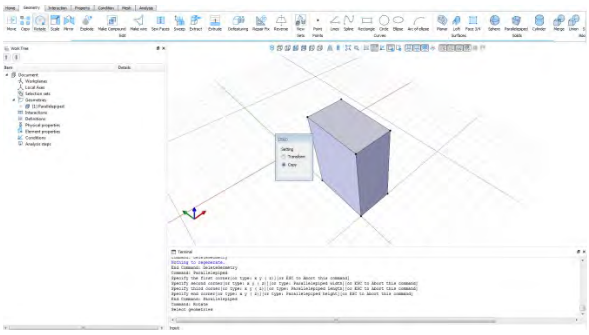


Figure 42. Rotating operation_part one

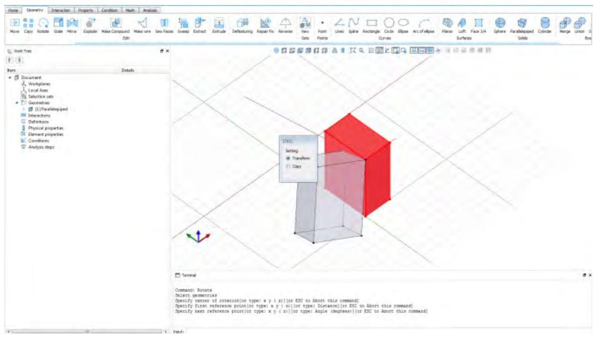


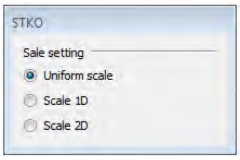
Figure 43. Rotating operation_part two

Scale

Geometry > Edit > Scale

This command changes the geometry's size in relation to one, two, or three dimensions. To activate the action, choose the command from the *Toolbar* or *Menu* and select the geometries. After activation, a menu will appear to allow the user to choose the type of scale **(1D, 2D, Uniform scale** [3D]**)**. The **Scale 1D** command changes the size of selected objects in one direction, **Scale2D** changes the size in two directions, and Uniform scald [3D] changes the size of selected objects uniformly in the x, y, and z directions.

1. *Click* **Scale** from the **Geometry > Edit** menu



2. Select the type of scale
After enabling the command, an interactive menu will appear with the scale parameters. Select the desired type of scale: **1D, 2D, or Uniform scale** (3D).
3. Select the geometries
Click any point on the geometries, the selected geometries will turn **red.** *Right-click* to confirm the selection**.**
4. Specify the reference base point
Click anywhere on the work plane, or on a point belonging to the geometry, or **type coordinate X, Y, or Z** to assign a reference base point for the scale.
5. Specify the scale factor
Type the value of the scale factor to create a new scaled object and press *[Enter]* to execute the command. If the user prefers to use on-screen graphics, this step may be skipped, and the user may proceed with the next steps. If instead, the user completes this step [5] they may ignore the following steps.
6. Specify the first point of reference, the unitary measure
Click on the work plane, or **type coordinate X, Y, or Z** to define the reference measure.
7. Specify the end point of reference, the value scale factor
Click on the work plane, or **type coordinate X, Y, or Z** to define the value scale factor. The command will execute upon selection of the third point.

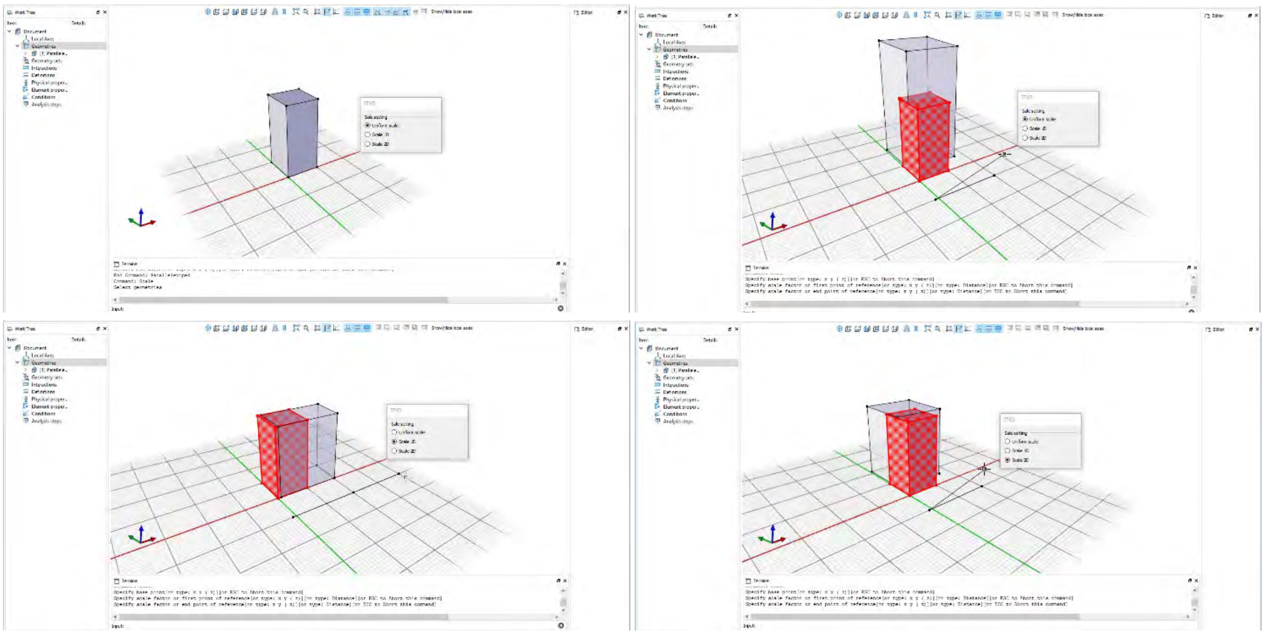
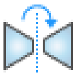


Figure 44. Example of the Scale command

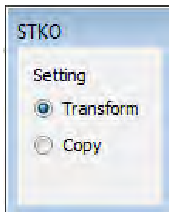


Mirror

Geometry > Edit > Mirror

This command mirrors or alternatively generates a specular copy of the selected geometry. The transformation obtained is an axial symmetry of the selected geometry. To initiate this action, choose the command from the *Toolbar* or *Menu* and select the geometries. The user can specify, using their mouse or keyboard, the **first** and **second points,** which will define the line that will become the **axis of symmetry** for **copying** and **mirroring** the objects. Users may use the *Terminal* to enter the **X, Y, and Z coordinates**, i.e., 100 -100 0, and after the first point has been set, the **distance value**, i.e., 100.5, to define the temporary line to be used as the axis of symmetry. The Z coordinate is optional, and by default, it is set to 0. After execution, if the user selected **Transform** the original geometry will be deleted. If the user selected **Copy**, the original geometry will remain on the work plane.

1. *Click* **Mirror** from the **Geometry > Edit** menu



After *click*ing **Mirror,** the user must choose between **Transforming** the selected object or creating a transformed **Copy** of it.
2. Select the geometry
Click on any point belonging to the geometry; the selected geometry will turn **red.** *Right-click* to enable the selection**.**
3. Specify the first point
Click anywhere on the work plane, or on a point belonging to the geometry, or **type coordinate X, Y, or Z** to assign the first reference point.
4. Specify the second point
Click on the work plane, or **type coordinate X, Y, or Z**, or **type** the **distance value** to define the line to be used as the **axis of symmetry**. If the user selected Transform in step 1, the transformed copy will appear in its new position, and the original will disappear. Instead, if the user selected Copy, the copy will appear in the position defined and the original will remain.

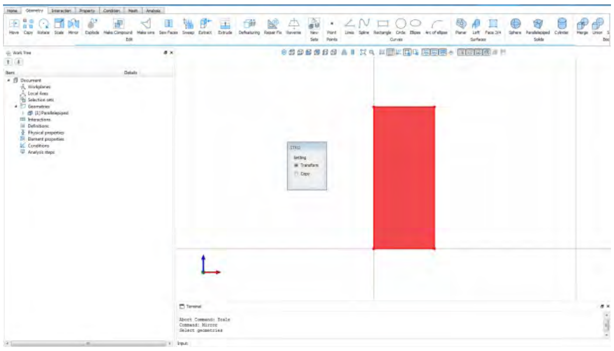


Figure 45. Mirror command activation

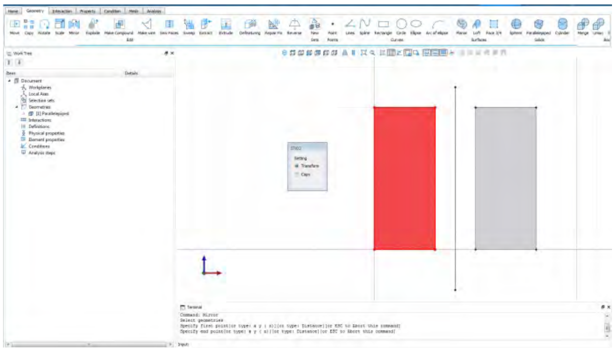
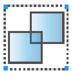


Figure 46.Mirror command in action

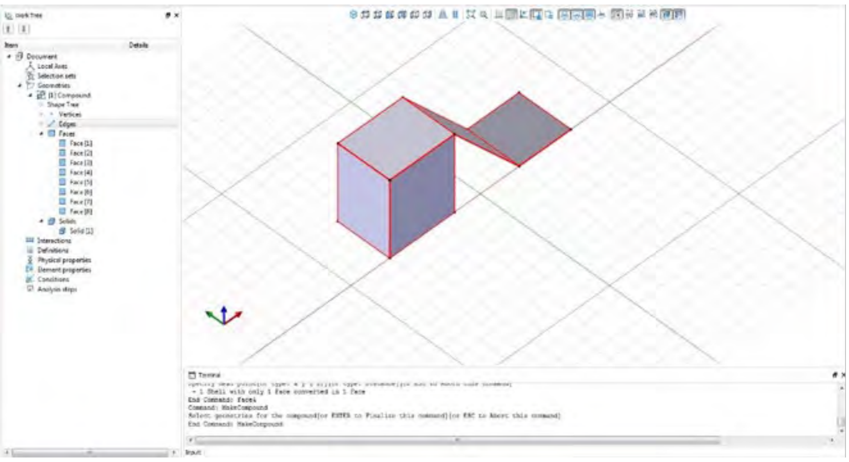
Explode
Geometry > Edit > Explode
This command identifies the subclasses of the object by splitting it into its components. A complex object can be divided into simpler parts by applying this command. For example, a parallelepiped can be split into its basic foundational geometries such as vertices, edges, and faces. To initiate the action, choose the command from the *Toolbar* or *Menu*, and select the desired geometries.
The user can **select** one or more **geometries** with their mouse and *Right-click* to execute the **explode** command.

1. *Click* Explode from the Geometry > Edit menu
2. Select the geometries
Click on any point belonging to the geometries to enable the selection; the selected geometries will turn **red.**
3. Explode
Press the *[Enter]* key or *Right-click* to confirm the command**.** After confirming, the geometry will be **exploded**.
4. Finish
The subdivided geometry will remain editable. Press the [***Esc***] key to preserve the geometries in the work plane, or select, using the mouse, the geometries on which to apply new actions. Repeat the flow to obtain more subdivisions.



Make Compound
Geometry > Edit > Make Compound
Compound – a group of any type of topological objects.
This command creates a compound of the selected geometries. Making a compound of curves, surfaces, and/or solids creates a "composite" geometry made up of different items that preserve their own geometrical features. In other words, this command allows the joining of different geometries. To initiate the action, choose the command from the *Toolbar* or *Menu* and select the geometries.
The user can **select** the **geometries** on which **apply** the **compound** function and execute the command by *Right-clicking*.

1. *Click* Make Compound from the Geometry > Edit menu
2. Select the geometries
Click on any point belonging to the geometries; the selected geometries will turn **red.**
3. Execute the command
Once the desired number of geometries are selected, press *[Enter]* on the keyboard or *Right-click* to execute the command. The geometries will be compounded.



Make Compound:

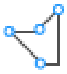
13 vertex

24 edges

8 faces

1 solid

Figure 47.Example of Compound



Make Wire

Geometry > Edit > Make Wire

This command creates a unique geometrical entity from two or more curves, i.e., a polyline. The procedure for building the wire from the selected edges is intuitive. To initiate the action, choose the command from the *Toolbar* or *Menu* and select the geometries.

The user can **select** the **geometries** on which they would like to **apply** the **Make Wire** function and execute the command by *Right-clicking*.

1. *Click* Make Wire from the Geometry > Edit menu

2. Select the geometries

Click on any point belonging to the geometries; the selected geometries will turn **red.**

3. Execute the command

Once the desired number of geometries are selected, press *[Enter]* on the keyboard or *Right-click* to execute the command. A polyline will be create from the selected geometries.

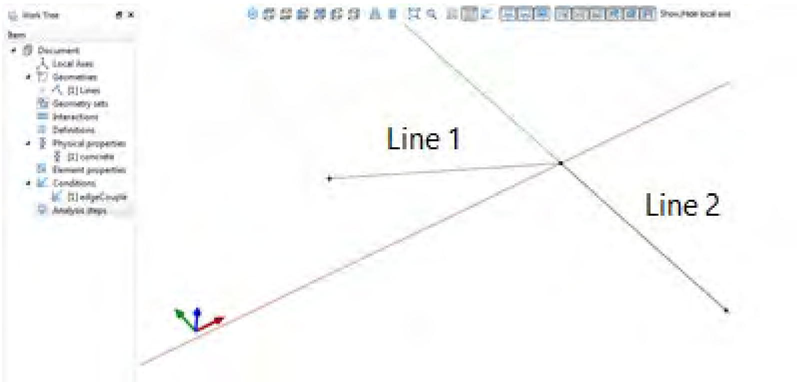


Figure 48. Before making wire- 2 lines

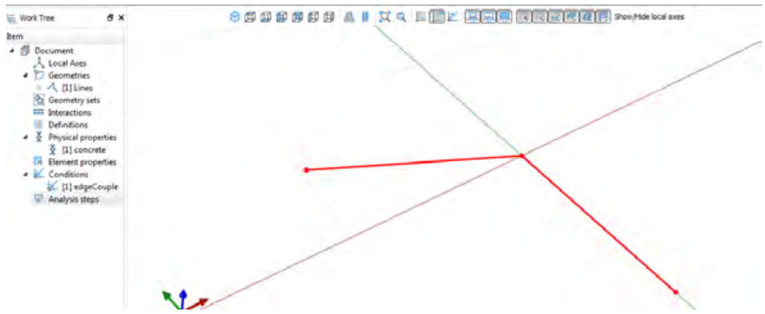
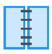


Figure 49. After making wire -1 polyline



Sew Faces

Geometry > Edit > Sew Faces

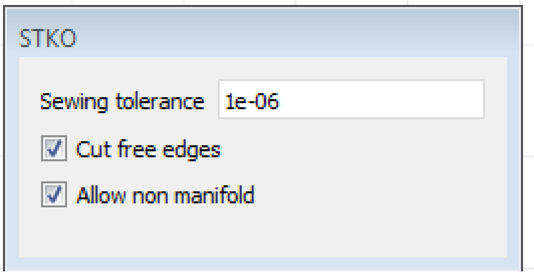
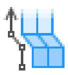


Figure 50. Sew Faces

The **Sew Faces** command creates a unique face from two or more faces of the starting objects. To initiate the action, choose the command from the *Toolbar* or *Menu* and select the geometries. After initiation, an interactive menu will open for selecting the parameters. The user can set the value of the **sewing tolerance** (by default set to 1e-06), enable the cutting of free edges, and enable the generation of non manifold elements.

After selecting the parameters, the user can **select** the **geometries** on which to **apply** the **Sew Face** function and execute the command by *Right-clicking*.

1. *Click* Sew Faces from the Geometry > Edit menu
2. Select the sewing parameters
After initiating the command, an interactive menu will open on the screen with the **parameters**. Define the ***sewing tolerance*** (by default the value is set to 1e-06). Select whether to enable ***cut free edges*** or ***allow no manifold***.
3. Select the geometries
Click on any point belonging to the geometries; the selected geometries will turn **red.**
4. Execute the command
Once the desired number of geometries are selected, press *[Enter]* on the keyboard or *Right-click* to execute the command. The selected geometries will be **sewn**.



Sweep

Geometry > Edit > Sweep

This command creates a geometry starting from a profile curve and moving along a rail to create an object. The starting curve, the **profile shape,** defines the surface cross-section, and the rail, the **spine shape**, defines the trajectory of the sweep. In this way, different categories of geometries can be obtained. In reference to a generic line track, vertices generate edges, edges generate faces, a face generates a solid, wire generates a shell, and a shell generates a compound solid. Solids and compound solids can not be extruded. Choose the command from the *Toolbar* or *Menu* and select the geometries desired to initiate the action. The *Terminal* allows the user to specify how to sweep the geometries using its interactive text interface. The user can first **select** the **geometry** class of interest to be used as the **profile shape to sweep,** and then the **reference** geometry, for the **spine shape**.

1. *Click* **Sweep** from the **Geometry > Edit** menu
2. Select the first geometry, the profile shape to sweep
Click on any point belonging to the geometry or on a subclass (vertex, edge, face, wire, or shell) to sweep; the selected geometry will turn **red**. *Right-click* to confirm the selection.
3. Select the second geometry, the spine shape
Click on any point belonging to the geometry; the selected geometry will turn **red.**
4. Execute the command

Press *[Enter]* on the keyboard or *Right-click* to execute the command. The command will generate the new geometry.

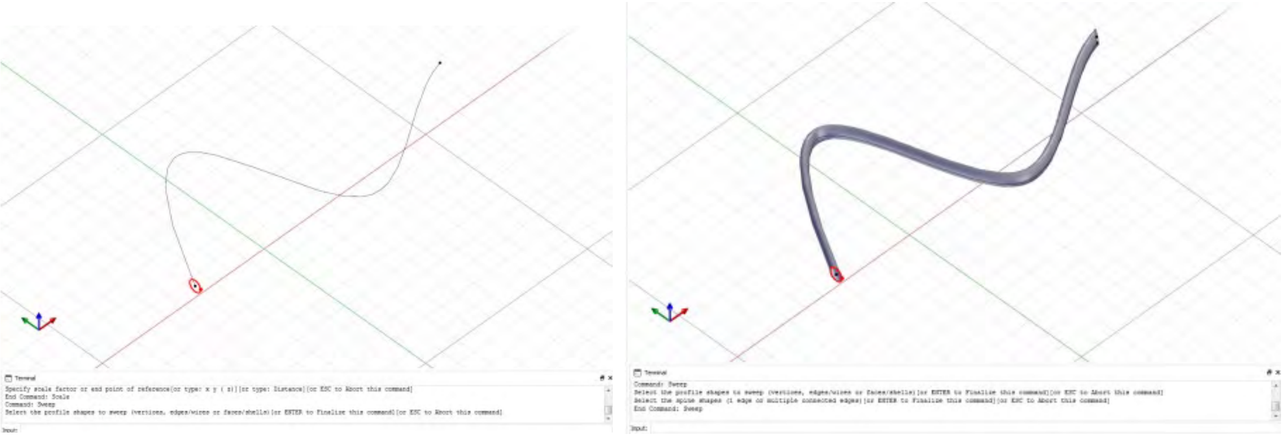
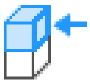


Figure 51.Example of sweep.



Extract

Geometry > Edit > Extract

The **Extract** command allows for the extraction of simple geometries from complex geometries.

1. *Click* Extract from the Geometry > Edit menu

2. Select the geometry to extract from

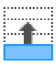
Click on any point belonging to the geometry desired to extract from. After *click*ing on the selection, the geometry will turn **red**.

3. Select the geometry to extract

Click on any point belonging to the geometry or on a subclass (vertex, edge, face, wire, or shell) to extract. After *click*ing on the selection, the geometry will turn **red**.

4. Execute the command

Press *[Enter]* on the keyboard or *Right-click* to execute the command. The simple geometry will be extracted from the complex geometry.



Extrude

Geometry > Edit > Extrude

1. *Click* Extrude from the Geometry > Edit menu

2. Select the geometry, the profile shape to extrude

Click on any point belonging to the geometry or on a subclass (vertex, edge, or face) to extrude. After *clicking* the selection, the geometry will turn **red**. *Right-click* to enable the function.

3. Define the height of the extrusion

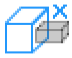
Click on the work plane, or **type coordinate X, Y, or Z,** or **type** the **distance value** to define the height of the extrusion of the geometry.

4. Define other directions (Optional)

After *Clicking* to **define the height** of the **extrusion,** the geometry (the profile shape to extrude) will remain selected. The user can continue extruding the geometry in different directions or heights. *Click* on the work plane, or **type coordinate X, Y, or Z,** or **type** the **distance value** to continue defining new heights for the extrusion of the geometry as desired.

5. Execute the command

Press *[Enter]* on the keyboard or *Right-click* to execute the command. The geometry will be extruded.



Defeaturing

This command involves the removal of one or more faces from a surface and then the consequent repair of that surface. The removed part is replaced by an approximation of it or an approximation of the faces connected to it. Some possible examples could be the replacement of a beveled angle with a sharp angle or the removal of a hole with consequent modification of the face in order to plug the hole.

Geometry > Edit > Defeaturing

1. *Click* Defeaturing from the Geometry > Edit menu

2. Select the geometry to defeature

Click on any point belonging to the geometry or on a subclass (vertex, edge, or face) to defeature. After *clicking* on the selection, the geometry will turn red.

3. Execute the command

Right-click to execute the command. The geometry will be defeatured as seen in the example below.

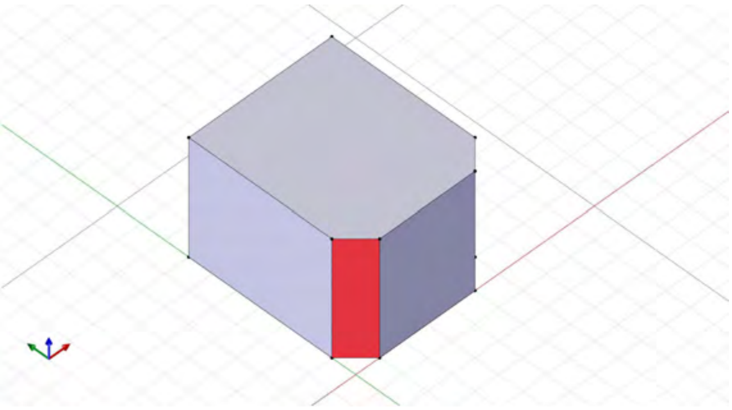


Figure 52.Geometry before defeaturing

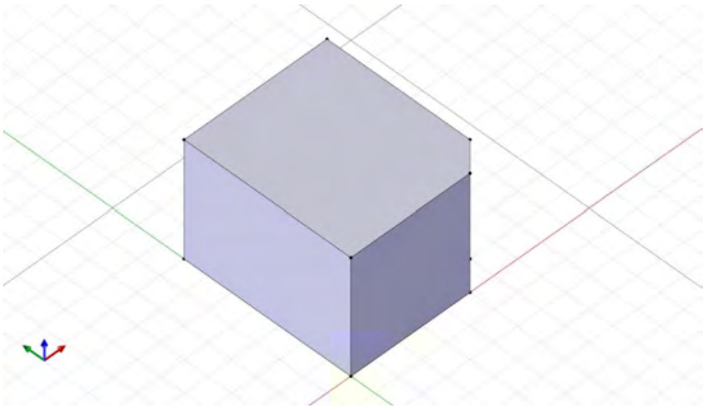
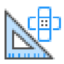


Figure 53.Geometry after defeaturing



RepairFix

This command fixes model geometry problems like a face with a missing seam edge, incorrect orientation of an entity, a self-intersecting wire, or a lacking edge. RepairFix adapts shapes to make them appropriate for use in STKO. The user can set the value of fix tolerance (by default set to 1e-06) and which elements to fix, such as faces, solids, etc.

Geometry > Edit > RepairFix

1. *Click* RepairFix from the Geometry > Edit menu
2. Setting Repair Fix

A *settings window* will open that allows the user to set the **tolerance** (by default set to 1e-06) and select the different elements to **fix**; i.e. Fix global, Fix faces, Fix small faces, Fix small edges, etc.

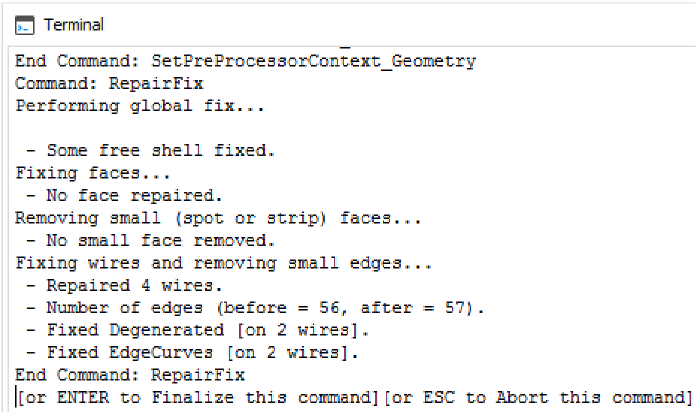
3. Select the geometries to RepairFix

Click on any point belonging to the geometry the user wishes to **RepairFix**; the selected geometry will turn **red**.

4. Execute the command

Press *[Enter]* on the keyboard or *Right-click* to execute the command.

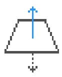
The terminal bar will show all repaired geometries:


Terminal

```
End Command: SetPreProcessorContext_Geometry
Command: RepairFix
Performing global fix...

 - Some free shell fixed.
Fixing faces...
 - No face repaired.
Removing small (spot or strip) faces...
 - No small face removed.
Fixing wires and removing small edges...
 - Repaired 4 wires.
 - Number of edges (before = 56, after = 57).
 - Fixed Degenerated [on 2 wires].
 - Fixed EdgeCurves [on 2 wires].
End Command: RepairFix
[or ENTER to Finalize this command][or ESC to Abort this command]
```

Figure 54. Terminal bar after RepairFix command



Reverse

This command displays the normal-direction of a geometry or surface and allows the user to invert it. The user can see the geometry's normal-direction arrows by activating the command Show/Hide local axis on the Quick Access Toolbar. Surface normals are represented by arrows perpendicular to the surface, and the u- and v-directions are indicated by arrows pointing along the surface. Closed surfaces always have surface normals pointing to the exterior. Obviously, this operation is not available for solids.

Geometry > Edit > Reverse

1. *Click* Reverse from the Geometry > Edit menu
2. Select the geometries to reverse

Click on any point belonging to the geometries or on a subclass (vertex, edge, or face) to **reverse**. After *clicking* the selection, the geometry will turn **red**.
3. Execute the command
Press *[Enter]* on the keyboard or *Right-click* to execute the command.

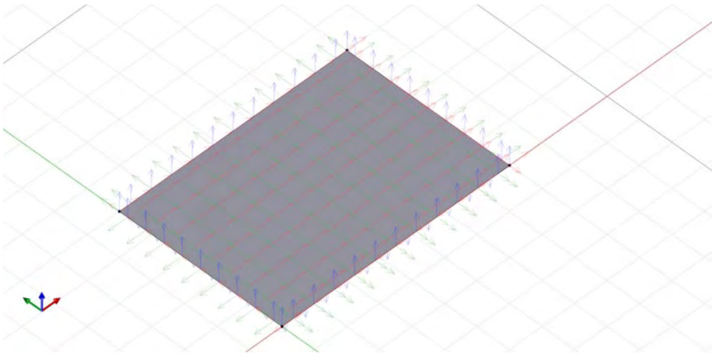


Figure 55. Surface with z-axis down

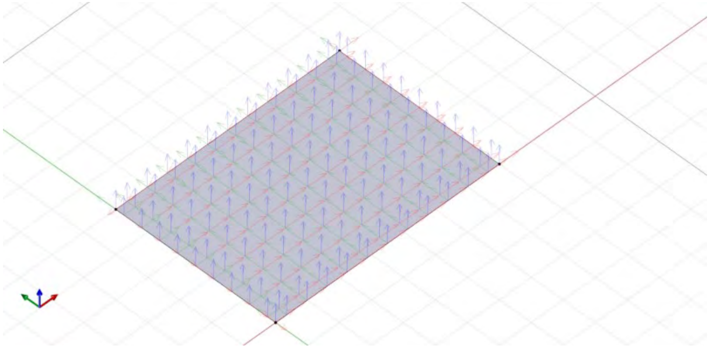


Figure 56. Surface with z-axis reversed up

2.1.3. Geometry – Sets

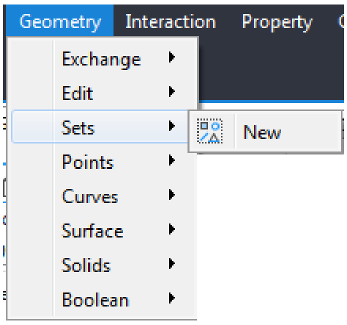


Figure 57. Set section from the Geometry drop-down menu

Geometry Sets

Geometry > Sets > New

1. *Click* **New** from the **Geometry > Sets** menu

2. Select the geometry to add to the selection set

Click on any point belonging to the geometry or on a subclass (vertex, edge, or face) to add to the selection. After *click*ing on the selection, the geometry will turn **red.** Press *[Enter]* on the keyboard or *Right-click* to confirm and name the selection set.

3. Add/remove geometry from the selection set

Right-click on the selection set in the Work Tree > **Add to selection set** > *click* on the geometries to add. Press *[Enter]* on the keyboard or *Right-click* to confirm the selection.

Right-click on the selection set in the Work Tree > **Remove from selection set** > *click* on the geometries to remove. > Press *[Enter]* on the keyboard> Confirm the selection set to remove the geometry from. Press **Ok**

2.1.1. Geometry – Points

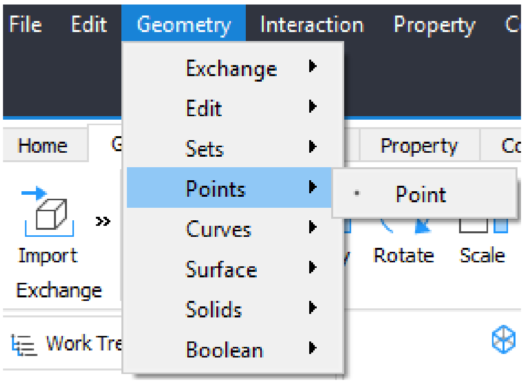


Figure 58. Point section in the Geometry drop-down menu

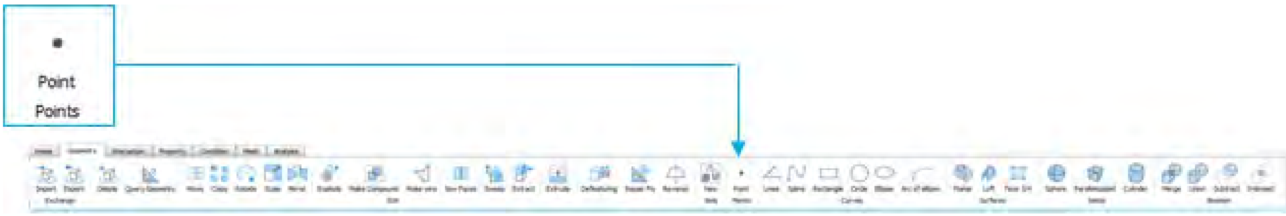


Figure 59. Point section in the Geometry main Toolbar

Point

Geometry > Points > Point

1. *Click* Point from the Geometry > Points menu

2. Execute the Command and assign the point

Click anywhere on the work plane or **type coordinate X, Y, or Z** to assign the point.

2.1.2. Geometry – Curves

The commands present in the **Curves** section are subsets of the **Geometry** category. Every command allows the generation of a specific geometric shape on the plane. To generate the desired shape, select the relative command from the Toolbar, and follow the indications shown in the textual dialogue interface of the Terminal for the insertion of the construction steps.

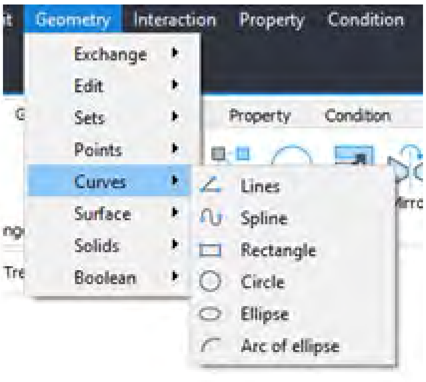


Figure 60. Curve section in the Geometry drop-down menu

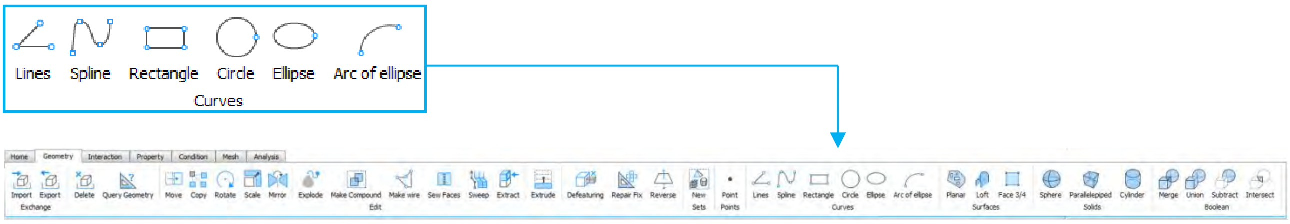
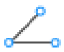


Figure 61. Curve section in the Geometry main Toolbar



Lines

Geometry > Curves > Lines

1. *Click* Lines from the Geometry > Curves menu

2. Specify the first point

Click anywhere on the work plane or **type coordinate X, Y, or Z** to assign the first point of the line.

3. Specify the second point

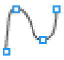
Click anywhere on the work plane or **type coordinate X, Y, or Z**, or enter the **scalar value** of the dimension of the segment to assign the second point of the line.

4. Specify the next point

If you wish to generate a polyline, continue to *Click* on the work plane, or **type coordinate X, Y, or Z,** or **type** the **scalar value** of the segment to define the position of the point.

5. Execute the command

Once the desired number of points **(Vertices)** and lines **(Edges)** of the geometry have been entered, press *[Enter]* or *Right-click* to execute the command.



Spline

Geometry > Curves > Spline

This command creates a spline in the space. To initiate the action, choose the command from the *Toolbar* or *Menu* and select the geometries. The user can set the construction of the spline **through control points** external to the line or **through points** internal to the line, and specify the value of **degree** within the range of 1-100 (by default set to 3) as an interpolating cubic spline. The textual dialogue interface of the *Terminal* defines the procedure to generate the geometries. After selection, the *Terminal* will open and allow the user to enter the **X, Y, and Z coordinates**, i.e., 100 -100 0, for the construction points, after which the user will be able to set the **distance value**, i.e., 100.5. The Z coordinate is optional, and by default, it is set to 0.

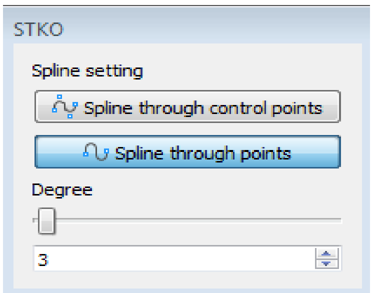
1. *Click* Spline from the Geometry > Curves menu

2. Select the construction criteria

After initiating the command, an interactive menu will appear on the screen with the construction criteria. Select the desired method:

-Select ***spline through control points*** to define the geometry built through points external to the line.

-Select ***spline through points*** to define the geometry built through internal points belonging to the line, and the value of **degree** (by default set 3) as an interpolating cubic spline.



3. Specify the construction points

Click on the work plane, or **type coordinate X, Y, or Z,** or **type** the **distance value** to define the construct points with which to build the spline.

4. Execute the command

Press *[Enter]* on the keyboard or *Right-click* to execute the command.

Rectangle

Geometry > Curves > Rectangle

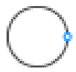
1. *Click* Rectangle from the Geometry > Curves menu

2. Specify the first point or first corner

Click anywhere on the work plane or **type coordinate X, Y, or Z** to assign the first point of rectangle.

3. Specify the second point or second corner

Click anywhere on the work plane, or **type coordinate X, Y, or Z**, or enter the **scalar value** of the dimension of the segment for the first and second side of the rectangle. The rectangle will automatically generate after the selection of the second point.



Circle

Geometry > Curves > Circle

1. *Click* Circle from the Geometry > Curves menu

2. Specify the first point, the center of the circle

Click anywhere on the work plane or **type coordinate X, Y, or Z** to assign the first point as center of the circle.

3. Specify the second point, the radius of the circle

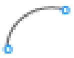
Click anywhere on the work plane, or **type coordinate X, Y, or Z**, or enter the **scalar value** of the dimension of the segment to assign the **radius** of the circle. The circle will automatically generate after the selection of the second point.

Ellipse

Geometry > Curves > Ellipse

1. *Click* Ellipse from the Geometry > Curves menu

2. Specify the first point, the center of the axes of the ellipse
Click anywhere on the work plane or **type coordinate X, Y, or Z** to assign the first point as center of the conic section identified by the intersection of the major and minor axis of the ellipse.
3. Specify the second point, the first semiaxis of ellipse
Click anywhere on the work plane, or **type coordinate X, Y, or Z**, or enter the **scalar value** of the dimension of the segment to define the first semiaxis, major or minor, of the ellipse.
4. Specify the third point, the second semiaxis of ellipse
Click anywhere on the work plane, or **type coordinate X, Y, or Z**, or enter the **scalar value** of the dimension of the segment to define the second semiaxis of the ellipse. The ellipse will automatically generate after the selection of the third point.



Arc of Ellipse
Geometry > Curves > Arc of Ellipse
1. *Click* Arc of Ellipse from the Geometry > Curves menu.
2. Specify the first point, the center of the ellipse axes
Click anywhere on the work plane or **type coordinate X, Y, or Z** to assign the first point as the center of the conic section identified by the intersection of the major and minor axis of the ellipse.
3. Specify the second point, the first semiaxis of ellipse
Click anywhere on the work plane, or **type coordinate X, Y, or Z**, or type the first ellipse axis radius value to define the first semiaxis, major or minor, of the ellipse.
4. Specify the third point, the second semiaxis of ellipse
Click anywhere on the work plane, or **type coordinate X, Y, or Z** type the second ellipse axis radius value to define the second semiaxis of the ellipse.
5. Specify the fourth point, the arc of ellipse
Click anywhere on the work plane, or **type coordinate X, Y, or Z**, or enter angle value. The arc of the ellipse will automatically generate after the selection of the fourth point.

2.1.6 Geometry – Surfaces

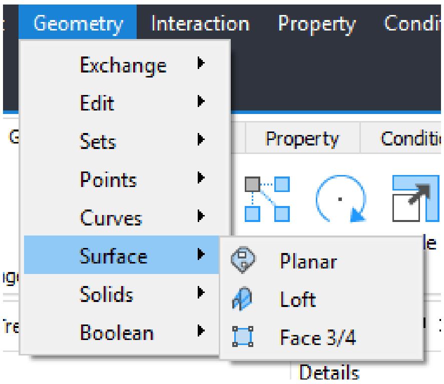


Figure 62. Surface section in the Geometry drop-down menu

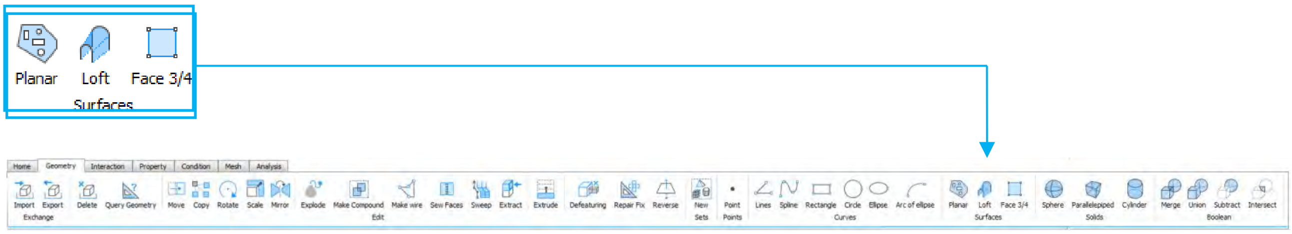


Figure 63. Surface section in the Geometry drop-down menu

Planar

Geometry > Surfaces > Planar

1. *Click* Planar from the Geometry > Surfaces menu

2. Select the geometries

Click on any point belonging to the geometries; the selected geometries will turn **red.**

3 Execute the command

Press *[Enter]* on the keyboard or *Right-click* to execute the command. The closed geometry will be transformed into a **surface.**

Loft

Geometry > Surfaces > Loft

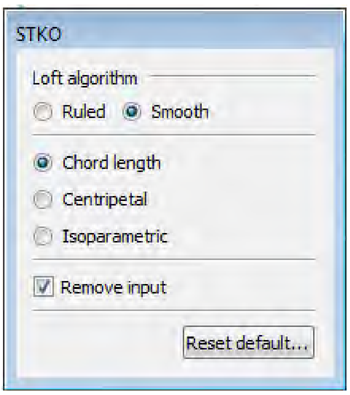


Figure 64. Loft Window

1. *Click* Loft from the Geometry > Surfaces menu
2. Choose the construction parameters

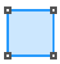
After enabling the command, an interactive menu will appear to allow the user to select the construction criteria. Select the desired type of algorithm. **Ruled** generates a surface with a linear profile. **Smooth** generates a surface with a curvilinear profile through the selection of either **Chord Length**, **Centripetal**, **or Isoparametric** methods. Selecting the **Remove Input** parameter removes the input construction geometries. *Click*ing **Reset Default** restores the default conditions.

3. Select the geometries

Click on any point belonging to the geometries; the selected geometries will turn **red.**

4. Execute the command

Press *[Enter]* on the keyboard or *Right-click* to execute the command.



Face 3/4

Geometry > Surfaces > Face 3/4

1. *Click* Face 3/4 from the Geometry > Surfaces menu
2. Specify the first point

Click anywhere on the work plane or **type coordinate X, Y, or Z** or to assign the first point.

3. Specify the second point

Click anywhere on the work plane or **type coordinate X, Y, or Z,** or enter the **scalar value** to assign the second point to define the first side of the polygon.

4. Specify the third point

Click anywhere on the work plane or **type coordinate X, Y, or Z** to assign the third point to define the second side of the polygon.

5. Specify the fourth point

Click anywhere on the work plane or **type coordinate X, Y, or Z** to assign the fourth point to define the third side of the polygon. The polygon will be closed automatically and a face will be generated from the figure.

Surface Modeling Examples

To create a surface from curves, choose the command **Geometry** > **Loft** from the Toolbar. Specify the **Loft Algorithm**: **Ruled** for broken surfaces or **Smoothed** (with three choices: *chord length, centripetal, isoparametric*). Select **Ruled** on the interactive menu. **Select** the first Wire/Vertex and then all other entities (wire, edge, or vertex). Press **Enter** to execute the command.

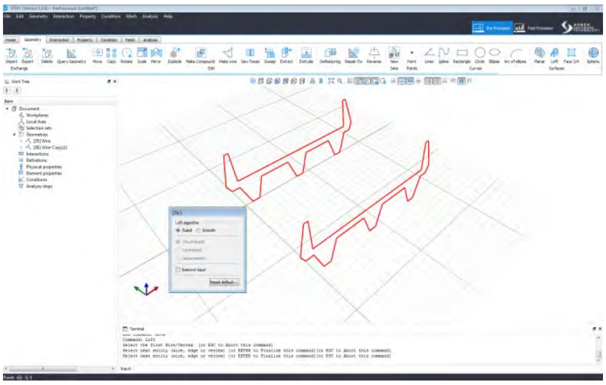


Figure 65. Start from two polylines as cross-sections.

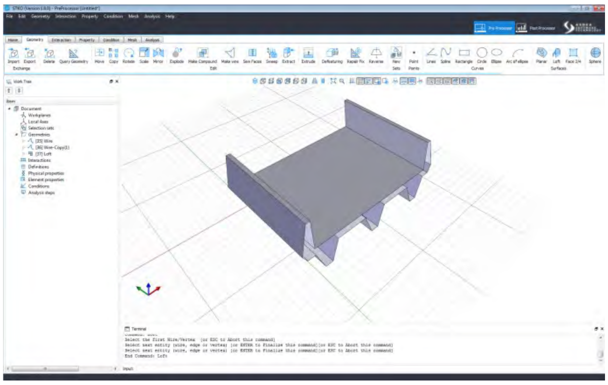


Figure 66.Loft between the cross-sections.

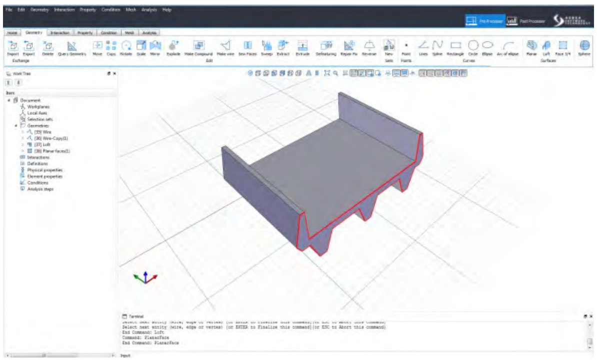


Figure 67. Then close with a Planar surface.

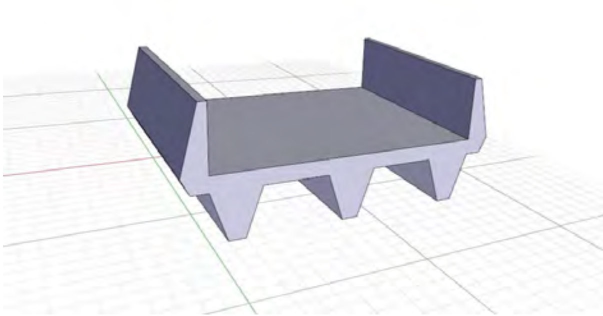


Figure 68. Example of a completed loft algorithm.

2.1.7. Geometry – Solids

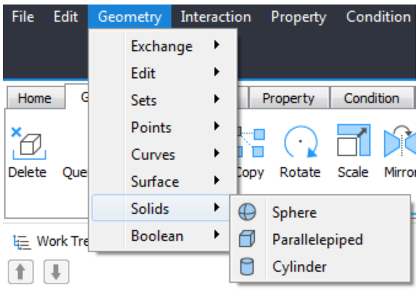


Figure 69. Solids section in the Geometry drop-down menu

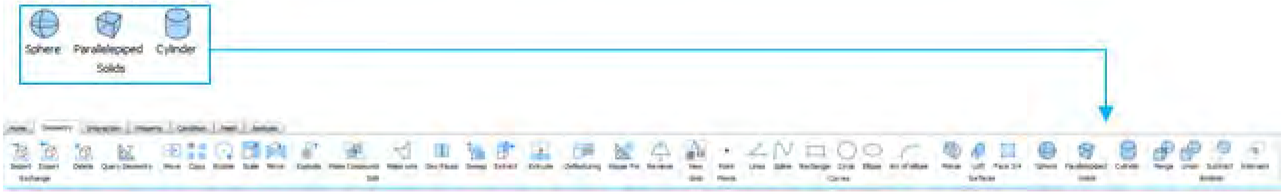
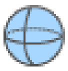


Figure 70. Solids Section on the Geometry main Toolbar



Sphere

Geometry > Solids > Sphere

1. *Click* Sphere from the Geometry > Solids menu

2. Specify the first point, the center of the sphere

Click anywhere on the work plane or **type coordinate X, Y, or Z** to assign the first point to define the center of the sphere.

3. Specify the second point, the radius of the sphere

Click anywhere on the work plane, or **type coordinate X, Y, or Z**, or enter the scalar **value** to define the **radius of the sphere**. After the second point is assigned, the command will execute and generate a sphere.

Parallelepiped

Geometry > Solids > Parallelepiped

1. *Click* Parallelepiped from the Geometry > Solids menu

2. Specify the first point, the first corner

Click anywhere on the work plane or **type coordinate X, Y, or Z** to assign the first point as the first corner of the parallelepiped.

3. Specify the second point, the first side
Click anywhere on the work plane, or **type coordinate X, Y, or Z**, or enter the **scalar value** of the dimension to assign the second point as the second corner, to define the first dimension of the polygon rectangle.
4. Specify the third point, the second side
Click anywhere on the work plane, or **type coordinate X, Y, or Z**, or enter the **scalar value** of the dimension to assign the third point to define the second side of the polygon rectangle.
5. Specify the fourth point, the height
Click anywhere on the work plane, or **type coordinate X, Y, or Z**, or enter the **scalar value** of the dimension to assign the fourth point to define the height of the parallelepiped. After the fourth point is assigned, the command will execute and generate a parallelepiped.

Cylinder
Geometry > Solids > Cylinder
1. *Click* Cylinder from the Geometry > Solids menu
2. Specify the first point, the circle center
Click anywhere on the work plane or **type coordinate X, Y, or Z** to assign the first point as the center of the circle for the base of the geometry.
3. Specify the second point, the cylinder radius
Click anywhere on the work plane, or **type coordinate X, Y, or Z**, or enter the **scalar value** of the dimension to assign the second point to define the radius of the circle**.**
4. Specify the third point, the cylinder height
Click anywhere on the work plane, or **type coordinate X, Y, or Z**, or enter the **scalar value** of the dimension to assign the height of the cylinder. After the height is assigned, the command will execute and generate a cylinder.

2.1.8. Geometry – Boolean

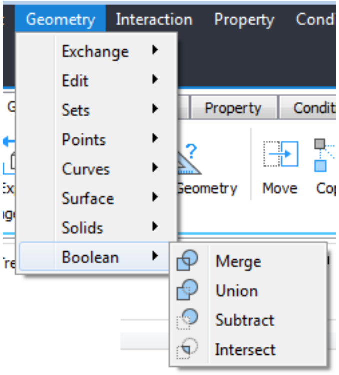


Figure 71. Boolean section in the Geometry drop-down menu

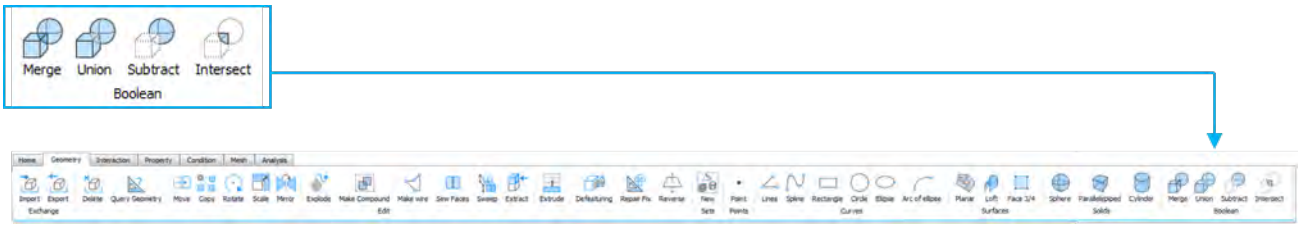
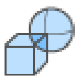
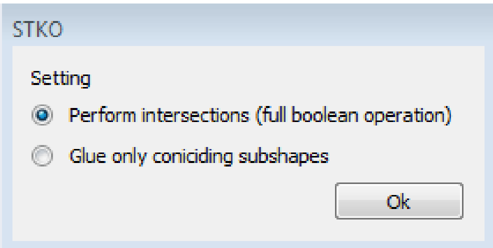


Figure 72. Boolean section in the geometry main Toolbar



Merge

Geometry > Boolean > Merge



1. *Click* Merge from the Geometry > Boolean menu

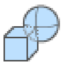
After *Clicking* **Merge**, STKO will open a setting window in which the user may choose either to execute a full Boolean operation or to glue the coinciding subshapes.

2. Select the geometries to Merge

Click on any point belonging to the geometries that you wish to merge; the selected geometries will turn red.

3. Execute the command

Press *[Enter]* on the keyboard or *Right-click* to execute the command. The geometries will be merged into a single geometry.



Union

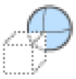
Geometry > Boolean > Union

1. *Click* Union from the Geometry > Boolean menu
2. Select the geometries to unite

Click on any point belonging to the geometries you wish to unite; the selected geometries will turn red.

3. Execute the command

Press *[Enter]* on the keyboard or *Right-click* to execute the command. The selected geometries will be united.



Subtract

Geometry > Boolean > Subtract

1. *Click* Subtract from the Geometry > Boolean menu
2. Select the first geometry, the constrained geometry

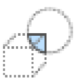
Click on any point belonging to the starting geometry/ies, defined as the **Retained geometry/s.** The selected geometries will turn **red.** To confirm the selection press the *[Enter]* key or *Right-click*.

3. Select the second geometry, the constrained geometry

Click on any point belonging to the second geometry/ies, defined as the **constrained geometry/s**. The selected geometries will turn **red**.

4. Execute the command

Press *[Enter]* on the keyboard or *Right-click* to execute the command.



Intersection

Geometry > Boolean > Intersection

1. *Click* Intersection from the Geometry > Boolean menu
2. Select the first geometry, the the constrained geometry

Click on any point belonging to the starting geometry/ies, defined as **the retained geometry/s**. The selected geometries will turn **red.** To confirm the selection press the *[Enter]* key or *Right-click*.

3. Select the second geometry, the constrained geometry
Click on any point belonging to the second geometry/ies, defined as **constrained geometry/s**. The selected geometries will turn **red**.
4. Execute the Command
Press *[Enter]* on the keyboard or *Right-click* to execute the command

2.2. Interaction modeling

Interaction Modeling defines the relationship among nodes or elements in order to simulate the interaction among them.

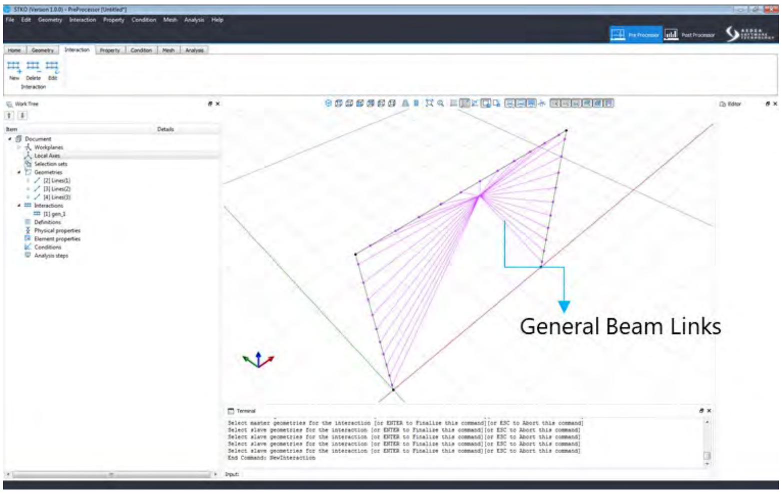


Figure 73. Example of General Beam Links

OpenSees supports three types of Interaction: **Node-to-Node links**, **Node to Element Links**, and **General Links**.
Choose the command **Interaction** > **New** from the main Toolbar, or *Right-click* on **Interaction** > **Add** from the Work Tree.
A new window (Interaction Editor) will appear, which allows the user to rename, edit, and specify the Interaction type.
By *click*ing on **Edit Shader…** the user will be able to assign colors and edge sizes to customize the visual appearance of the materials.
Select the default Interaction Type: **Node-to-Node links**. *Click* **OK** to confirm the selection. The output bar will guide the user to first select the Restrained Element. *Right-click* to confirm the first choice and to select the Constrained Element. After that, the user will see the new Interaction between nodes as shown below.

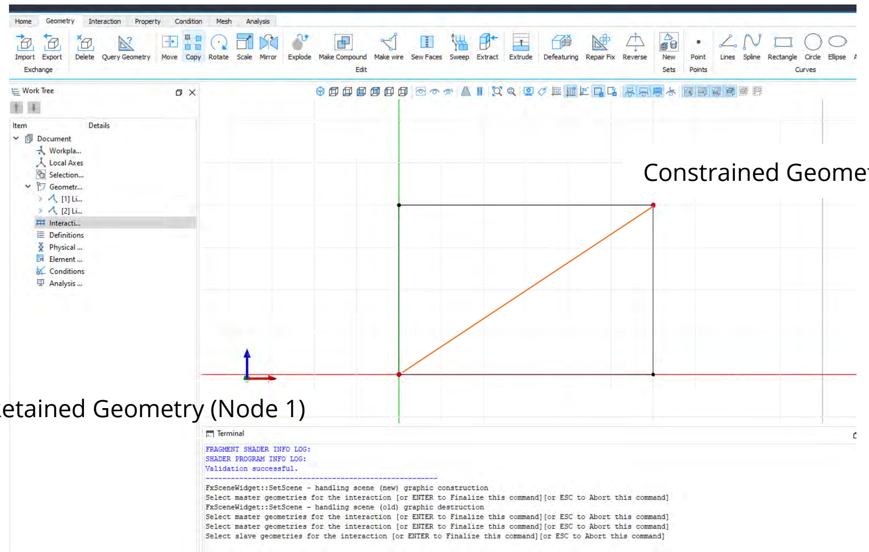


Figure 74. Example of Node to node links

For **Node-to-Node links**, see the OpenSees website where the example *ZeroLength Element* explains this Interaction Type.
For more info visit the webpage:
http://opensees.berkeley.edu/wiki/index.php/ZeroLength_Element

If the user wishes to work with **General Links** between elements, for example, between a surface and a line:
Choose the **Interaction** command > **New** from the Toolbar, or *right-click* on **Interaction** > **Add** from the Work Tree.
A new window (Interaction Editor) will appear, which allows the user to rename, edit, and specify the Interaction type. In this example, select **General Link**. *Click* **OK** to confirm the settings.
First, select the Retained Geometry and then the Constrained Geometries. *Right-click* to confirm each choice. Something similar to the example below will appear:

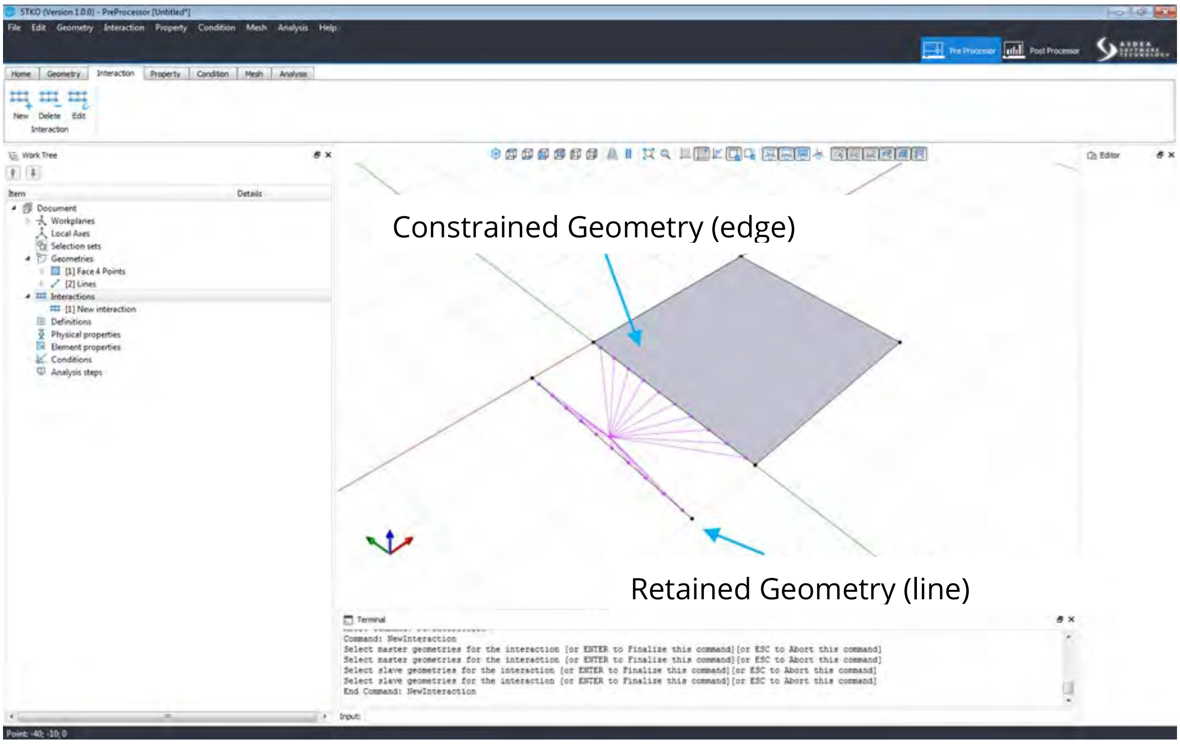


Figure 75. Example of General Links

For **General Links** see the OpenSees website, where the example *ZeroLengthContactNTS2D* explains this Interaction Type.
For more info visit the webpage:
http://opensees.berkeley.edu/wiki/index.php/ZeroLengthContactNTS2D
If the user wants to work with **Node to Element links** between elements, for example, a surface and a line:
Choose the command **Geometry-Interaction** > **New Interaction** from the Toolbar, or *Right-click* on **Interaction** > **Add** from the Work Tree.
A new window (Interaction Editor) will appear, which allows the user to rename, edit, and specify the Interaction type. In this example, choose **Node to Element links**. *Click* **OK** to confirm the settings.
First, select the Retained Geometry and then the Constrained Geometries. *Right-click* to confirm each choice. Something similar to the example below will appear:

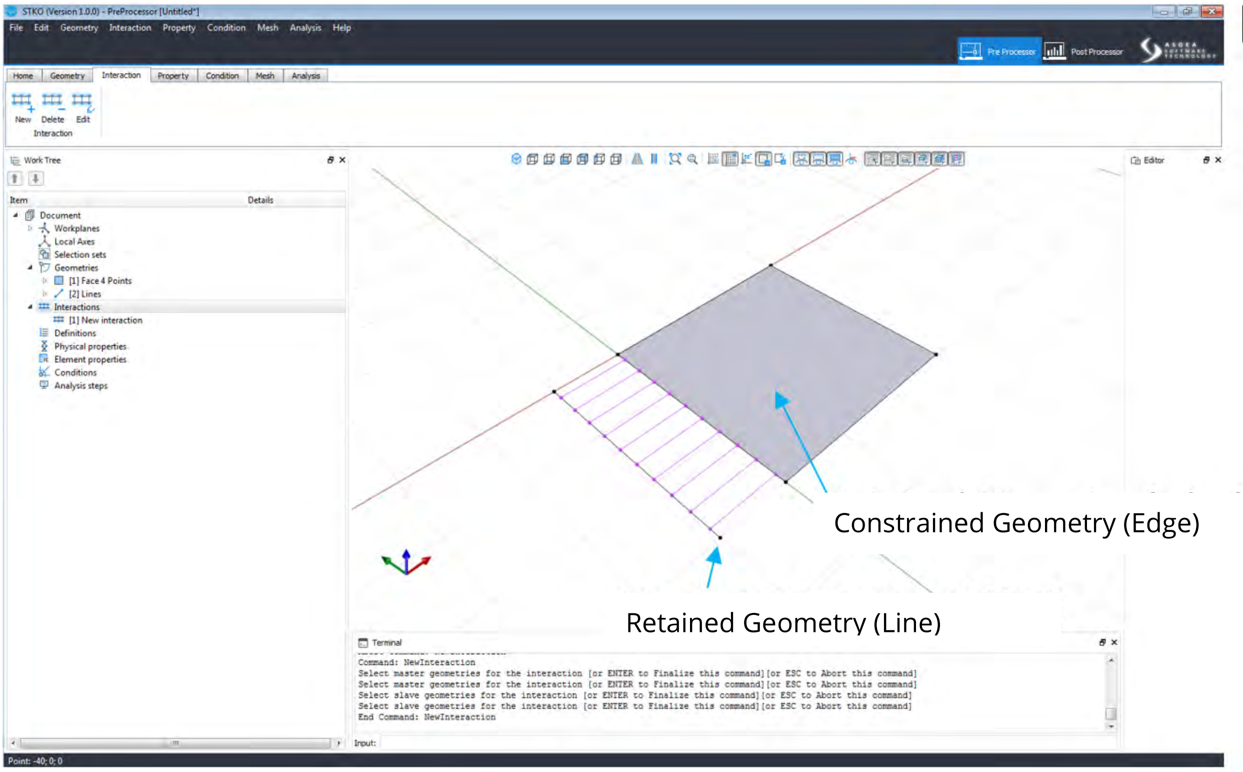


Figure 76. Example of Node to Element Link

Meshing the surface using structured mesh yields the following results:

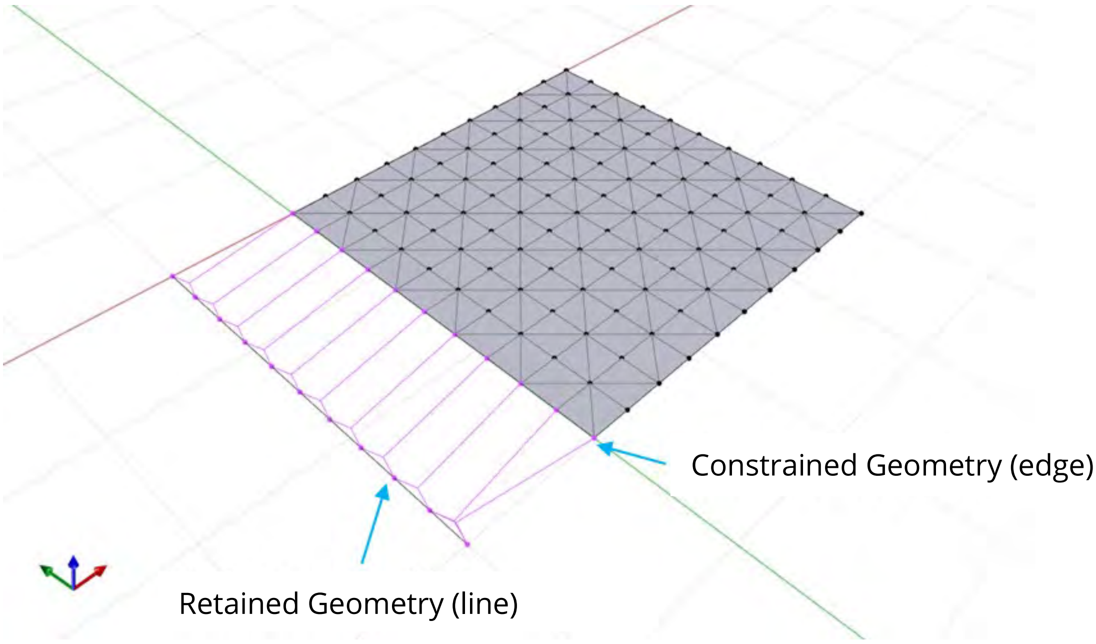


Figure 77. Example of Node to element link

For **Node to Element links,** see the OpenSees website, where the example *BeamContact3D* explains this Interaction Type.

For more, visit the webpage:

http://opensees.berkeley.edu/wiki/index.php/BeamContact3D

Another Example of Interaction Modeling

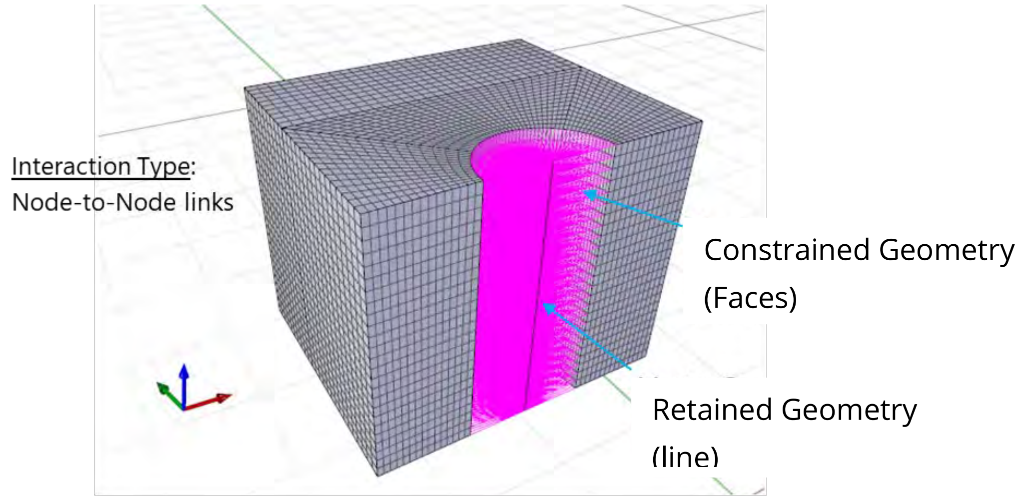


Figure 78. Example of Node-to-Node Links Interaction

NOTE: **EqualDOF**, **rigidLink**, and **rigidDiaphragm** can be only applied to the Interaction.

For this example: choose **Interaction** > **General link** > select the **retained node** > select the **constrained nodes** > **Ok** to confirm. After a General link has been created, the user can apply a **rigidDiaphragm** to the interaction**.**

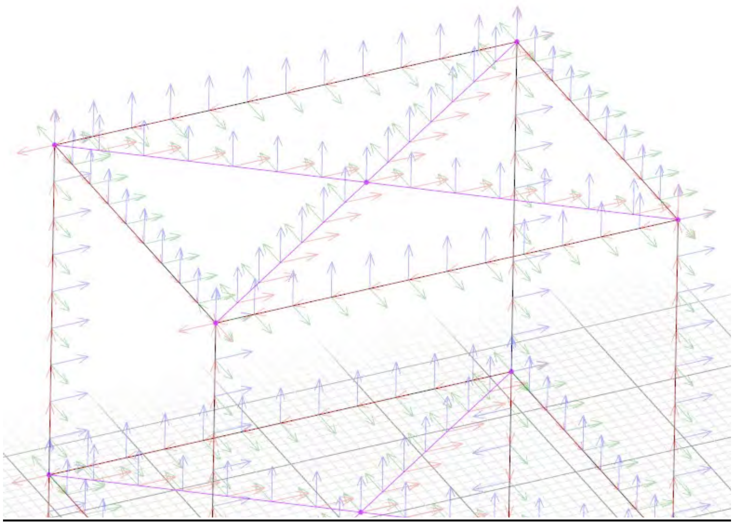


Figure 79. Rigid diaphragm

NOTE: Interaction will also influence the meshing and partition phase. For more information on **Meshing,** please consult [§ 2.8Meshing](). For more information on **Partition,** please consult [§ 1 Partition]().

2.3. Defining and Assigning Local Axes

The local coordinates system is defined such that the local x-axis will correspond to the tangent of a curve. The other axes (y-axis and z-axis) will be defined as being perpendicular to this local x-axis. The first step is to select the elements (*Figure 70*). In STKO, the Local Axes command translates to the **Geometric Transformation command** in OpenSees.
(*For more Information, please consult:*
http://opensees.berkeley.edu/wiki/index.php?title=Geometric_Transformation_Command&oldid=2626).

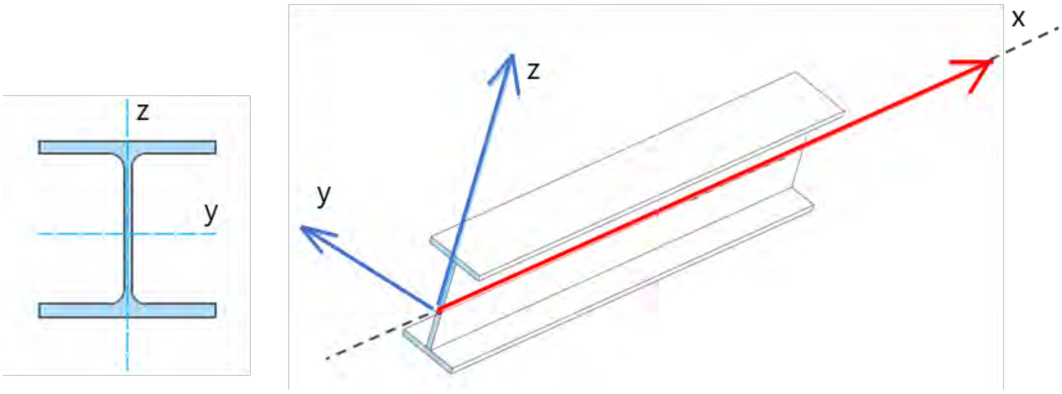


Figure 80 . Configuration of local axes in a double T section

Defining and assigning Local Axes to a figure can also be used to apply loads to the local coordinate system of a surface. The following image provides an example.

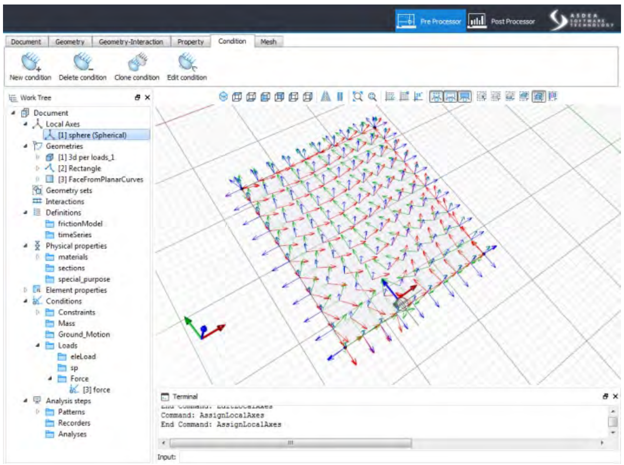


Figure 81. Local Axes Assignment

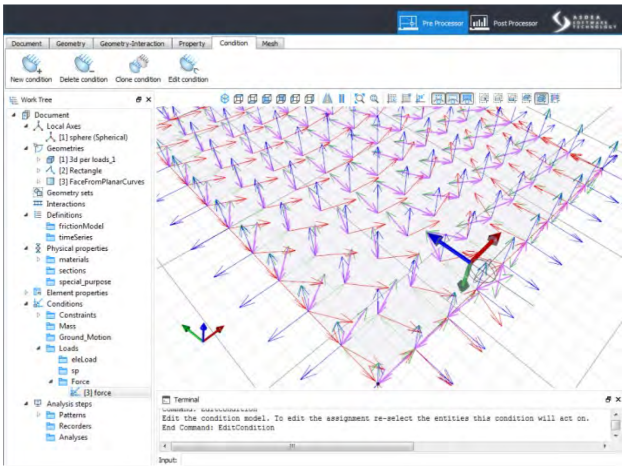


Figure 82. FaceForce Assignment

STKO allows local axes data to be exported to the Postprocessor, where results can be viewed according to the global and local axes previously assigned.
The following sections will explain how to use the rectangular, cylindrical, and spherical local axes commands.

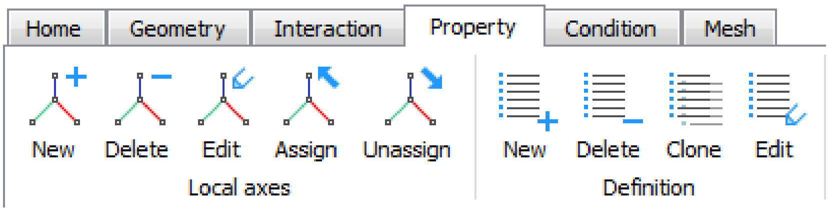


Figure 83. Local Axes toolbar.

2.3.1. Rectangular Local Axes

Rectangular Local Axes for a three-dimensional space are an ordered triplet of axes (e_x, e_y, and e_z vectors) that are pair-wise perpendicular. When local axes are displayed in the STKO window, the e_x vector is red, the e_y vector is green, and the e_z vector is blue and is always orthogonal to the local x-y plane.

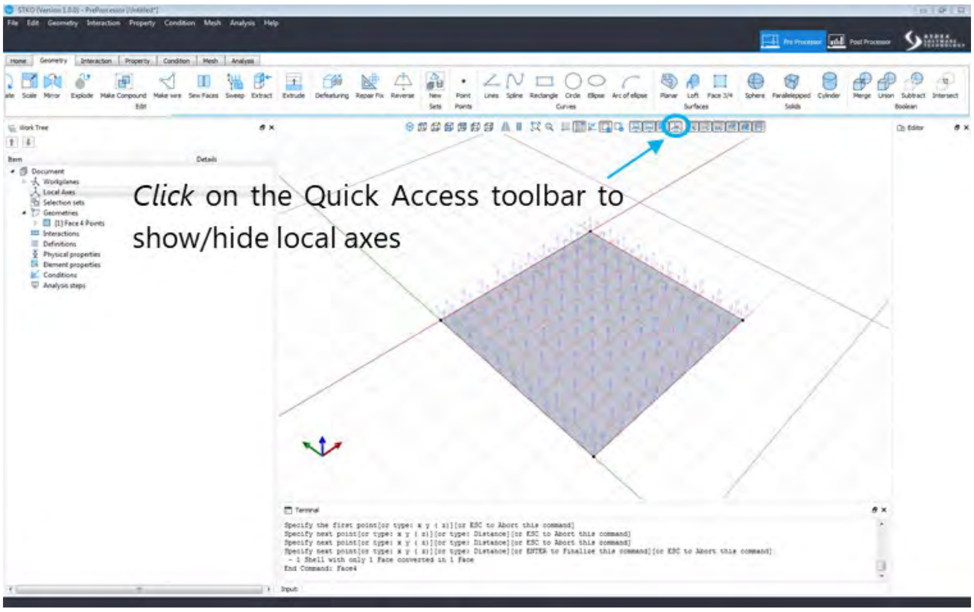
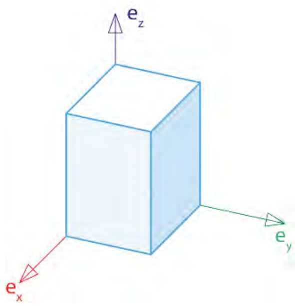
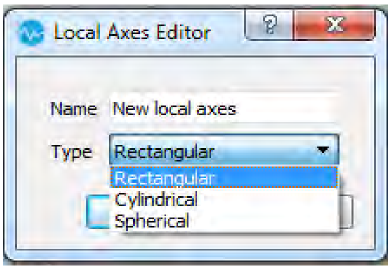


Figure 84. Visualization of local axes

To define a different direction from the *default* Local Axes:



Select **Property** > **Local Axes** > **New Local Axes** from the Toolbar or *Right-click* on **Local Axes** > **Add** from the Work Tree. Assign a name to the new local axes and select **Rectangular** from the **Type** drop-down menu.



Click on **OK** to confirm the Editor selection. Specify the first point (origin), by *clicking* on the work plane or by typing x, y, or z coordinates. Then, specify the second point (the X-axis), by *clicking* on the work plane or by typing x, y, or z coordinates. Follow the same step for the third point (X-Y plane).

To assign the New Local Axes to a surface, edge, or solid, *right-click* on the **new local Axes** > **Assign**, then select the geometry and *right-click* to confirm.

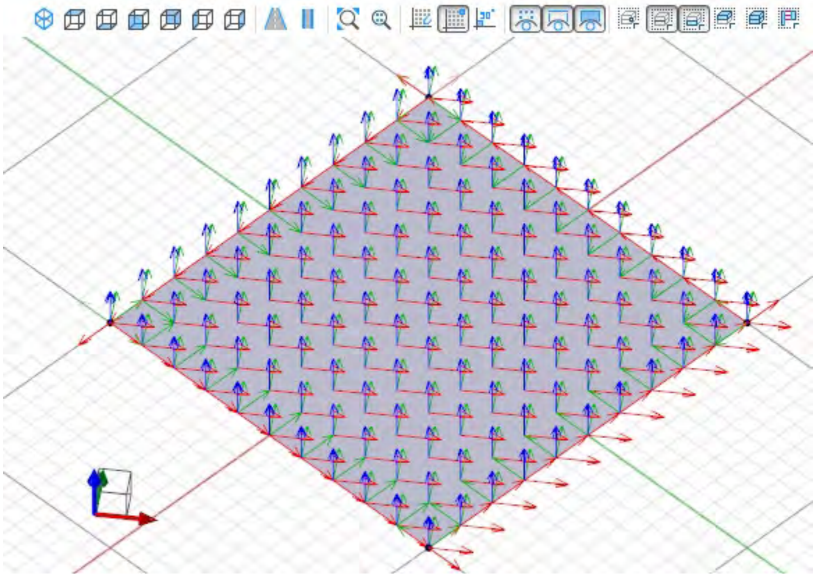


Figure 85. Visualization of quadratic local axes

Users may also directly *click* on the New Local axes in the Work Tree and drag it to the geometry. To unassign the new local axes, click the geometry to which the user previously assigned the new local axes. Then, drag the new local axes from the Work Tree to the selected geometry while holding the CTRL button.

To **Edit** the rectangular local axes, *right-click* **New Local Axes** > **Edit** on the Work Tree and the Editor panel will appear. The Editor Panel allows the user to assign a load to the geometry in the direction of the new local axes.

2.3.2. Cylindrical Local Axes

Select **Property** > **Local Axes** > **New** or *Right-click* **Local Axes** > **Add** from the Work Tree. Assign a name to the new local axes and select **Cylindrical** from the **Type** drop-down menu.

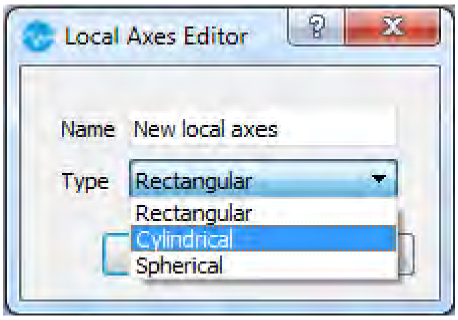


Figure 86. Local axes editor

Click **OK** to confirm the Editor selection. Specify the first point by *clicking* on the workplane or by typing x, y, or z coordinates. The first point will be the origin around which all the vectors rotate. Then, specify the second point (the X-axis) by *clicking* on the workplane or by typing x, y, or z coordinates Follow the same steps to assign the third point (X-Y plane).

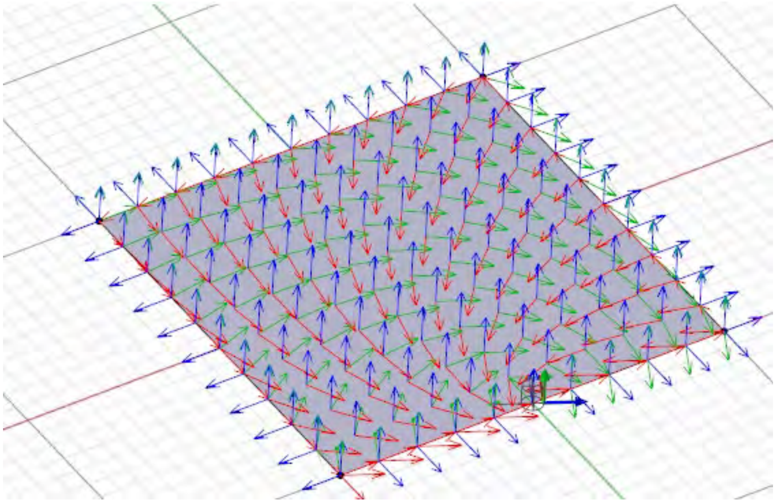


Figure 87. Visualization of cylindrical local axes to the edge of the surface

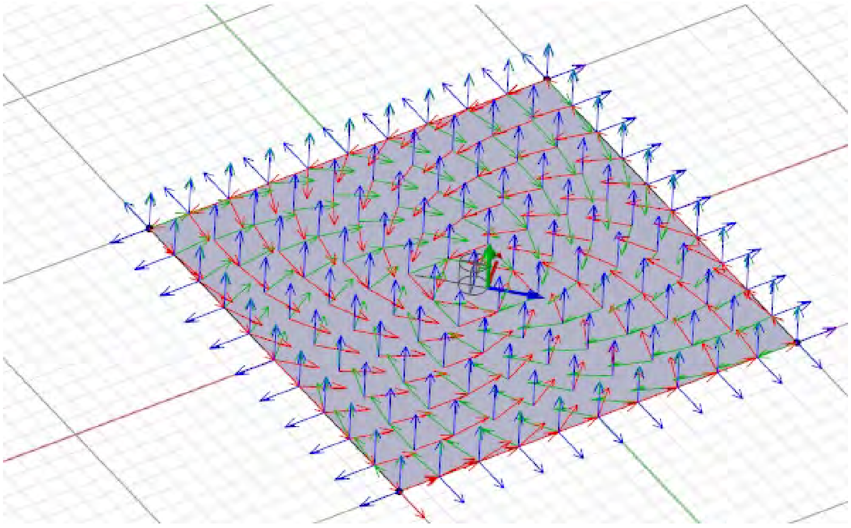


Figure 88. Visualization of cylindrical local axes to the center of the surface

To assign the New Local Axes to a surface, edge or solid, *Right-click* on the **new local Axes** > **Assign** > select the geometry > *Right-click* to confirm.
Or, users may *click* directly on the New Local axes and drag it to the geometry from the Work Tree. To unassign the new local axes, click the geometry to which the user previously assigned the new local axes. Then, drag the new local axes from the Work Tree to the selected geometry while holding the CTRL button.

To Edit the cylindrical local axes, *Right-click* on the New **Local Axes** > **Edit** and the Editor panel will appear.
Then it will be possible to assign a load to the geometry in order to direct it according to the new local axes.

2.3.3. Spherical Local Axes

Select **Property** > **Local Axes** > **New** or *right-click* **Local Axes** > **Add** from the Work Tree. Assign a name to the new local axes and select **Spherical** from the **Type** drop-down menu.
Click **OK** to confirm the Editor selection. Specify the first point by *clicking* on the workplane or by typing x, y, or z coordinates. The first point will be the origin around which all the vectors rotate. Then, specify the second point (the X-axis) by *clicking* on the workplane or by typing x, y, or z coordinates Follow the same steps to assign the third point (X-Y plane).
To assign the New Local Axes to a surface, edge, or solid, *right-click* on the **new local Axes** > **Assign** > select the geometry > *Right-click* to confirm.
Or, users may *click* directly on the New Local axes and drag it to the geometry from the Work Tree. To unassign the new local axes, click the geometry to which the user previously assigned the new local axes. Then, drag the new local axes from the Work Tree to the selected geometry while holding the CTRL button.

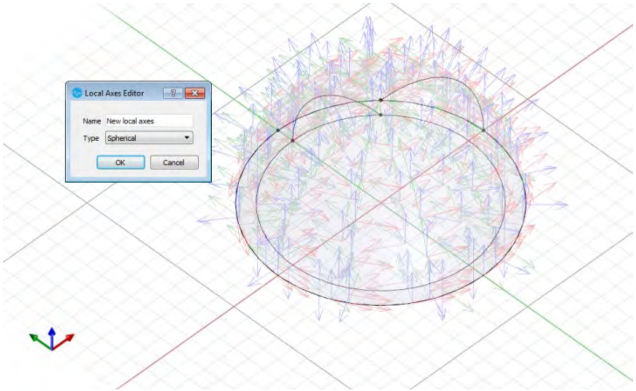


Figure 89.Visualization of spherical local axes

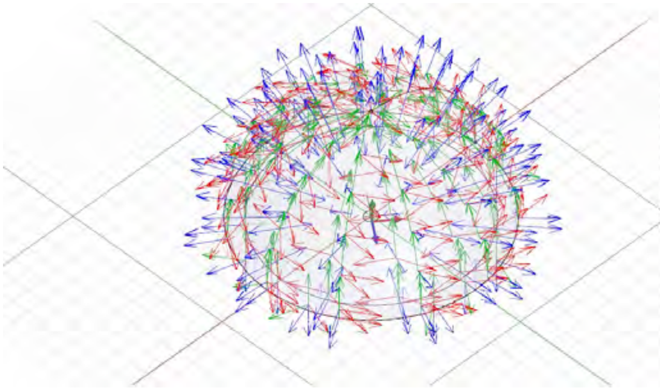


Figure 90. Visualization of spherical local axes

2.4. Defining and Assigning Physical Properties

To define and assign **Physical Properties** to the model, choose **Property** from the Toolbar.

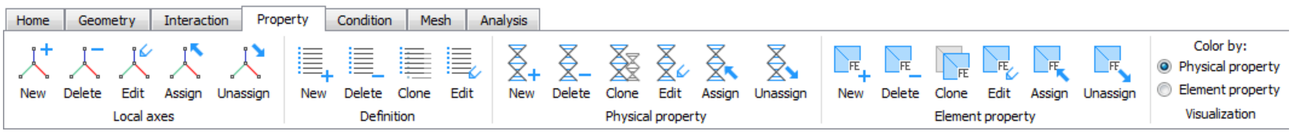


Figure 91. Property section in the main Toolbar

2.4.1. Materials

Choose the command **Property > New physical property** from the Toolbar, or directly *Right-click* **Physical properties** > **Add** on the Work Tree Panel.

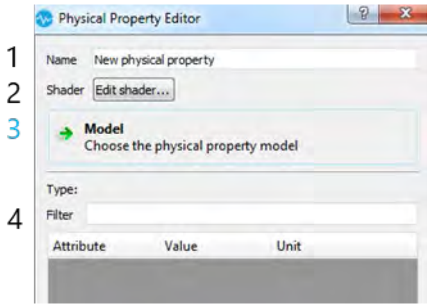


Figure 92. Physical Property Editor

Choose a **Name** to attribute to the New Physical Property (**1**), and customize its appearance using the command **Edit shader** (**2**) to enter the **Visual Material Editor**.

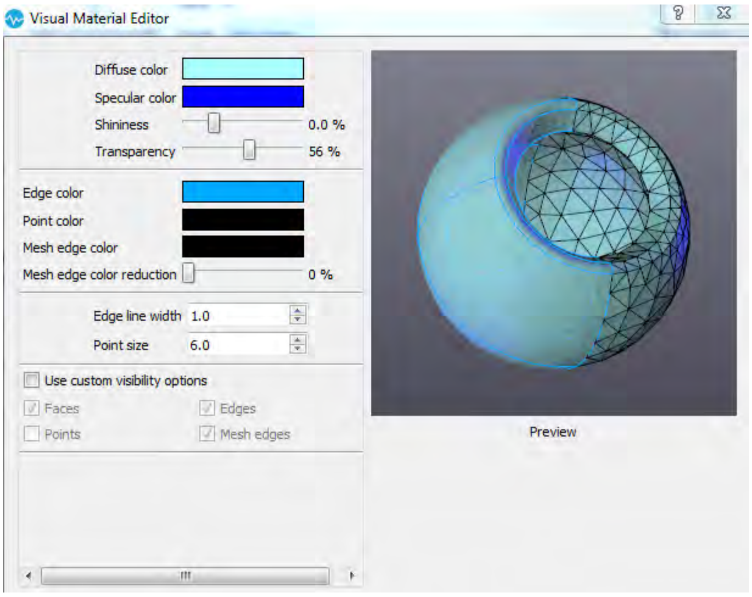


Figure 93. Visual Material Editor

Choose the **Diffuse** and **Specular colors** and give **Shininess** and **Transparency** to the new material. It is also possible to assign **Edge** and **Point colors** with different widths and sizes. The **Mesh edge color** and **Mesh edge color reduction** are also customizable. *Click* **Ok** to confirm the selections made in the **Visual Material Editor**.

Click **Model** (**3**) to choose the physical property model, and select **materials** > **uniaxial**, or **nD Opensees materials.**

Once the “Type” is selected, the user can insert **Values** for **Attributes** and read the description of each Attribute through an external link to the Opensees website.

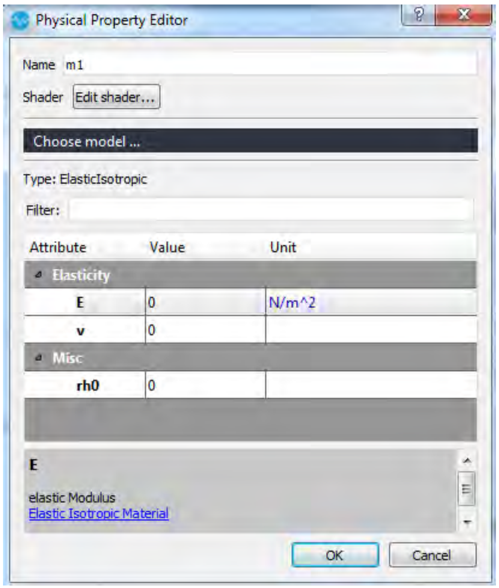
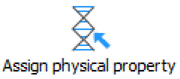


Figure 94. Physical Property Editor

To quickly select an Attribute, *click* on Filter (**4**) and type the first letter of the attribute desired. *Click* **Ok** to confirm the **Physical Property Editor** selections. Then, *Click*



Assign physical property on the Toolbar to attribute the new Material to the desired geometry. *Click* OK, select the geometry, then *Right-click* or press *[Enter]* to confirm.

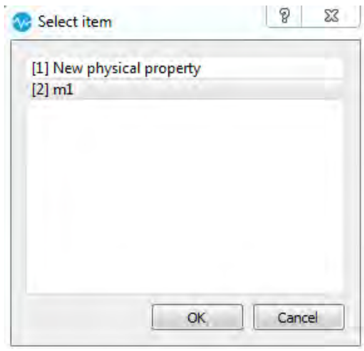


Figure 95. Window of item selection

The user may also right-click the new physical property on the Work Tree, select assign, click the desired geometry, then right-click to confirm the assignment.
To **Edit**, **Delete,** or **Clone** the created material, *click* **Property** and select the corresponding command from the Toolbar.

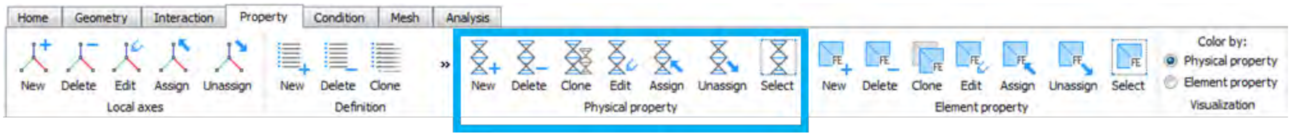


Figure 96. Delete, Clone and Edit physical property

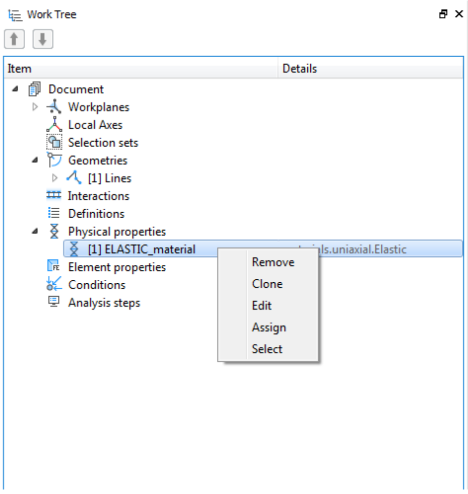


Figure 97. Remove, Clone, Edit, Assign, and Select a Material from the Work Tree

The user may also directly *Right-click* the new Physical Property on the Work Tree, and select the desired command from the drop-down menu.
To summarize, there are three ways that physical properties can be assigned to geometries:
-*Click* **Assign** on the main toolbar, select the physical property, press OK, *Click* on the desired geometry, and *Right-click* or press *[Enter]* to assign the property
-*Right-click* on the physical property on the **work tree** and select assign from the **drop-down** menu, then *Click* on the desired geometry, and *Right-click* or press *[Enter]* to assign the property
-*Drag and drop* the physical property from the Work Tree to the desired geometry
NOTE: Once the material has been assigned to the geometry, it will become the color that was set as the material color.

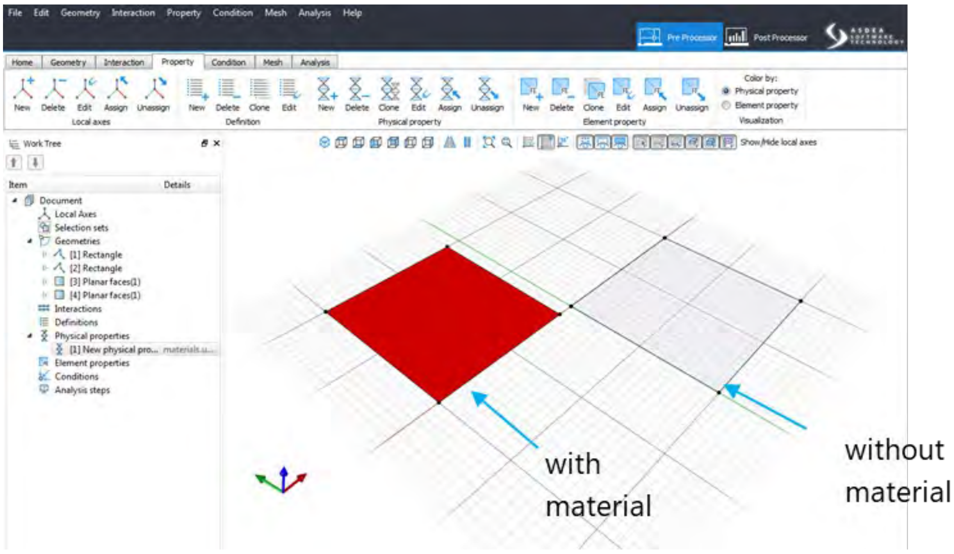


Figure 98. Example of surface with or without material

Let’s examine a few examples on how to set new physical properties in **STKO**.
Choose the command **Property** > **New Physical property** from the Toolbar.
As seen before, the **Physical Property Editor** will appear. Type the name, edit the Material colors, and select **Model.** This command allows the user to select different **Materials** or **Sections.**
Select, for instance, Model > materials > uniaxial > Steel and Reinforcing Steel Materials > Steel04

Click **Ok** to confirm the Material. It will appear in the Work Tree under the corresponding item.
The same process can be used if the user wishes to select a **Concrete Material**.
Select, for example, **New Physical Property > Model > materials > uniaxial > Concrete Materials > ConfinedConcrete01**. This command is used to construct a uniaxial material object of confined concrete.
For more information, please consult:
http://opensees.berkeley.edu/wiki/index.php/ConfinedConcrete01_Material

Click **OK** to confirm the new material. It will appear in the Work Tree under physical properties.
The same process can be followed if the user wishes to create other materials, such as an **nD Material** like **FaFourSteelPCPlaneStress**. This command is used to model a Prestressed Concrete Plane Stress material object.
For more information, consult:
http://opensees.berkeley.edu/wiki/index.php/Plane_Stress_Concrete_Materials

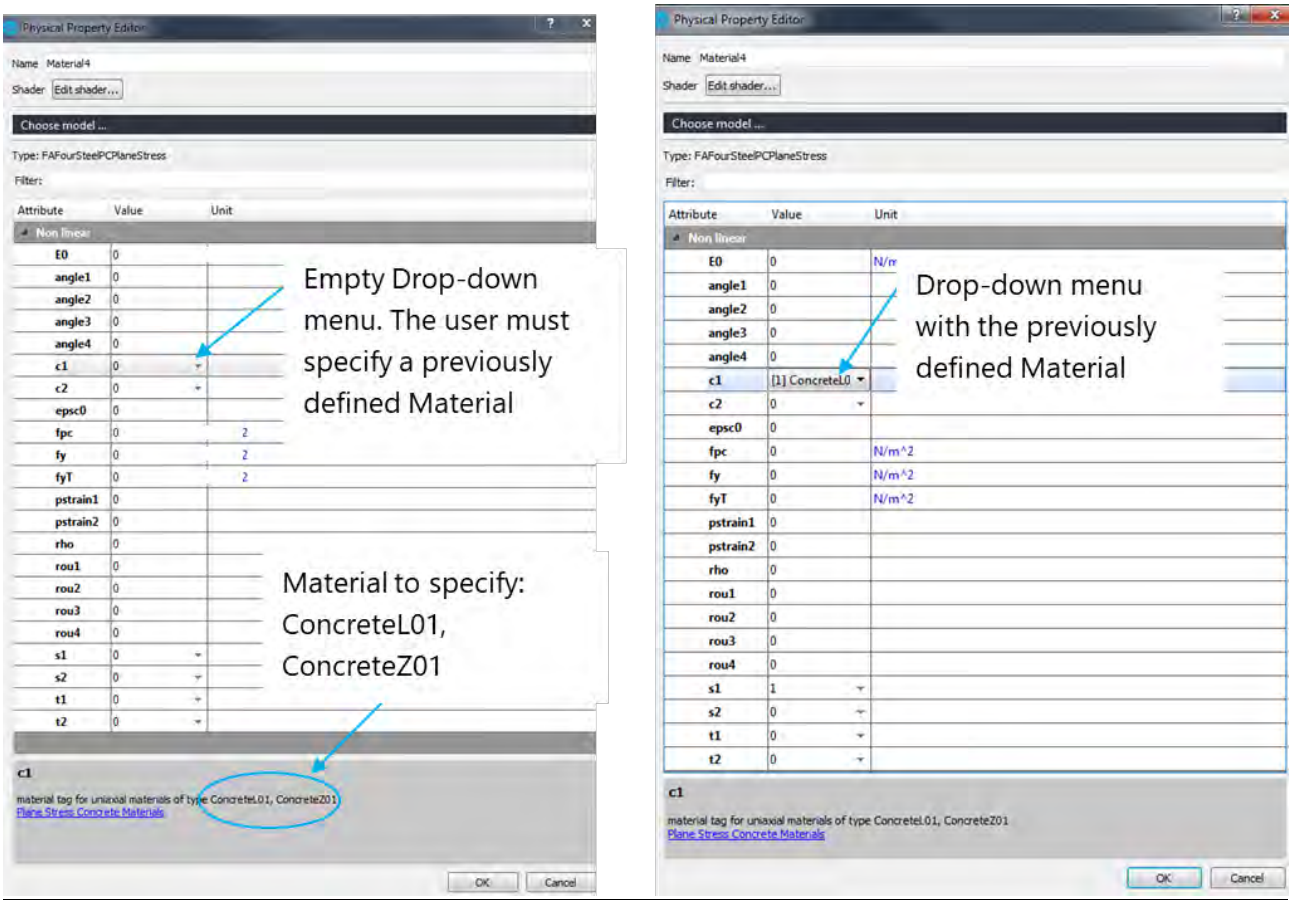


Figure 99. Physical Property Editor of FAFourSteelPCPlaneStress

Click **Ok** to confirm the new Material. It will appear in the Work Tree under the corresponding item.

Users may also create a **Parallel Material**. This command is used to construct a parallel material object made up of an arbitrary number of previously constructed **Uniaxial Material objects**.

To do so the user should *click* New Physical Property > Model > Materials Uniaxial > Some Standard Uniaxial Materials > Parallel.

For more information, consult:

http://opensees.berkeley.edu/wiki/index.php/Parallel_Material

A new Dialog box will appear which allows the user to select **Attributes**, insert **Factor Values**, and specify **Materials**.

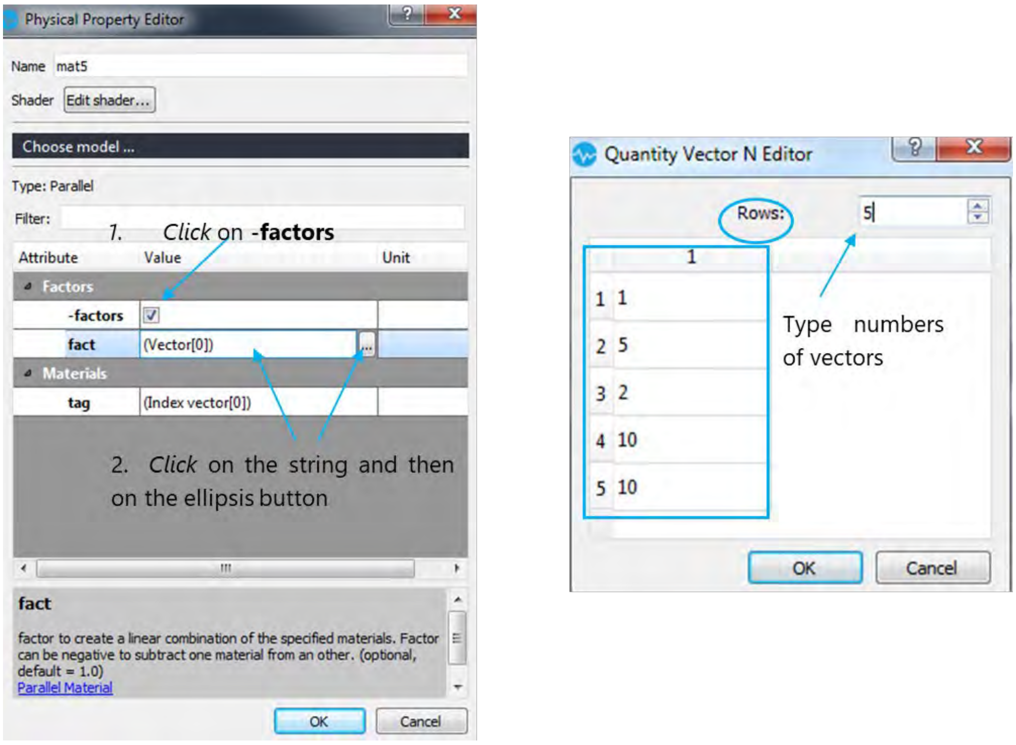


Figure 100. Physical Property Editor of Parallel material

After defining the **Quantity Vector,** the **Physical Property Editor** will update the new input. Repeat the same process for **Material Tags:** *Click* Index vector (0), then *Click* on the ellipsis button that will appear. The Index Vector Editor Window will allow the user to set the number of rows. The user should also select a previously defined Material for each row (i.e. ConcreteL01).

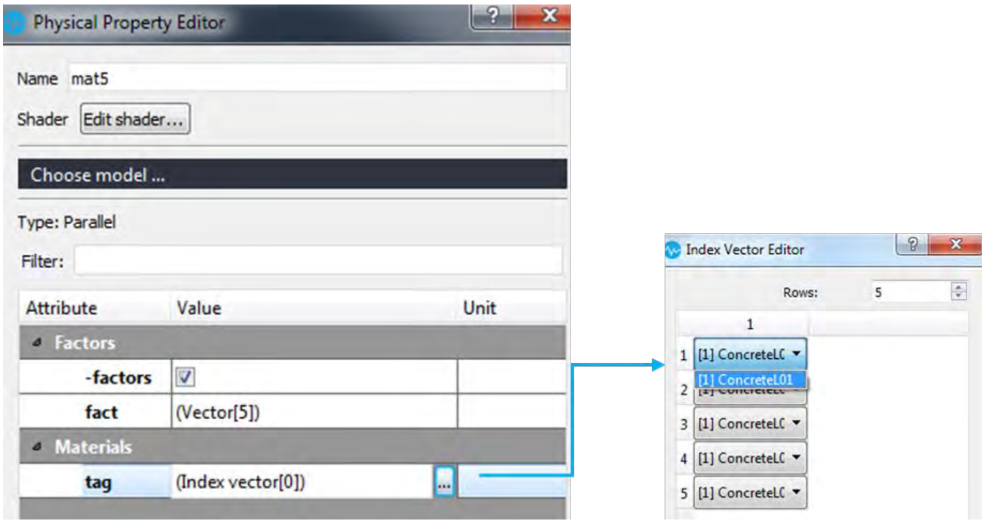


Figure 101. Index Vector Editor

2.4.1.1. Material Tester

The Material Tester is a tool added to all uniaxial materials available in OpenSees, and some of the 3d materials. After defining the material parameters, the user can choose a strain history and run different type of tests (monotonic, cyclic, custom, or in reference to a curve history). It is possible to customize the number of cycles to run in the test, the division of cycle, the target strain and the boundaries of the strain history curve. Once all the parameter are defined, the user can obtain the stress strain history of the material defined, under the chosen strain history, by pressing the *Test* button.

This tool is really useful for comparing the input parameters assigned to the materials in the numerical model and the ones from experimental testing.

In version 2.0.2, a new feature was added to the tester. Once the test has been performed, the user can click on the button *Data* to access a table widget for viewing and copying the data from the generated stress and strain curve.

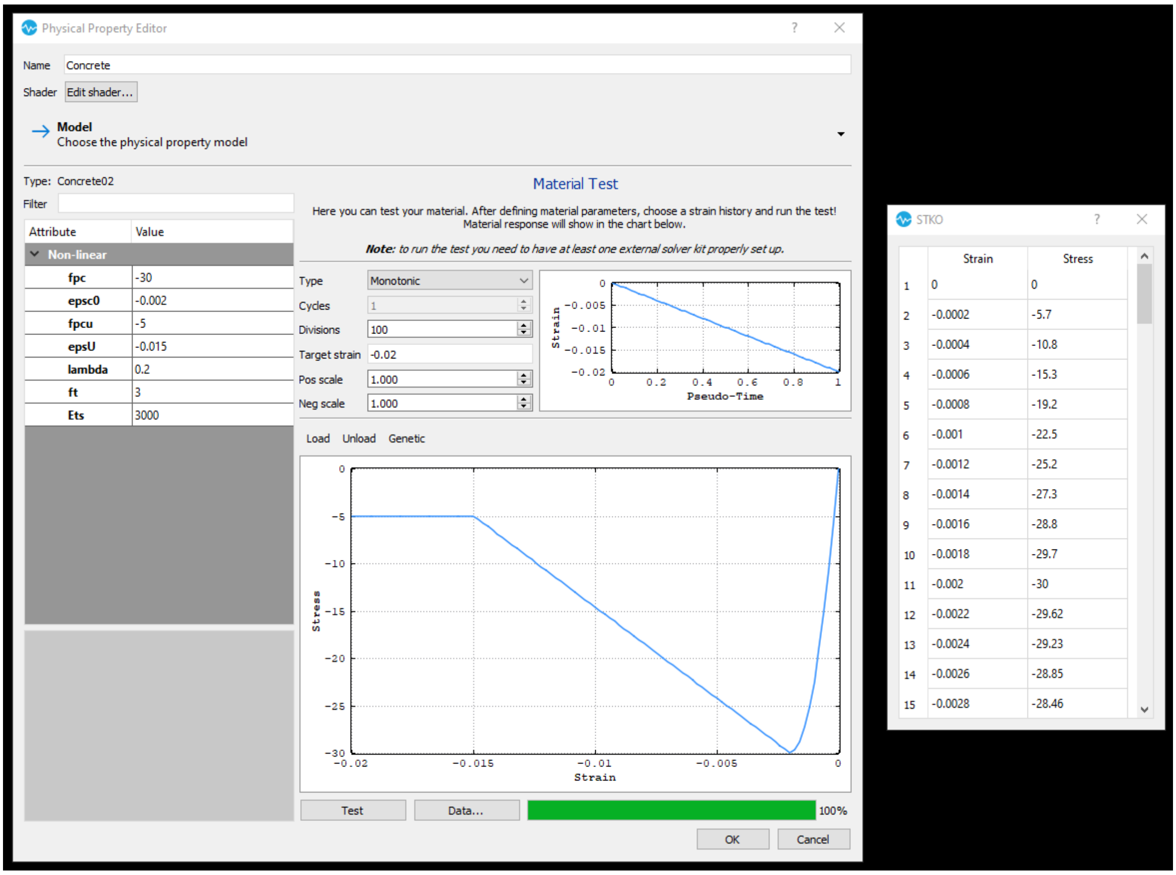


Figure 102. Material Tester

2.4.2. Cross-Sections

2.4.2.1. Elastic Beam Cross-Sections

To create an Elastic Beam Cross-Section, *click* **Property** > **New physical property** from the Toolbar, or directly *right-click* **Physical properties** > **Add** on the Work Tree Panel.
Then, the physical property **Model** can be chosen in the new interface. *Click* **Sections** from the drop-drown menu and select, for instance, **Elastic** as shown below.

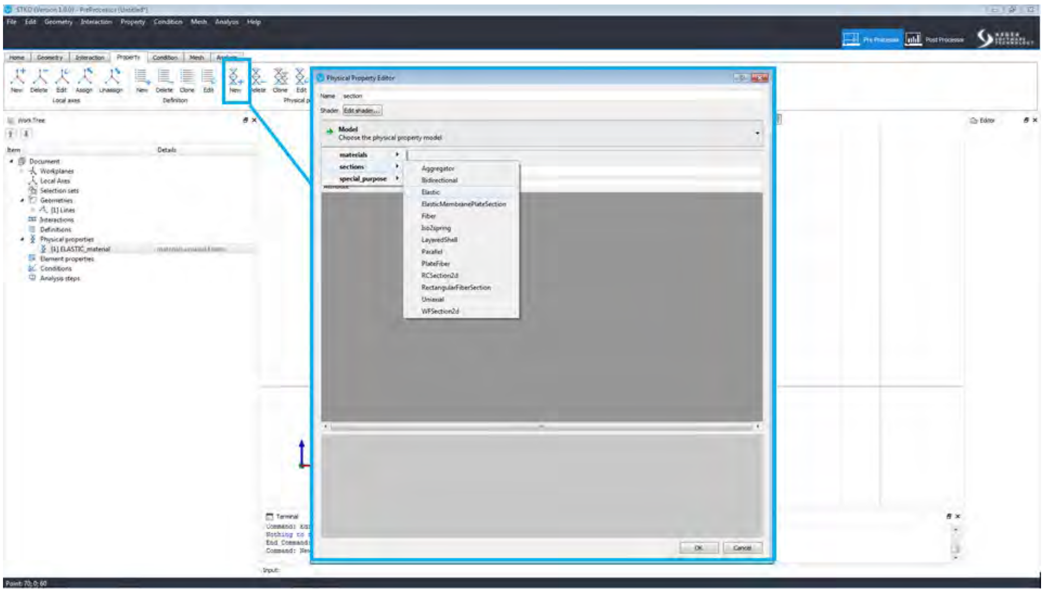


Figure 103. Physical Property Editor of an Elastic Beam Cross Section

An informative table relating to Elastic Sections will appear. *Click* where it says **Undefined** in the **Section** row, then *Click* the ellipsis button that will appear. A **BeamCrossSection Editor** will open for selecting the section type from the **Database** (1), the section preset with the sizes from **Section** (2), and the **Unit** (3) to assign to the elastic section. All available sections may be customized by users in terms of dimensions. Otherwise, there is the database available with standard sizes.

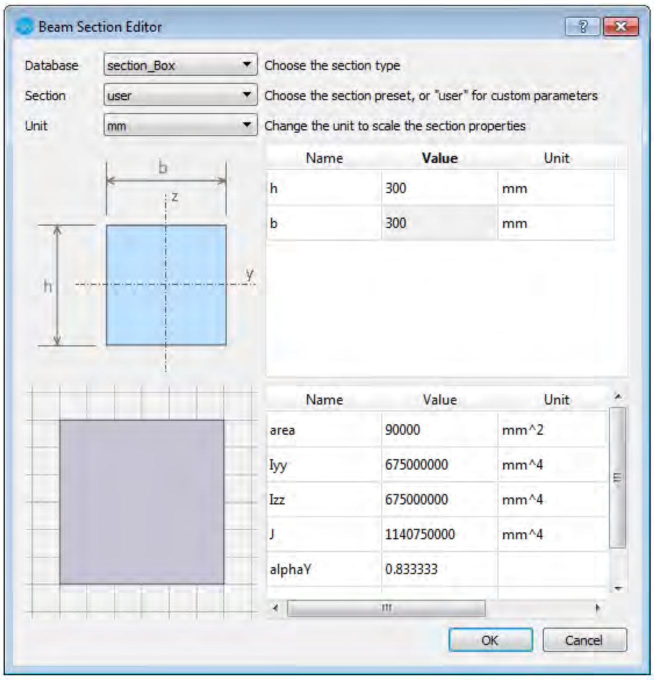


Figure 104. Beam Section Editor

Click **OK** to confirm the settings. The new section will appear under **Physical Properties** on the Work Tree. Now it is ready to be assigned to the model.

The new Elastic Section can be modified at any time. *Click* on **Edit Physical property** from the Toolbar and select the item from the list, or *Right-click* on the desired **Physical property** on the Work Tree and select **Edit.**

2.4.2.2. Fiber Beam Cross-Sections

After defining the Materials, to create a **Fiber Beam Cross-Section,** *click* **Property** > **New physical property** from the Toolbar, or directly *Right-click* **Physical properties** > **Add** from the Work Tree Panel.

Then the physical property **Model** can be chosen from the new interface. *Click* on **sections** from the drop-drown menu and select, for instance, **Fiber** as shown below.

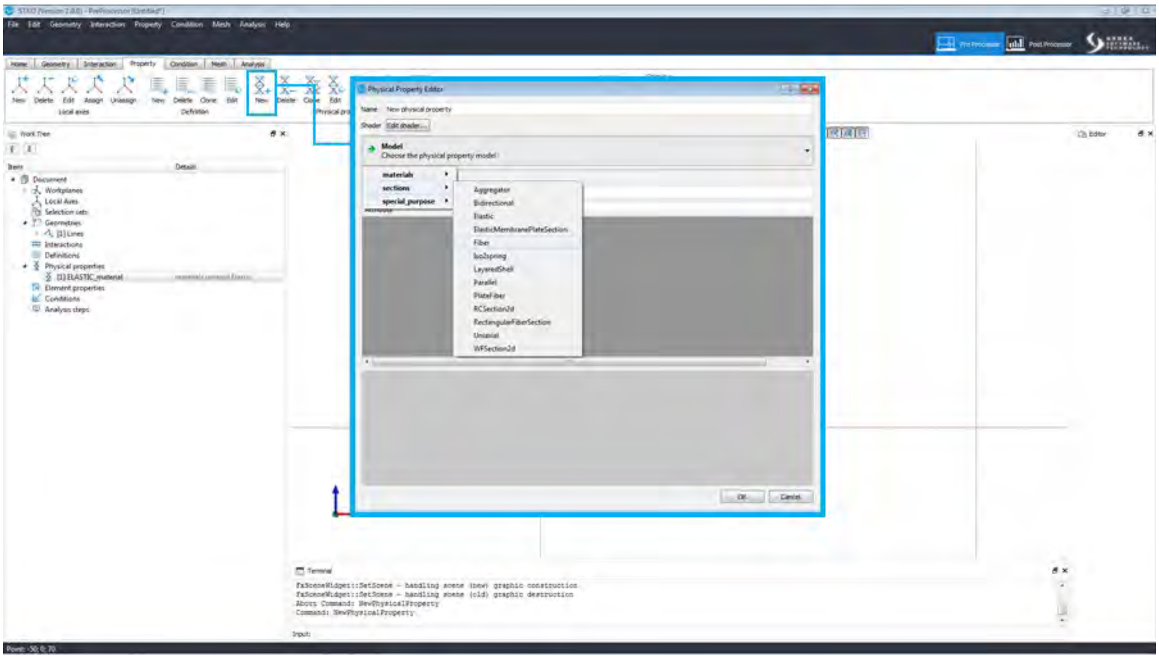


Figure 105. Physical Property editor of a Fiber

An informative table relating to Fiber Section will appear. *Click* on the **Fiber section** row, then *Click* on the ellipsis button which will appear. Like in STKO's pre- and postprocessor interfaces, the **Terminal** (3) contains two text editors: the first is what the software is generating according to the user's input, the second is the **Input** bar in which elements like the coordinates of points can be entered.

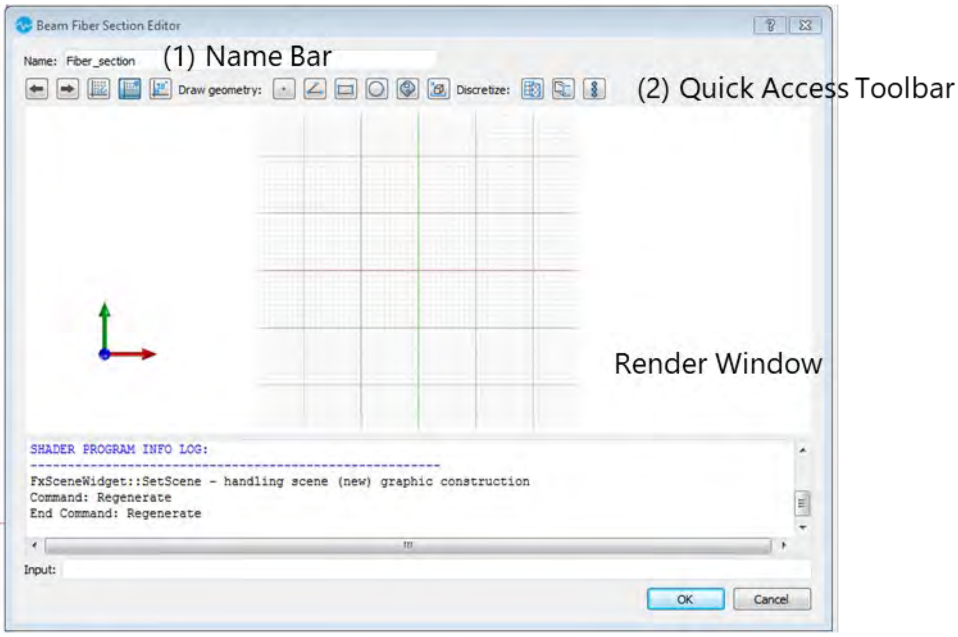


Figure 106. Beam Fiber Section Editor

On the Quick Access Toolbar, users have the option to import elements into the Beam Fiber Section Editor. *Clicking* the **Import** button on the toolbar, users can either choose to import an Elastic Section or a Geometry they have previously created on their work plane.

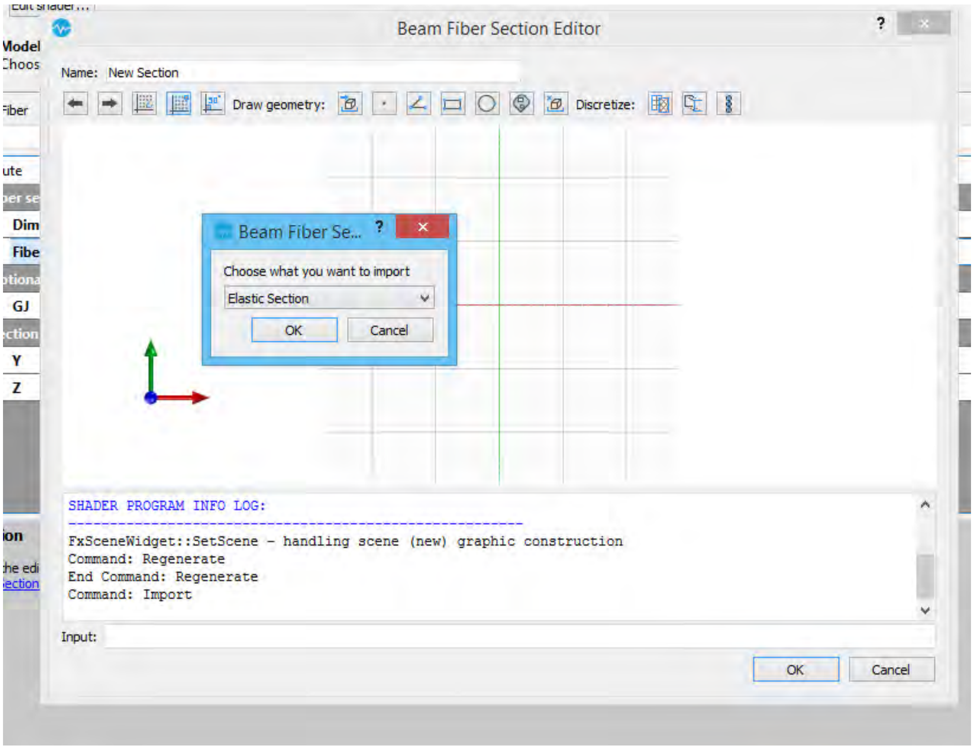


Figure 107. Import Example

If the user chooses Elastic Section, they will then be given the option to choose from a database of Elastic Section options. Once the user has chosen their desired section and set the units and parameters, they should *click* OK.

Then, the Elastic Section will be automatically imported into the Beam Fiber Section Editor where they can make further modifications.

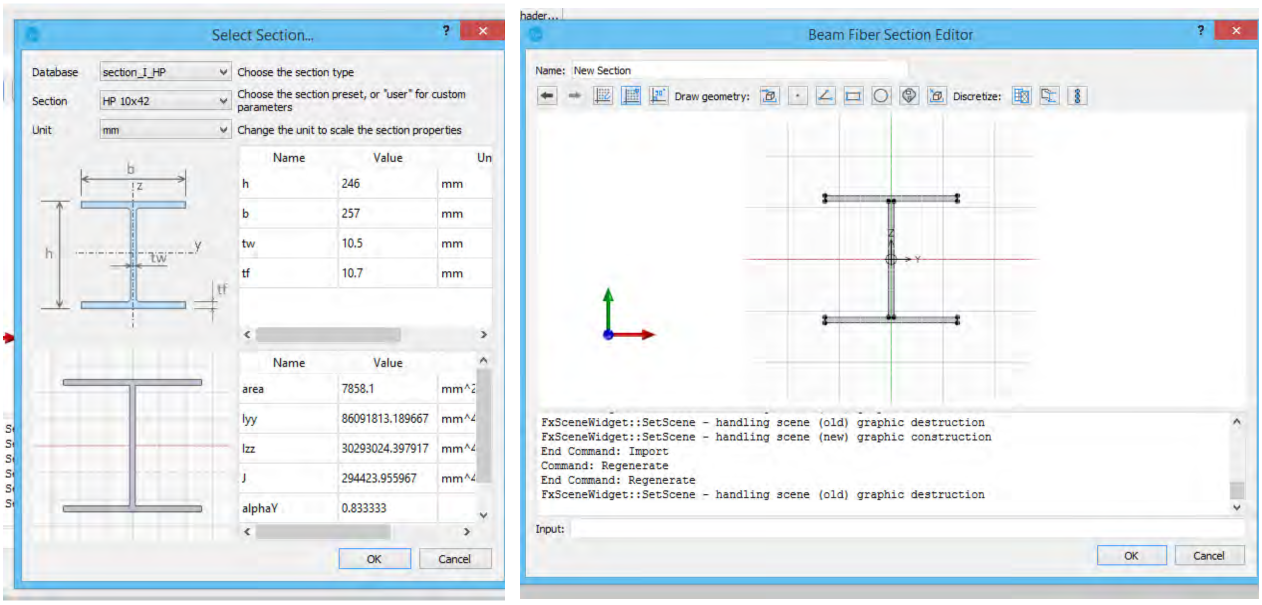


Figure 108. Elastic Section Import

The other option is to import a geometry that the user already has on their work plane. When the Import Geometry option is activated, the user will see a screen with the various geometries available listed. They will then be able to select the appropriate geometry, which will appear in the Beam Fiber Section Editor.

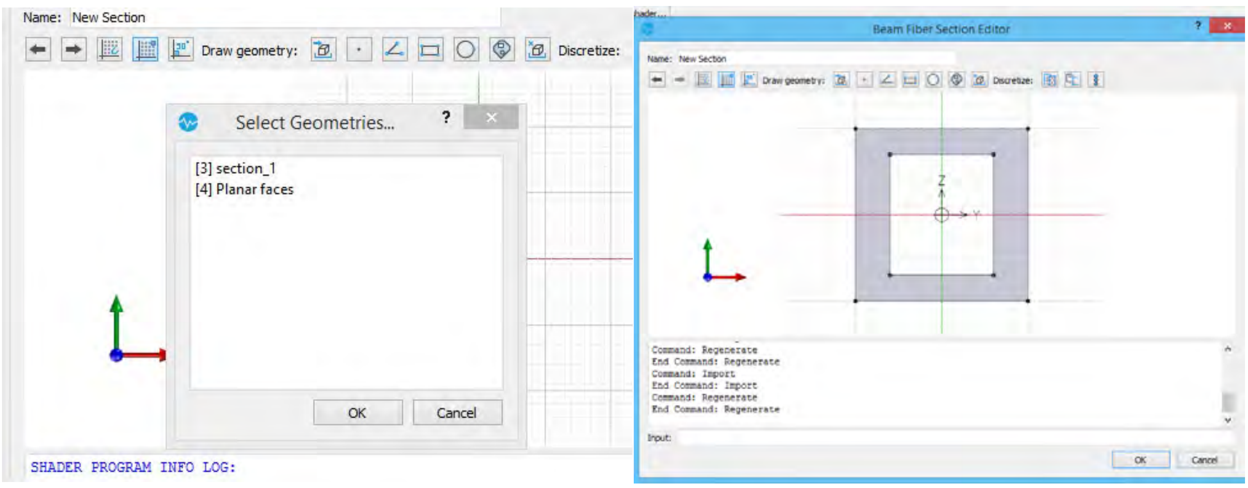


Figure 109. Import Geometry

<u>Example</u>: This example shows a Fiber Section with four different materials: **concrete cover**, **concrete core**, **rebars,** and Fiber Reinforced Polymers (**FRP**).

Select the Rectangle command from the Quick Access Toolbar to draw a Beam Section that will be the concrete cover (outer rectangle).

The Terminal bar specifies ways to design the rectangle: manually (specify the first and second corner), or by typing the **x, y**, and **z** (optional) coordinates, i.e.: 10 10 0, after the first corner. The user can also specify **rectangular width** and **height** after the first corner.

The user should now draw a second inner rectangle for the application of the second material: the concrete core.

Figure 110. Inner and outer rectangle of a Beam Section

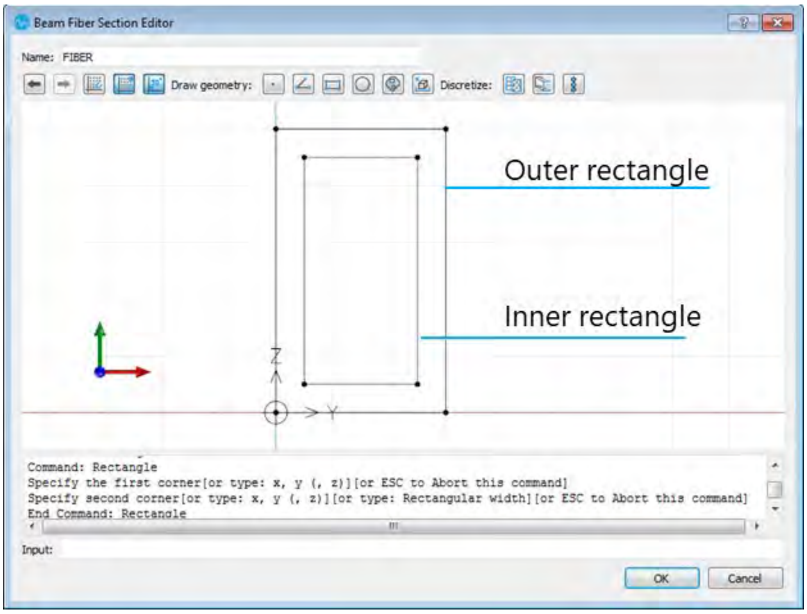


Figure 111. Inner and outer rectangle of a Beam Section

Select the **Surface from planar curves** command from the Quick Access

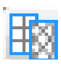
Toolbar to create a surface between the rectangles.

Select the outer rectangle and then the inner rectangle and *Right-click* to execute the command.

Select **Surface from planar curves** once more to create the surface of the inner rectangle.

Select the inner rectangle and *Right-click* to execute the command.

After creating the two surfaces, select **Make Surface Fiber** from the Quick Access Toolbar.



Automatically a **Dialog** box will appear in which the user can rename the surface fibers, insert a **Mesh Size,** and assign **Materials** (in our case, *concrete cover* for the outer surface and *concrete core* for the inner surface). *Click* OK and select the Outer Rectangle to apply the Surface Fiber.

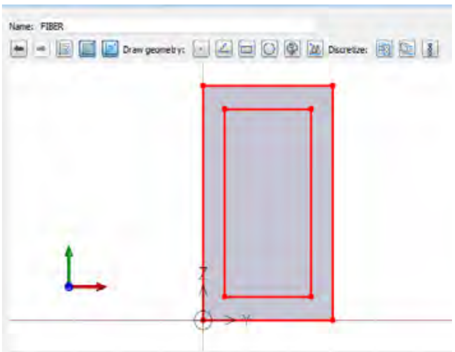


Figure 112. Inner and outer surfaces

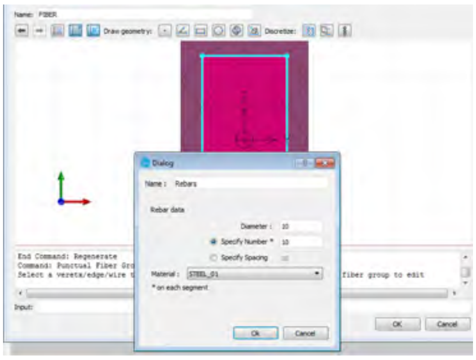


Figure 113. Assignment of Mesh Size and Material

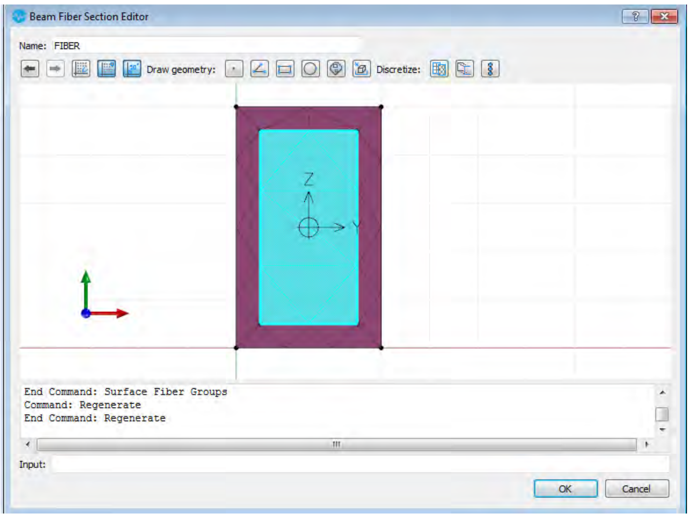


Figure 114. Fiber Beam Cross Section

The way the mesh can be applied to the Beam Fiber Section has been updated to include the quadrilateral structured mesh, inserting the input type by number. As shown in the picture below.

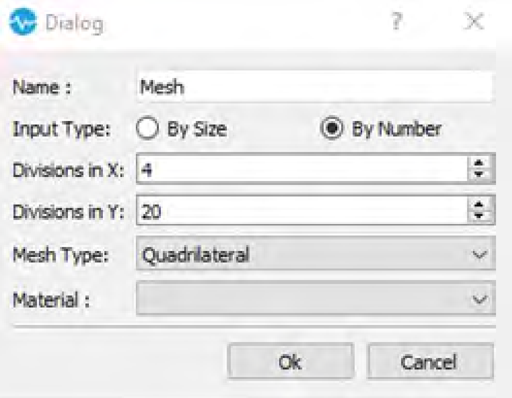


Figure 115. Meshing a Beam Fiber Section

If the section is not aligned with the XY axis as shown in the dialog, STKO will recognize X as the direction of the section closer to the X-axis and vice versa. The example below shows how X has been determined according to this criterion, assigning the same mesh by number.

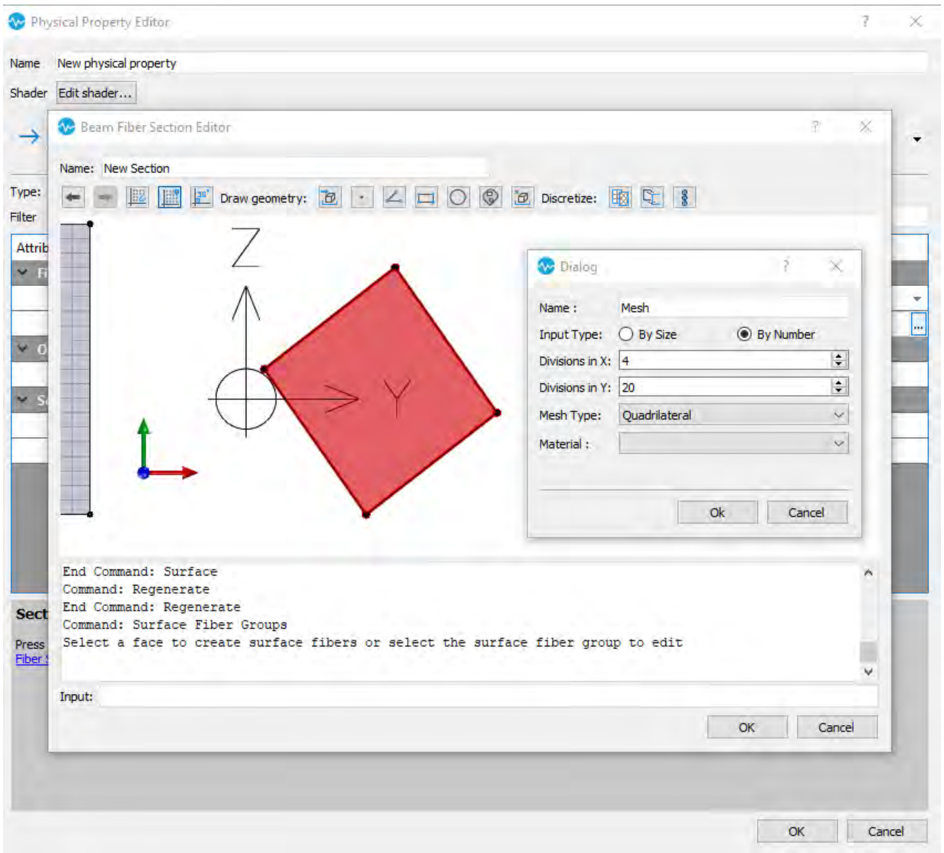


Figure 116. Meshing a Beam Fiber Section- 2

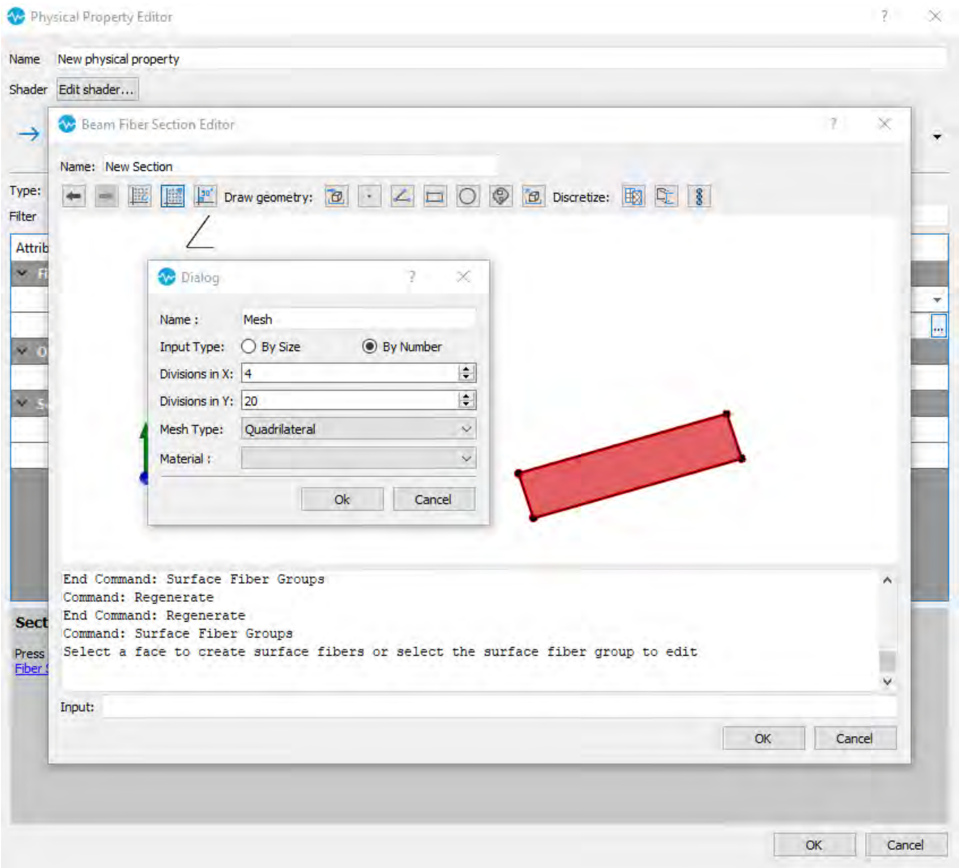


Figure 117. Meshing a Beam Fiber Section- 3

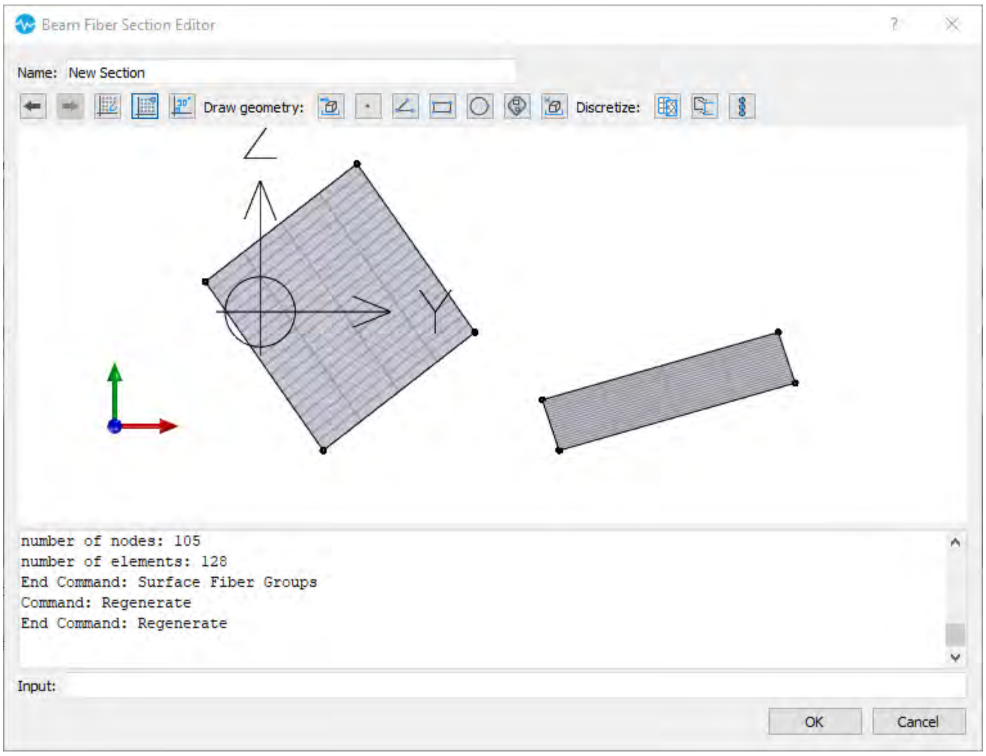


Figure 118. Meshing a Beam Fiber Section- 4

To add rebar Materials, draw another inner rectangle and select the **Make Punctual Fibers** command.

Automatically a **Dialog** box will appear in which the user may rename the rebars, insert **Rebar data,** specify the diameter, the number of rebars on each segment, and the spacing between the rebars. Assign the **Material** and *click* **Ok** to confirm.

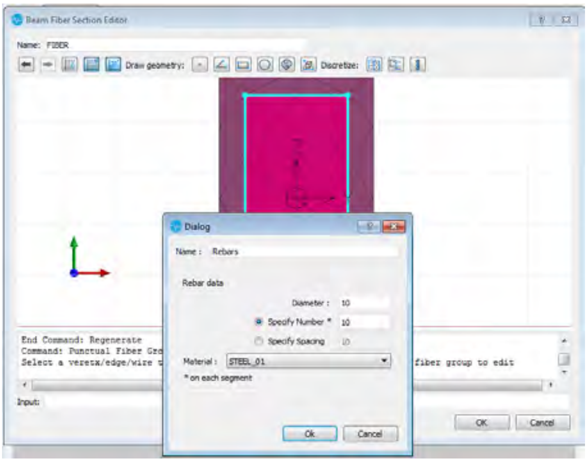


Figure 119. Dialog of the Rebar Material

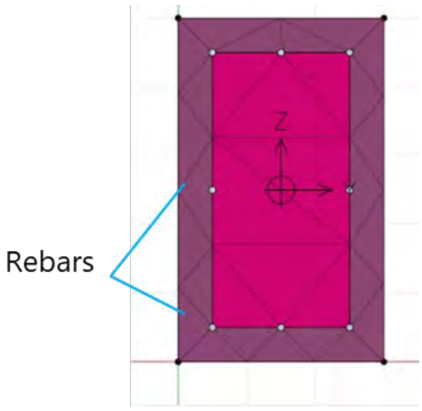


Figure 120. Rebar Detail

To create an FRP Material, draw a line to simulate it and select the **Linear Fibers** command from the Quick Access Toolbar.

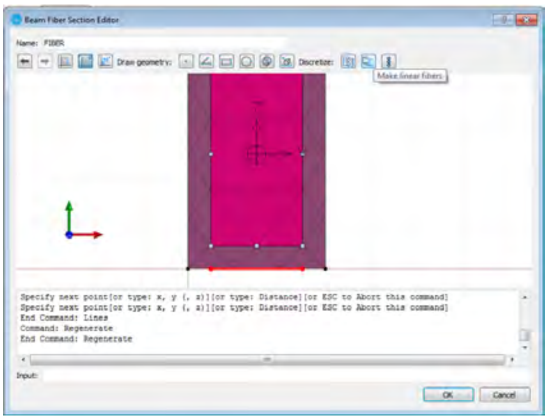


Figure 121. Dialog to size mesh, thickness, and material

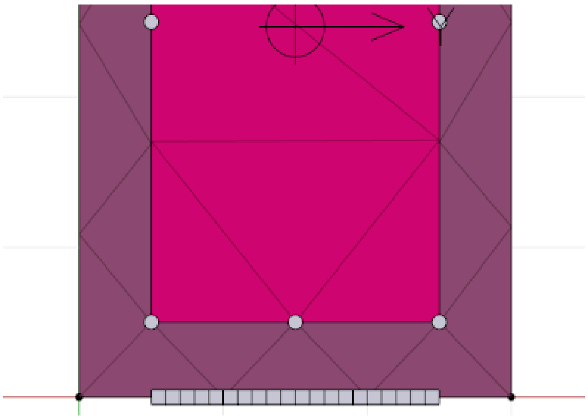


Figure 122. Detail of the linear fiber mesh

Click **Ok** to confirm the Section Fiber and to confirm the Physical Property Editor selections. The new section will appear in **Physical Properties** > **sections** on the Work Tree. It is ready to be assigned to the model.

To remove, clone, edit, or assign the Fiber Section, *click* on **Edit Physical property** from the Toolbar and select the item from the list, or *Right-click* the Physical Property on the Work Tree and choose the command.

2.4.3. Special Purpose for Physical Properties

STKO does not support the assignment of more than one physical property to a geometry. Assigning another property to the geometry will overwrite the previous assignment. This is generally an accepted assumption for a finite element. For this reason, **STKO** enables the user to create a **Special Purpose**, that is, a “container” in which more physical properties can coexist. After creating more physical properties, or element properties, such as sections, materials, etc., it is possible to create a special purpose like a BeamSection property to be added to the

previously-defined sections. Furthermore, the user can create different Special Purposes to assign to different geometries (e.g. beams and columns).
After creating sections (for example IPE100 and IPE120), it is now possible to create two **Special Purposes** from **Property** > **New physical property** from the Toolbar, or by *right-click*ing **Physical properties** > **Add** on the Work Tree Panel.
Select, for example, **Model** > **special_purpose** > **Beam-Column** > **BeamSectionProperty**. An informative table relating to BeamSection Property will appear. Insert the previously created section type.

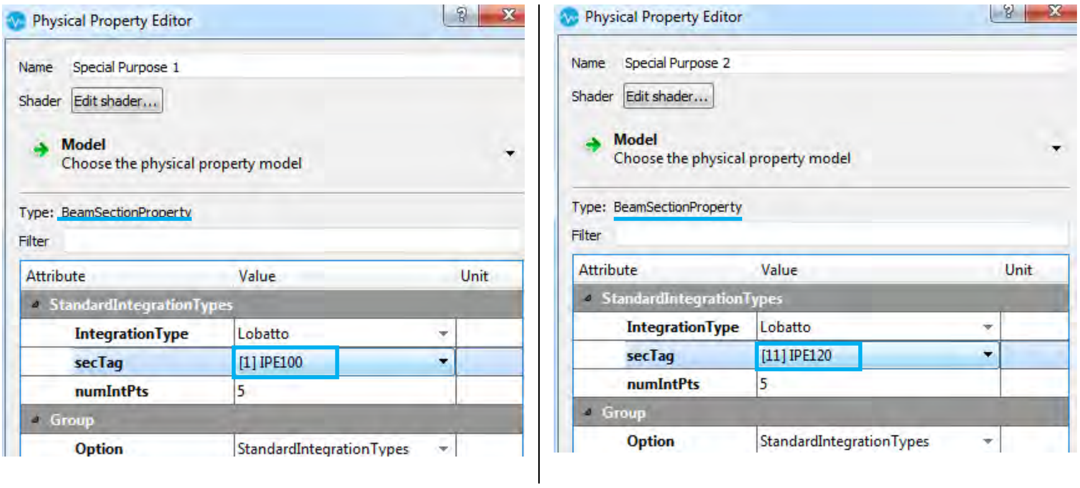


Figure 123. Special Purpose for Physical Properties (IPE100 and IPE120 of BeamSectionProperty)

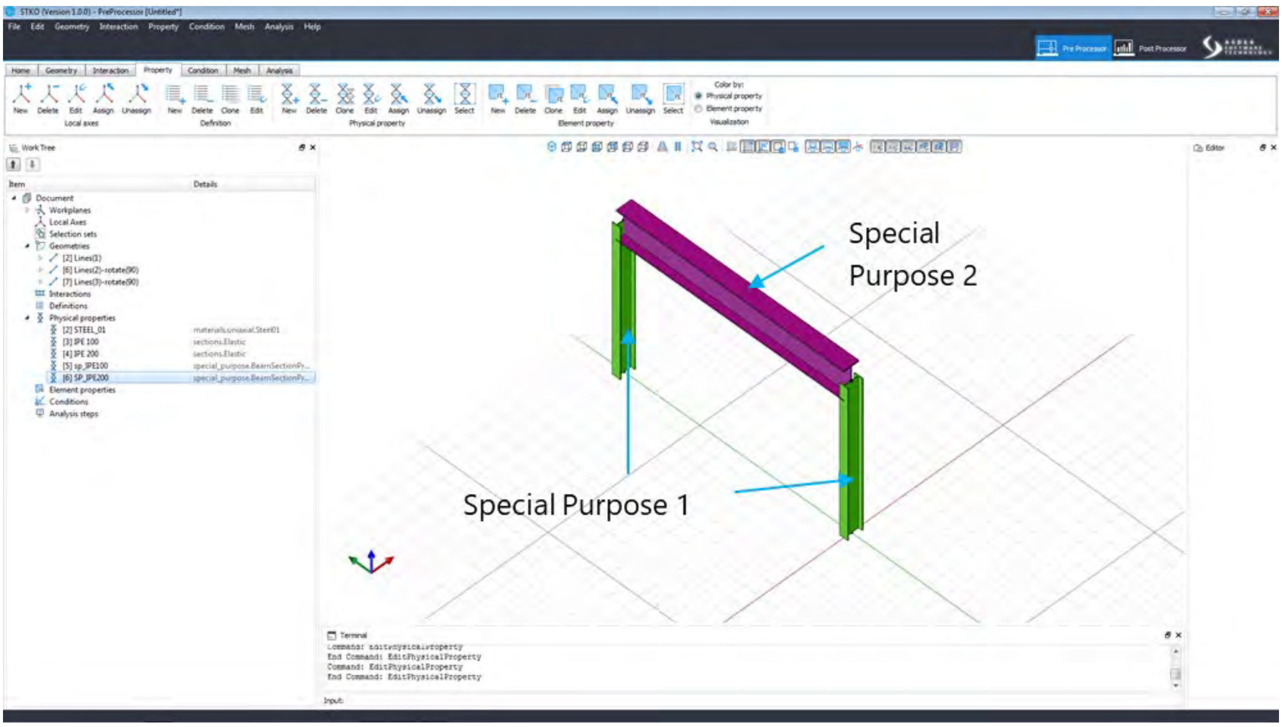


Figure 124. Visualization of Special Purpose

Click **OK** to confirm the settings.

An example is the **HingedBeam** Special Purpose. Create uniaxial materials such as *Steel and Reinforcing-Steel Materials* (Steel01). After that, create zeroLength special purposes to insert into the HingedBeam:

Select New Physical Property > special_purpose > zero-Length Material > zeroLengthMaterial from the Toolbar, or *Right-click* on special_purpose > Add from the Work Tree.

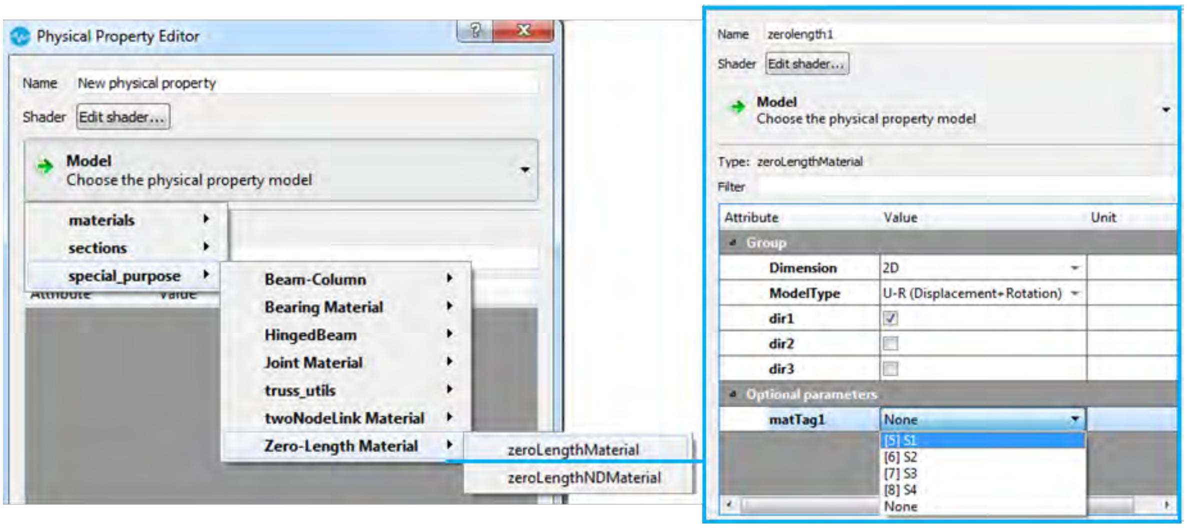


Figure 125. Physical Property Editor of a zeroLengthMaterial

After creating the **zeroLengthMaterials**, create a new special purpose **HingedBeam**

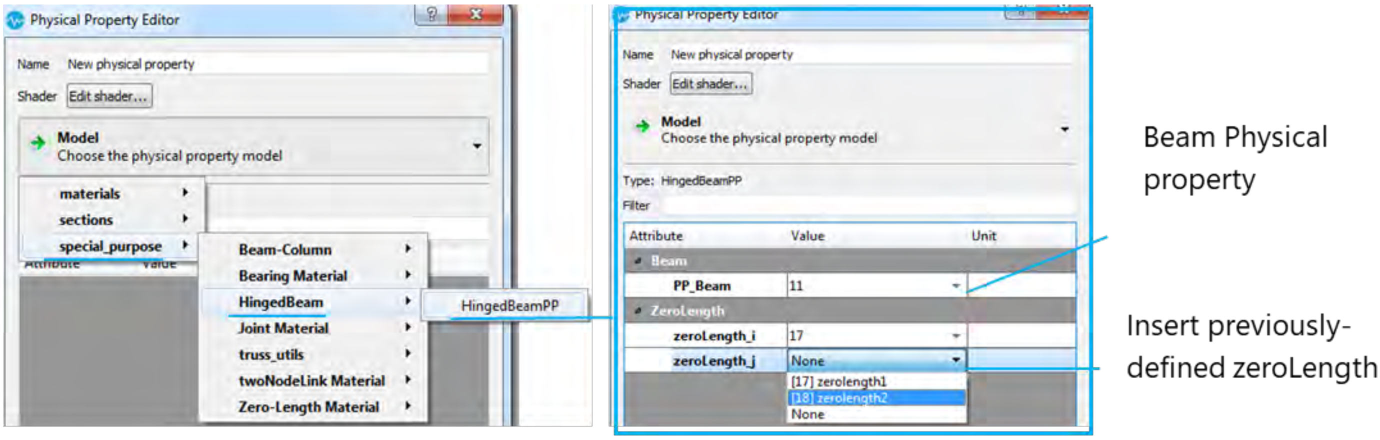


Figure 126. HingedBeam

Click **Ok** to confirm the settings. Select the element and assign the new Special Purpose:

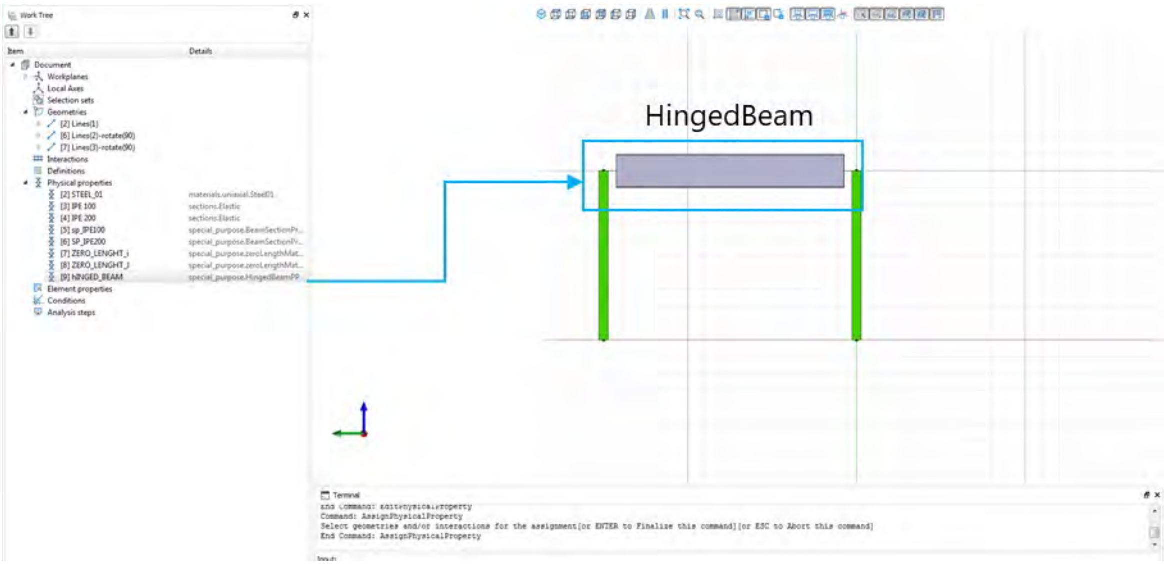


Figure 127. Visualization of HingedBeam

The next figure shows the scheme of the element relationship between OpenSees and STKO.

Figure 128. Scheme of the element relationship that occurred between OpenSees and STKO

<u>NOTE</u>: After creating a Special Purpose for Physical Properties, **STKO** allows for the generation of **Regions** before launching the Analysis (see § 0 Regions).

2.5. Defining and Assigning Element Property

Choose the command **Property** > **element Property** > **New** from the Toolbar, or by directly *Right-clicking* **Element properties** > **Add** on the Work Tree Panel.

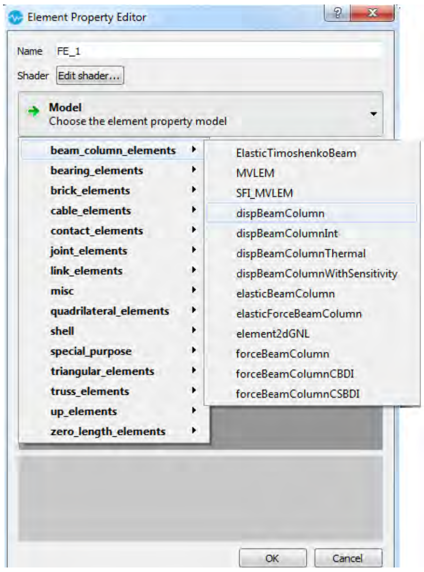


Figure 129. Element Property Editor

A new interface with the same Physical Property layout will appear, as shown above. Select the finite element that will define the geometry and assign values to the table.

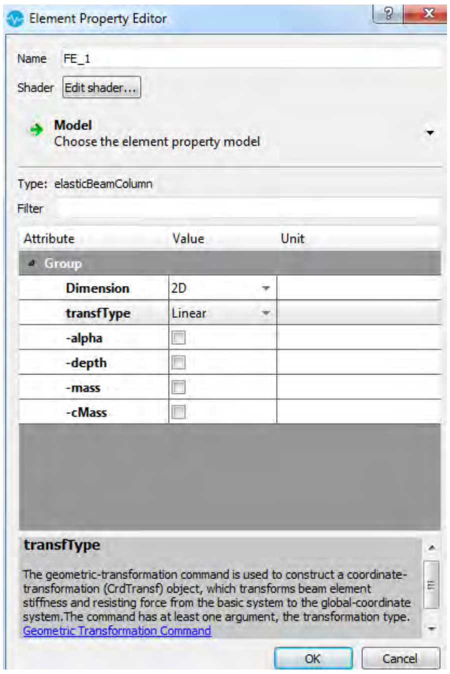


Figure 130. Element Property Editor of an elasticBeamColumn

Click **OK** to confirm the settings. The new element will appear under **Element Properties** > **beam_column_element** (in this example) on the Work Tree.

It is possible to modify the new element at any time. *Click* on **Edit Element property** from the Toolbar and select the item from the list, or *Right-click* on the item and select **Edit** on the Work Tree.
NOTE: Do not forget to mesh the geometry to definitively assign the Element properties. *Click* **Mesh** > **Build mesh** from the Toolbar.

Example: Draw a Frame Structure, define, and assign physical properties and finite elements.

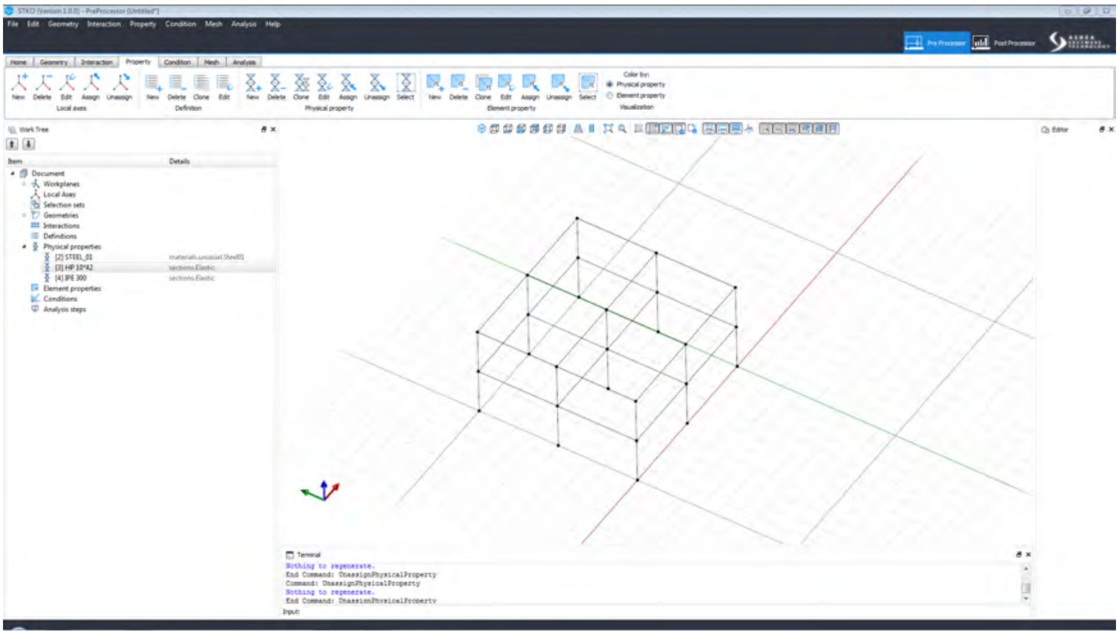


Figure 131. Example of a Frame Structure

Choose the command **Property** > **New physical property** from the Toolbar, or by directly *Right-clicking* **Physical properties** > **Add** on the Work Tree Panel. **Edit shader** to define the appearance of geometry, then select, for example, a **uniaxial material**, like **Concrete01** shown in the example.

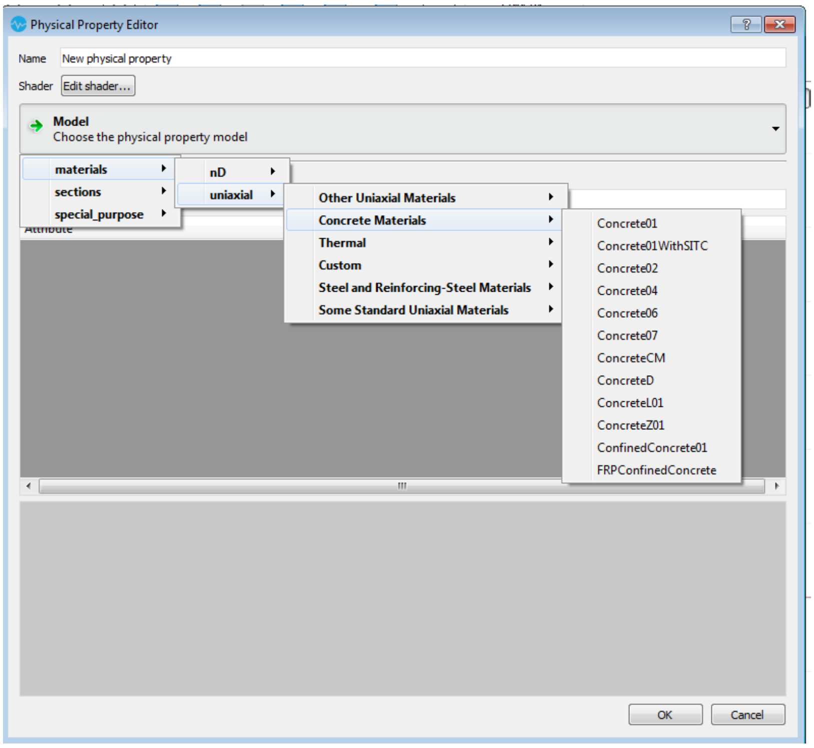
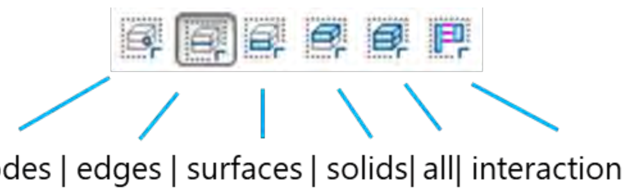


Figure 132. Example of creation of Uniaxial Material

Assign the new Physical property to the geometries with the aid of selection boxes in the Quick-Access menu. Selecting the buttons controls the different types of entities that can be selected.



After assigning the new physical property to the geometries, the color of the geometries will change, as shown in the figure.

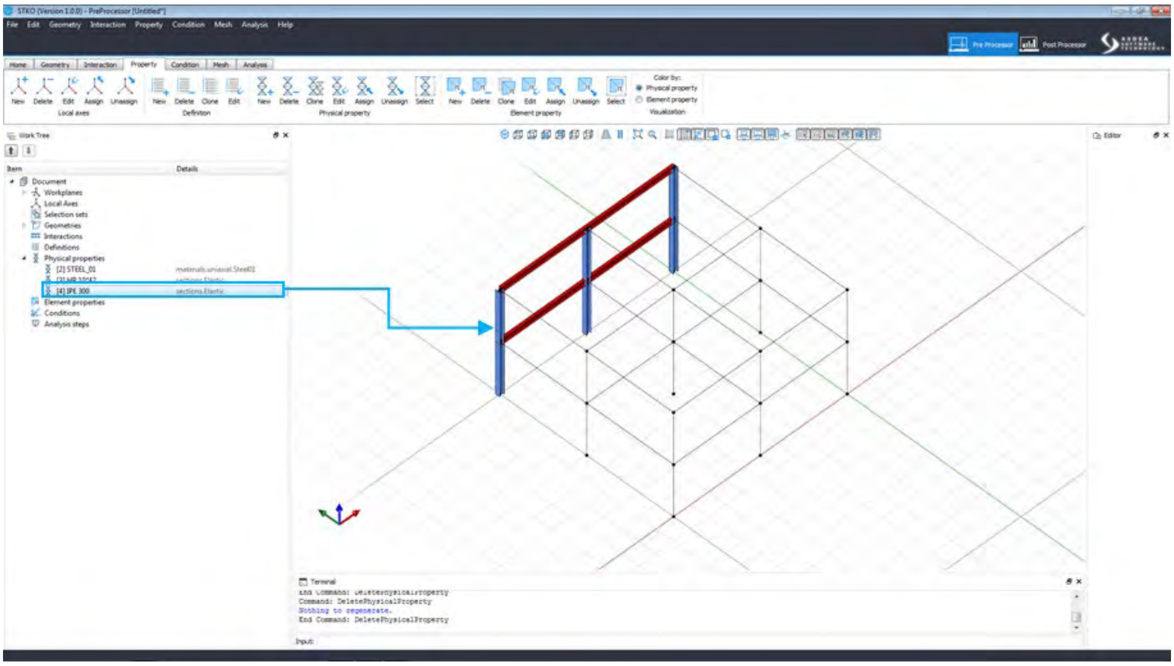


Figure 133. Example of assignment of uniaxial material to the columns

Each New Physical Property will have a different color.
Now, add a **New element property** to assign to the geometries. Select the geometry desired, then *click* **Assign element property** to attribute the new element to the geometry. This can also be completed by directly *Right-clicking* on the element in the Work Tree and *clicking* **Assign.** The model is ready to be meshed and exported. Select **Mesh** > **Global seed** > Choose the "Type" and "Division", and then select **Assign**.

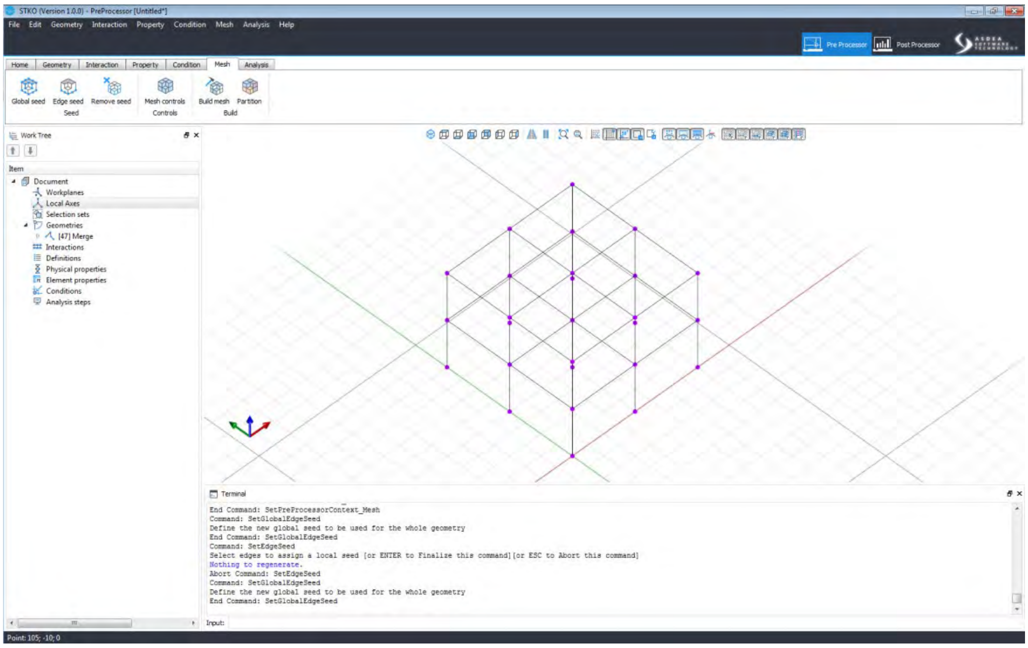


Figure 134. Mesh of the frame structure

It is also possible to assign a **Condition** model, as explained in [§ Defining and Assigning Boundary Conditions and Loads]().

2.5.1. Special Purpose for Element Properties

Like the Special Purpose for the Physical Properties, STKO can create a Special Purpose for Element Properties (**HingedBeamColumn**), which automatically applies the zeroLength to the element that will generate **extra nodes**.

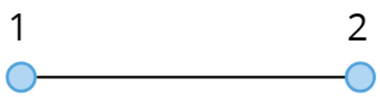


Figure 135. Scheme of a beam

Figure 136. Scheme of a beam with extra nodes

To create an Element Property, *click* on **Property** > **New element property** from the Toolbar, or by directly *Right-clicking* **Element Property** > **Add** on the Work Tree Panel.
Then, choose **Model** > **zero_length_elements** > **ZeroLength** and repeat the process for other elements.

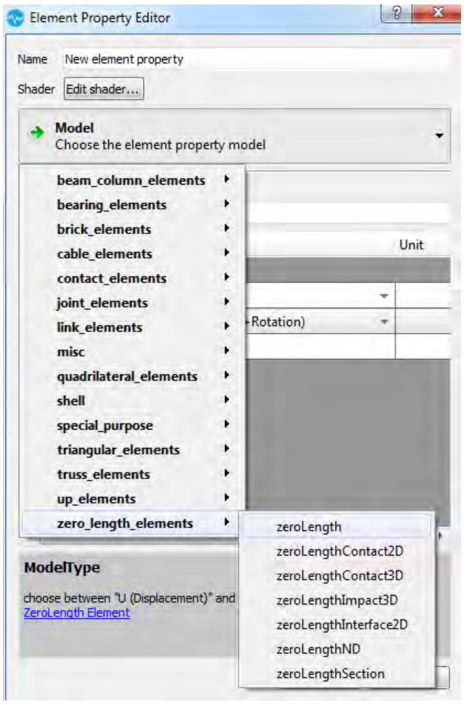


Figure 137. New element property - zeroLength

After creating the elements, it is possible to create a **Special Purpose** by *clicking* **Property** > **New element property** from the Toolbar, or by directly *Right-clicking* on **Element properties** > **Add** from the Work Tree Panel. Then select **Model** > **special_purpose** > **HingedBeam**.

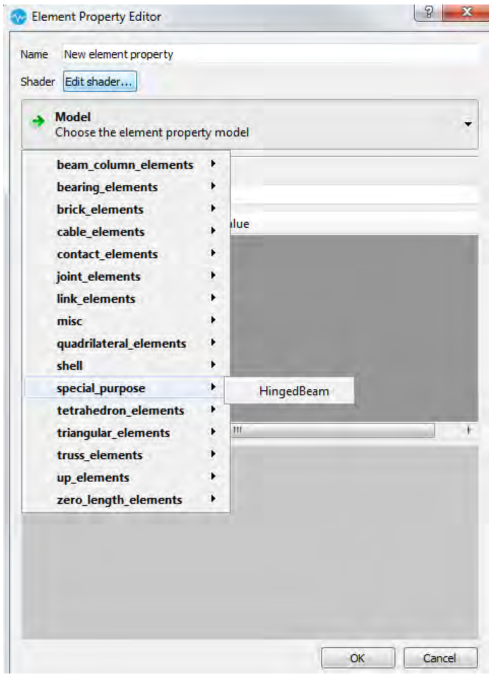


Figure 138. New element property - special_purpose

Insert the Beam or Column Element

Insert zeroLength previously created.

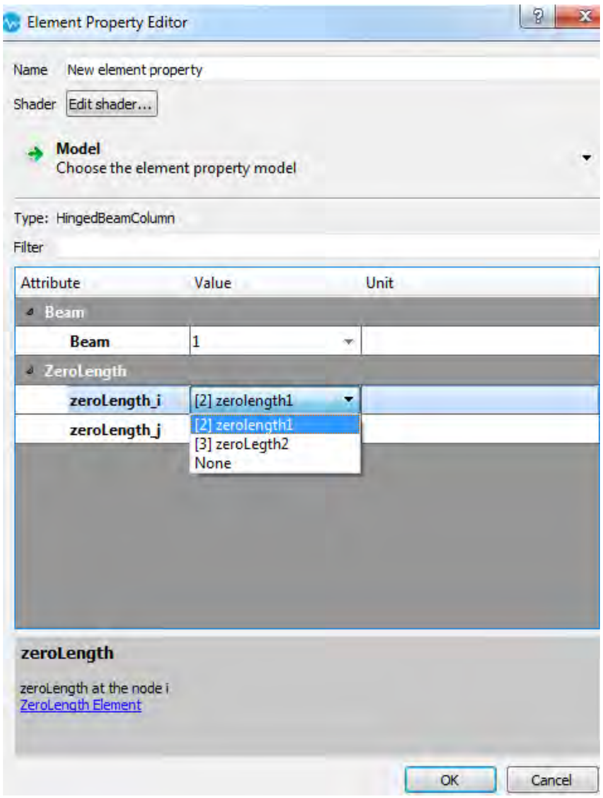


Figure 139. New element property - HingedBeamColumn

After inserting **zeroLength_i** and **zeroLength_j**, *Click* **OK** to confirm the settings. Assign the new **Special Purpose** to the Beam or Column.

2.6. Defining and Assigning New Definition

Choose the command **Property** > **New Definition** from the Toolbar, or by directly *right-clicking* **Definition** > **Add** on the Work Tree.

A new interface will appear, as shown in the figure. The user can choose between different **frictionModels,** like Coulomb, VelDepMultiLinear, etc. With this interface, the user can also choose between different **timeSeries,** like Linear, constant, Path, etc. (as shown in the example below). **Definitions** are general purposes that are not assigned to the geometries or to the interactions. For this reason, Definitions do not allow the command "Assign".

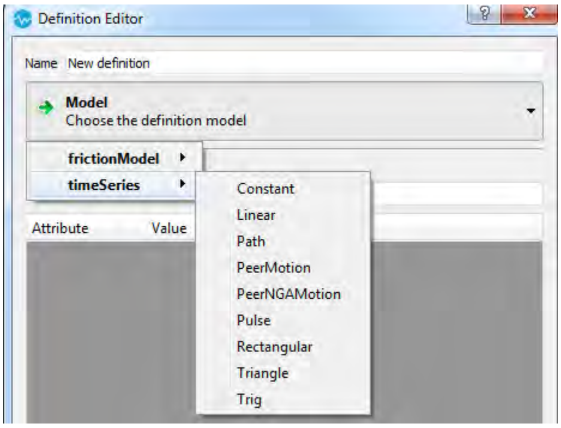


Figure 140. Definition Editor

Choose **frictionModel** or **timeSeries** and assign values to the table, like in this example (Multi-Linear Velocity Dependent Friction) from the frictionModel.

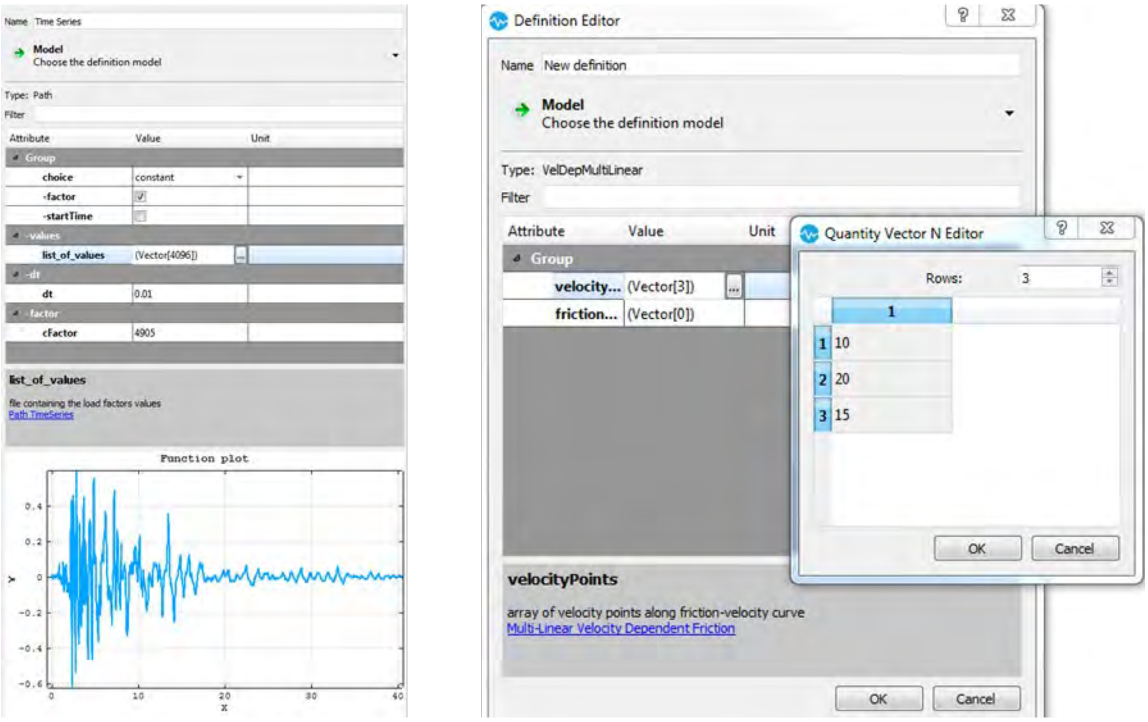


Figure 141.Definition editor Path TimeSeries (left) and VelDepMultiLinear (right)

Every group that belongs to each definition model contains a link to the OpenSees website (http://opensees.berkeley.edu/wiki/index.php/Main_Page) to the description of each item.

2.7. Defining and Assigning Boundary Conditions and Loads

From the Preprocessing interface, choose the command **Condition** > **New Condition** from the Toolbar. Here, the user can add a new Boundary Condition to assign to the geometry.

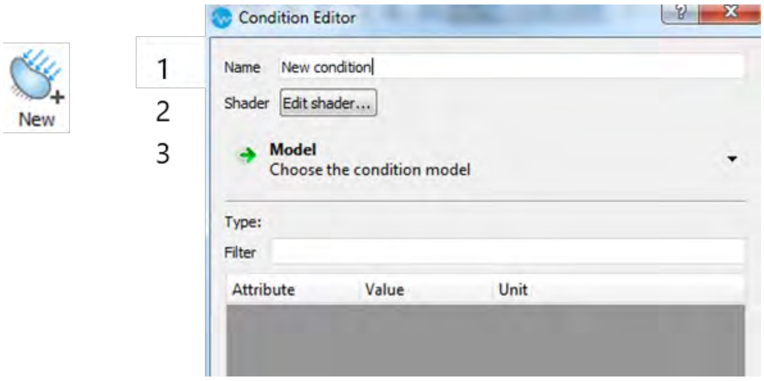


Figure 142.Condition editor

Choose a **Name** to attribute to the Boundary Condition and customize its appearance in the **Visual Material Editor** by *clicking* **Edit shader**.

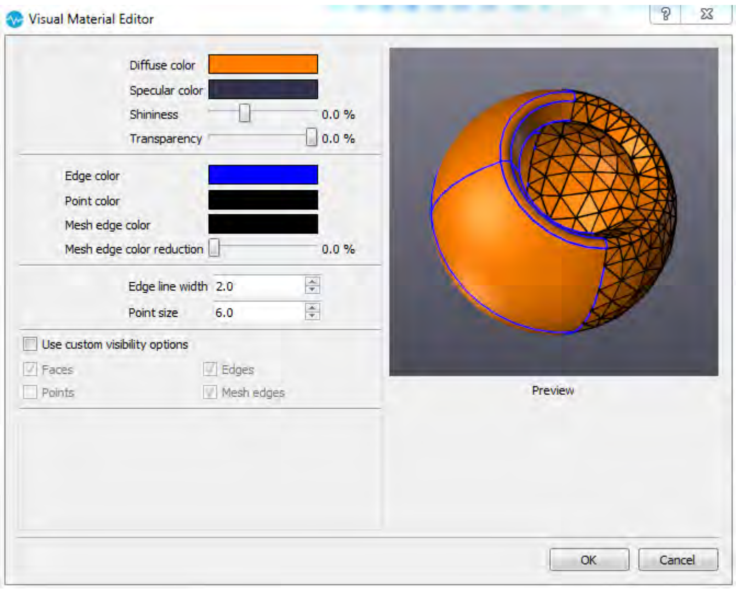


Figure 143. Shader

Choose the **Diffuse** and **Specular colors** and give **Shininess** and **Transparency** to the condition. Also, choose the **Edge** and **Point color**s with different widths and sizes. **Mesh edge color** and **Mesh edge color reduction** are also customizable.

Click **Ok** to confirm the **Visual Material Editor**.

Click Model to select the Constraints, Ground_Motion, Loads, or Mass.

To assign Constraints to the model:

Choose the command Condition > New Condition from Toolbar.

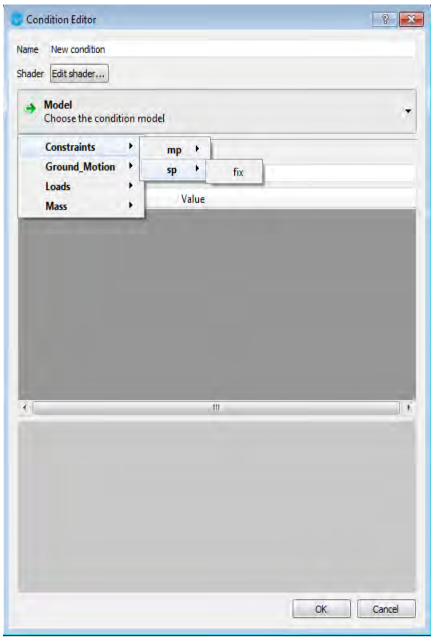


Figure 144. Constraints window editor

Select Constraints from the drop-down model menu and choose between mp and sp constraints.
After selecting a constraint, in this example, the sp constraint, *click* **sp> fix** and an editing window will appear.
In this section, the user can set the degree of freedom to assign to an element and in which direction it will act.
NOTE: conditions must be attributed to the part of the geometry by selecting it while the editing window is still open. Select the geometry part, which will turn red [with the editing window still open], meaning that STKO is assigning the condition to the selected part. Then, *click* OK to assign the condition to the selected geometry.
To modify the assigned geometry (vertex, edge, or surface), edit the condition and select the new geometry.

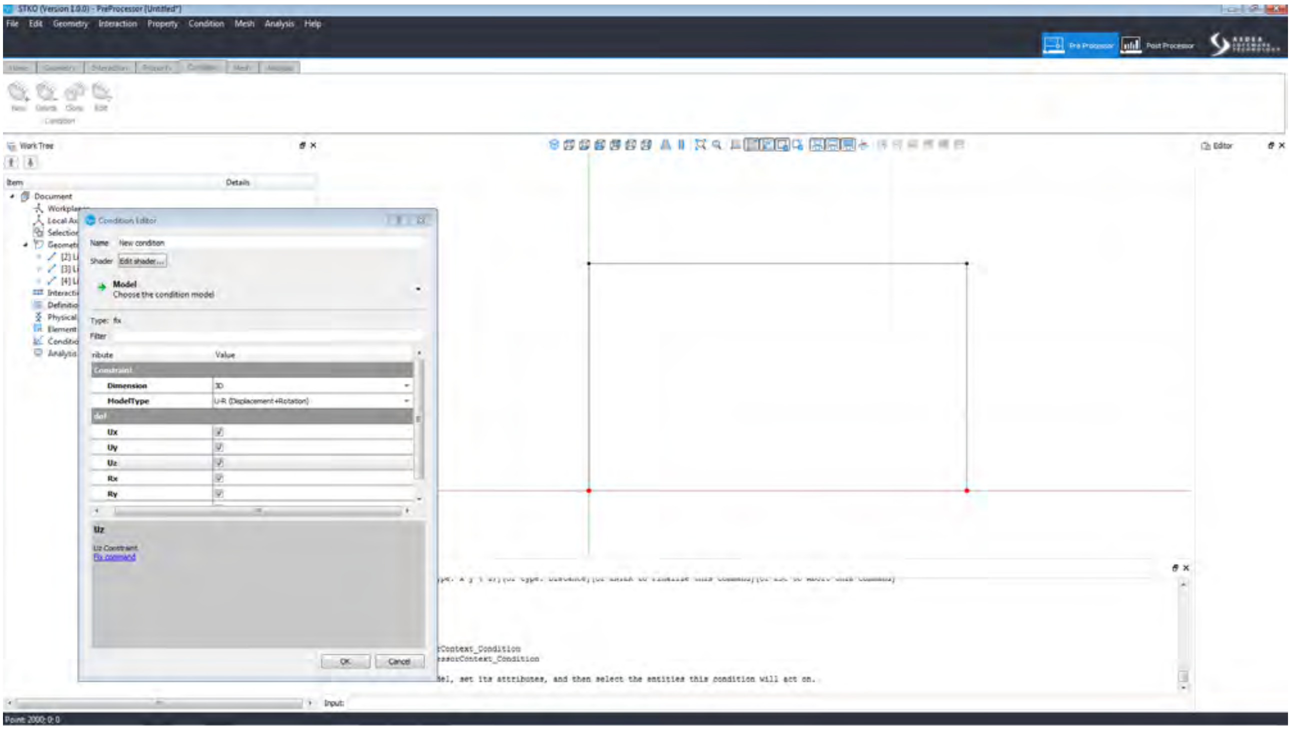


Figure 145. Example of attribution of constraints

To define **Loads**, after selecting loads from the drop-down menu, the user must choose between:

- force > edgeCouple, nodeforce, faceForce etc.
- eleLoad > eleLoad_beamuniform*(see figure below)*
- sp > prescribed nodal values

Once the load type is defined, the user can specify the **mode** of application and the quantity of **force** in all directions (in 2D and 3D).
Like constraints, the loads must be attributed to an element by selecting it while the editing window is still open.

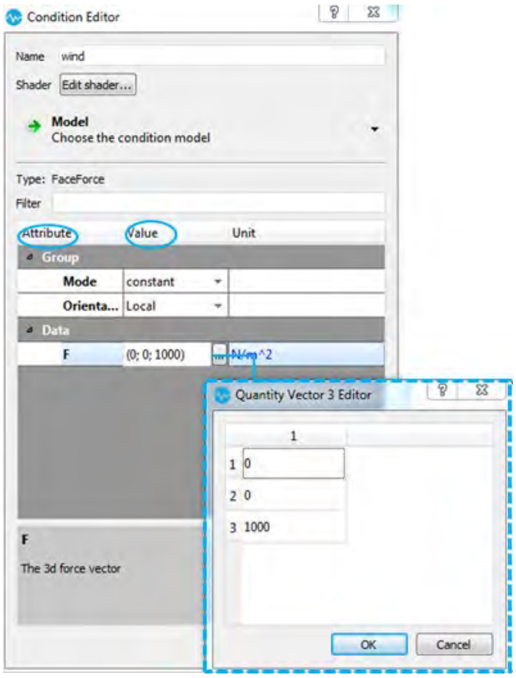


Figure 146.Load definition

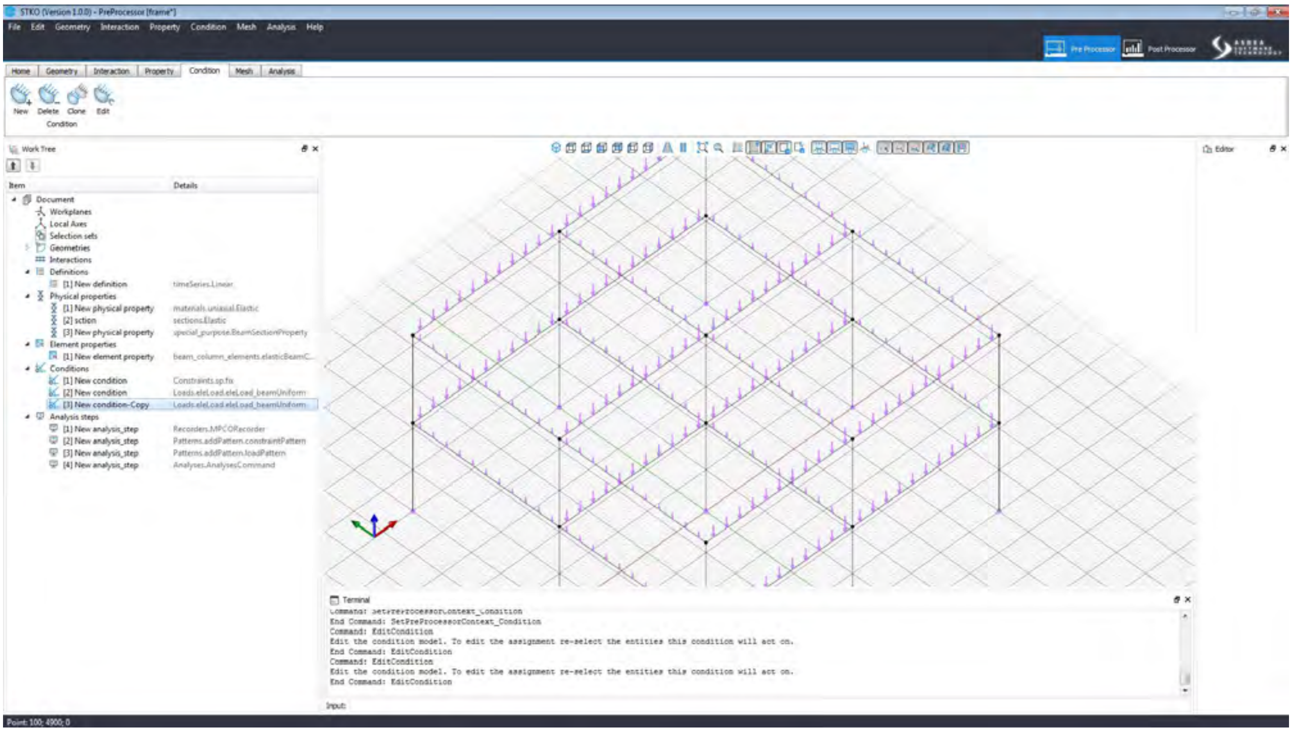


Figure 147.Visualization of model load attribution, eleLoad_beamuniform

For example, to assign a **load** to the model:

Choose the command **Condition** > **New Condition** from the Toolbar.

Type a **Name** to attribute to the Load and customize its appearance using the **Visual Material Editor**. Then, *Click* **Model** and Select **Loads** > **Force** > **NodeForce**.
Edit it, and assign it to an element.

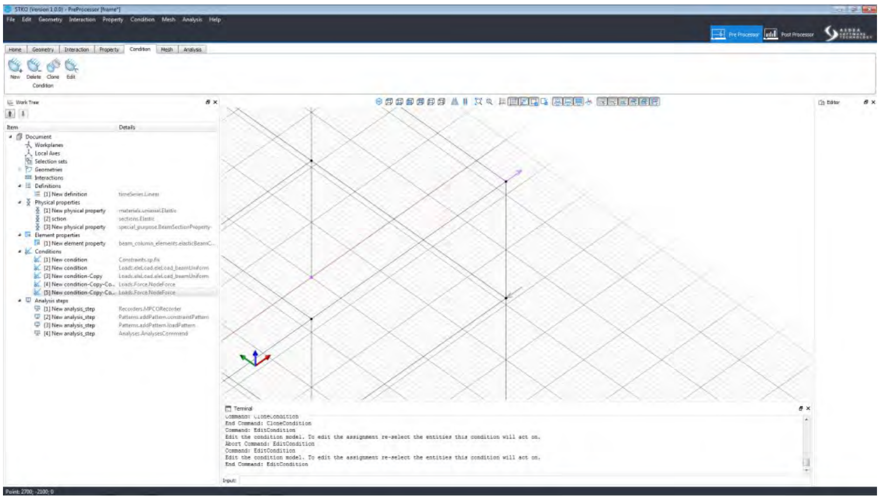


Figure 148. Visualization of positive and negative Nodal Force

To **Edit**, **Delete**, or **Clone** the new condition, *Click* **Condition** and select the corresponding command from the Toolbar.

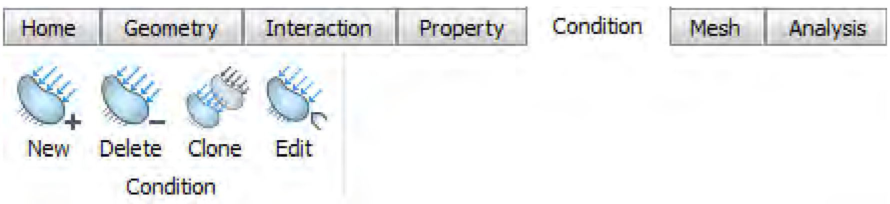


Figure 149. New, Delete, Clone, Edit condition from Condition section of the main Toolbar

Or directly *Right-click* the item under **Conditions** on the Work Tree Panel and select the command.
In addition to FaceForce load, Opensees provides other types of loads as shown in the figure.

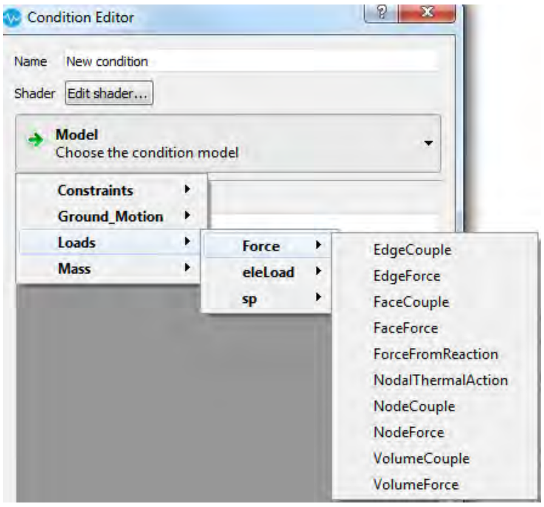


Figure 150. STKO loads

Despite different available applications, Opensees only processes nodal forces. For this reason, **STKO** displays these Forces on nodes by means of *lumping*.

Lumping is the transposition of a uniform distributed load into **nodal loads**. To better understand this "transposition", see the following example:

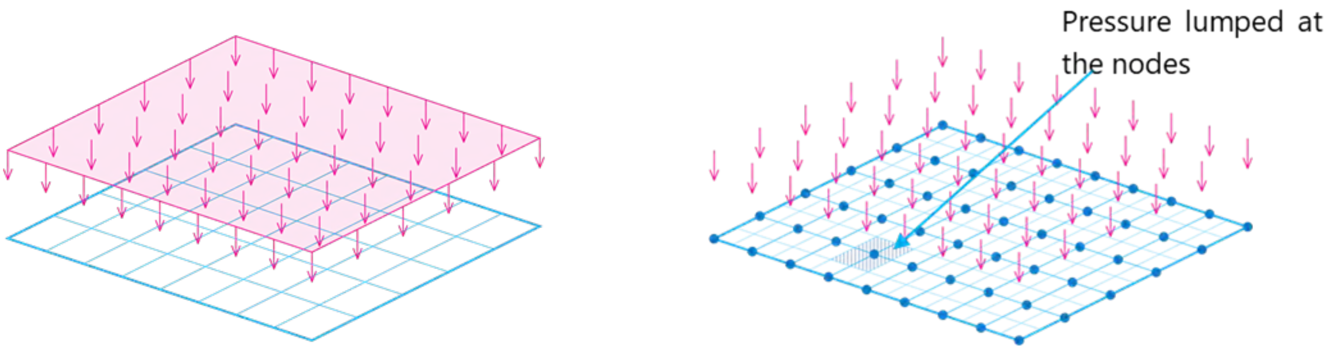


Figure 151. Uniform distributed load over a meshed surface (left) and example of lumping (right)

Before we saw a FaceForce example on a 3D solid, with constant mode, a local axis orientation, and a 3D force vector.
The user may also choose a **function mode** of every Force type from the Condition Editor.

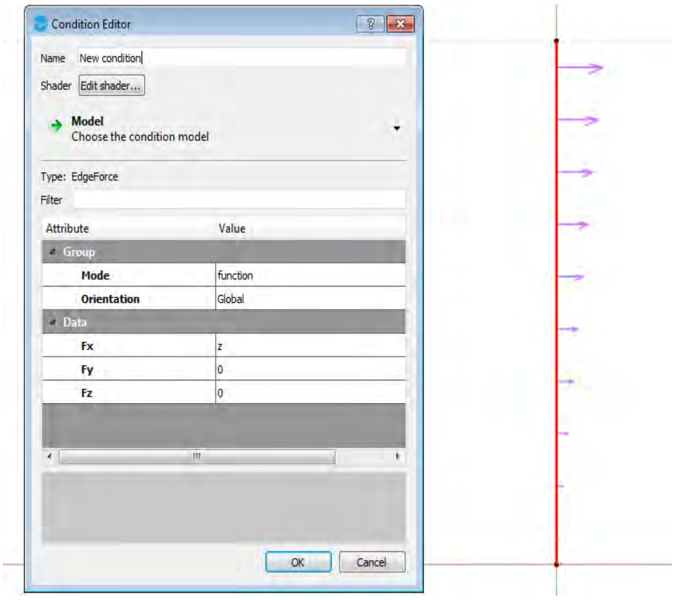


Figure 152. Example of EdgeForce with function mode over a Beam

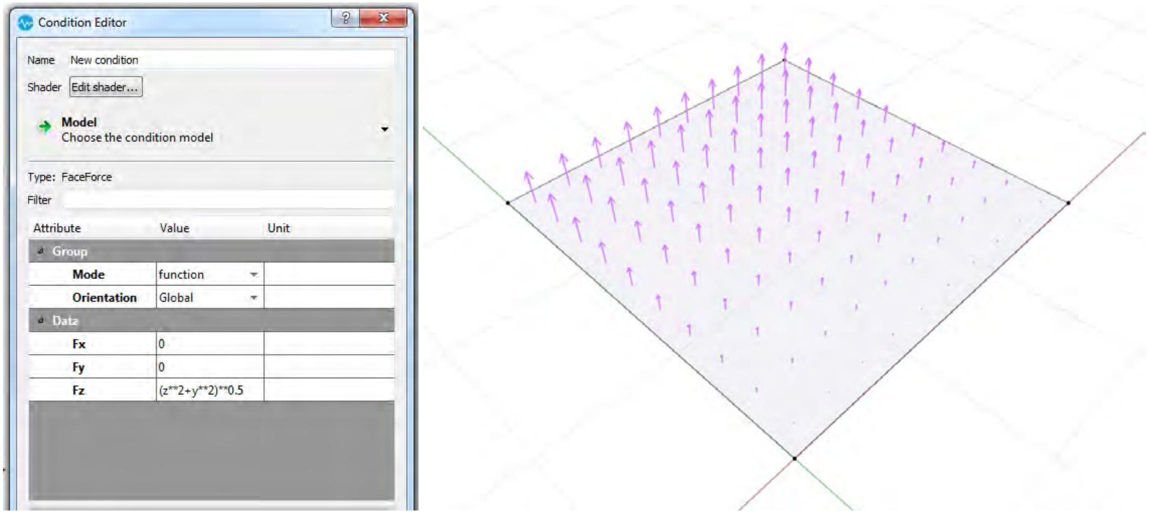


Figure 153. Example of FaceForce with function mode over a Surface

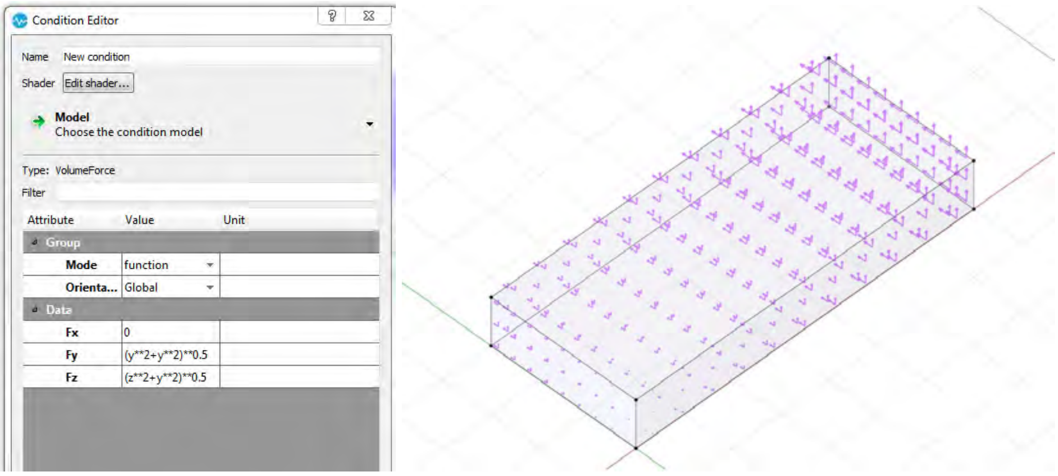


Figure 154. Example of VolumeForce with function mode over a Solid 3D

2.7.1. BeamSolidCoupling

BeamSolidCoupling is a constraint element that unifies a solid element (n_{dm}= 3; n_{df}= 6) with a beam element (n_{dm}= 3; n_{df}= 3) through **Interactions** (§2.5 Interaction Modeling). These interactions will create two mp constraints: **EqualDOF** and a **RigidLink**. The equalDOF links the solid nodes to extra nodes and the RigidLink connects the extra nodes to the beam node. The beam node is the constrained node of the Rigidlink and its constrained nodes are the extra nodes. Also, extra nodes are the retained nodes of the equalDOF and its constrained nodes are the solid nodes. To better understand the **beamSolidCoupling** command, see the following sketch.

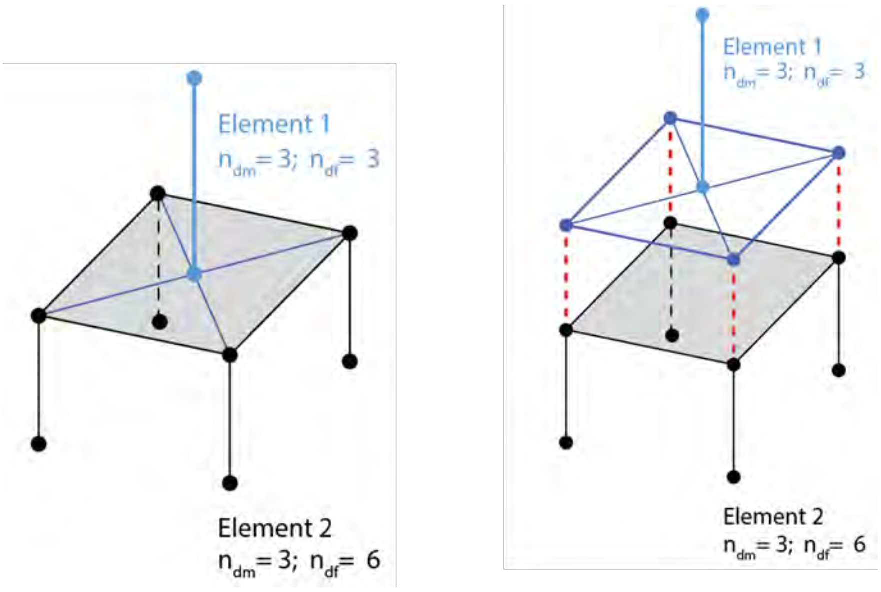


Figure 155. Interactions between a solid and a beam element

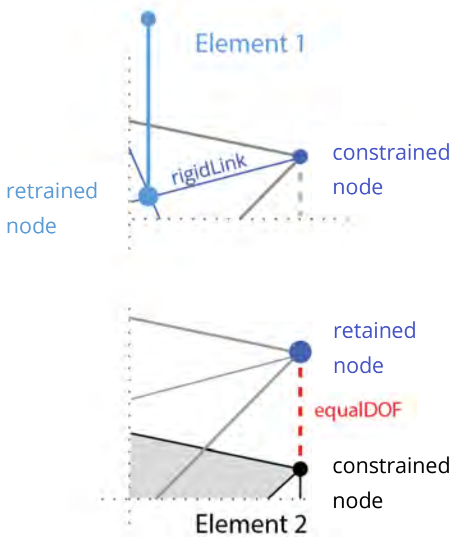


Figure 156. Identification of the retained nodes and constrained nodes in the RigidLink and EqualDOF

2.7.2. BeamToSolidBarSlip

BeamToSolidBarSlip is a tool used to connect a beam element with 6 DOFs to the surrounding solid's faces. The displacement DOFs of the beam in the local transversal directions are coupled with those of the solid with an EqualDOF constraint. The displacement DOF in the local axial direction of the beam is connected to the one of the solid with a zeroLength element, which is equipped with a nonlinear uniaxial material that simulates the nonlinear relationship of the bar-slip.

The **BeamToSolidBarSlip** is not a component built-in to OpenSees. To resolve this, STKO creates two extra nodes with six DOFs located in the same position of the solid node. One of them is connected to the beam node with a rigidLink. The second one is connected to the solid node with an EqualDOF in all three translational DOFs. Then, the first node and the second node are connected with an EqualDOF in all DOFs except the one along the beam's axial direction. Finally, STKO creates a zeroLength element between the first and the second node with a nonlinear uniaxial material along the beam's axial direction to simulate the bar-slip nonlinear relationship.

Note: When the beam's local directions are aligned with the global coordinate system, STKO uses EqualDOF multipoint constraints (as described above) to create the kinematic connections. However, when the beam's local directions are not aligned with the global coordinate system, EqualDOF constraints cannot be used. In that case, STKO uses an equivalent stiff uniaxial material the stiffness of which is the K penalty value. Note that this value should be stiff enough to enforce the constraint but not too stiff, otherwise the system will be ill-posed. For example, you can choose a value that is a couple of orders of magnitude larger than the bar's axial stiffness.

Node Constrained (Solid)

Figure 157. BeamToSolidBarSlip Interaction Diagram

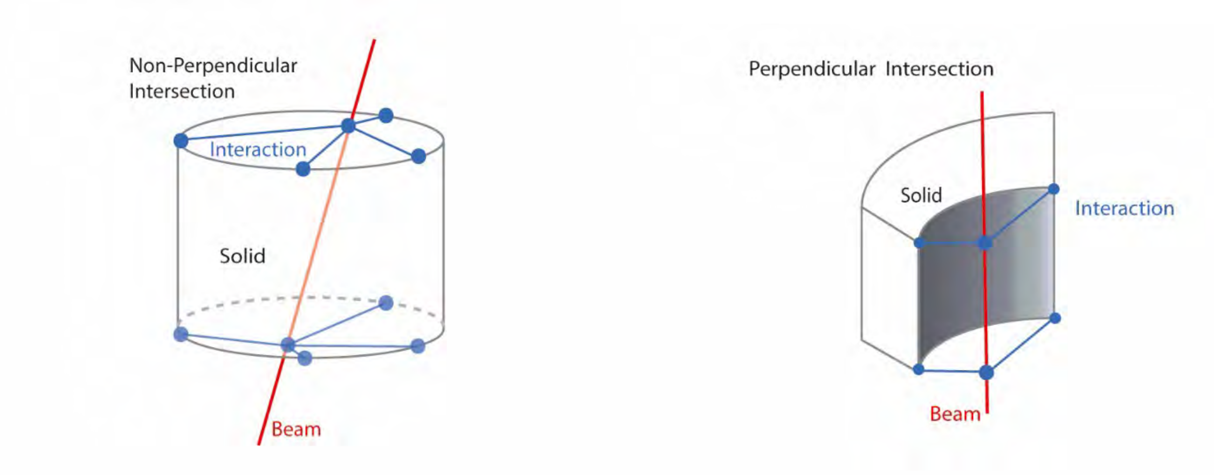


Figure 158. Beam Directions using BeamToBarSlip

2.7.3. Automatic RC Beam-Column Joint Element with Scissor Model

The Automatic RC Beam-Column Joint Element with Scissor Model[1] is a tool that allows users to create a beam-column joint assembly that follows the scissor model by simply assigning a physical and an element property to the vertices of the frame model.

[1] Alath S, Kunnath SK. (1995) "Modeling inelastic shear deformations in RC beam–column joints". Engineering mechanics proceedings of 10th conference, May 21–24, University of Colorado at Boulder, Boulder, Colorado, vol. 2. New York: ASCE: p. 822–5.

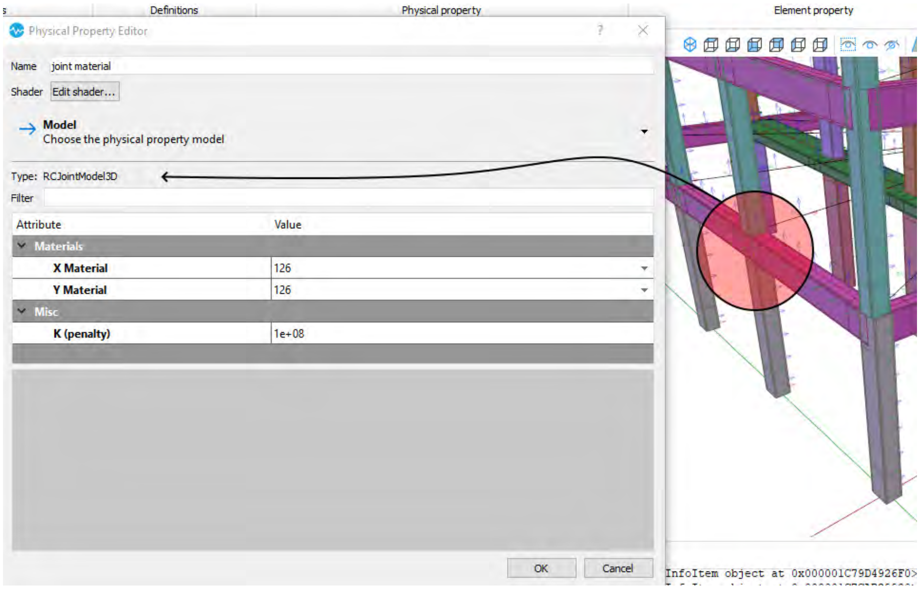
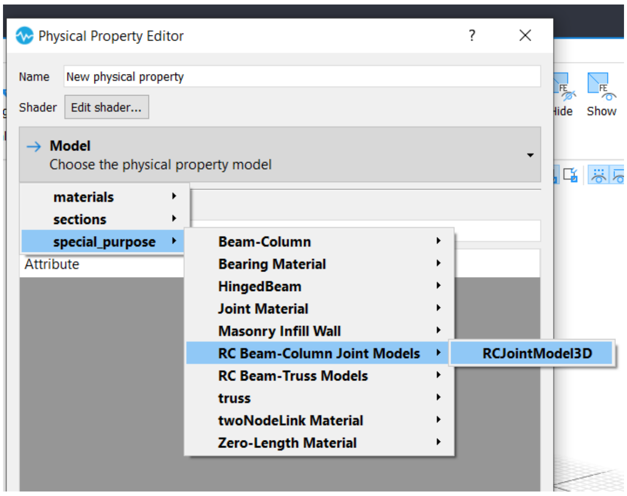


Figure 159. Automatic RC Beam-Column Joint Element with Scissor Model

To access the tool, add a new physical property, then click Model>special_purpose>RC Beam-Column Joint Models>RCJointModel3D



2.7.4. ForcefromReaction

STKO enables the user to create more than one Analysis step. Once the first Analysis has been concluded, it is possible to read all resultant forces on nodes and apply them like Force. This means that a **reaction may be converted into a Force** (ForcefromReaction command). The following example shows two Analysis steps:

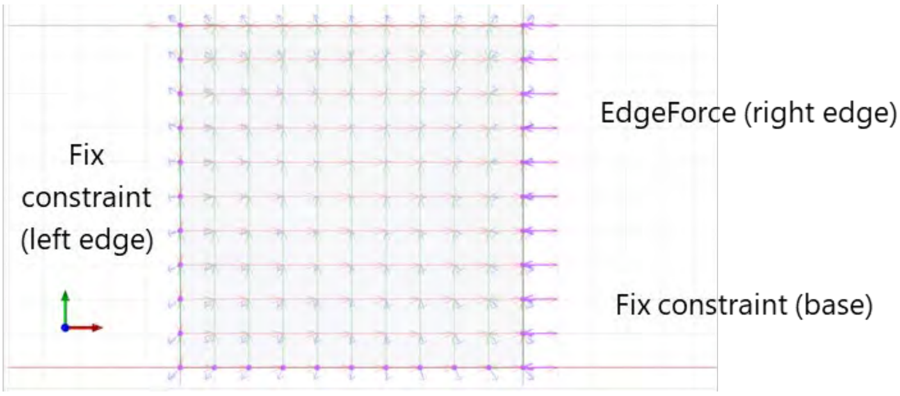


Figure 160. First Analysis Step

Analysis Step 1:

After defining **Constant timeSeries (1)** from Definitions. Attribute **Physical properties (2)** (material elastic_isotropic, plate_fiber etc.) and **Element properties (3)** (ShellMITC4). Then assign constraints to the shell: **fix constraint (4)** to the base and fix constraint to the left edge of the shell. In addition, assign an **EdgeForce (5)** to the right.

Then, define a **Recorder (6)** checking all the nodal results to display after. Before creating a new Analysis step, do not forget to set **Patterns (7)**, two for constraints and one for load. Setting the first sp constraint (fix constraint on the base) in the first **constraintPattern** and the second sp constraint (fix constraint on the left edge) in the second **constraintPattern**. Then creating a third Pattern for the load and inserting the edgeForce in load and the timeSeries previously defined in tsTag. Finally setting the first **Static Analysis Step (8)** specifying the number of analysis steps to perform (in this case **numIncr= 1**).

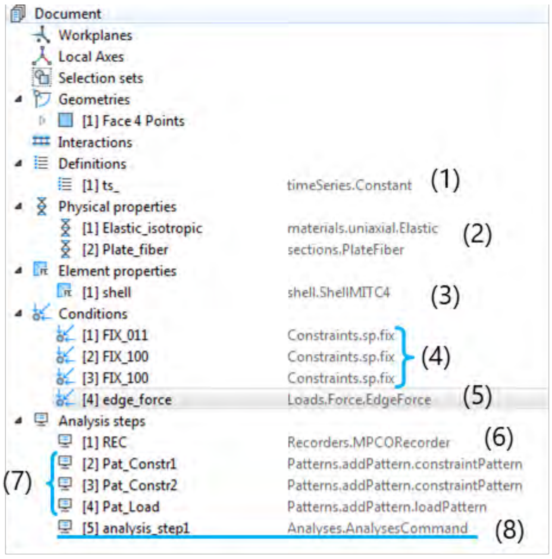


Figure 161. Analysis Steps

Analysis Step 2:

Then, define a new fix constraint to apply to the right edge of the shell instead of the force. Then creating a new Force: **ForcefromReaction** (9).

Choose the command **New condition** from the main Toolbar > **Model** > **Load** > **ForcefromReaction** and set the Dimension (2D or 3D). Assign the Force to the left edge of the shell.

Use the **removePattern** (10) to remove Load Pattern and spContraint Pattern (only the fix constraint of the left edge) previously created. Once removing previously Patterns, creating new **Patterns**: one for the ForcefromReaction (LoadPattern) and one for the fix constraint (spConstraint Pattern). Applying the LoadPattern to the left edge and the spConstraint Pattern to the right edge, keeping the fix constraint to the base.

Finally setting the second **Static Analysis step** (11), specifying the number of analysis steps to perform (*numIncr ≠0* − in this case **numIncr= 1**−).

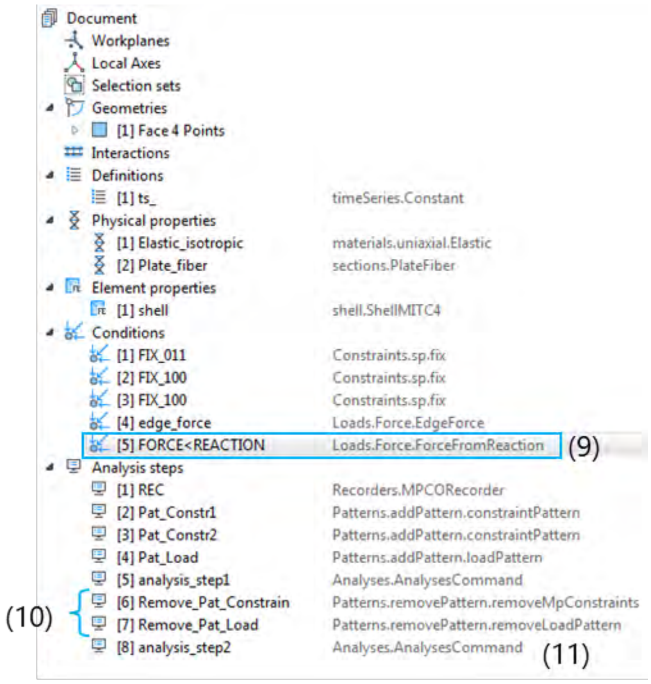


Figure 162. Analysis Step 2

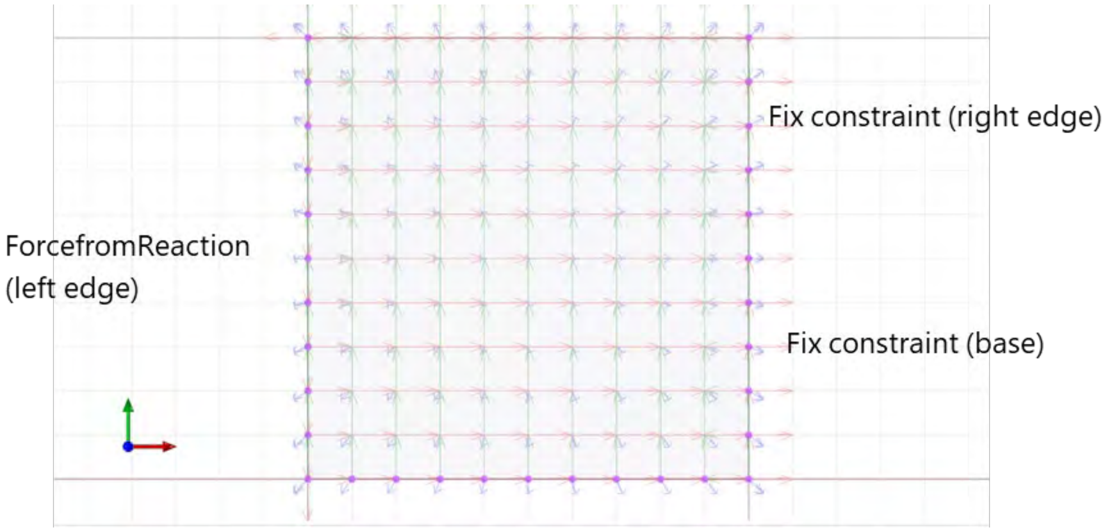


Figure 163. Second Analysis Step

Launch the Analysis to generate an .mpco file. Open the **STKO Postprocessor** interface and open the new Database (.mpco).

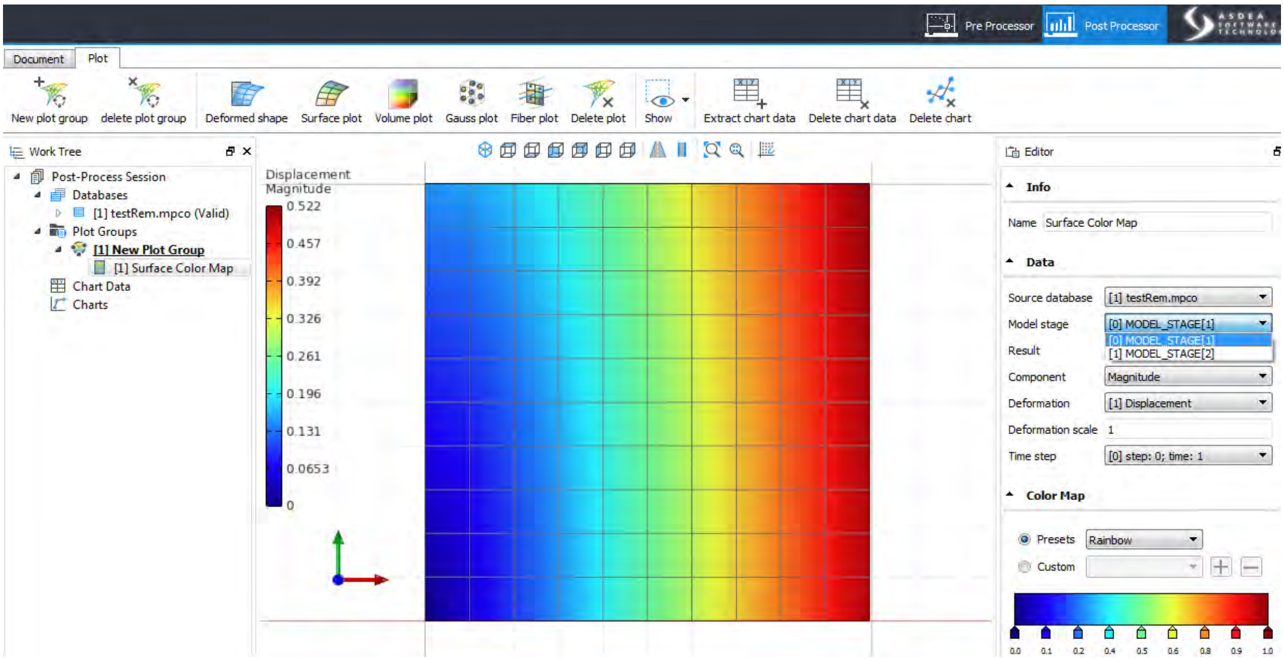


Figure 164. Post-processor

Then *right-click* **New Plot Group** and *click* **Surface Color Map**, or select **Surface Plot** form the Toolbar. In the editor menu, you can select **MODEL STAGE 1** or **MODEL STAGE 2** to see the output results.

2.7.5. DistributedLK

Choose the **new condition** command from the main Toolbar and then select **Model** > **Constraints** > **mp** > **DistributedLK2d** or **DistributedLK3D**.

This enables the user to insert two materials (an Axial material [default], and a Tangential material). This command respectively generates “extra” nodes and uniaxial material (Viscous) applied to the ZeroLength.

1. **“Extra” nodes**:

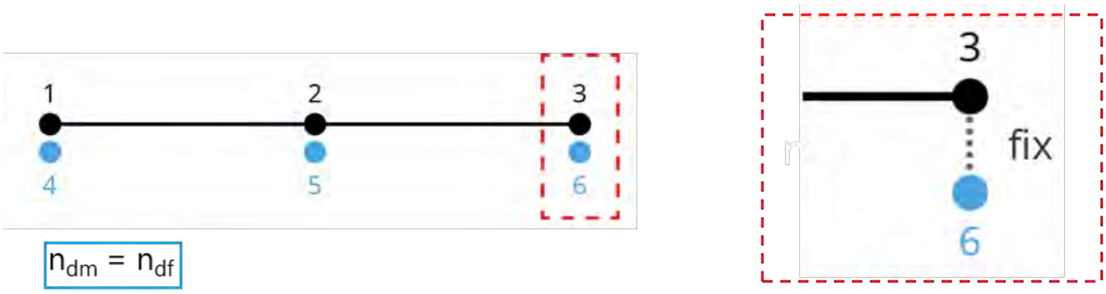


Figure 165. Fix constraint

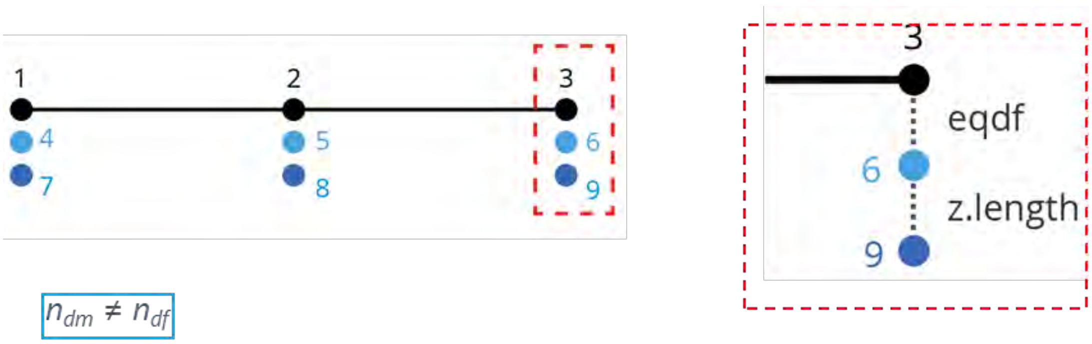


Figure 166. equalDOF+fix

2. Uniaxial **material (Viscous)** applied to the **ZeroLength**:

The following example explains the application of the **DistributedLK2D** command to an element.

After defining the **physical properties** (ElasticIsotropic material) and the **element properties** (SSPbrick element and VS3D4 element), to assign it to the geometry desired, choose the conditions to apply. Select **Condition** > **New condition** > **Model** > **Constraint** > **mp** > **DistributedLK2D** from the main Toolbar:

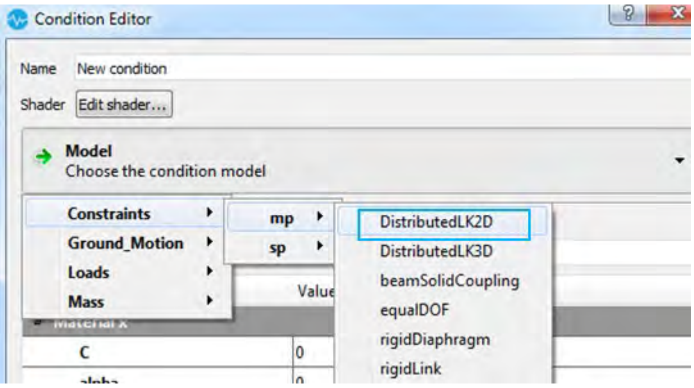


Figure 167. Condition editor of a DistributedLK2D

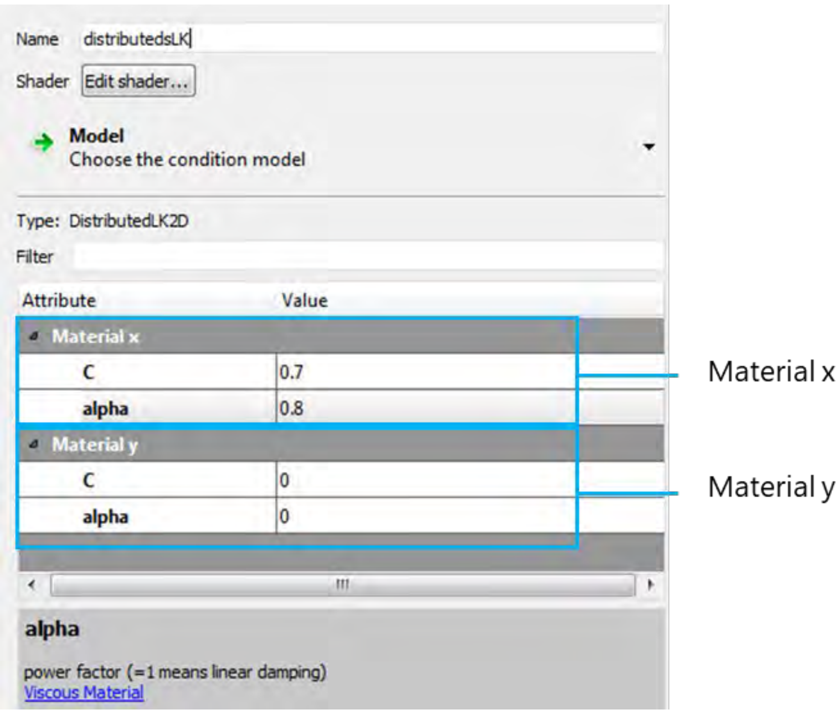


Figure 168. Assignment of axial material and tangential material

Select the edge and assign the **mp constraint**. *Click* **OK** to confirm the settings.
After that, add a new load. **Condition** > **New condition** > **Model** > **Loads** > **Force** > **FaceForce**. Set parameters and assign it to the element. Now create the **Timeseries** from **Definitions** (for example a **Trig** Timeseries) setting the period, the start time, and the end time.
Then, it is necessary to define a **Recorder** to write the nodal results, which can be viewed with the postprocessor. Before creating a new Analysis step, do not forget to set **Patterns**, one for the constraint and one for the load. Set the first mp constraint (for the edgeDistributedLK) in the first **constraintPattern** and then the **loadPattern** [selecting the **Timeseries** and the

FaceForce previously defined]. After that, it is possible to set the **Analysis Step** specifying the analysis Type (for example Transient) and all the parameters.

2.8. Meshing the Geometrical Model

Meshing a Geometrical Model is the most fundamental step in the 3D development process. **STKO** supports many Mesh Modeling both structured and unstructured algorithms. In this section, the user will learn the parts of a mesh and how to build 2D and 3D mesh models through Global and Local Mesh Controls.

Figure 169. Example of meshed solids

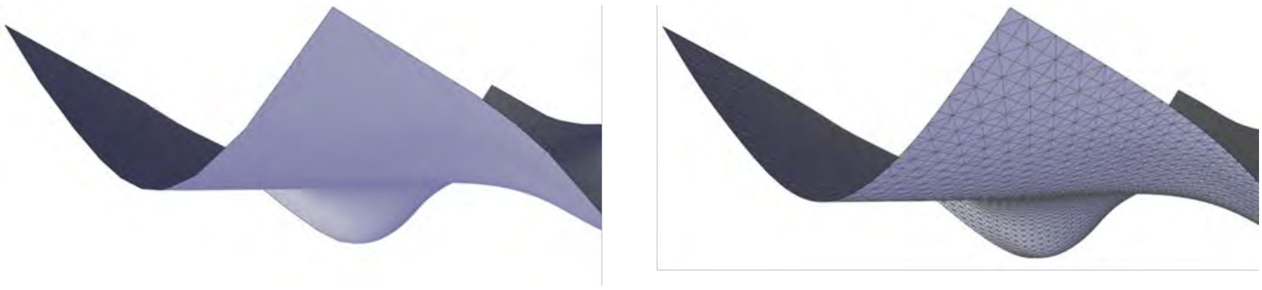


Figure 170. Example of meshed solids

2.8.1. Element Type

STKO uses a hybrid mesh generation approach that allows for complex 2D and 3D geometries to be optimally meshed by using different mesh elements. STKO supports 5 different Element

Types: Linear Type (2 or 3 nodes), Triangular Type (3 or 6 nodes), Quadrilateral Type (4, 8, or 9 nodes), Tetrahedral Type (4 or 10 nodes) and Hexahedral Type (8, 20, or 27 nodes –Serendipity).

Linear Type (2 or 3 nodes) is composed of two nodes at each extremity of the line. The quadratic form has an additional node in the middle of the line.

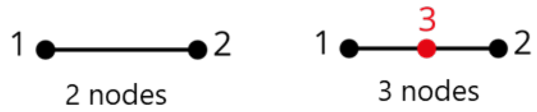


Figure 171. Linear Type with 2 and 3 nodes

Triangular Type is composed of 3 nodes located at the three geometric corners of the triangle. Six node **Triangle Type** has three additional nodes located in the middle of each of the three edges.

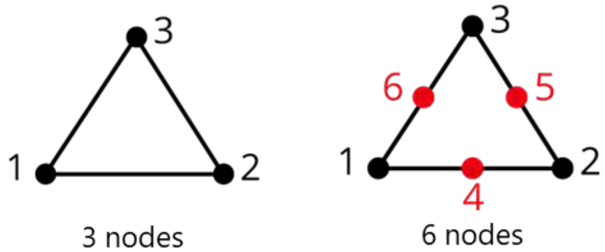


Figure 172. Triangulsr Type with 3 and 6 nodes

Quadrilateral Type is composed of 4 nodes located at the four geometric corners of the quadrangle. In addition, Quadrilateral Types with 8 and 9 nodes both have four mid-edge nodes; the 9 node version includes a mid-face node.

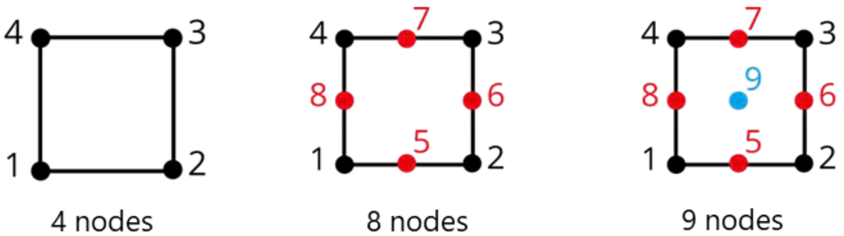


Figure 173. Quadrilateral Type with 4, 8 and 9 nodes

Tetrahedral Type is composed of four nodes located at the four geometric corners of the tetrahedron. Tetrahedral Type also has six additional nodes in the middle of each of the six edges.

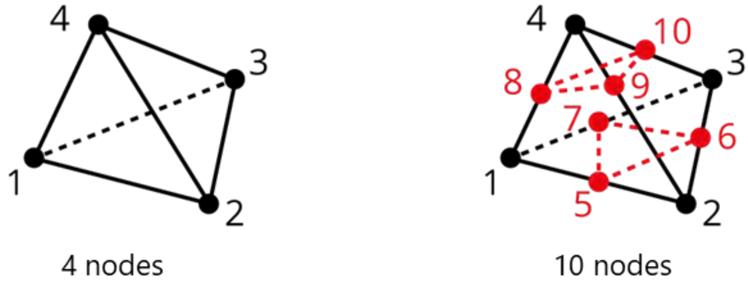


Figure 174. Tetrahedral Type with 4 and 10 nodes

Hexahedral Type (Serendipity) is composed of eight nodes located at the eight geometric corners of the hexahedron. In addition, Hexahedral Type with 20 nodes has a node at the middle of each of the twelve edges. Hexahedral Type with 27 nodes adds a node in the middle of each of the six faces, and one at the cell center.

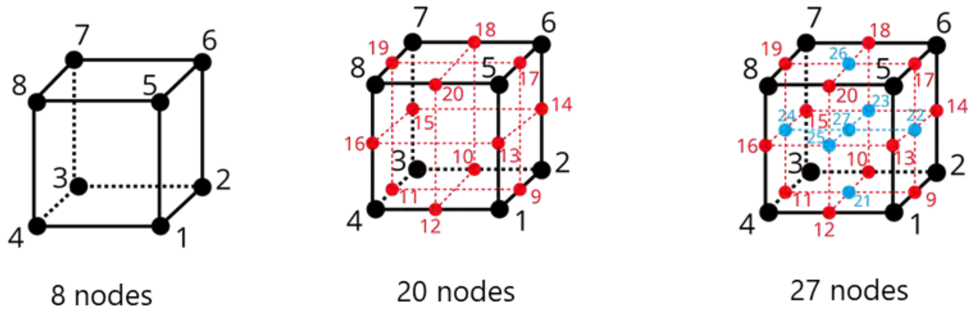


Figure 175. Hexahedral Type with 8, 20 and 27 nodes

2.8.2. Global and Local Mesh Controls

The user can manipulate the mesh settings globally [**Global seed**] and locally [**Edge seed**] using the commands located in the Mesh section of the Toolbar.

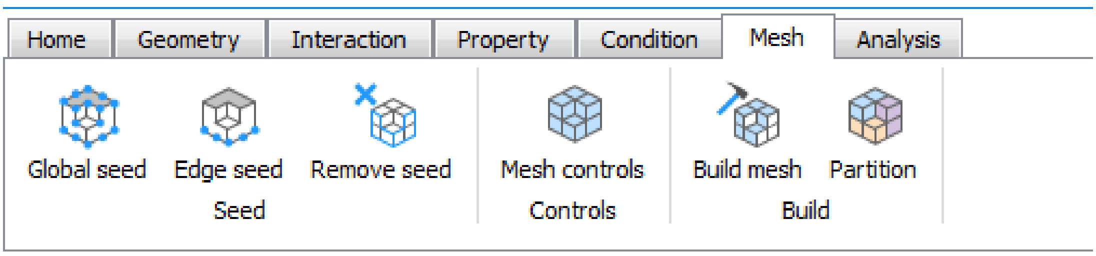


Figure 176. Global and Local Mesh Control in Mesh section of the main toolbar

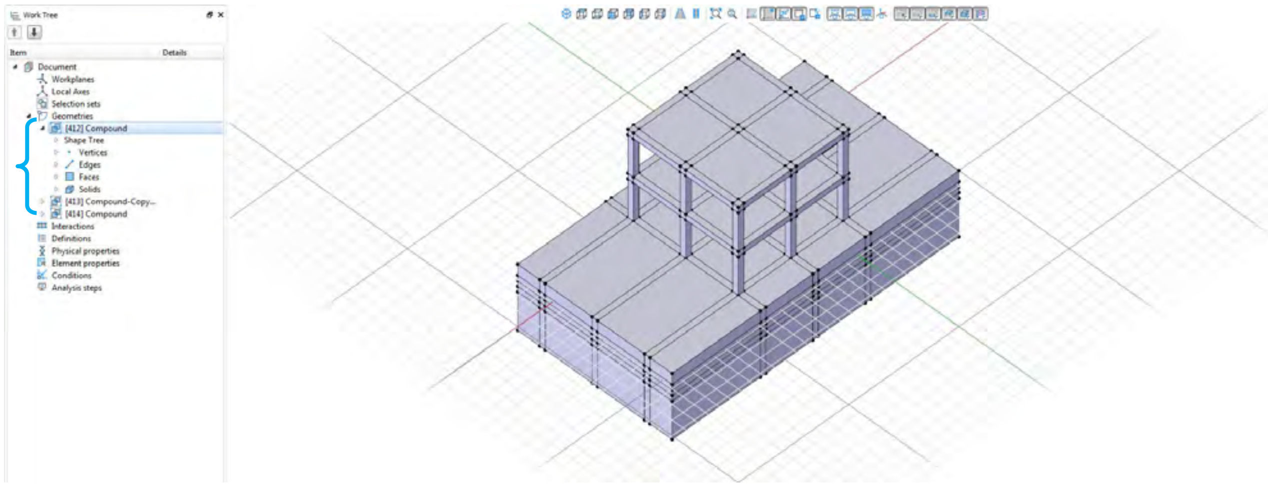


Figure 177. The Work Tree shows all solid features of the model.

Before defining the mesh controls, connect all solids with the Merge command. *Click* **Geometry** > **Boolean** > **Merge** and select all geometries to join. A folder of the **[n] merged** geometries will appear in the Work Tree.

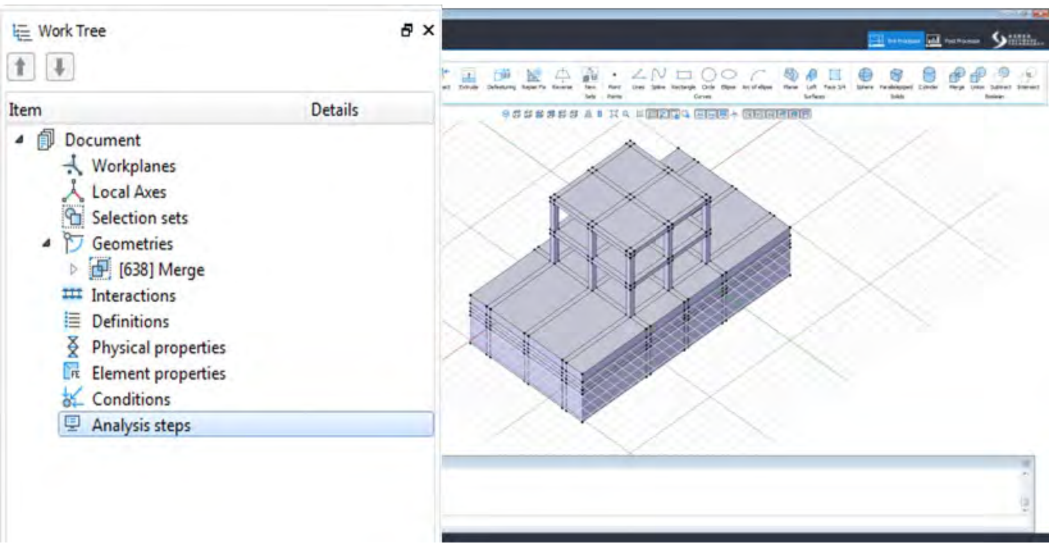


Figure 178. The work tree shows all the merged solids of the model

The commands **Global Seed**, **Edge Seed**, and **Remove Seed** in the **Mesh** section set the dimensions of finite elements.

The **Global Seed** command allows the user to make global adjustments in the meshing phase, and it includes **uniform biased** and **dual biased** distributions by **size** and **divisions**.

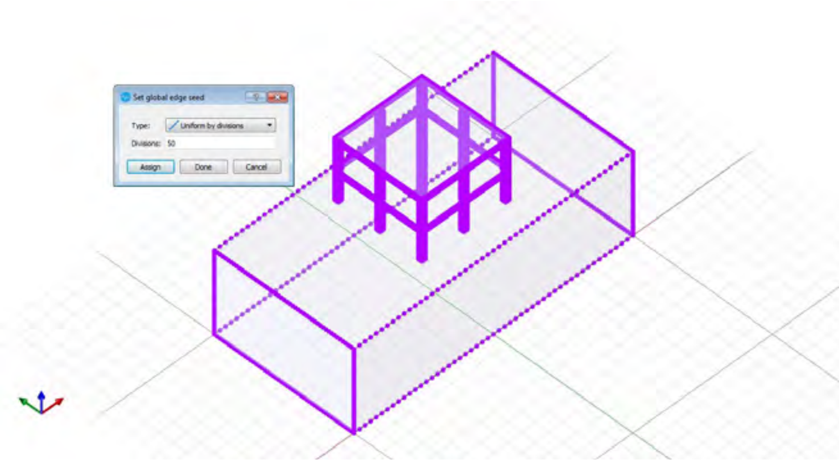


Figure 179. Example of Global seed with uniform type by divisions

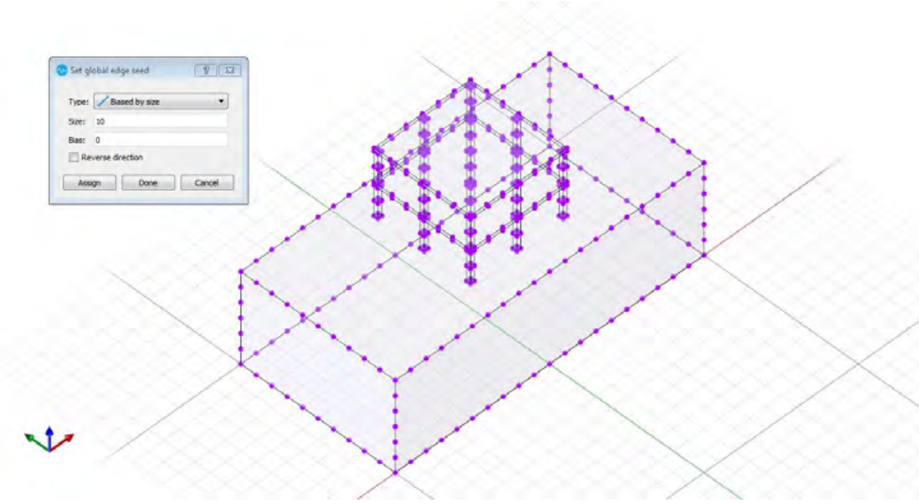


Figure 180. Example of Global seed with biased type by size

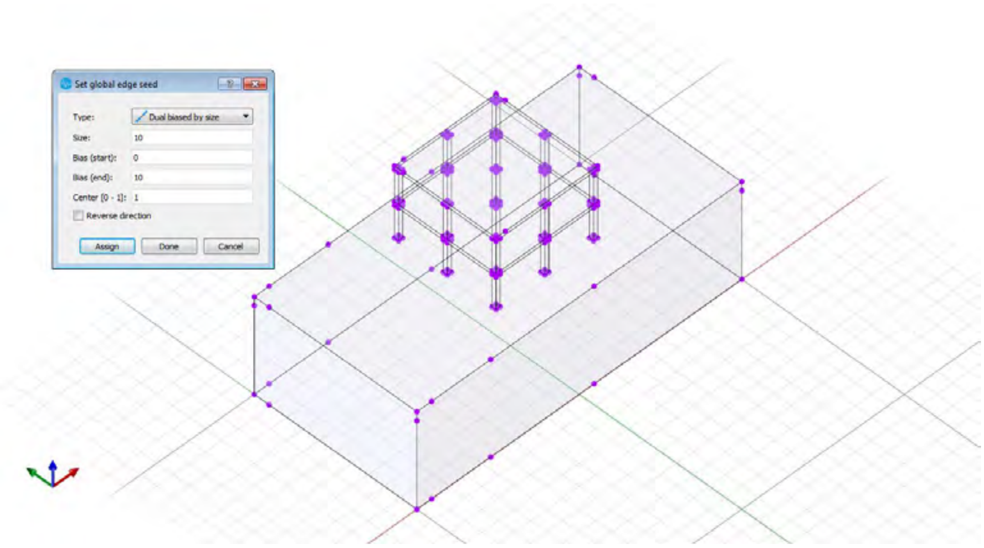


Figure 181. Example of Global seed with dual biased type by size

To confirm the Global Seed settings, *click* **Done**. Next, select **Mesh Controls** to define the elements of selection (Edges, Faces, or Solids), the Algorithm (Unstructured or Structured), the Topology (Tri/Tetra or Quad/Hexa for Faces/Volume), and the Order of the Mesh (Linear, Quadratic, or Quadratic/Serendipity). The colors in Color Mode on the Mesh Control editor will indicate the user's choice of mesh by changing the of the selected geometry.

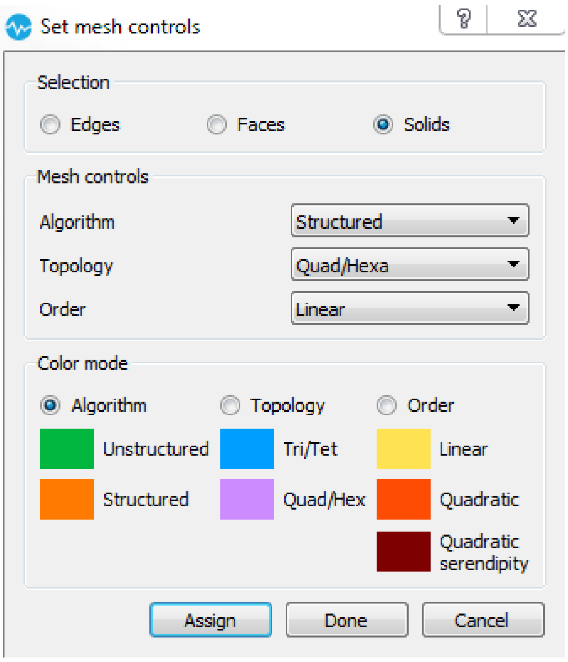


Figure 182. Set mesh control window

Click **Assign** to preview, and **Done** to confirm the Mesh Controls.

After defining the **Global seed** and the **Mesh Controls,** select the **Build mesh** command. The geometric 3D model will automatically mesh according to the user's input.

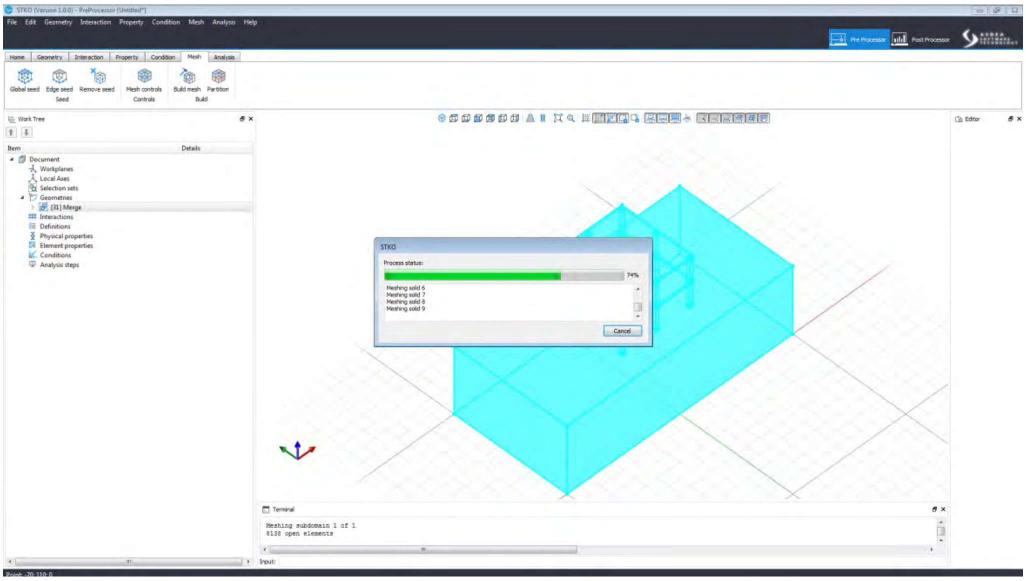


Figure 183. Example of model while meshing

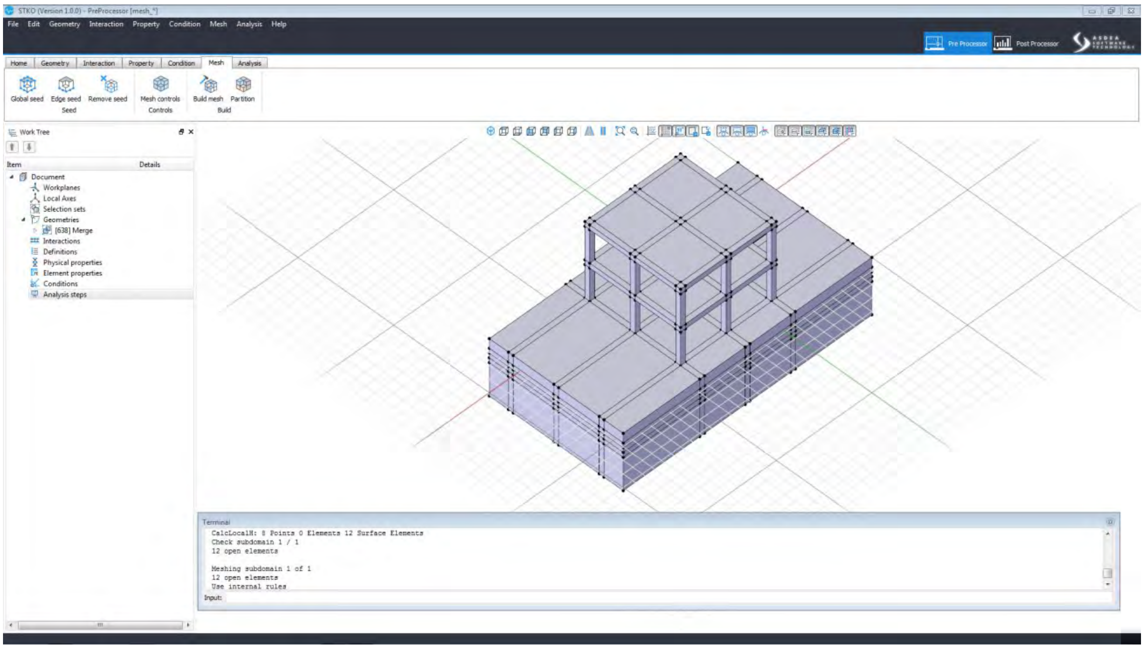


Figure 184.Example of model after meshing

To change the Global edge seed to a larger or smaller size, *Click* **Global seed**, set the new size, and *Click* **Build Mesh**.

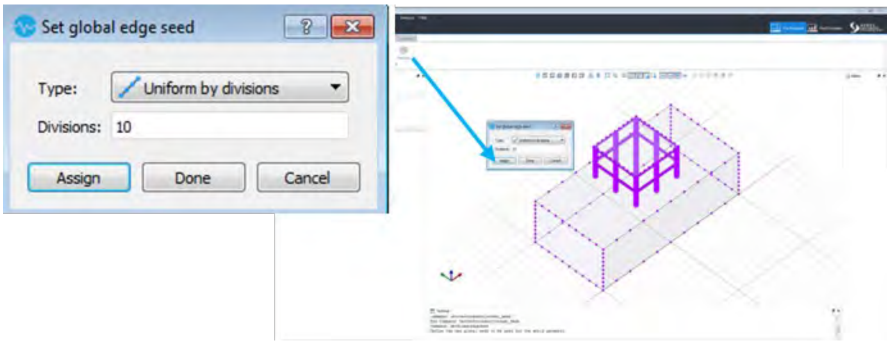


Figure 185.Example of global edge setting.

NOTE: To make a **Structured Mesh**, ensure that the imported 3D model is solely composed of solid geometries like cubes (or geometries with six faces), or, in the case of surfaces, that they have four edges. If not, the imported model will be meshed with **Unstructured Mesh.**

STRUCTURED MESH LAYOUT

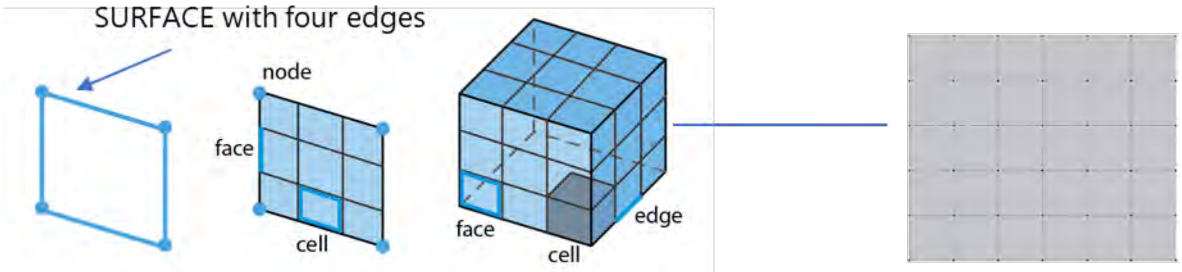


Figure 186. Hierarchical Order of Structured Mesh with quadrilateral grid

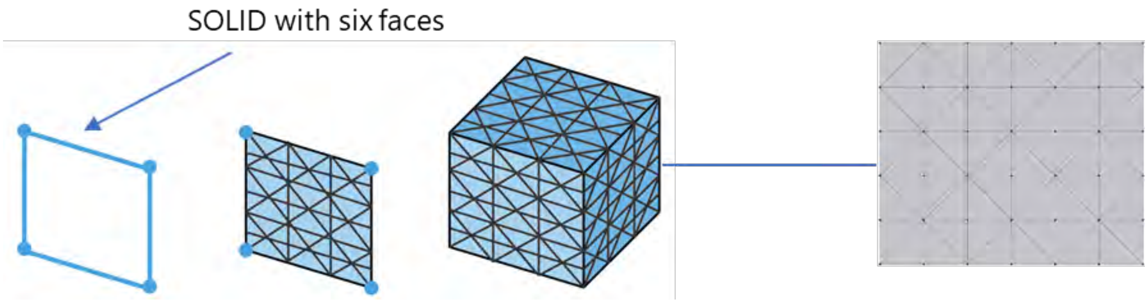


Figure 187. Hierarchical Order of Structured Mesh with triangular grid.

STKO TIPS

As mentioned above, with **Structured Mesh** the user can create a triangular or quadratic mesh. There are some restrictions on the **seed** that the user can define. **STKO** automatically generates a warning to inform the user that the number of nodes on a surface edge has been altered without varying the logarithm (the type of Seed attributed to it either *uniform by size, uniform by division, biased by size etc.*) in order to successfully obtain a structured mesh.

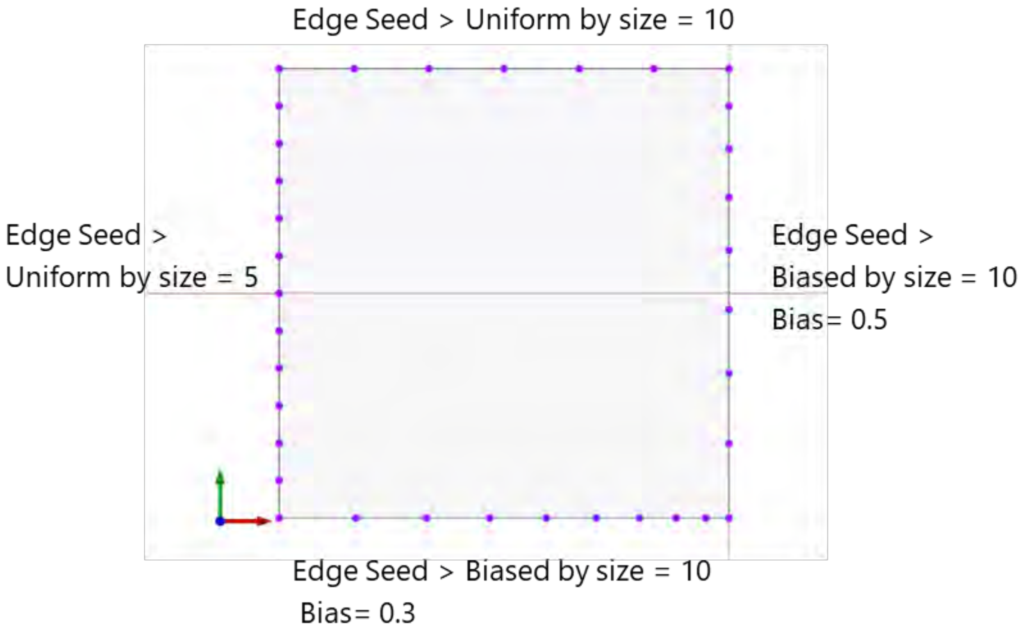


Figure 188. Example with different edge seed types

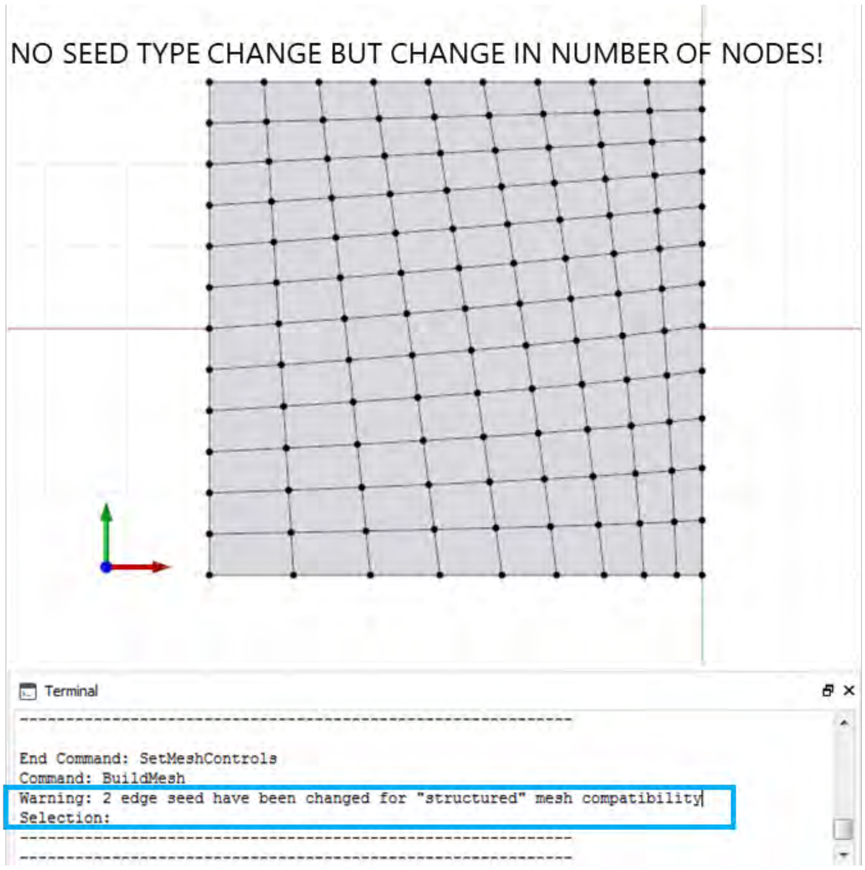


Figure 189. Changing of the number of nodes in order to achieve "structured" mesh compatibility

STKO TIPS

In **Unstructured Mesh**, STKO will generate a warning if the user sets up a hexagonal unstructured mesh and it will automatically convert the user input into a tetrahedral mesh to successfully obtain the unstructured mesh.

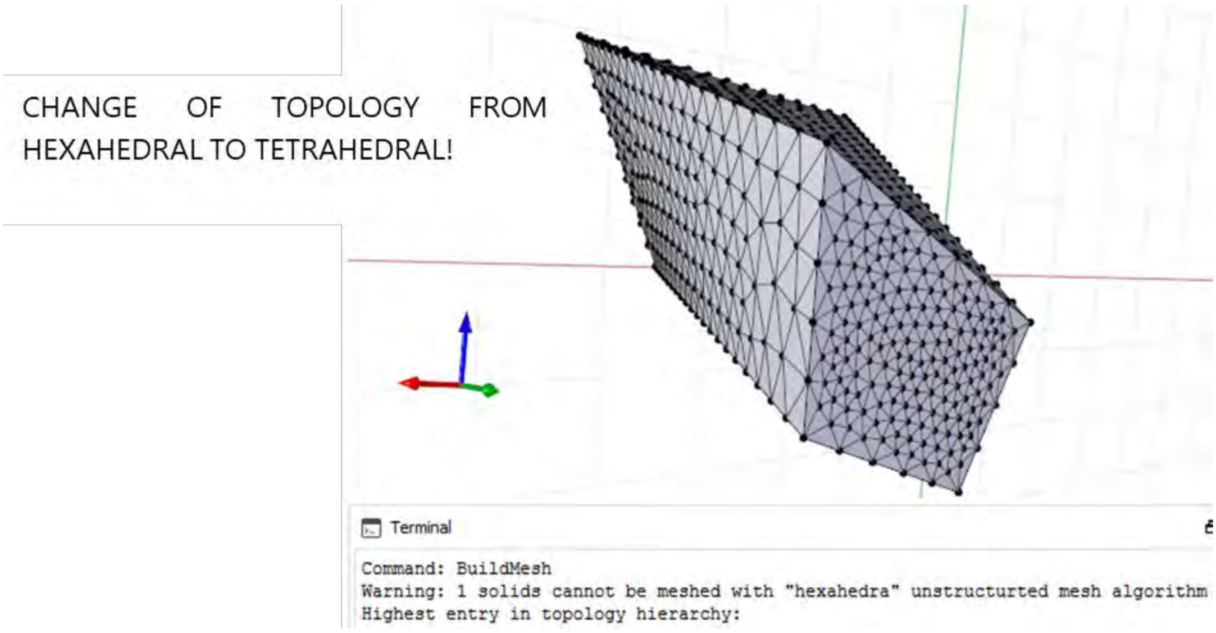


Figure 190. Changing of hexahedral mesh algorithm into tetrahedral

2.8.3. Curve Meshing

After creating curved shapes in STKO, the user can set the **Edge Seed** using the command from the Toolbar.

The user can choose the type of Edge seed to attribute to the curve:

Select the edge > **Assign** > **Done** if the Type Division is right for the edge.

Click on **Mesh Control** from the Toolbar to set the Selection in **Edges**. Select the curve > **Assign** > **Done** and/or directly *click* on **Build Mesh**.

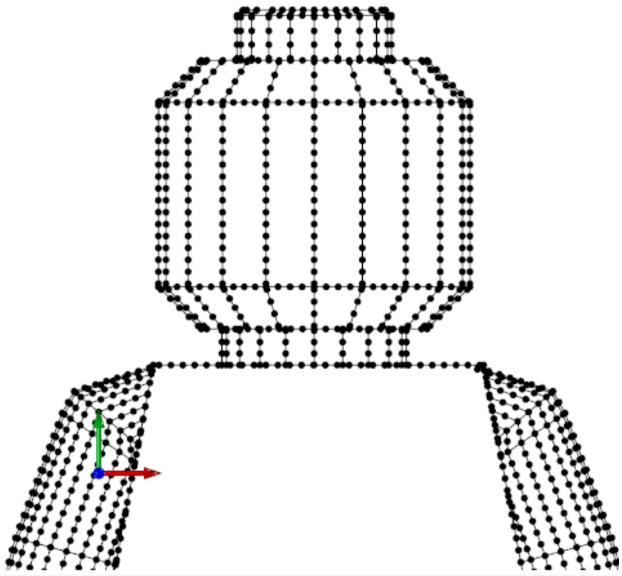


Figure 191. Example of a curve meshing

2.8.4. Unstructured Surface Meshing

The parameterization of unstructured surface meshing is of fundamental importance in STKO, especially for complex surface geometries. Triangular and quadrilateral patterns in linear and quadratic forms have been used to define the surface geometry in the examples below.

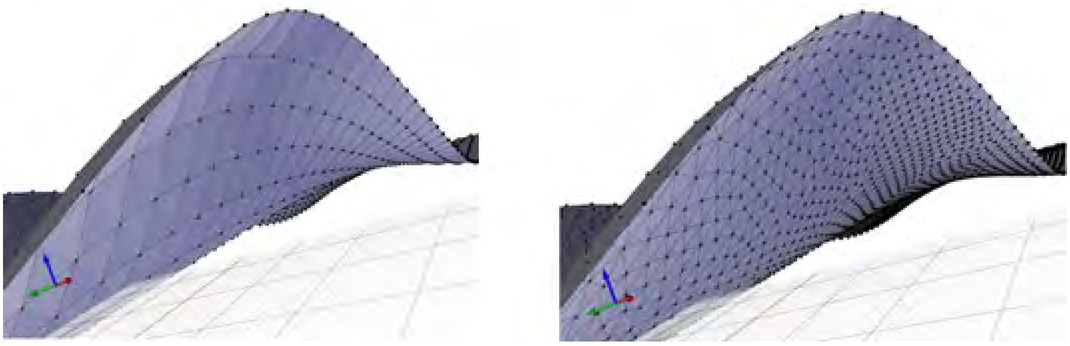
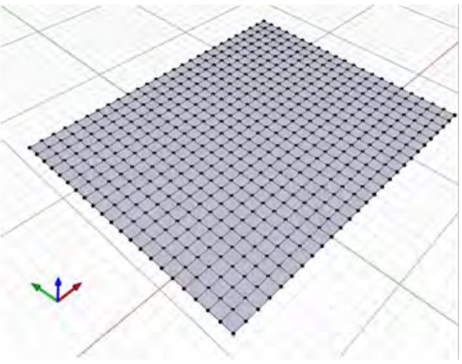


Figure 192. Quadrilateral and Triangular Unstructured Mesh of a Surface

2.8.5. Structured Surface Meshing

STKO can generate structured quadrilateral or triangular surface meshing for any 4-sided surface. The mesh spacing can be easily defined or automatically computed using the Mesh Control command.



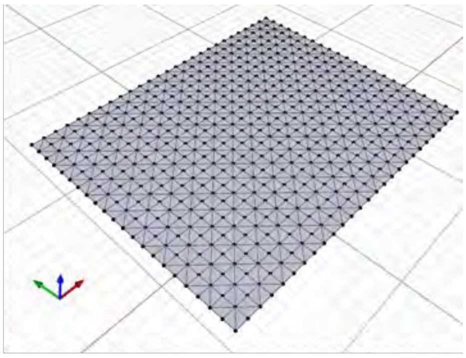


Figure 193. Quadrilateral and Triangular Structured Mesh of a Surface

2.8.6. Unstructured Solid Meshing

Unstructured meshes have an irregular topology, meaning that they do not follow a uniform pattern.

Figure 194. Example of the meshing of an unstructured solid

2.8.7. Structured Solid Meshing

Mesh is said to be **structured** if its **topology** is regular, meaning that it has a well-known pattern (such as cubes with regular spacing dx, dy, dz between nodes).

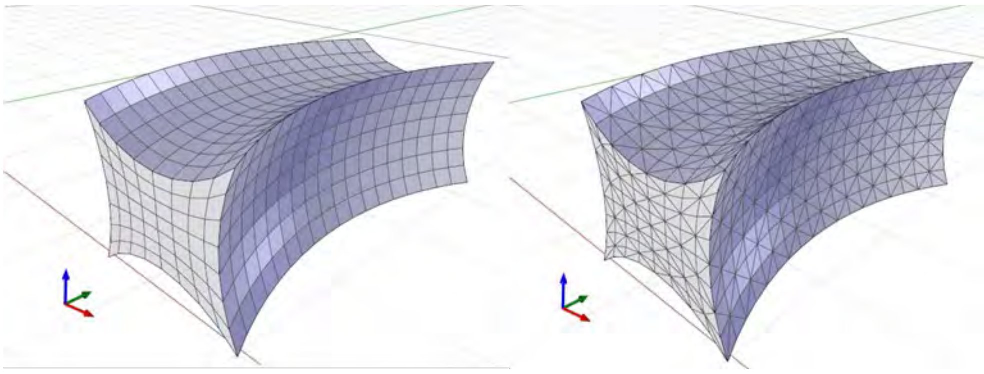


Figure 195. Hexahedral and Tetrahedral Structured Mesh of a solid

2.8.8. Partition

For parallel computations, **STKO Multiprocessor** provides facilities to partition the mesh. *Click* **Mesh** from the Toolbar and select the **Partition** command.

Partition

A Partition Mesh window will appear for setting the Number of partitions.

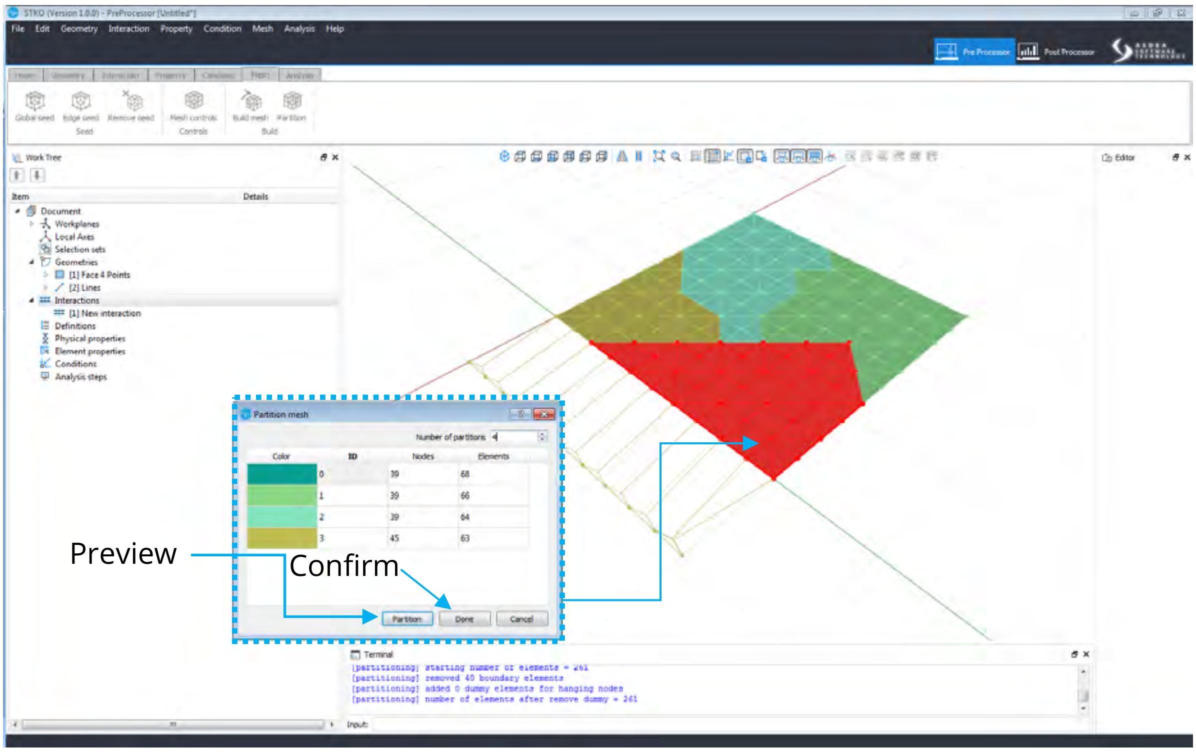


Figure 196. Partition mesh window to define the number of partitions

As in *Figure 190*, *Click* **Partition** to preview the effect the new partitions will have on the selected geometry. After *Clicking* **Partition**, a colorful legend will appear in the partition window, which allows the user to preview the colors, IDs, nodes, and elements of the new partitions. *Click* **Done** to execute the partition command.

After the execution of the command, the geometry will not appear colorful like in the preview; however, the command was executed. The user can review the partitions by *Clicking* the **Partition** command again. The partition window will open and highlight the partitions, allowing the user to review the partition information.

To delete the previously input partitions, press CTRL+Z (apple+Z for Mac) on the keyboard.

To change the previously input partitions, *click* **Partition** and simply change the number of Partitions and *Click* **Done** to confirm.

NOTE: every **Partition** will create a **Processor** and a **File**:

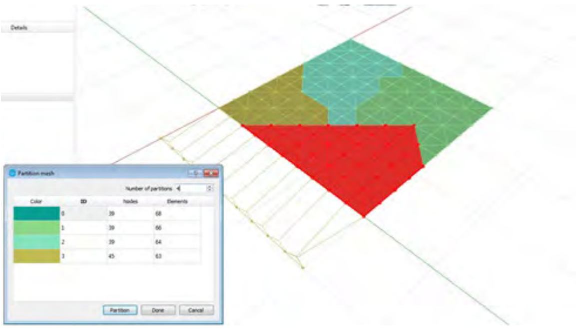


Figure 197. Visualization of the Partition with three processors (0, 1, 2)

Launching the Analysis in Opensees will create .tcl files with different processors and IDs for the number of partitions previously defined, as in the example.

Launching the Analysis in Opensees will create .tcl files with different processors and IDs for the number of partitions previously defined, as in the example.

```
if {$process_id == 0} {
# tag x y z

# nodes without assigned properties

model basic -ndm 3 -ndf 3
source definitions.tcl
    node 1 -7.8418324815891 2.68273875832474 17.0980394135998
    node 2 -7.8418324815891 2.80248744949622 17.3039225498954
    node 3 -7.8418324815891 -0.961681702644319 17.3039225498954
    node 4 -7.8418324815891 -0.841933011472845 17.0980394135998
    node 5 -7.8418324815891 2.94858480577239 12.9326749321528
    node 6 -7.8418324815891 -1.10777905892049 12.9326749321528
    node 11 -7.8418324815891 -2.0266149357764 11.8038740777807
    node 12 -7.8418324815891 3.8674206826283 11.8038740777807
    node 13 -8.151613421261073 2.776818268992782 17.3039225498954
    node 14 -8.452944344437439 2.7005109158120413 17.3039225498954
    node 15 -8.73760572904503 2.5756468556303456 17.3039225498954
    node 16 -8.997832754568604 2.4056320545131387 17.3039225498954
    node 17 -9.226527106113101 2.1951040730393903 17.3039225498954
    node 18 -9.417450597930072 1.9498055657382134 17.3039225498954
```

Figure 198. Process 0 with all relative nodes

```
} elseif {$process_id == 1} {
# tag x y z

model basic -ndm 3 -ndf 3
source definitions.tcl
    node 7 -7.8418324815891 4.82291105675198 0.0
    node 8 -7.8418324815891 -2.98210530990008 0.0
    node 9 -7.8418324815891 -3.69084558846896 8.38317442609098
    node 83 -7.419897127416489 4.800034386575861 0.0
    node 84 -7.002908579492379 4.731672584088979 0.0
    node 85 -6.595755647284915 4.618627128921701 0.0
    node 86 -6.203211826660753 4.462223375677416 0.0
    node 87 -5.829879335010474 4.2642950153616015 0.0
    node 88 -5.480135154457523 4.027162577031332 0.0
    node 89 -5.158079715744733 3.753606221713776 0.0
    node 90 -4.867488824432914 3.4468331475600693 0.0
    node 91 -4.611769393032912 3.1104399883793685 0.0
    node 92 -4.393919498071385 2.748370646395868 0.0
    node 93 -4.216493230384699 2.364870053601169 0.0
    node 94 -4.081570750739097 1.964434403807852 0.0
    node 95 -3.990733901847625 1.551758438887939 0.0
```

Figure 199. Process 1 with all relative nodes

```
} elseif {$process_id == 2} {
# tag x y z

model basic -ndm 3 -ndf 3
source definitions.tcl
    node 10 -7.8418324815891 5.53165133532086 8.38317442609098
    node 199 -7.8418324815891 4.6995360089745795 10.093524251935841
    node 200 -7.8418324815891 4.9075648405611485 9.665936795474627
    node 201 -7.8418324815891 5.115593672147718 9.238349339013414
    node 202 -7.8418324815891 5.323622503734288 8.810761882552
    node 261 -8.340396138998088 5.504619999265339 8.38317442609098
    node 262 -8.833114593365057 5.423842908799039 8.38317442609098
    node 263 -9.314211171309328 5.290267101452481 8.38317442609098
    node 264 -9.778045455325303 5.105458631437278 8.38317442609098
    node 265 -10.219179412520626 4.871584209092362 8.38317442609098
    node 266 -10.632441150579805 4.591385798186303 8.383174426090978
    node 267 -11.012985553472905 4.268148468899125 8.38317442609098
    node 268 -11.35635108601107 3.905661883378011 8.38317442609098
    node 269 -11.658512101267393 3.5081758654135524 8.383174426090978
    node 270 -11.915926037606441 3.0803505751414475 8.383174426090978
    node 271 -12.12557495198045 2.627201872925746 8.38317442609098
    node 272 -12.285000902552211 2.154042512982116 8.383174426090978
```

Figure 200. Process 2 with all relative nodes

The **Partition** command enables the user to use multiple processors [OpenseesMP].

2.9. Analysis

Learn how to structure and order your analysis steps so that your analysis runs smoothly.

2.9.1. Defining Analysis Steps

This section describes how to create the analysis steps.

2.9.1.1. Patterns

The user must define **Patterns** and **Recorders** before launching the analysis in OpenSees.

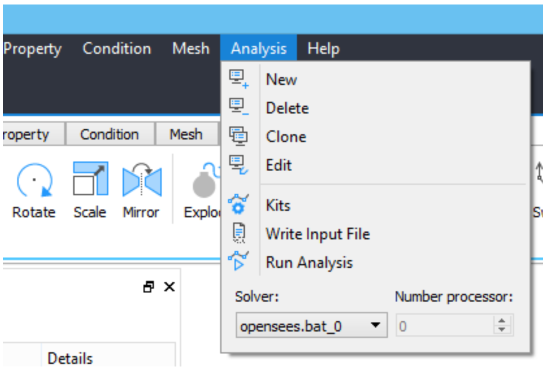


Figure 201. Analysis steps in the main menu

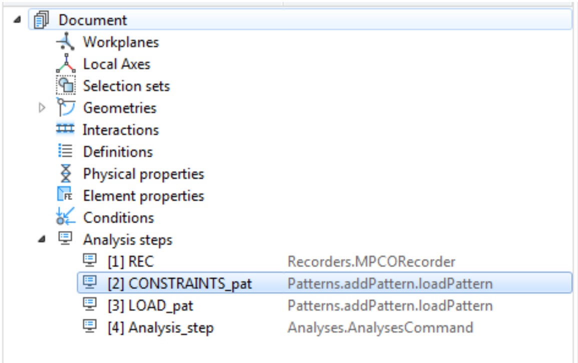


Figure 202. Analysis steps in the work tree

To define the **Patterns**, *Right-click* on **Analysis steps** > **Add** from the Work Tree panel, or **Analysis** > **New** from the main Toolbar.

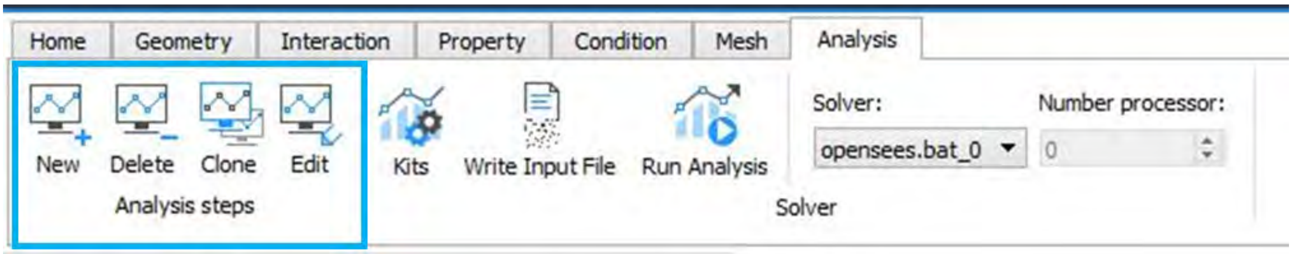


Figure 203. Analysis steps in the main menu

Choose the AnalysisStep model by *Clicking* **Model** > **Patterns** > **AddPattern** and selecting, for instance, **loadPattern**:

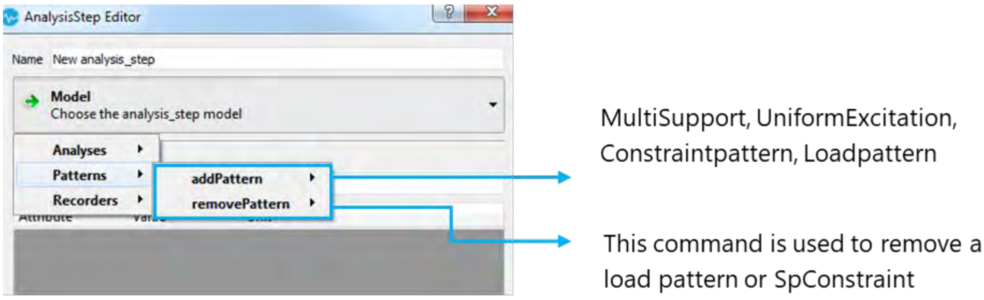


Figure 204. Add or Remove Pattern using the AnalysisStep editor

To add a load use **loadpattern**:

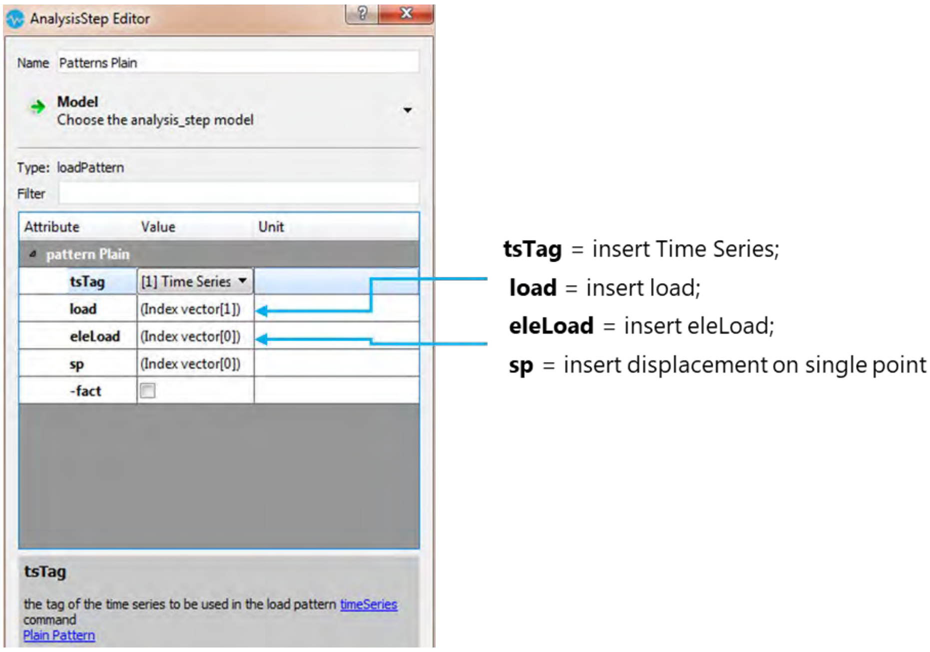


Figure 205. Add a load and a pattern

Another important pattern type is the **constraintPattern**. After defining constraints, **STKO** allows the user to create **Patterns** which contain commands to generate **sp constraints** and **mp constraints**, as shown in the following example.

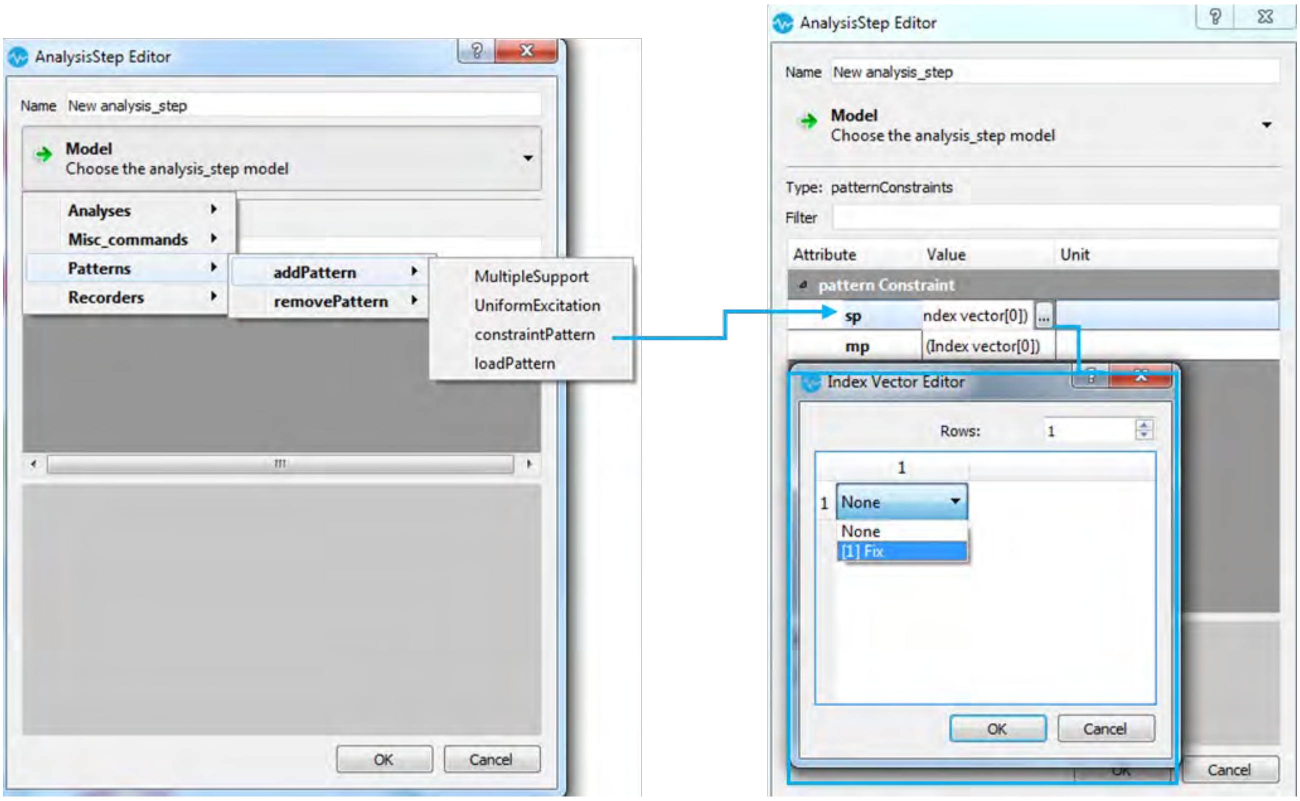


Figure 206. Creation of a constraintPattern for a single point SP (or multiple point MP) constraint.

Select **OK** to confirm the settings.

2.9.1.2. Misc_commands

Misc_commands is an analysis step that includes several commands. One of them is the **Region** command.

Region

Region is used to label a group of nodes and elements. This command is also used to assign Rayleigh damping parameters to the nodes and elements in this region. The region is specified by either elements or nodes, not both. If elements are defined, the region includes these elements and the all connected nodes.

First, create a **Selection Set** to use during the **region** command. Simply choose **Selection sets** from the Work Tree Panel > **Add** and select the geometry or groups of geometries to add to the selection set.

Then, to begin creating a Region, *Right-click* on **Analysis Steps** > **Add**> **Model** >**Misc_commands** > **Region** from the Work Tree Panel, or **Analysis** > **Model** >**Misc_commands** > **Region** from the main Toolbar.

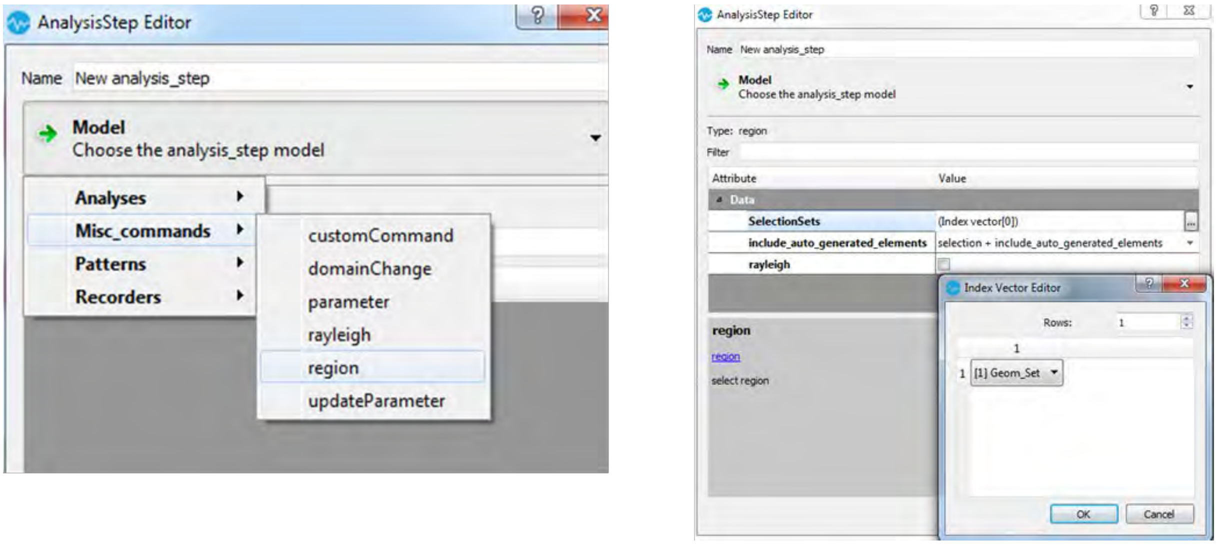


Figure 207. Creation of region from AnalysisStep editor with previously created Selection Set

Selection Set: To add a previously created selection set, *Click* Index Vector in the AnalysisStep Editor, then *Click* the ellipsis button that appears. The Index Vector Editor window will open. After the user types the desired number of rows (1, in the example), a drop down menu will appear allowing the user to choose from among their previously defined selection sets. *Click* **OK** to confirm the selection.

Include_auto_generated_elements:

The user may then define the inclusion of auto generated elements. By *Clicking* on the drop-down menu include_auto_generated_elements, the user may select one of the three options shown in the image above. After selecting whether or not to use auto generated elements, *Click* **OK** to confirm the settings.
It is not necessary for the user to create a Region for the completion of an analysis. Its use is up to the discretion and needs of the user.

Rayleigh Damping Widget

The **Rayleigh Damping Widget** is a feature of **Misc_commands** that allows the user to input the two frequencies of interest and it will automatically compute the alpha and beta Rayleigh damping parameters. To access the widget, click to add a new step, then *click* **Model** >**Misc_commands** > **Rayleigh**.

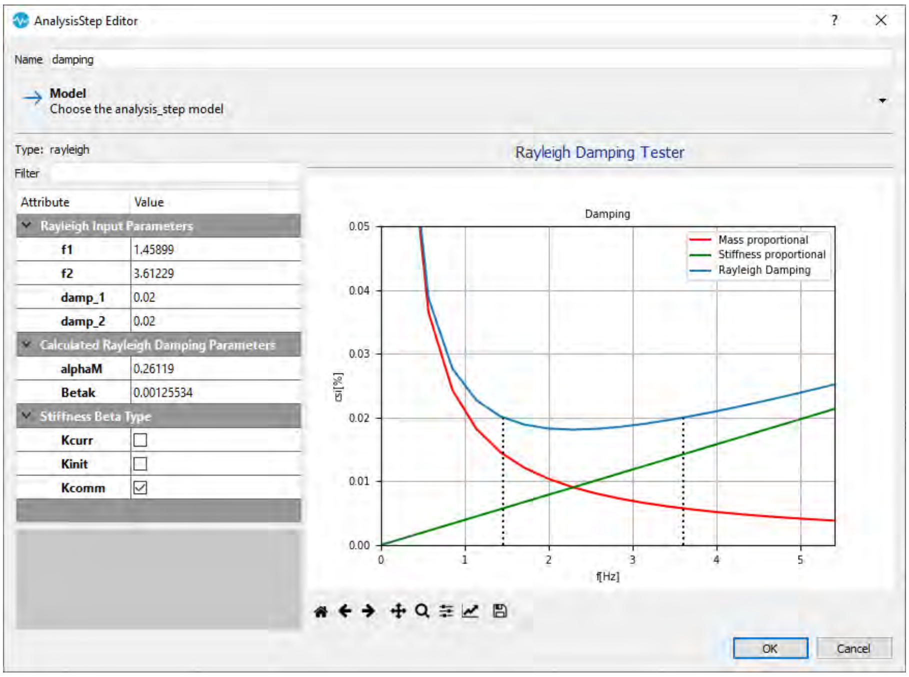


Figure 208. Rayleigh Damping Widget

Monitor

The Analysis Monitor with Real-Time Plot tracks the statistics of the analysis and draws real-time plots of the variables that the user wishes to monitor during the analysis. To use the Analysis Monitor, *click* **Analysis** > **Model** >**Misc_commands** > **Monitor**, then check the box next to Monitor Plot to open the editor.

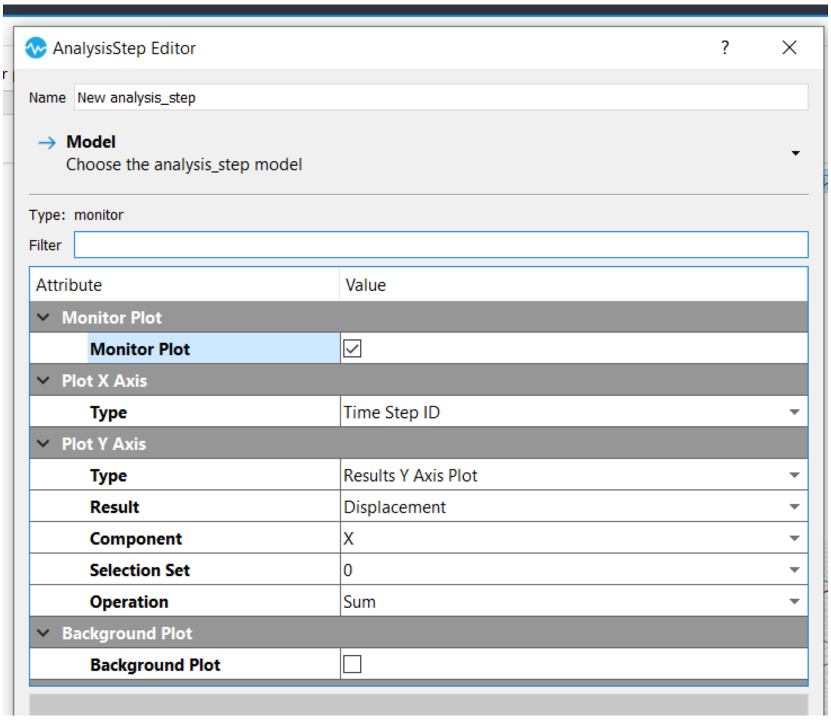
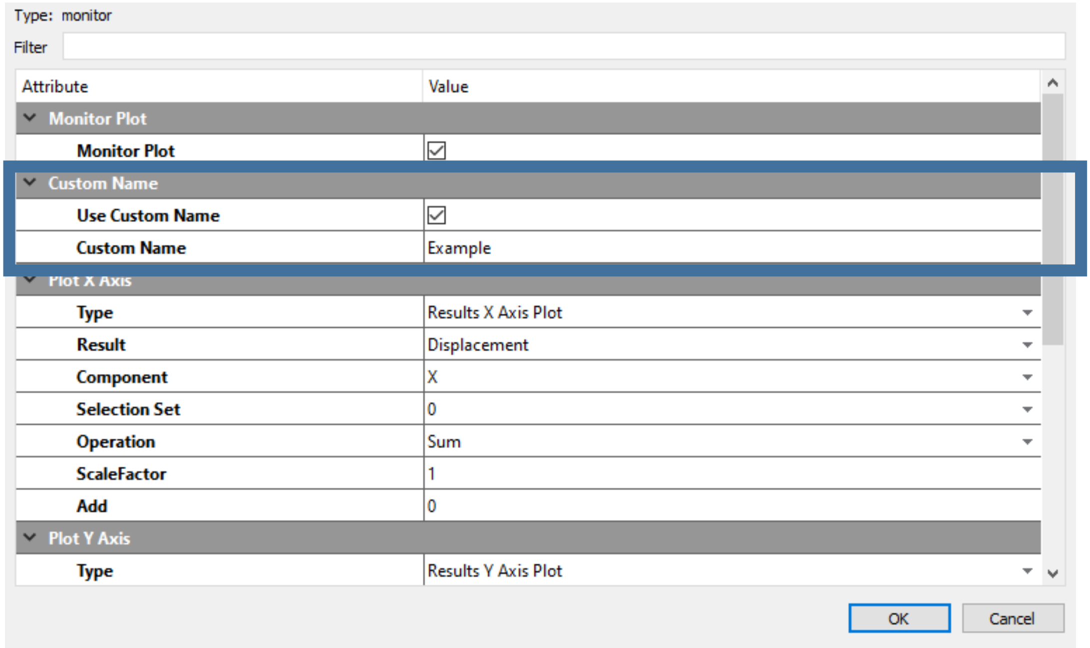


Figure 209. Monitor Plot

Don’t forget to name your monitor plot. Check *Use Custom Name* and you can name the plot:



The monitor plot can be used in two ways. It can be used simply to monitor data during the analysis. In this case, it doesn’t matter where the Monitor plot is in the list of analysis steps, it will open when the analysis is running to show the data.

If instead, the user wishes to monitor something in particular, the monitor should be created and the desired parameters selected. Then, make sure it is listed in the work tree after the elements of the models the monitor refers to (usually, after the Analysis Step).

When you begin the analysis, the monitor will activate, and you will be able to monitor the data in real-time, like in the image shown below.

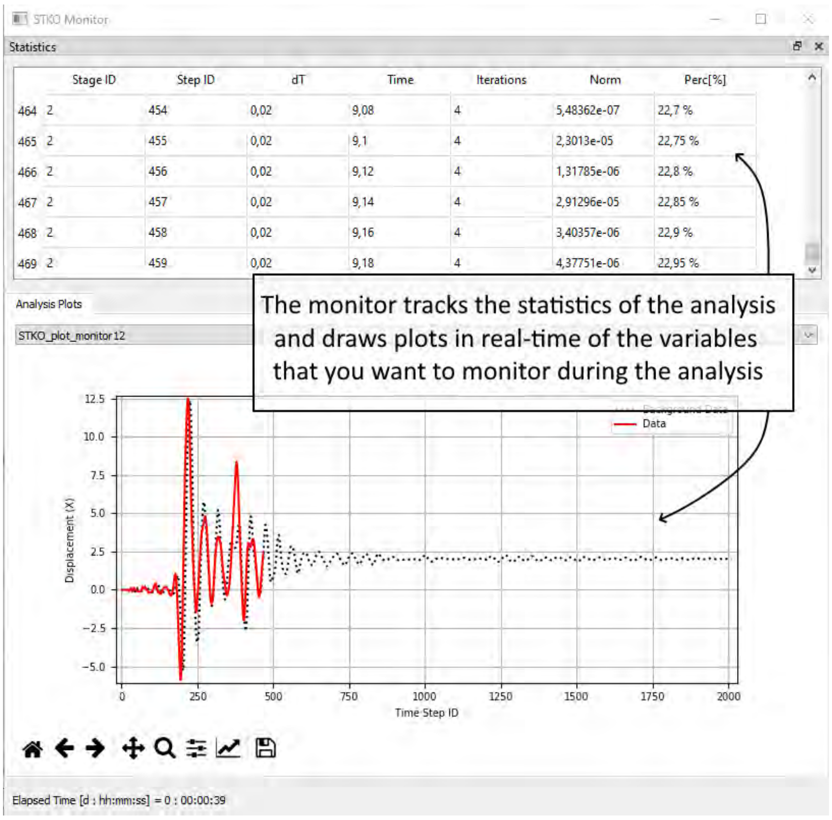


Figure 210. Monitor Plot

Our monitor tool has been updated to allow you to superimpose plots generated by the same analysis, before this option was not available, as you could just select one graph at a time to be visualized on your monitor.

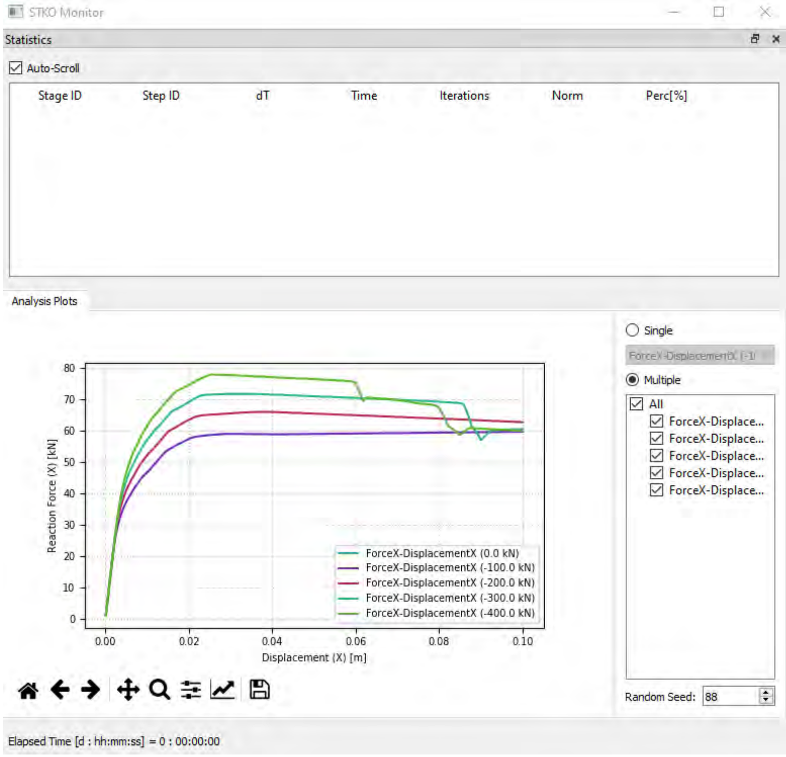
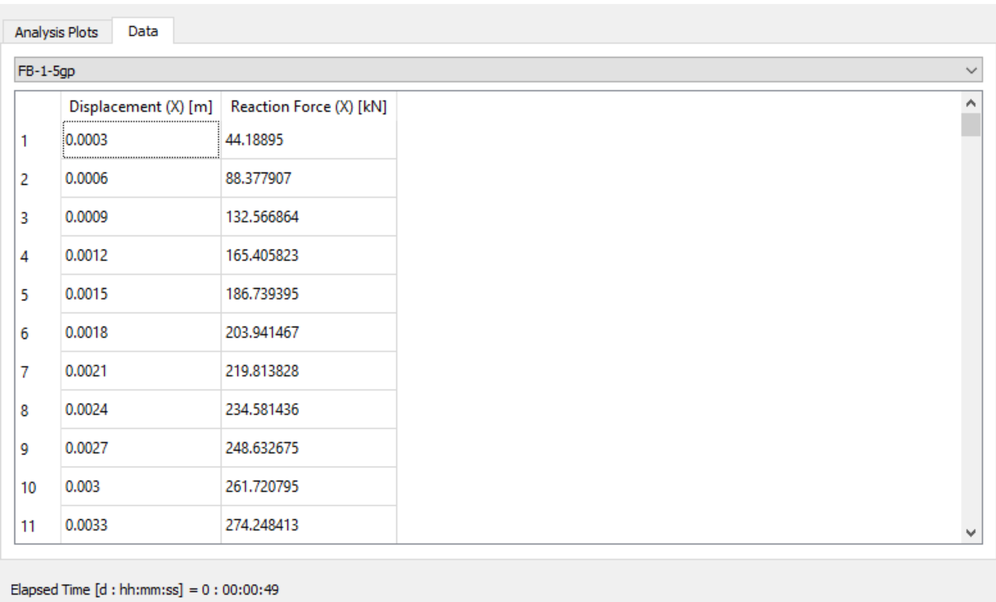


Figure 211. Superimposed Plots generated by the same analysis

The Data Tab of the Analysis Monitor allows you to view the data contained in the plots featured on the Analysis Plots Tab. Simply click the dropdown list to choose the plot whose data you wish to review.



	Displacement (X) [m]	Reaction Force (X) [kN]
1	0.0003	44.18895
2	0.0006	88.377907
3	0.0009	132.566864
4	0.0012	165.405823
5	0.0015	186.739395
6	0.0018	203.941467
7	0.0021	219.813828
8	0.0024	234.581436
9	0.0027	248.632675
10	0.003	261.720795
11	0.0033	274.248413

Figure 212. Example of the Data Tab

2.9.1.3. Custom Command

With the release of *Version 2.0.2*, now one or more custom procedures defined by the user can be added internally in STKO. In OpenSees, this is easily achieved by calling the **analyze command** from a loop and inserting specific functions to be performed before or after the analysis, for a set number of increments. Given that in STKO the **analyze command** is called by inserting the **AnalysisCommand** in the **analysis steps**, an internal list called **all_custom_functions** has been inserted and made accessible from the **customCommand**. To create a **customCommand,** *Right-click* **Analysis Steps** > **Add** > **Model** > **Misc_commands** > **customCommand** from the Work Tree Panel, or **Analysis** > **New** > **Model** > **Misc_commands** > **customCommand** from the main Toolbar

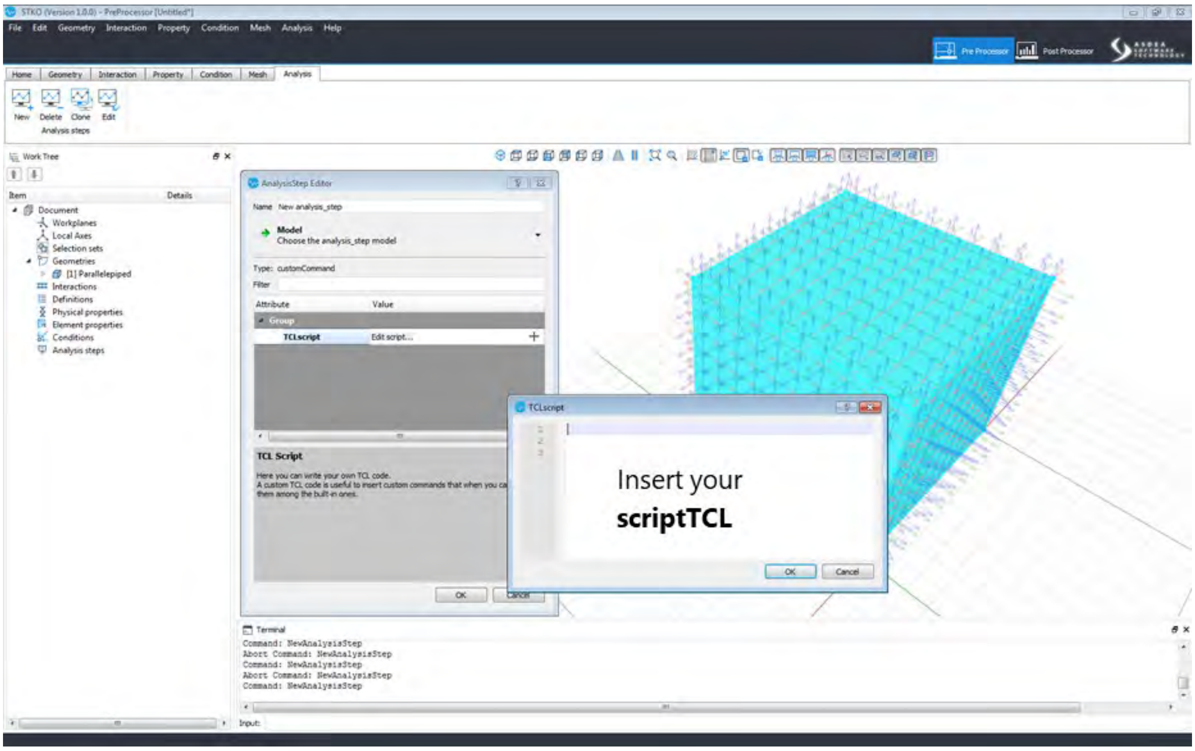


Figure 213. Customable scriptTLC interface from the customCommand

For example, if you are performing a nonlinear time history analysis, it may be useful to have a report of the result of each iteration as a progression, using a script inserted in a CustomCommand (as shown in the image below), right before the desired AnalysisCommand in the Work Tree.

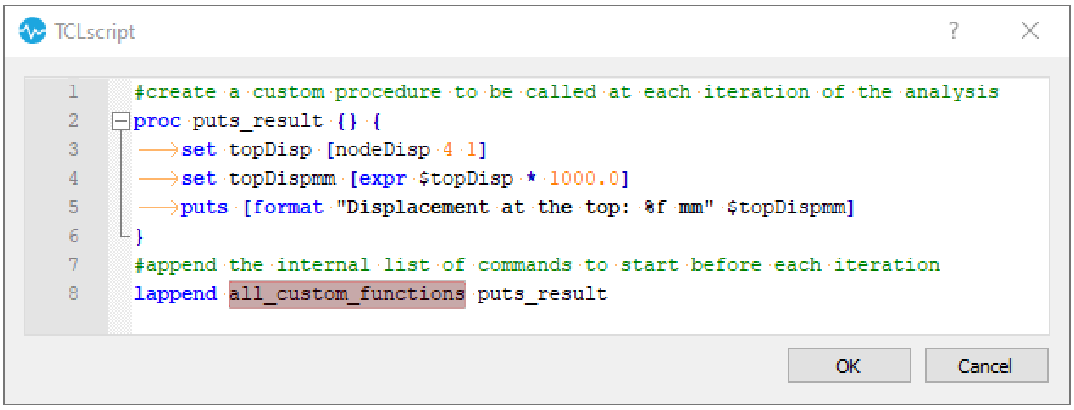


Figure 214. Sample TLC Script for a customCommand

The script shown in the image above would give a result like the following:

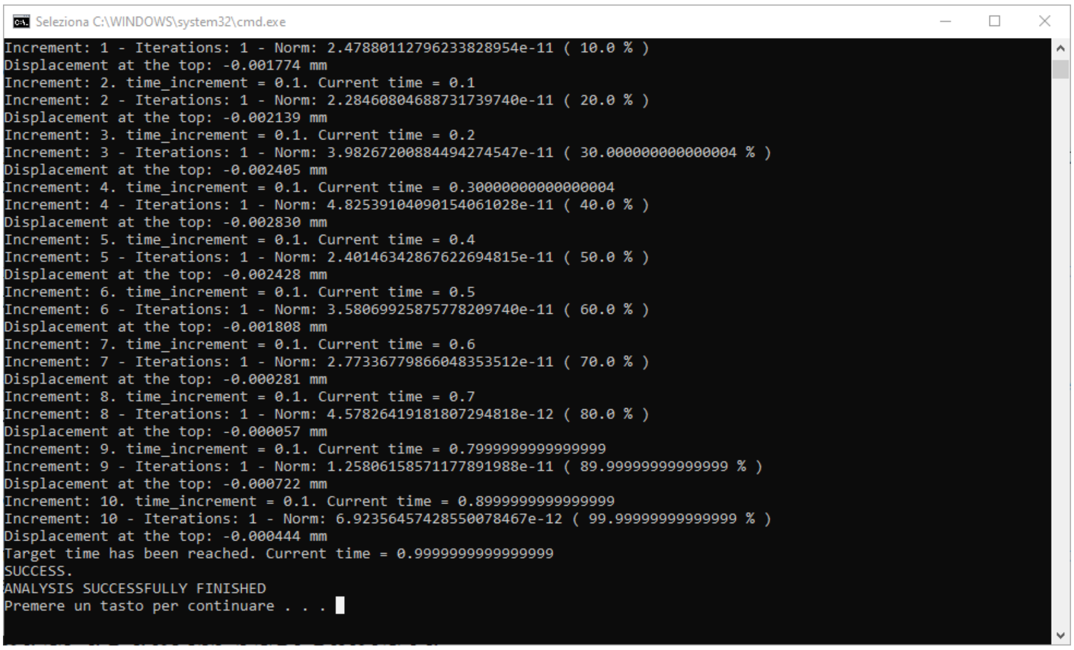


Figure 215. CustomCommand Result Sample

Custom commands allow the user a lot of flexibility in creating the analysis. For example, you could operate a specific variables update in a parametric analysis. However, to set up a custom command, a user needs to be acquainted with tcl programming.

2.9.2. Recorders

To set the recorders, *Right-click* **Analysis steps** > **Add** > **Model** > **Recorders > MPCORecorder** from the Work Tree Panel. A list of nodal results and element results will appear in the AnalysisStep Editor, as shown in the following image.

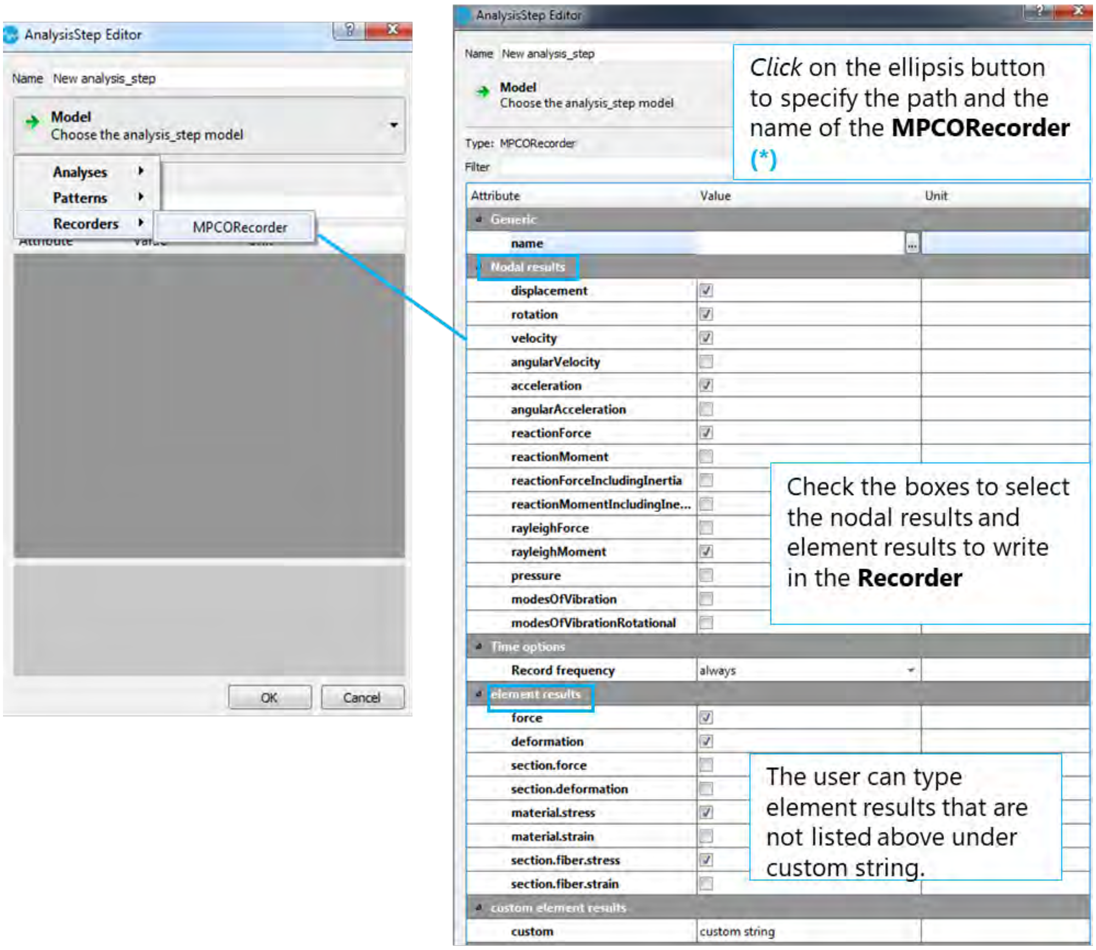


Figure 216. Creation of a MPCORecorder from the AnalysisStep Editor

NOTE (*): The user must create a name (or name and file location) for the MPCORecorder; otherwise the recorder will not be recognized by STKO when writing the results, and nothing will be recorded.
Click **OK** to confirm the settings.

2.9.3. Analysis

STKO supports three types of analyses:

1. Static – for static analysis
2. Transient – for dynamic analysis with constant time steps
3. Variable Transient – for transient analysis with variable time steps.

Begin by *Right-clicking* **Analysis Steps** > **Add** > **Model** > **Analyses** > **AnalysesCommand** from the Work Tree Panel, or **Analysis** > **Model** > **Analyses** > **AnalysesCommand** from the main Toolbar.

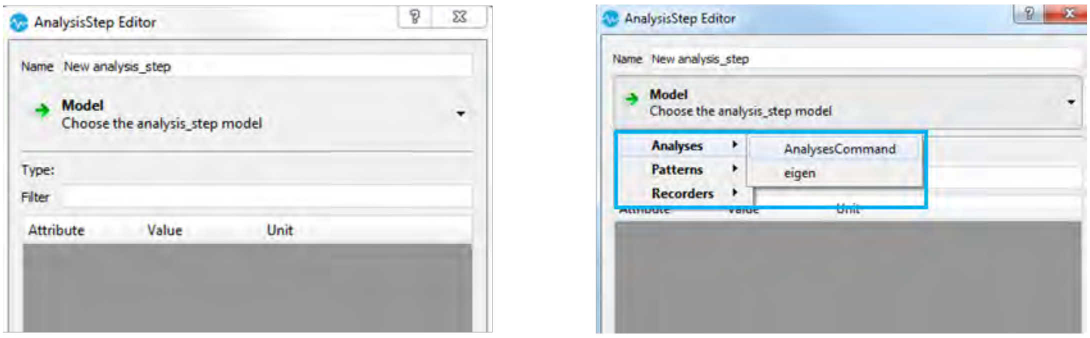


Figure 217. Creation of Analysis Command from the Analysis Step Editor

Depending on the Analyses Type selected by the user, the parameters available will vary.

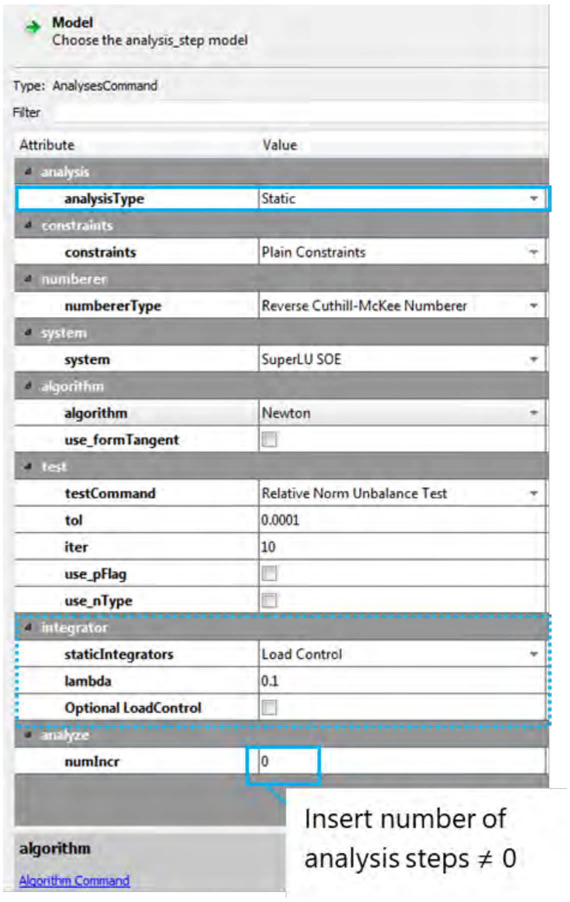


Figure 218. Example of Static Analysis Type

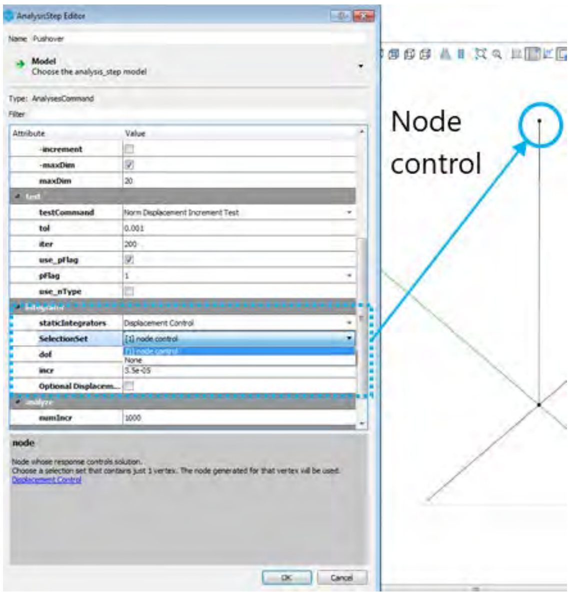


Figure 219.Example of Static Analysis with Displacement control

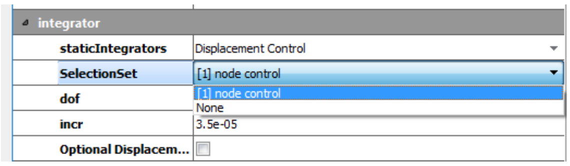


Figure 220. Integrator window

STKO allows the user to control the analysis according to a precise element or group of elements defined by a previously created **Selection Set** (see § 2.1.3 geometry set), available in the drop-drown menu in the **Integrator** of the analysis.
Unlike in OpenSees, **Variable Transient** is not listed as a selectable analysisType. Instead, if the user desires to run a **Variable Transient** analysis, they must first select Transient for the analysisType**.** The user will then be given the option to check a box to run a **VariableTransient** analysis, as shown in the image below.

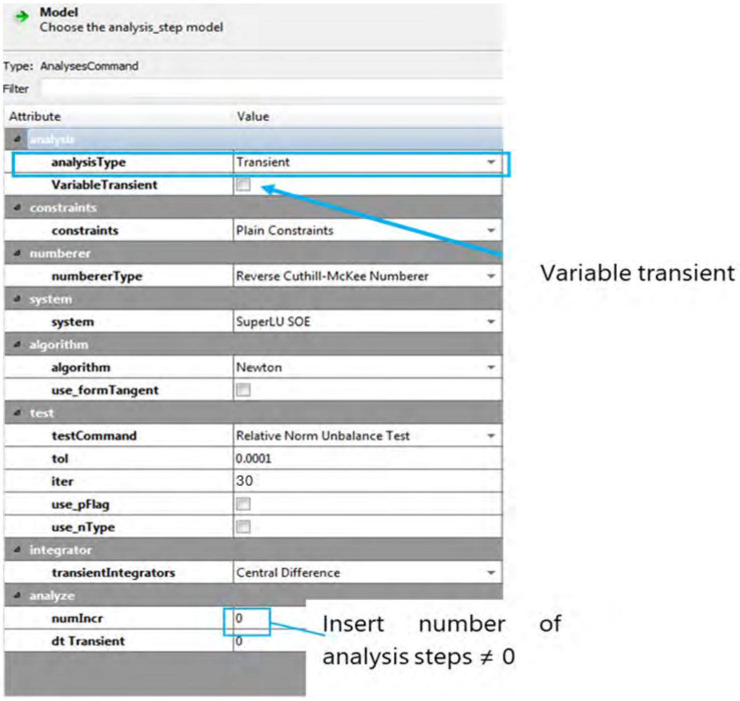


Figure 221.Example of Transient (and variable) Analysis

Users can also choose to set an **adaptive time step,** which can automatically resize the time step size to improve the convergence of highly nonlinear problems.

The adaptive time step requires the definition of a maximum number (iter in the test command in Figure 205) and desired number of iterations. The latter is set automatically to 1/3 of the maximum number of iterations, for this reason the maximum number of iterations is recommended to be equal or larger than 20.

An adaptive time step can be set by navigating to the analyze section of the editor, clicking the drop down menu, and selecting adaptive time step.

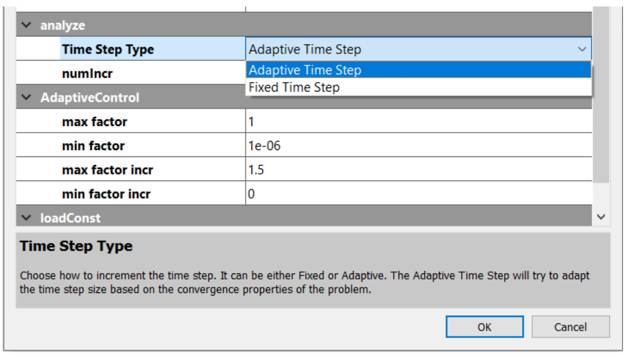


Figure 222. Adaptive Time Step

Once selected, users will be able to set the different **AdaptiveControls**, which will open below the analyze section. To gain more information about these controls, users can click on their names and a brief description will appear at the bottom of the editor window.

Like in OpenSees, the analysis-definition allows the user to select from among the different linear and nonlinear analysis tools available. The user must define the following items for each analysis created. We recommend defining the items in the following order:

- **Analysis Type Command** (http://opensees.berkeley.edu/wiki/index.php/Analysis_Command);
- **Contraints Command** (http://opensees.berkeley.edu/wiki/index.php/Constraints_Command);
- **Numberer Command** (http://opensees.berkeley.edu/wiki/index.php/Numberer_Command);
- **System Command** (http://opensees.berkeley.edu/wiki/index.php/System_Command);
- **Algorithm Command** (http://opensees.berkeley.edu/wiki/index.php/Algorithm_Command);
- **Test Command** (http://opensees.berkeley.edu/wiki/index.php/Test_Command);
- **Integrator Command** (http://opensees.berkeley.edu/wiki/index.php/Integrator_Command);
- **Analyse Command** (http://opensees.berkeley.edu/wiki/index.php/Analyze_Command).

STKO provides links to OpenSees explanations to help users understand the functions of the items and subitems available.

NOTE: Remember to respect the order of the Analysis steps. The user may use the up and down arrow buttons above the Work Tree to place the steps into the correct order to successfully launch the Analysis.

2.9.3.1. Eigen Analysis

Eigen Commands can be created in STKO, allowing the user to perform the analysis and generate Eigen values (http://opensees.berkeley.edu/wiki/index.php/Eigen_Command).

Right-click **Analysis Steps** > **Add** > **Model** > **Analyses** > **Eigen** from the Work Tree Panel, or **Analysis** > **Model** > **Analyses** > **Eigen** from the main Toolbar.

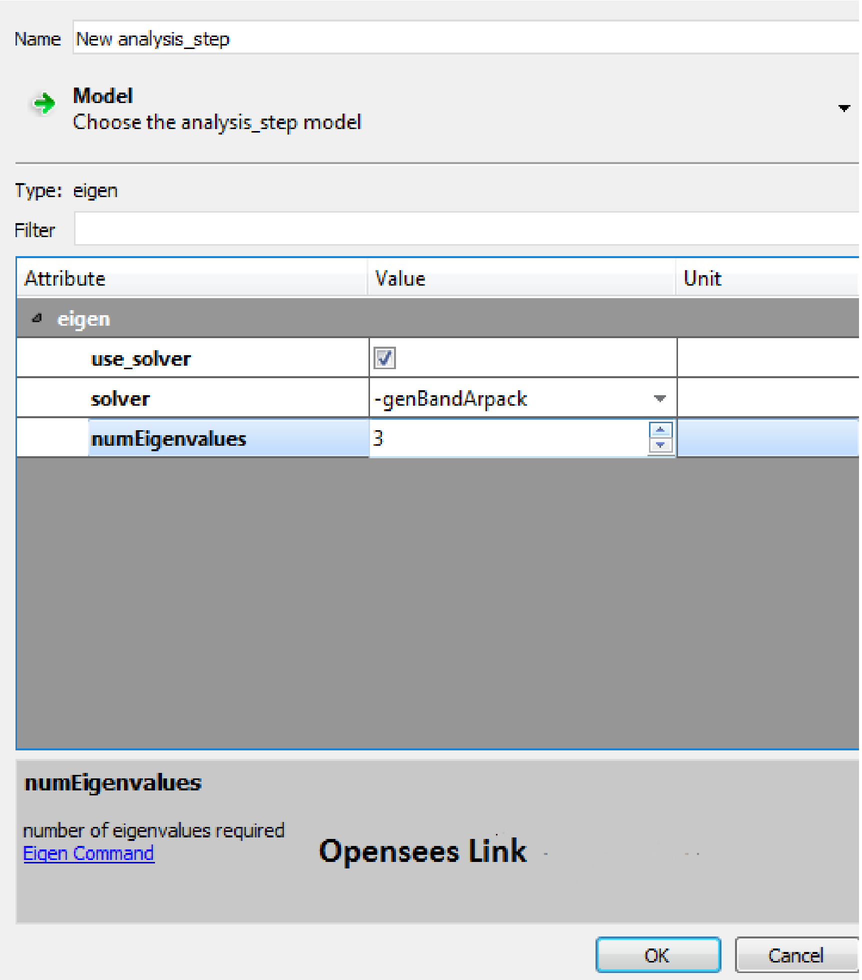


Figure 223. Creation of an Eigen command in the AnalysisStep Editor

After selecting the settings for the Analysis, *Click* **OK** to confirm. The new Analysis will appear in the Work Tree.

2.9.3.2. ModalProperties Command

The **modalProperties** command is not an analysis, it's a tool for the computaion of modal properties based on a previously set eigen analysis.

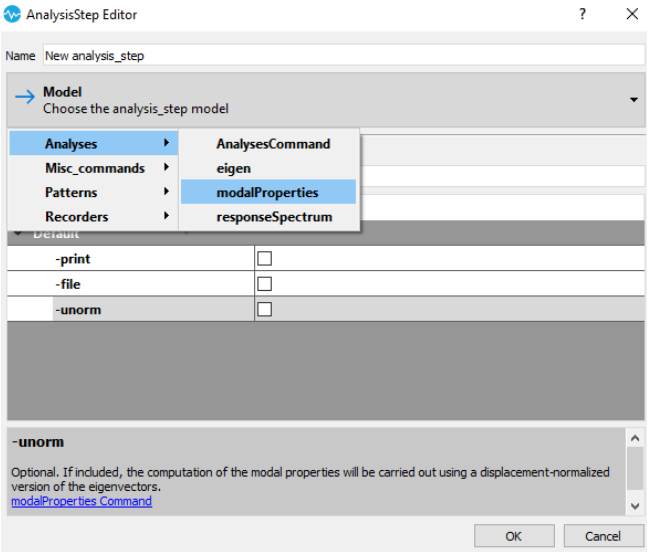


Figure 224. Modal Properties Command

Therefore, it order to run the **modalProperties** command, you **must** already have an **eigen command** set, otherwise it won't work. The command can be used for both 2D and 3D problems.

```
# MODAL ANALYSIS REPORT

* 1. DOMAIN SIZE:
# This is the size of the problem: 2 for 2D problems, 3 for 3D problems.
3


* 2. EIGENVALUE ANALYSIS:
#          MODE        LAMBDA         OMEGA     FREQUENCY        PERIOD
# ------------- ------------- ------------- ------------- -------------
              1        7578.8       87.0563       13.8554     0.0721738
              2       8484.47       92.1112       14.6599     0.0682131
              3       10518.5        102.56       16.3229     0.0612636
              4         85779       292.881       46.6134     0.0214531
              5       89260.1       298.764       47.5498     0.0210306
              6        101089       317.945       50.6025     0.0197619
              7   1.71885e+06       1311.05        208.66    0.00479249


* 3. TOTAL MASS OF THE STRUCTURE:
# The total masses (translational and rotational) of the structure
# including the masses at fixed DOFs (if any).
#            MX            MY            MZ           RMX           RMY           RMZ
# ------------- ------------- ------------- ------------- ------------- -------------
           1600          1600          1600          7200         10000         10000


* 4. TOTAL FREE MASS OF THE STRUCTURE:
# The total masses (translational and rotational) of the structure
# including only the masses at free DOFs.
#            MX            MY            MZ           RMX           RMY           RMZ
# ------------- ------------- ------------- ------------- ------------- -------------
           1600          1600          1600          7200         10000         10000
```

Figure 225. Sample of a modalProperties command report

The command computes:

- The total mass of the structure for each DOF (both fixed and free)
- The center of mass
- The generalized mass matrix
- The modal participation factors
- The modal participation masses

- The cumulative modal participation masses
- The modal participation mass ratios
- The cumulative modal participation mass ratios.

2.9.3.3. Response Spectrum Command

The Response Spectrum command generates a response spectrum analysis based on the eigen value analysis and the modal properties command previously set.

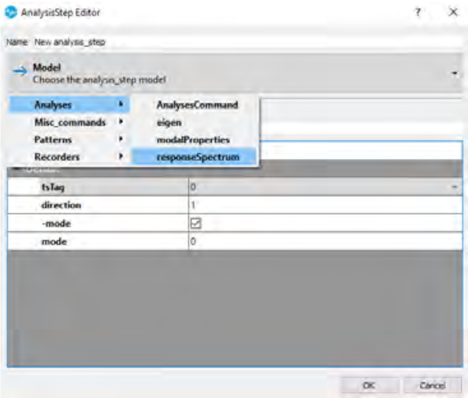


Figure 226. Response Spectrum Command

The response spectrum analysis performs N linear analysis steps, where N is the number of eigenvalues requested in a previous call to the eigen Command. For each analysis step, it computes the modal displacements. The modal combination of these modeal displacements it up to the user and can be done via TCL or Python Scripting.

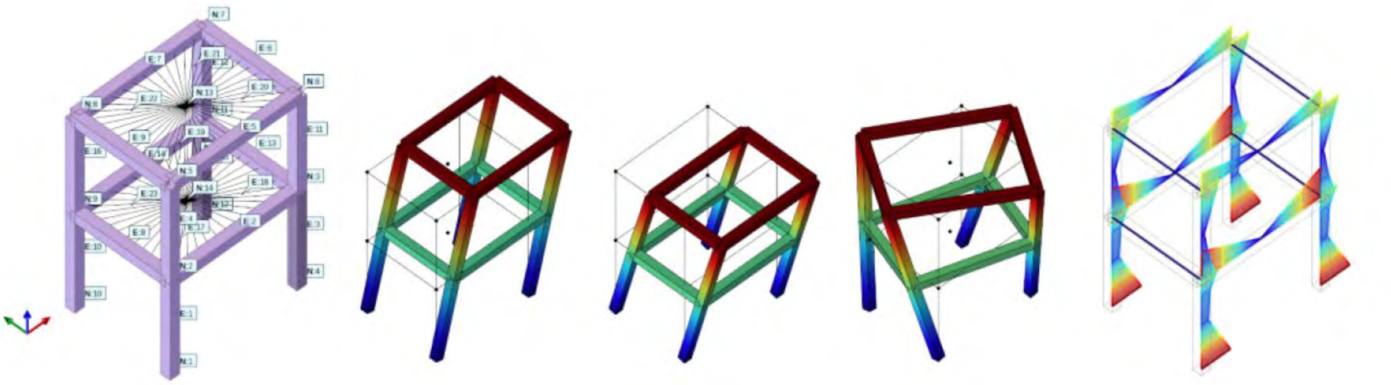


Figure 227. Response Spectrum Structural Example

2.9.4. Solver

2.9.4.1. Kits

Once the user has defined the analysis steps, they can assign the solver.

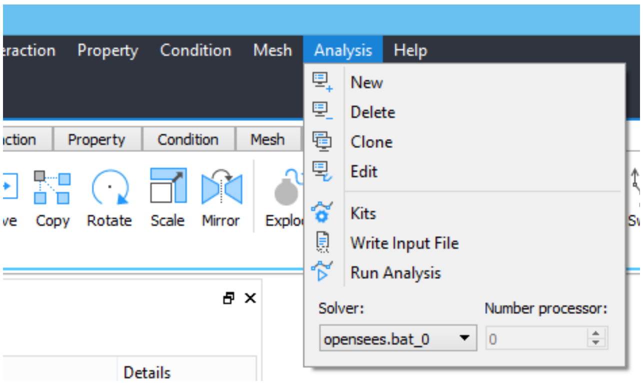


Figure 228. Solver part in the main menu.

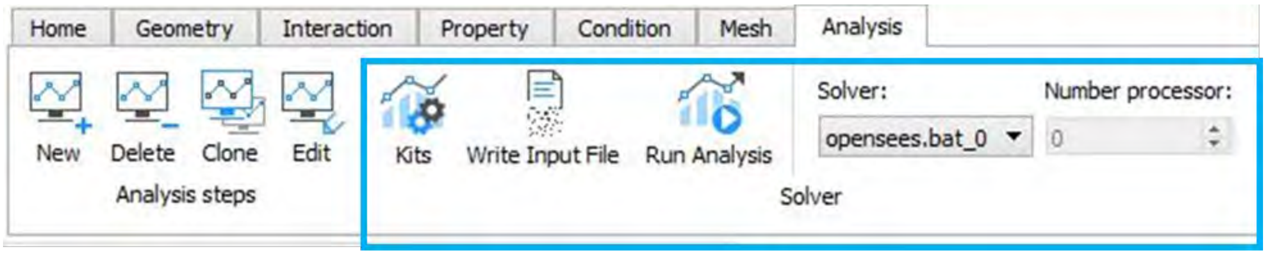


Figure 229.Solver part in the toolbar

Select **Analysis** > **Kits** from the main Toolbar. The **Edit Solver Kits window** will appear. In this window, the user can **Find or Import a solver.** *Click***ing** on **Find solver** automatically identifies the available solvers and lists them in the auto-detected area. The **Auto-detected solvers** may be: Opensees, Opensees Mp, and/or Opensees Sp.

As Opensees is an open source program, it can be customized by the user. STKO allows the user to Import Solvers from Opensees by *Clicking* **Import Solver** and selecting the solver they wish to use. The imported solver will appear listed under **Manual** in the Edit Solver Kits window.

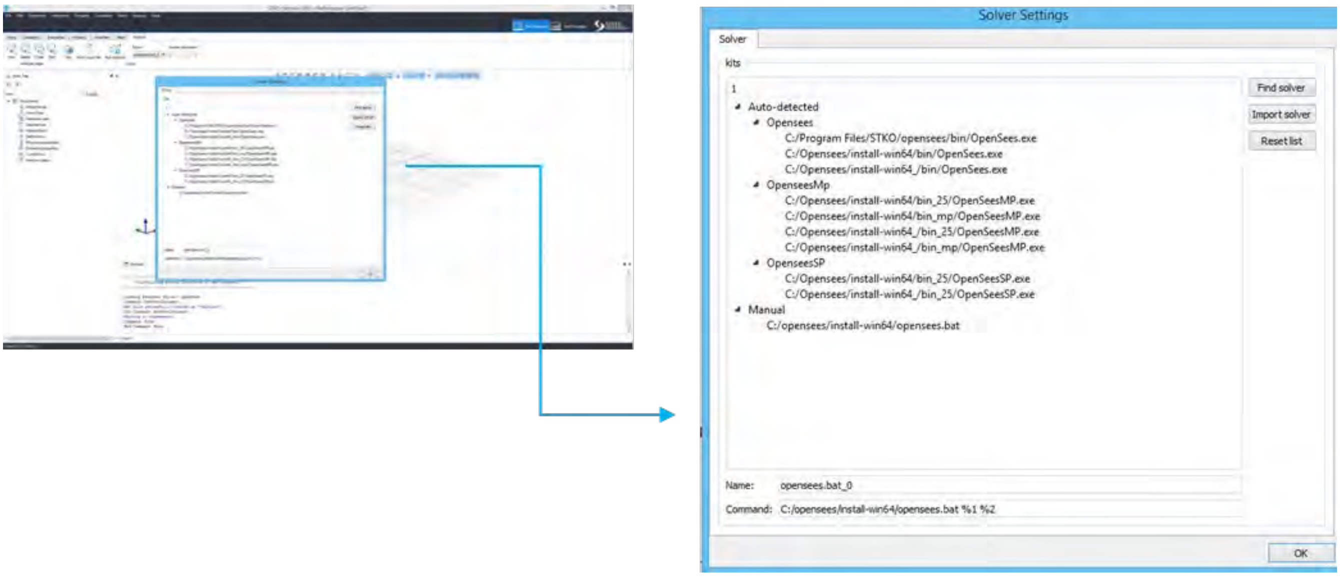


Figure 230.Solver settings window

2.9.4.2. Write Input File

After defining a solver, the user can write an Input file. *Clicking* **Write Input File** on the toolbar will generate 7 **ticle (.tcl)** files for solving.

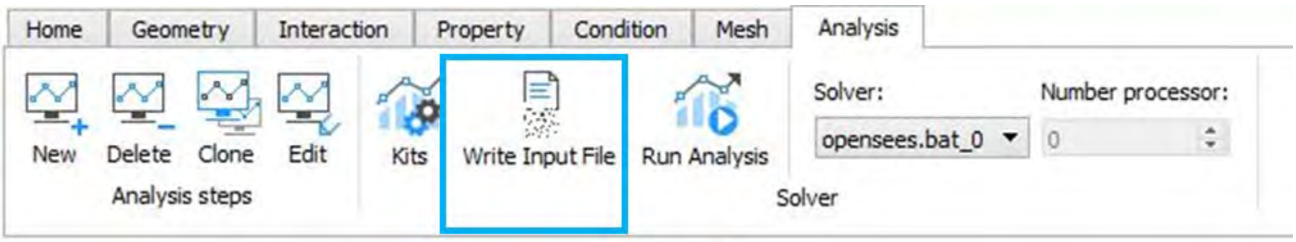


Figure 231.Write input file

The write Input file generates **ticle (.tcl)** files with the nodes, elements, forces, etc. The name of the files are: **main.tcl; definitions.tcl; materials.tcl; sections.tcl; nodes.tcl; elements.tcl.**

1. **main.tcl:** is the main file, which calls all the other **.tcl files**.

```
wipe

model basic -ndm 2 -ndf 3
source definitions.tcl
source materials.tcl
source sections.tcl
# source node
source nodes.tcl
# source element
source elements.tcl
# source analysis_steps
source analysis_steps.tcl

wipe
```

```
# beam_column_elements elasticBeamColumn

model basic -ndm 3 -ndf 6
source definitions.tcl

# Geometric transformation command
geomTransf Linear 1 0.0 0.0 1.0
element elasticBeamColumn 1 1 2 2.8791800000000004e-05 1000.0 0.2

model basic -ndm 2 -ndf 2
source definitions.tcl

# quadrilateral_elements quad
element quad 6 3 4 5 6 1.0 PlaneStrain 2
```

Figure 232. Script of main.tlc

2. **definitions.tcl:** describes **Timeseries** and **friction models** with all the parameters previously defined.

```

# timeSeries Constant
timeSeries Constant 1

# timeSeries Linear
timeSeries Linear 2

# timeSeries Path
set timeSeries_list_of_values_3 {-0.001219107 -0.001174382 -0.001200978 -0.001160988 -0.001187677 -0.001130464 -0.001142031 -0.001086745 -0.001110704 -0.001058737 \
                                 -0.001117877 -0.001083785 -0.001103846 -0.001054804 -0.001022829 -0.0009210969 -0.0009857138 -0.001016127 -0.001052384 -0.0009796386 \
                                 -0.0009052866 -0.0007784561 -0.0008150074 -0.0008270079 -0.0009953287 -0.0007569791 -0.0006789728 -0.0006281026 -0.0006604477 -0.0005554428 \
                                 -0.000798364 -0.0008538866 -0.0006053459 -0.0002852234 -9.234265e-05 8.304891e-05 3.811512e-05 7.049374e-05 -0.0007523362 -0.0006814194 \
                                 -0.0002601149 -0.0001916987 -0.0006974825 -0.0009492605 -0.001258791 -0.0007975182 -0.0007634738 -0.0007832481 -0.001349468 -0.00155521 \
                                 -0.001578003 -0.00117561 0.0002309789 0.002451741 0.002370209 0.001141321 6.727348e-05 -0.0005350975 -0.0006291217 0.0001383487 \
                                 0.0006619764 0.001517668 0.0008724106 -0.001289447 -0.002767579 -0.002993438 -0.002231119 -0.001865167 -0.00321937 -0.005534245 \
```

Figure 233. Script of all timeSeries values

```
timeSeries Path 3 -dt 0.01 -values $timeSeries_list_of_values_3  -factor 386.0
```

Figure 234. Script of timeSeries Path

3. **materials.tcl:** contains the materials.

```
uniaxialMaterial Bilin 2 42746000.0 0.0009612886464016636 0.000961288
uniaxialMaterial Bilin 3 53432500.0 19.896026762738035 19.89602676273
uniaxialMaterial Bilin 4 19246333.333333332 0.11235757780625887 0.112
uniaxialMaterial Elastic 5 29000.0
uniaxialMaterial Elastic 6 1e-09
```

Figure 235. Script of material.tcl previously defined

4. **sections.tcl:** contains the sections.

```
section Elastic 1 29000.0 38.0892 4435.484519239999

section Elastic 11 29000.0 30.644 3903.2259366666667

section Elastic 12 29000.0 990.0 106920.0
```

Figure 236. Script of section.tcl

5. **nodes.tcl:** contains the nodes and the masses of the model.

```
# tag x y
node 1 0.0 0.0
node 2 0.0 180.0 -mass 0.692646 1e-09 1e-09
node 3 -360.0 180.0 -mass 0.692646 1e-09 1e-09
node 4 0.0 324.0 -mass 0.6796996374935267 1e-09 1e-09
node 5 -360.0 0.0
node 6 -360.0 324.0 -mass 0.6796996374935267 1e-09 1e-09
node 7 360.0 324.0
node 8 360.0 180.0
node 9 360.0 0.0
```

Figure 237. Script of node.tcl with previously defined nodes and masses

6. **elements.tcl:** contains all the element properties**,** including the extra nodes, such as those created for Zerolength elements.

```
# Extra nodes for zeroLength
# node tag x y z
node 10 0.0 0.0 0.0
node 11 0.0 180.0 0.0

# beam_column_elements elasticBeamColumn

# Geometric transformation command
geomTransf PDelta 1
element elasticBeamColumn 1 10 11 38.0892 29000.0 4435.484519239999 1

# zero_length_elements zeroLength
equalDOF 1 10 1 2
element zeroLength 15 1 10 -mat 2 -dir 3 -orient 0.0 1.0 0.0 -1.0 0.0 0.0
equalDOF 2 11 1 2
element zeroLength 16 2 11 -mat 2 -dir 3 -orient 0.0 1.0 0.0 -1.0 0.0 0.0

# Extra nodes for zeroLength
# node tag x y z
node 12 0.0 180.0 0.0
node 13 -360.0 180.0 0.0

# beam_column_elements elasticBeamColumn

# Geometric transformation command
geomTransf PDelta 2
element elasticBeamColumn 2 12 13 30.644 29000.0 3903.2259366666667 2

# zero_length_elements zeroLength
equalDOF 2 12 1 2
element zeroLength 17 2 12 -mat 4 -dir 3 -orient -1.0 0.0 0.0 0.0 -1.0 0.0
equalDOF 3 13 1 2
element zeroLength 18 3 13 -mat 4 -dir 3 -orient -1.0 0.0 0.0 0.0 -1.0 0.0

# Extra nodes for zeroLength
# node tag x y z
node 14 0.0 180.0 0.0
node 15 0.0 324.0 0.0

# beam_column_elements elasticBeamColumn

# Geometric transformation command
geomTransf PDelta 3
element elasticBeamColumn 3 14 15 38.0892 29000.0 4435.484519239999 3
```

Figure 238. Script of all element properties

7. **analysis_step:** contains constraints (in this case fix constraint and equalDOF), Patterns (for loads, constraints, and UniformExcitation), misc_commands (region), and the last required step is the analysis command. This structure may be repeated for additional analyses, like in the example below.

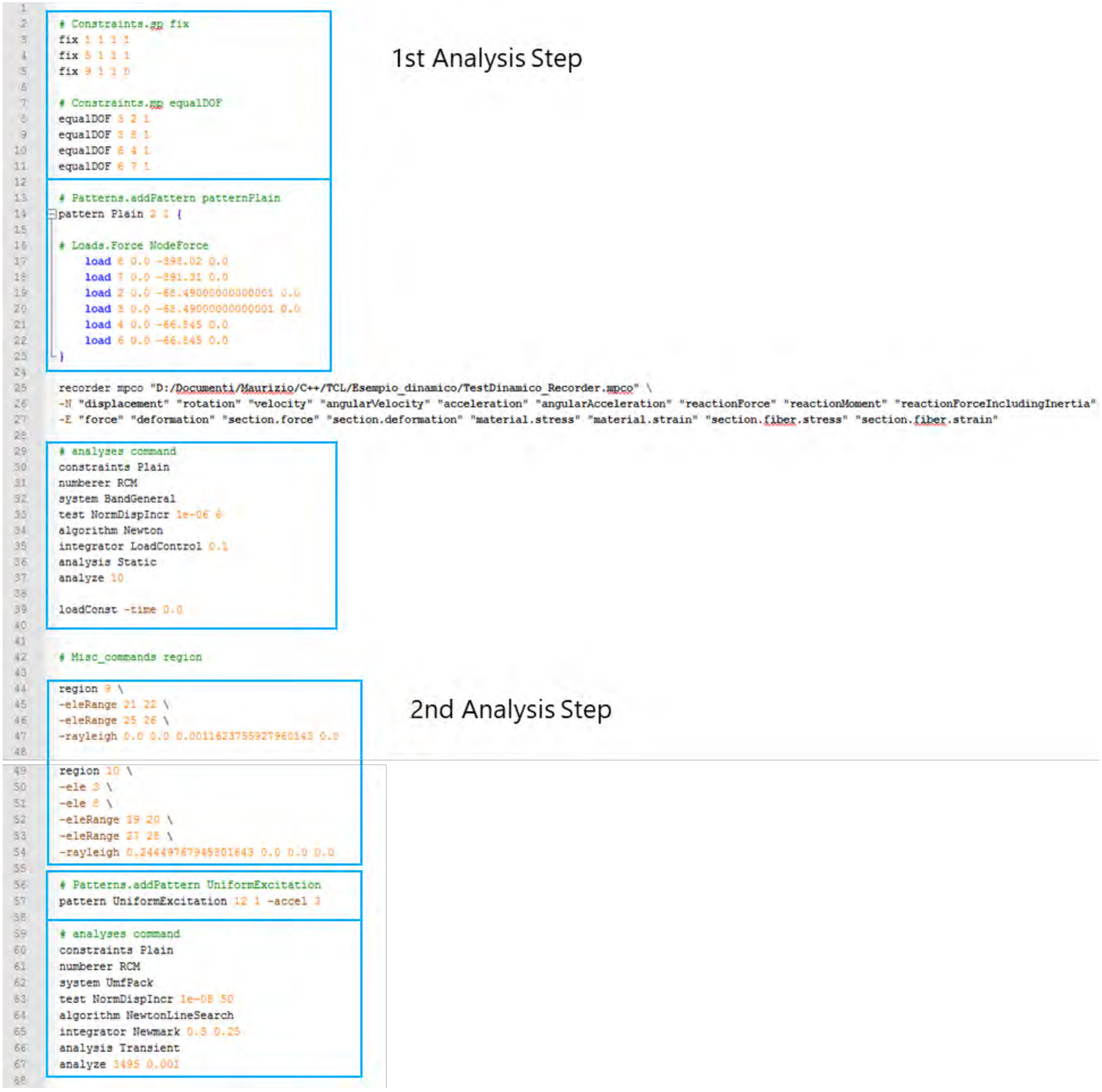


Figure 239. Analysis steps

2.9.4.3. Run analysis

Once the solver has been selected, the user can choose whether to write the input file or to directly run the analysis. To run the analysis, *Click* **Run analysis** from the main Toolbar.

After *clicking* **Run Analysis**, **STKO** will connect to Opensees and generate 7 ticle (.tcl) files, an .mpco.cdata file, and an .mpco file. The ticle (.tcl) files contain the description of the model to be solved by Opensees. These 7 files will be, the main.tcl, definitions.tcl, materials.tcl, sections.tcl, nodes.tcl, and elements.tcl. The .mpco.cdata translates the graphic properties of the model created in the preprocessor and makes them visible in the postprocessor.

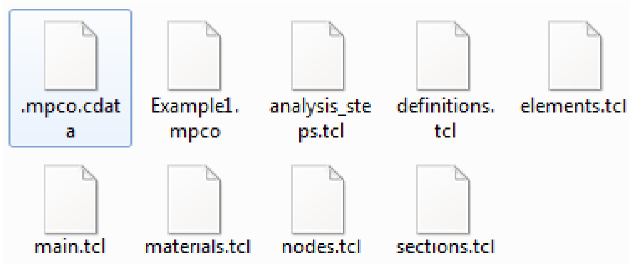


Figure 240.Example of the creation of ticle, .mpco.cdata, and .mpco files

The analysis toolbar also has two drop down menus:

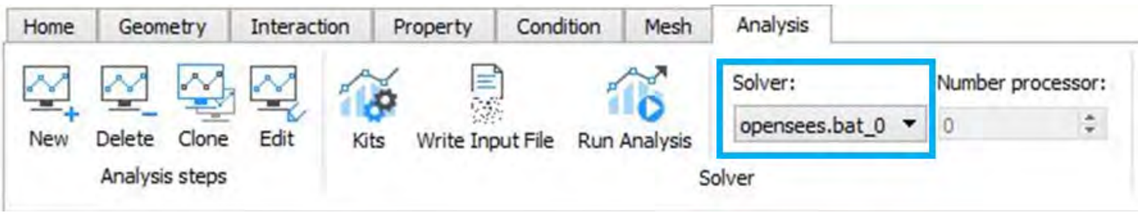


Figure 241. Solver list

The first one lists all the possible solvers that can be used which are **OpenSees, OpenSeesMp, and OpenSeesSp.**

When choosing **OpenSees** as the solver, the number of processors used in the analysis will be automatically determined.

When choosing **OpenSeesMp** as the solver, the number of implied processors will be stated according to the number of partitions (Refer to § 1 Partition).

When choosing **OpenSeesSp** as the solver, the number processor drop-down menu will appear in which the user can select the number processor to use in the analysis.

3. POSTPROCESSING MODULE

The user can visualize the results of the analyzes with the Postprocessing module and the Opensees Databases. The STKO Postprocessor allows the user to analyze the results through the creation of plots, graphs, and other tools. This section of the manual will explain how to use the key features.

3.1 Working with Databases

To Open an OpenSees Database from the Postprocessor interface, *Right-click* **Databases** > **Open Database** on the Work Tree panel, or *Click* **Open DB** on the main Toolbar.

Select the **MPC Output Database** file (*.mpco).

The imported Database will appear in the Work Tree panel, as shown below.

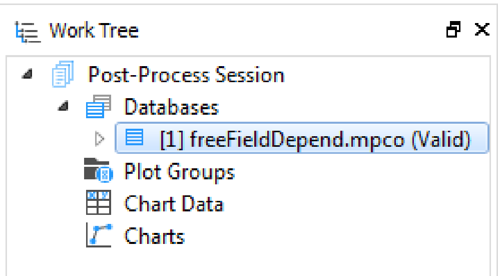


Figure 242. Creation of a new mpco output database

To close the Database, select **Close DB** from the Toolbar, or *Right-click* the Database on the Work Tree panel and *Click* **Close Database**.

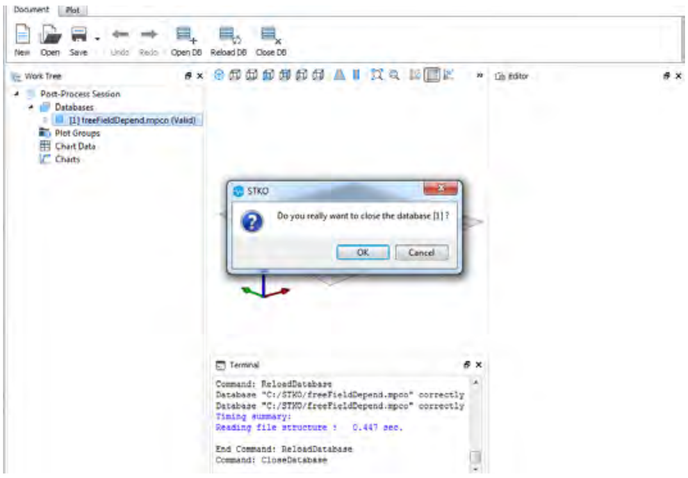


Figure 243. STKO close database window

To reload the Database, *Click* **Reload DB** on the Toolbar, or *Right-click* the Database on the Work Tree panel. Select **Reload Database**.

3.2 Creating Plot Groups

To analyze results in the postprocessor, choose the command **Plot** > **New plot group** from Toolbar or *Right-click* **Plot Groups** > **New Plot Group** from the Work Tree panel.

To delete a plot group choose the command **Plot** > **Delete plot group** from the main Toolbar, or *Right-click* the desired plot group on the Work Tree panel and select **Delete Plot Group.**

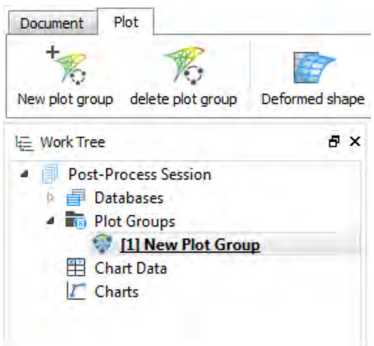


Figure 244. Creation of the New Plot Group

3.2.1 Deformed Shape Plot

To create a Deformed Shape Plot, *Right-click* **New Plot Group** > **Deformed Shape** on the Work Tree panel, or *Click* **Plot** > **Deformed Shape** from the Toolbar.

Click on the **Deformed Shape** on the Work Tree Panel to open the editor.

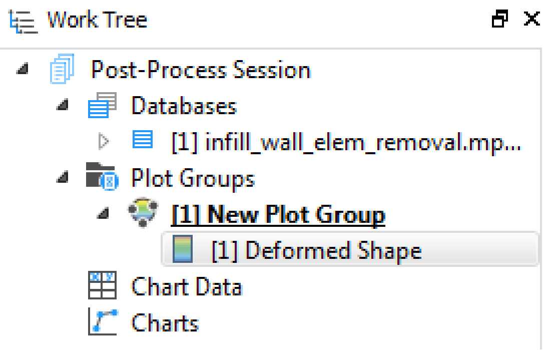


Figure 245. Creation of a Deformed Shape plot

An Editor panel will appear allowing the user to modify and select the different elements of the deformed shape, as shown below.

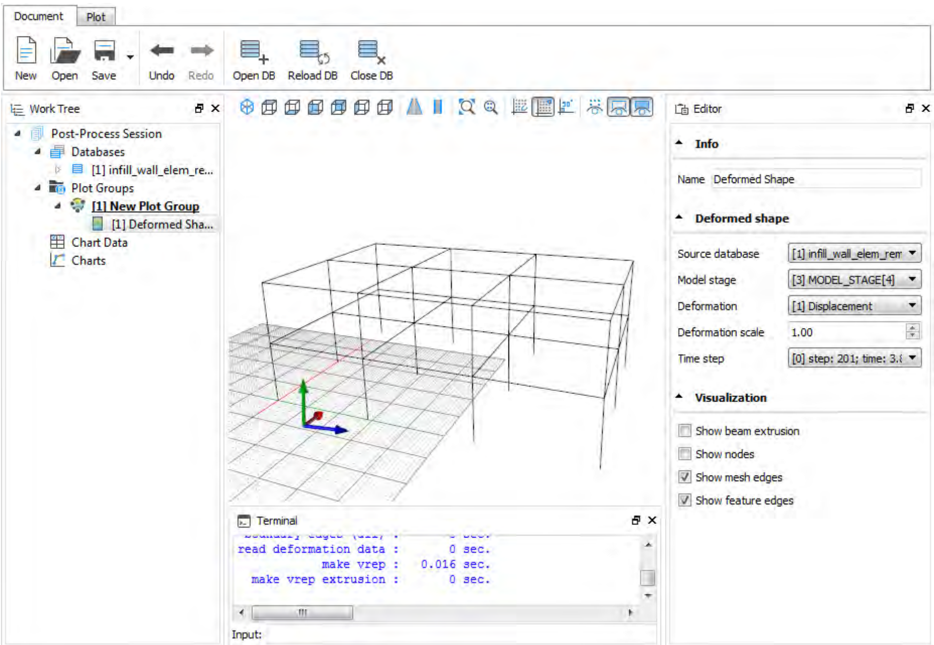


Figure 246. Selection of deformation scale

The next image shows what happens when the Deformation scale is changed to a value of 20.

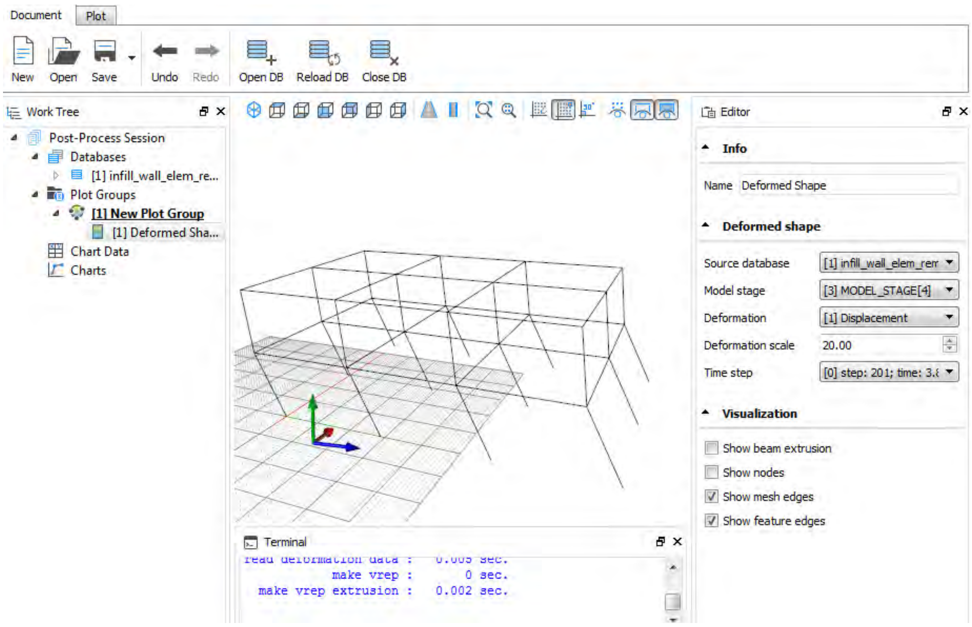


Figure 247. Amplified deformation

To Delete the **Deformed Shape**, *Right-click* on **Deformation Shape** > **Delete Plot** on the Work Tree Panel.

To Add new Plots to analyze, *right-click* **New Plot Group** on the Work Tree panel and choose the desired Plot type.

The Deformed Shape Plot editor now includes an option called custom color scheme. It changes the visualization in the render view and can show, for example, physical or element property assignments. It can also be used for SolverElementType, Geometry, and Geometry Subshape. This feature is particularly useful when STKO is only used to postprocess a model as it allows users to view the preprocessing assignments made in the tcl script.

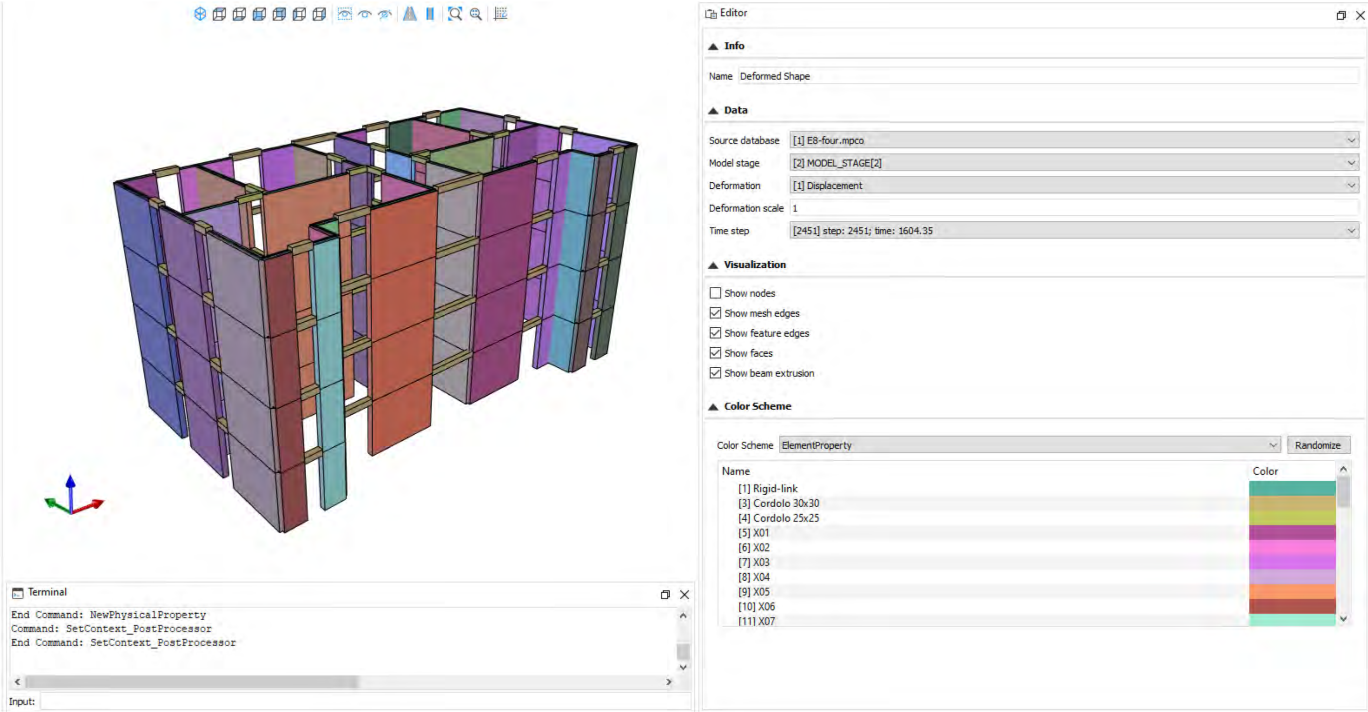


Figure 248. Custom Color Scheme for Deformed Shape Plot

3.2.2 Surface Color Map Plot

To create a Surface Color Map Plot, *right-click* on **New Plot Group** from the Work Tree panel and Select **Surface Color Map** or choose the command **Plot** > **Surface Plot** from the Toolbar.

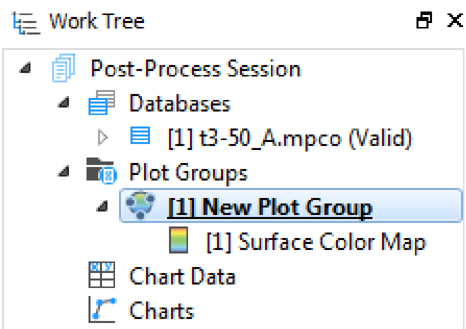


Figure 249. Creation of a Surface Color Map

Clicking on the **Surface Color Map** in the Work Tree will open the Editor Panel, allowing the user to modify any editable information of the **Surface Color Map Plot**, as shown in the following image.

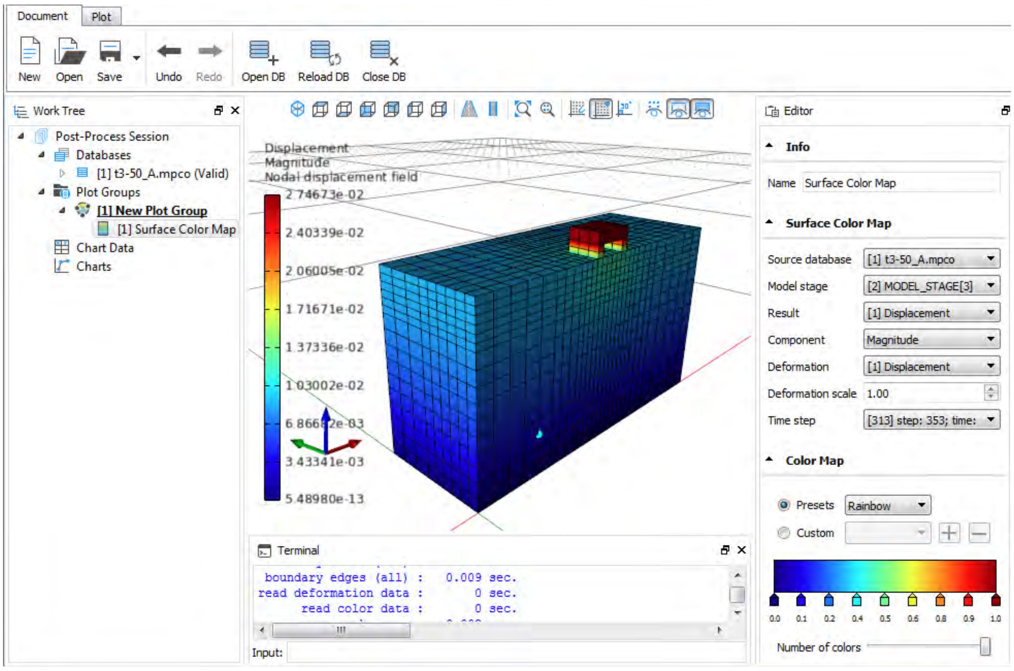


Figure 250. Surface Color Map editor panel

To Delete the Plot, *Right-click* on **Surface Color Map Plot** > **Delete Plot**. To Add new Plots to analyze, *Right-click* on **New Plot Group** and choose the desired Plot type.

3.2.3. Volume Color Map Plot

To create a **Volume Color Map Plot**, *Right-click* on **New Plot Group** on the Work Tree panel and Select **Volume Color Map** or choose the command **Plot** > **Volume Plot** from the Toolbar

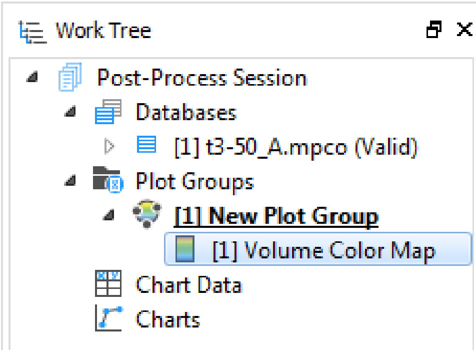


Figure 251. Creation of a Volume Color Map

Clicking on the **Volume Color Map** in the Work Tree will open the Editor Panel, allowing the user to modify any editable information of the **Volume Color Map Plot**, as shown in the following image.

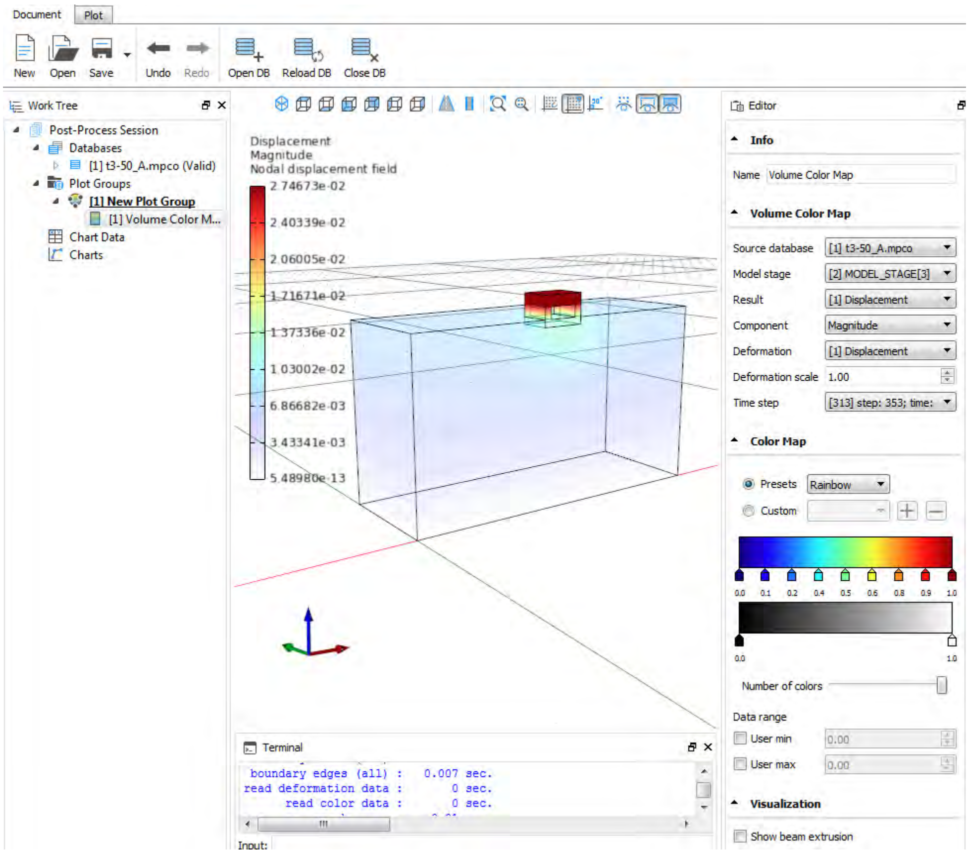


Figure 252. Editor of the Volume Color Map

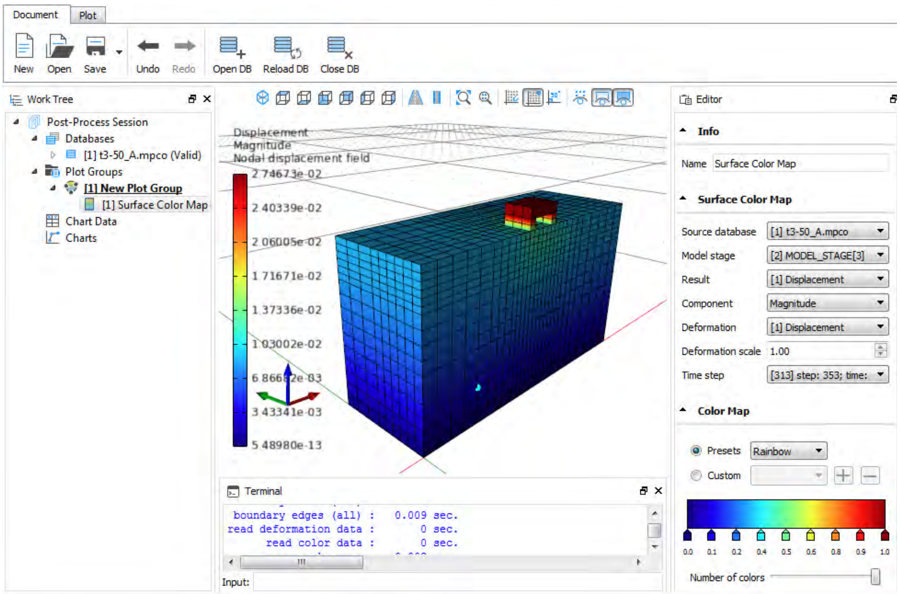


Figure 253. Surface Color Map editor panel

To Delete the Plot, *Right-click* on **Volume Color Map Plot** > **Delete Plot**. To Add new Plots to analyze, *Right-click* on **New Plot Group** and choose the desired Plot type.

3.2.4. Vector Plot

To create a Vector Plot, *Right-click* **New Plot Group** on the Work Tree panel and Select **Vector Plot** or choose the command **Plot** > **Vector Plot** from the Toolbar.

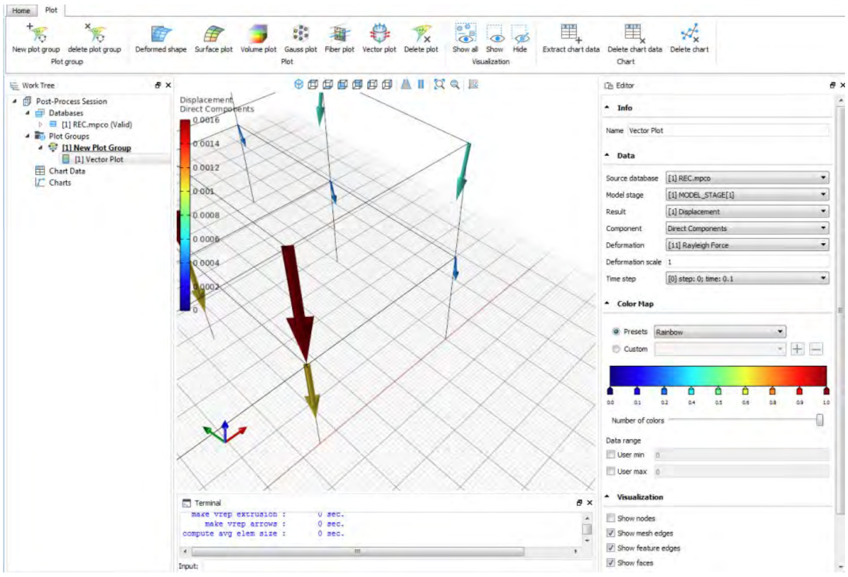


Figure 254. Vector Point Plot

Clicking on the **Volume Color Map** in the Work Tree will open the Editor Panel, allowing the user to modify any editable information of the **Volume Color Map Plot**, as shown in the following image.

To Delete the Plot, *Right-click* on **Volume Color Map Plot** > **Delete Plot**. To Add new Plots to analyze, *Right-click* on **New Plot Group** and choose the desired Plot type.

3.2.5. Gauss Point Plot

To create a Gauss Point Plot, *Right-click* **New Plot Group** on the Work Tree panel and Select **Gauss Point Plot** or choose the command **Plot** > **Gauss Plot** from the Toolbar.

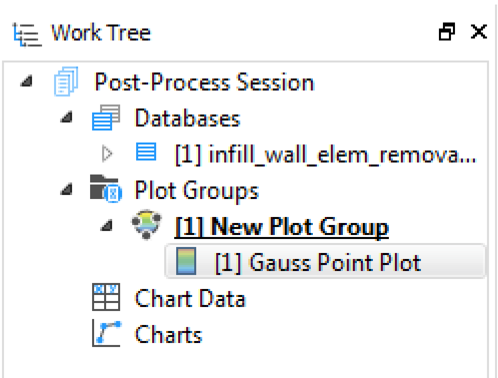


Figure 255. Creation of the Gauss Point Plot

Clicking on the **Gauss point plot** in the Work Tree will open the Editor Panel, allowing the user to modify any editable information of the **Gauss Point Plot**, as shown in the following image.

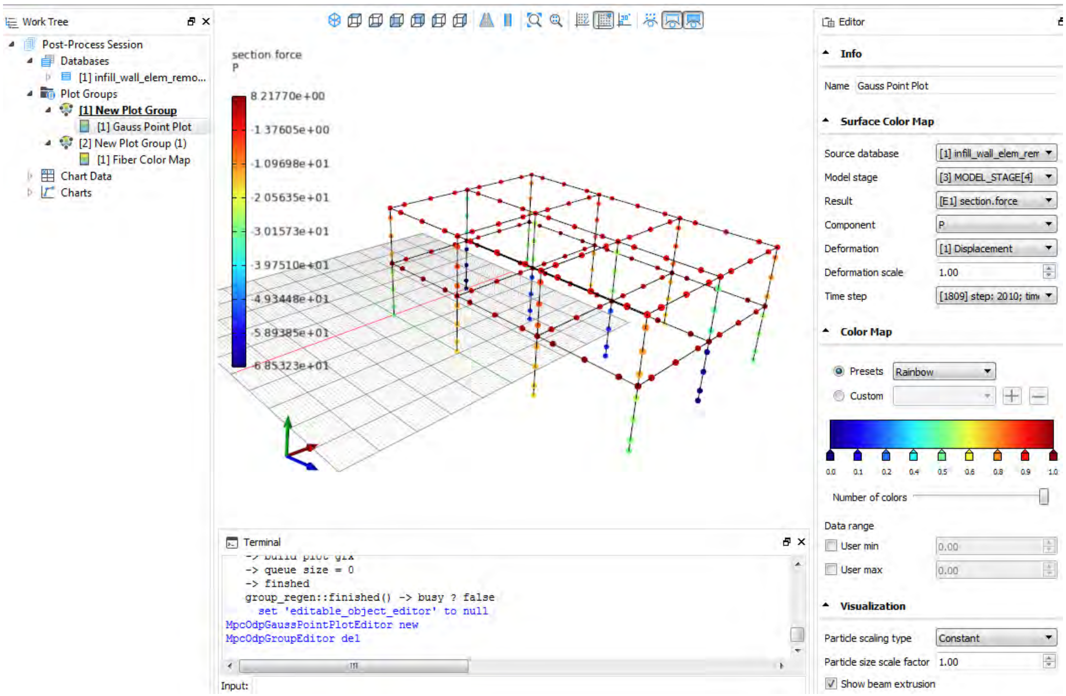


Figure 256. Gauss Point Plot Editor

Using the Editor Panel, users can insert a shrink factor to avoid the overlap of gauss points on the plot, as modeled in the figures below.

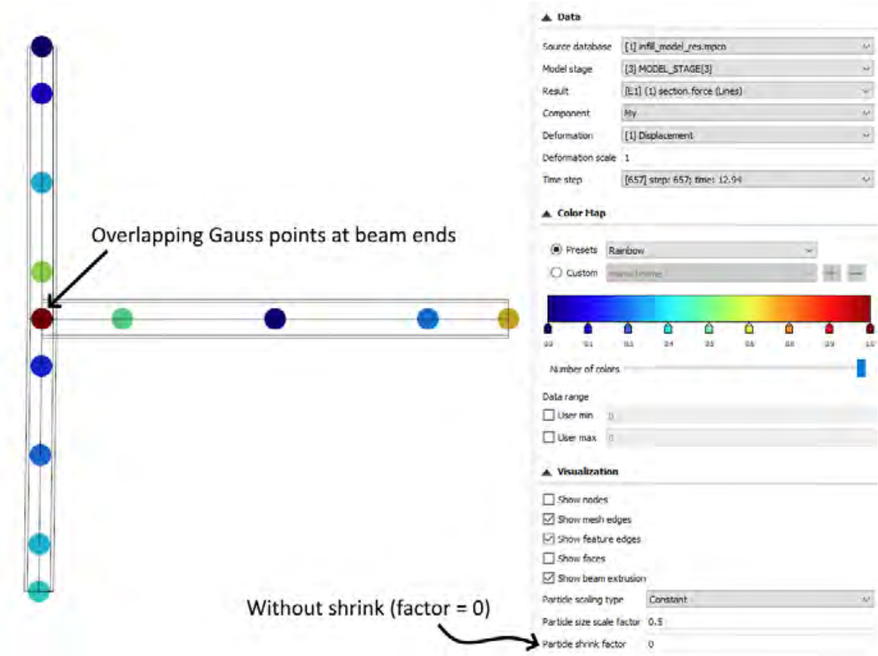


Figure 257. Overlapping Guass Points

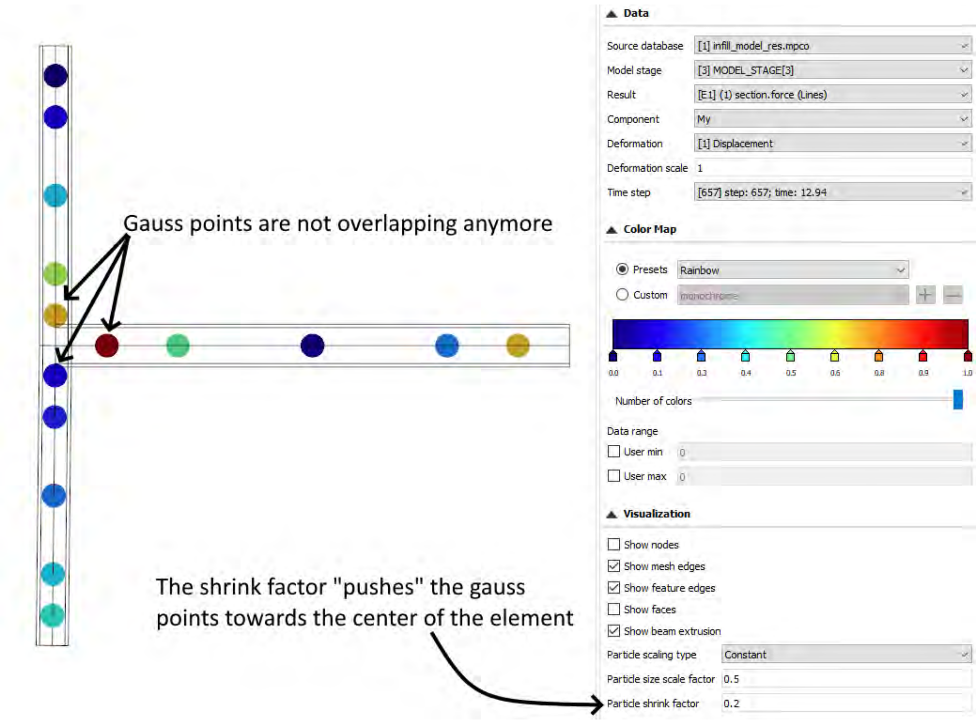


Figure 258. Setting the Particle Shrink Factor

To Delete the Plot, *Right-click* on **Gauss Point Plot** > **Delete Plot**. To Add new Plots to analyze, *Right-click* on **New Plot Group** and choose the desired Plot type.

3.2.6. Fiber Section Plot

To create a Fiber Section Plot, *Right-click* on **New Plot Group** on the Work Tree panel and Select **Beam/Shell Fiber Color Map** or choose the command **Plot** > **Fiber Plot** from the Toolbar.

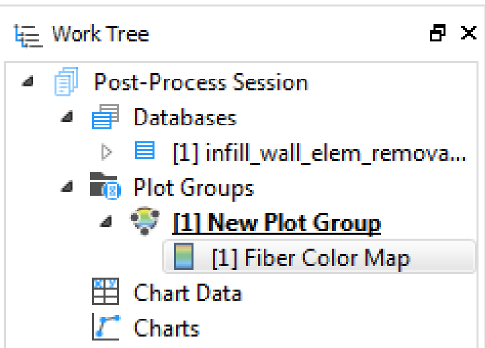


Figure 259. Creation of the Fiber Color Map

Clicking on the **Fiber Color Map** in the Work Tree will open the Editor Panel, allowing the user to modify any editable information of the **Fiber Color Map Plot**, as shown in the following image.

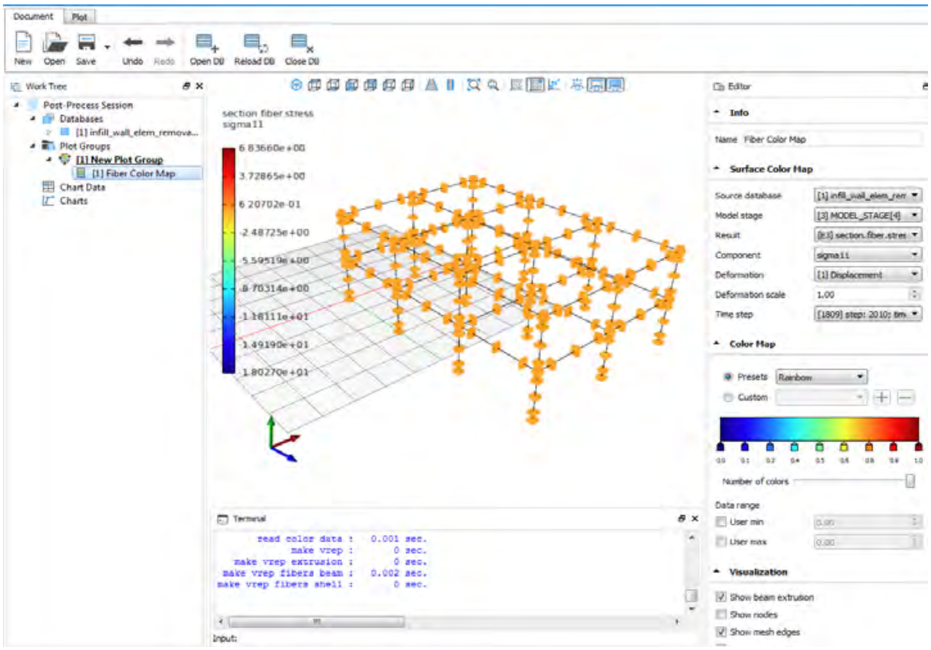


Figure 260. Fiber Color Map Editor

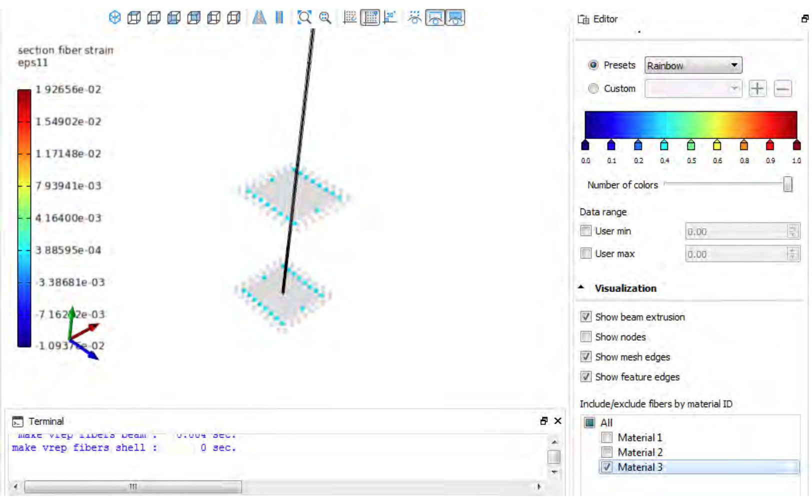


Figure 261. Selection of the steel material

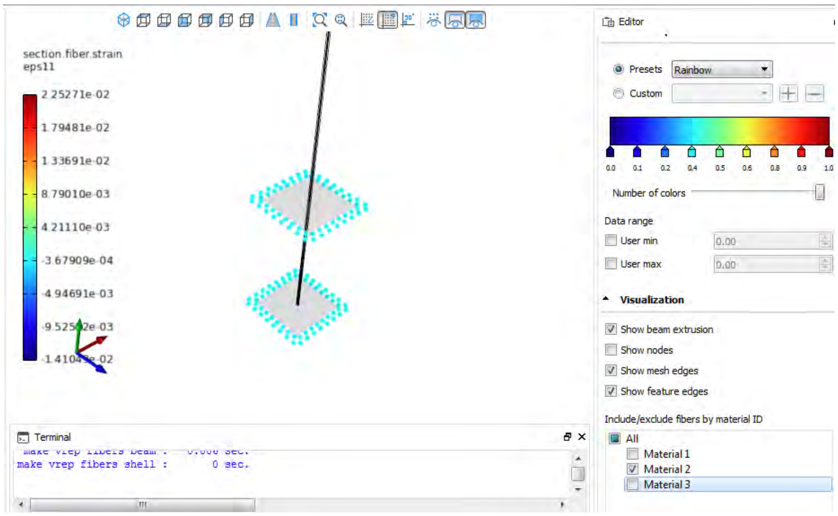


Figure 262. Selection of the concrete cover

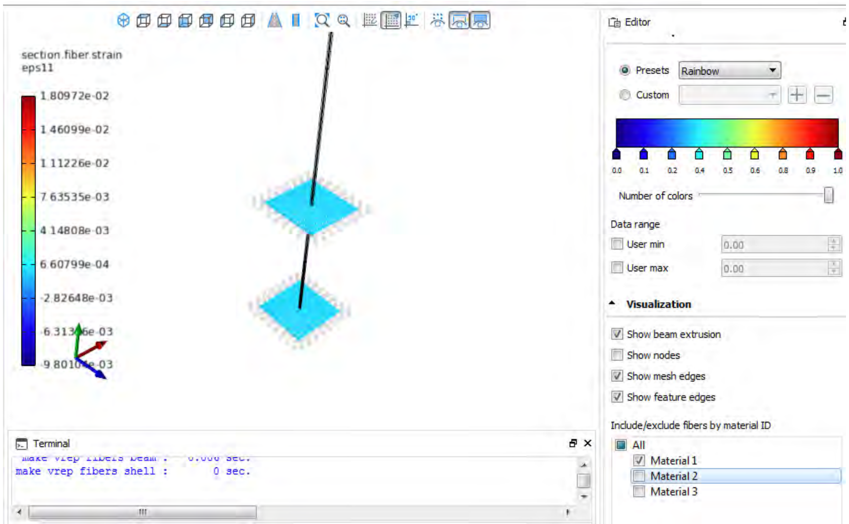


Figure 263. Selection of the concrete core

To Delete the Plot, *Right-click* on **Fiber Color Plot** > **Delete Plot**. To Add new Plots to analyze, *Right-click* on **New Plot Group** and choose the desired Plot type.

3.2.7. Beam Diagram Plot

STKO Version 1.1.5 introduces the Beam Diagram Plot. The Beam Diagram Plot allows users to view results on beam elements. The results referred to are localForce, which is a result on the element nodes, as well as section.force and section.deformation, which are results on element Gauss points. Results on element nodes have opposite signs at the two beam end-nodes, while results on Gauss points always have the correct sign.

Previously, if users attempted to view localForce (for example) by using a Surface Color Map, they would obtain results with inverted signs, making the results less easy to read and interpret. The Beam Diagram Plot internally changes the signs of results at the element end-nodes.

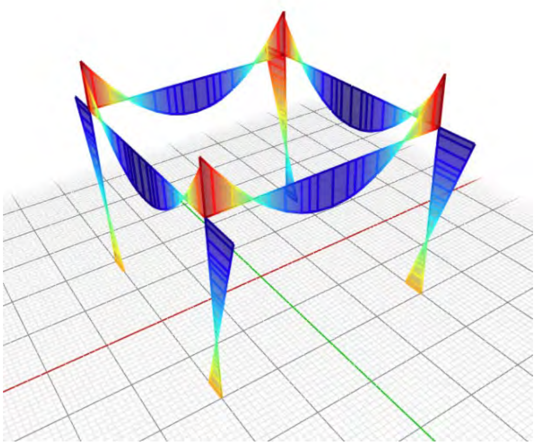


Figure 264. Example Beam Diagram Plot

As results on element nodes, such as local force, are stored at the element nodes, the diagram along the element length can at most be a line that linearly interpolates the two end values. To view a smoother result, the user should discretize the element. On the other hand, if you visualize results on Gauss points, the beam diagram plot will create a piece-wise linear interpolation along all the Gauss points, making the diagram naturally smoother because the number of Gauss points is typically larger than the number of the nodes.

To create a Beam Diagram Plot, *Right-click* on **New Plot Group** on the Work Tree panel and Select **Beam Diagram Plot** or choose the command **Plot** > **Beam Diagram Plot** from the Toolbar.

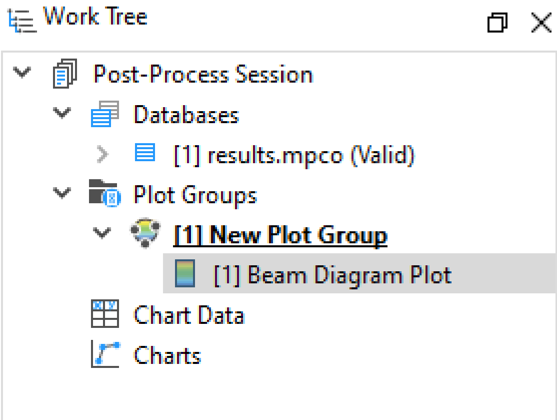


Figure 265. Creation of the Beam Diagram Plot

Clicking on the **Beam Diagram Plot** in the Work Tree will open the Editor Panel, allowing the user to modify any editable information, as shown in the following image. The result selections have been renamed from the previous S1, S2, S3, etc., to N, My, Mz, T, etc., to make them more easily identifiable.

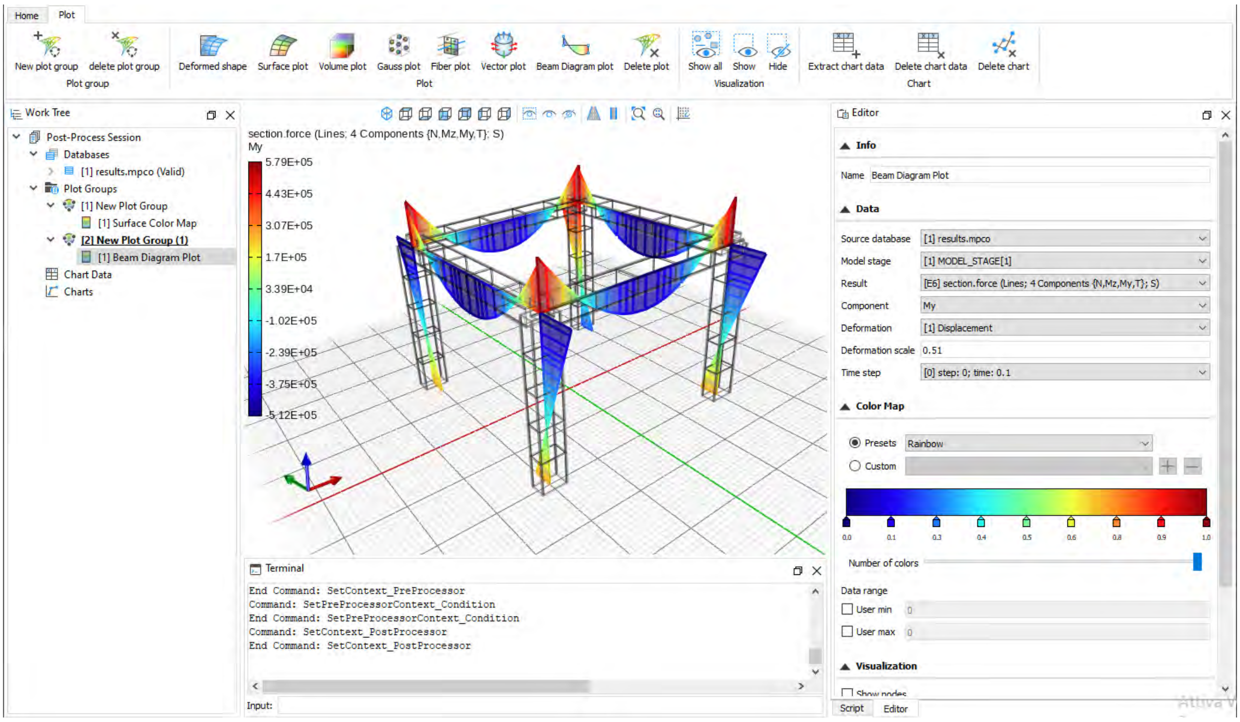


Figure 266. Beam Diagram Plot Editor

To Delete the Plot, *Right-click* on **Beam Diagram Plot** > **Delete Plot**. To Add new Plots to analyze, *Right-click* on **New Plot Group** and choose the desired Plot type.

3.2.8. Animation

STKO allows for the recording of a **video** of the time steps of a postprocessor model after plots have been defined. *Click* on the **New Plot Group** in the Work Tree Panel and the Editor panel will open the animation function, as shown in the following image.

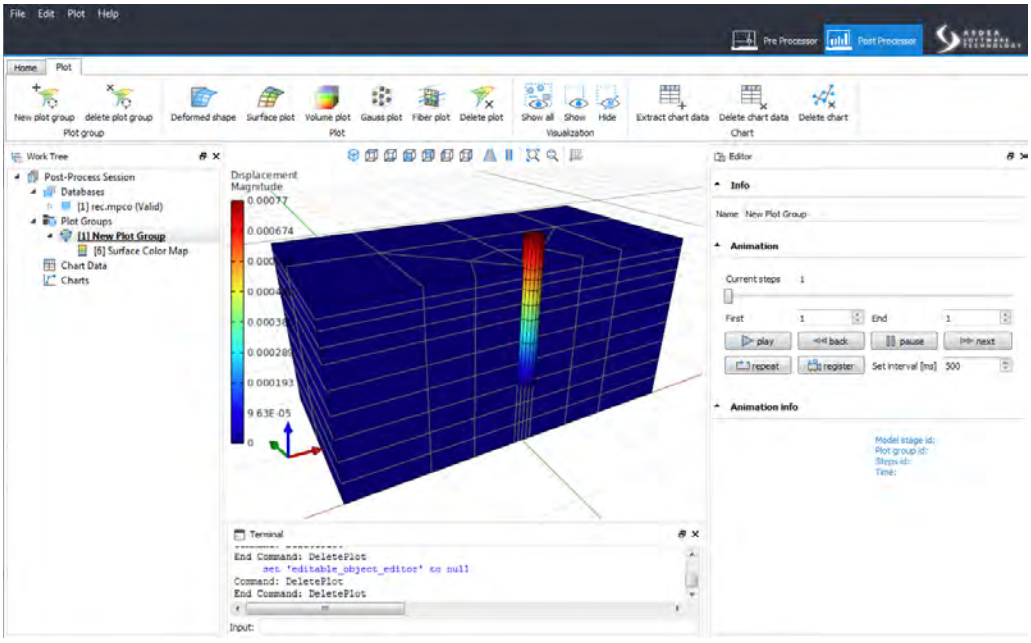


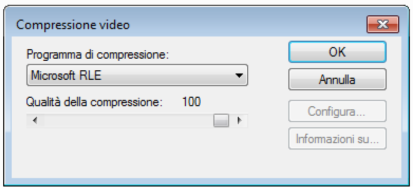
Figure 267. Recording a video

- Set the **Start** and the **End** frames
- *Click* **play** to view the animation.

- When you are ready to record the animation, *click* record.

- The Play button will convert into a **record** button.

- The user will need to name the file, and choose the codec. They can then press the **rec** button and begin recording; every motion of the model will be recorded by STKO.



- *Click* **Stop** to interrupt the recording. The video will automatically save to the previously defined path.

The user may also choose to record the video with or without the **write info** (Model stage Id, Plot group id, Steps Id, Time), by checking the box, as modeled in the following images.

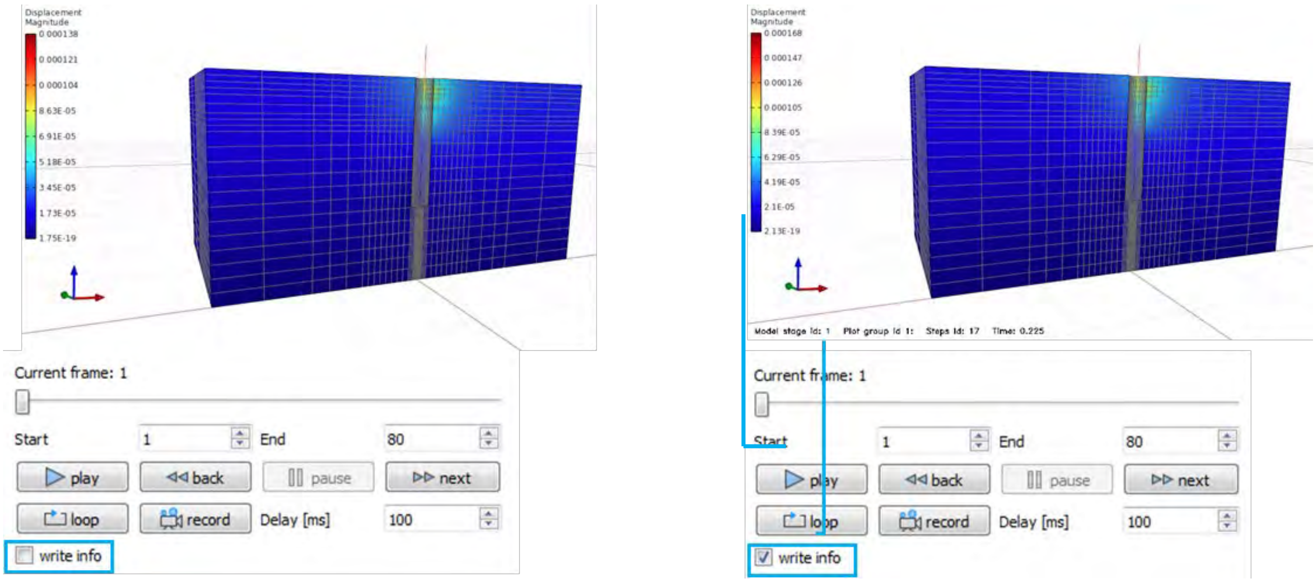


Figure 268. Video with and without information

3.2.9. Managing Tensor and Vector Results

In general, Opensees Elements do not export dynamic invariants. **STKO** overcomes this thanks to the Postprocessor. **Tresca**, **Von Mises**, **J_2, Octahedral tension** are calculated by means with invariants:

SYMBOL	DESCRIPTION	EXPRESSION
$\bar{I}_1^{\varepsilon}$	First modified invariant of Green strain tensor	$\bar{I}_1^{\varepsilon} = \frac{1}{2}(\bar{I}_1 - 3)$
$\bar{I}_2^{\varepsilon}$	Second modified invariant of Green strain tensor	$\bar{I}_2^{\varepsilon} = \frac{1}{4}((\bar{I}_2 - 3) - 2(\bar{I}_1 - 3))$
$\bar{I}_3^{\varepsilon}$	Third modified invariant of Green strain tensor	$\bar{I}_3^{\varepsilon} = \frac{1}{8}((\bar{I}_1 - 3) - (\bar{I}_2 - 3))$
σ_{tresca}	Tresca stress	$\max(\max(\lvert\sigma1 - \sigma2, \lvert\sigma2 - \sigma3\rvert), \lvert\sigma1 - \sigma3\rvert)$
σ_{mises}	Von Mises stress	$\sqrt{3 \cdot J_2}$
τ_{oct}	Octahedral tension	$\sqrt{2/3 \cdot J_2}$

The following example shows three types of geometries in the Postprocessor: a Solid, a Shell, and a Beam element.

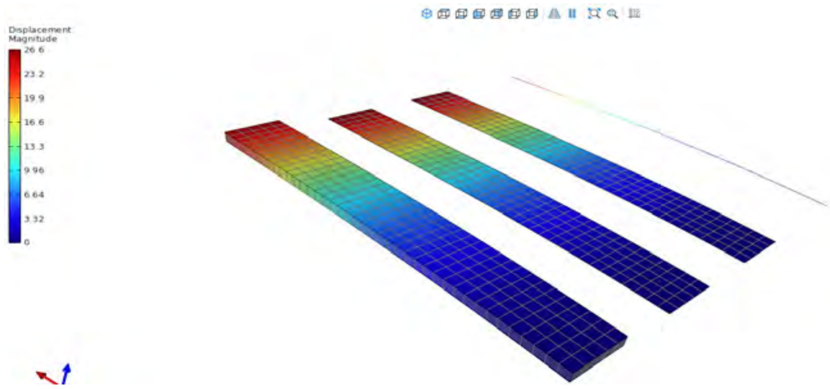


Figure 269. Post processor examples

It is important to highlight how each geometry is associated to a result - i.e. E2 section force (Surfaces):

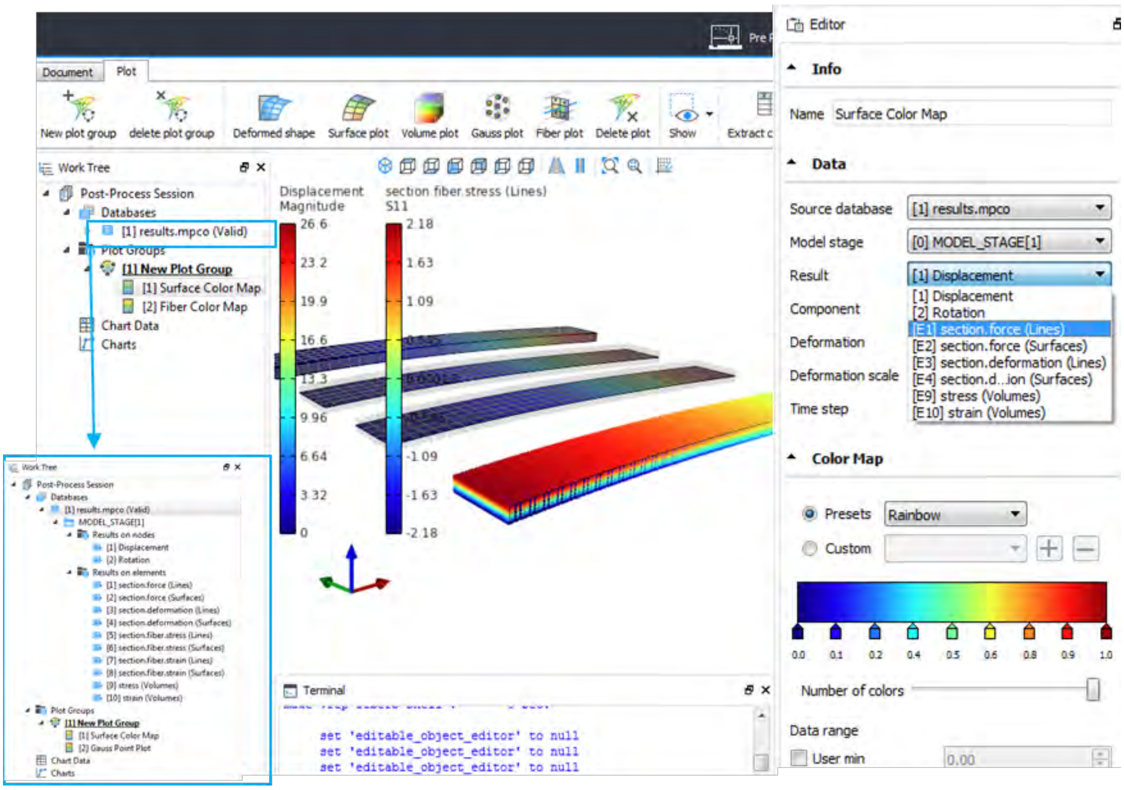


Figure 270. Geometries are associated with the results

Varying the results (E1, E2, E3, E4, E9, etc.) will only change the color map of the geometries related to the result.

Element	Visualization
Shell 	Fiber Plot
Beam/Column 	Fiber Plot
Solid 	Face Plot

Figure 271. Element Type and Visualization Chart

STKO may give scalar, vector, and tensor results. These results must be expressed in a coordinate system. Vector and tensor results are oriented according to three reference systems: Global Axes, Local Axes, and Element Local.

3.2.9.1. Local Result Orientation

STKO may give scalar, vector, and tensor results. These results must be expressed in a coordinate system. Vector and tensor results are oriented according to three reference systems: Global Axes, Local Axes, and Element Local.

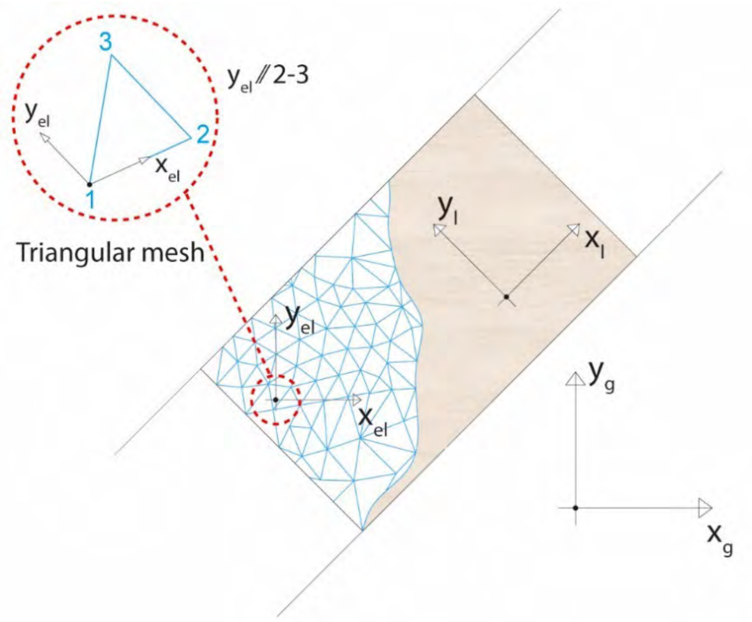


Figure 272. Local orientation results

Opensees saves the results in .mpco, considering:

- x_g, y_g, z_g = **Global Axes** (default);
- x_l, y_l, z_l = **Local Axes** (user defined);
- x_{el}, y_{el}, z_{el} = **Element Local** (typically of the element).

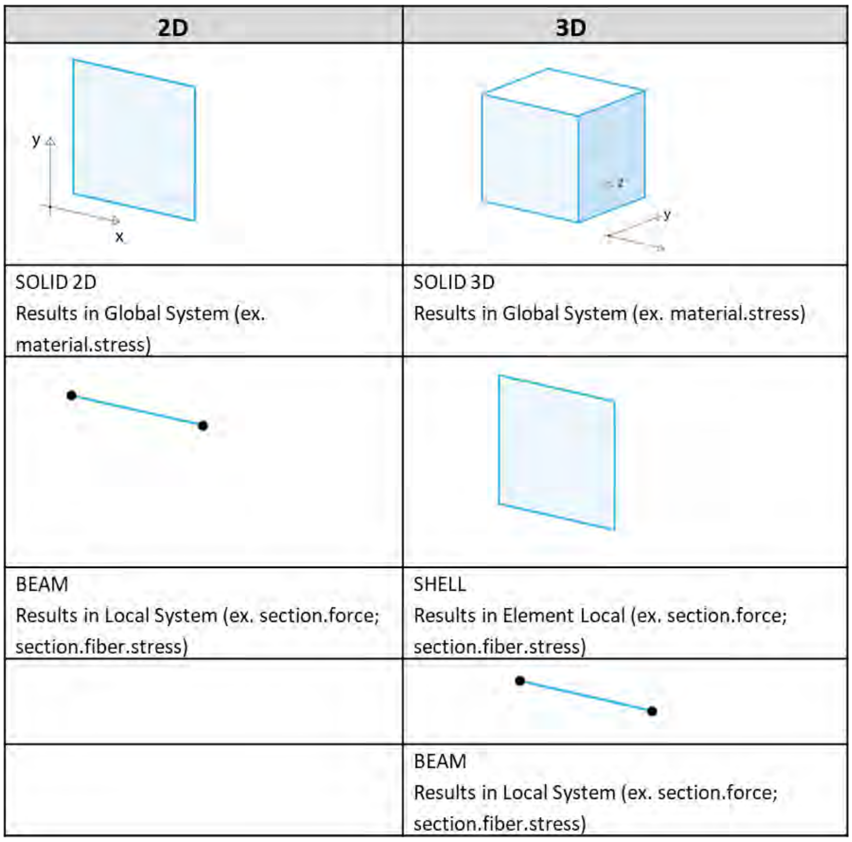


Figure 273. Result Types

The vector and tensor results in the local system are explained in the chart below.

Element	**<u>OpenSees</u>**	STKO Scientific ToolKit for OpenSees
2D and 3D Solids	Global System	Global System
Beam	Local System	Local System
Shell	Element Local	STKO rotates axes according to Local System

Figure 274. Vector and Tensor Results

The example below shows this important feature using two plates with same dimensions and same reference system (global axes [x-y]) but with different element properties: a shell and a 3D solid.

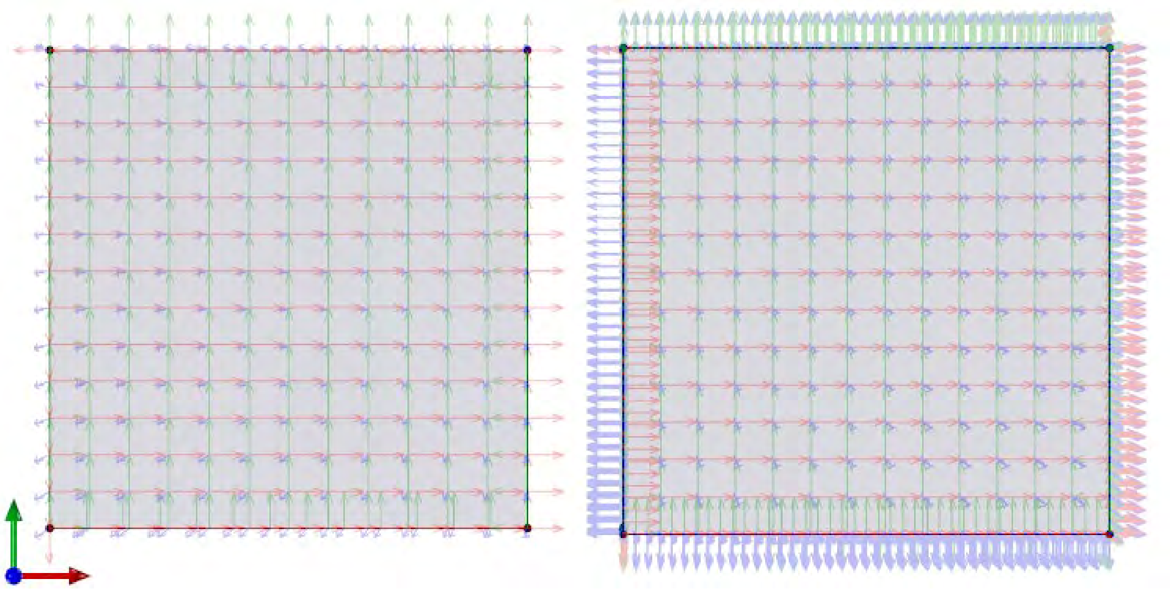


Figure 275. Shell (left) and 3D ELEMENT with a vertical Pressure (right)

Rotating the plates by 30°, the global axes remain unchanged in their initial configuration (x-y):

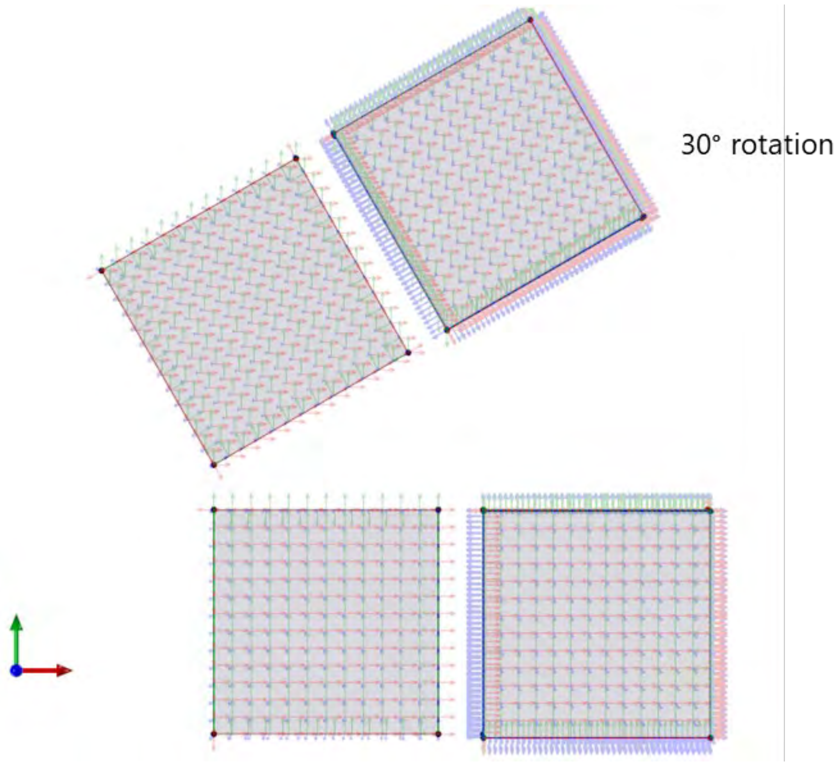


Figure 276. Global axes

The user will see different results, even if the elements are the same, between the non-rotated plates and the rotated ones because rotation changes the coordinate system. The user can view these differences in the postprocessor after launching the analysis.

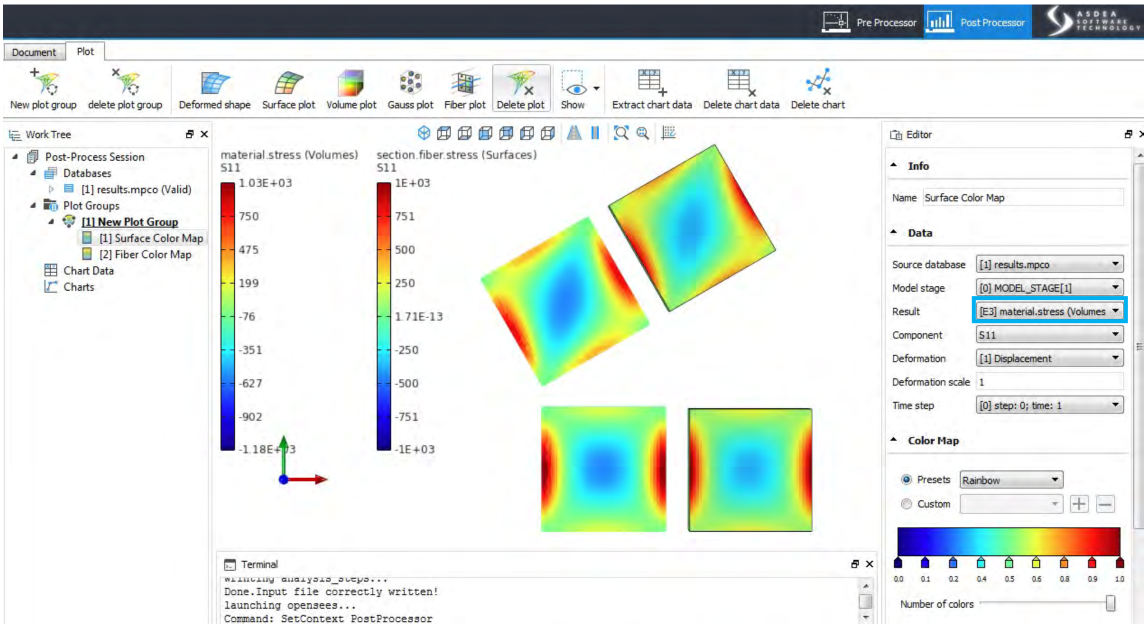


Figure 277. material.stress Color Map

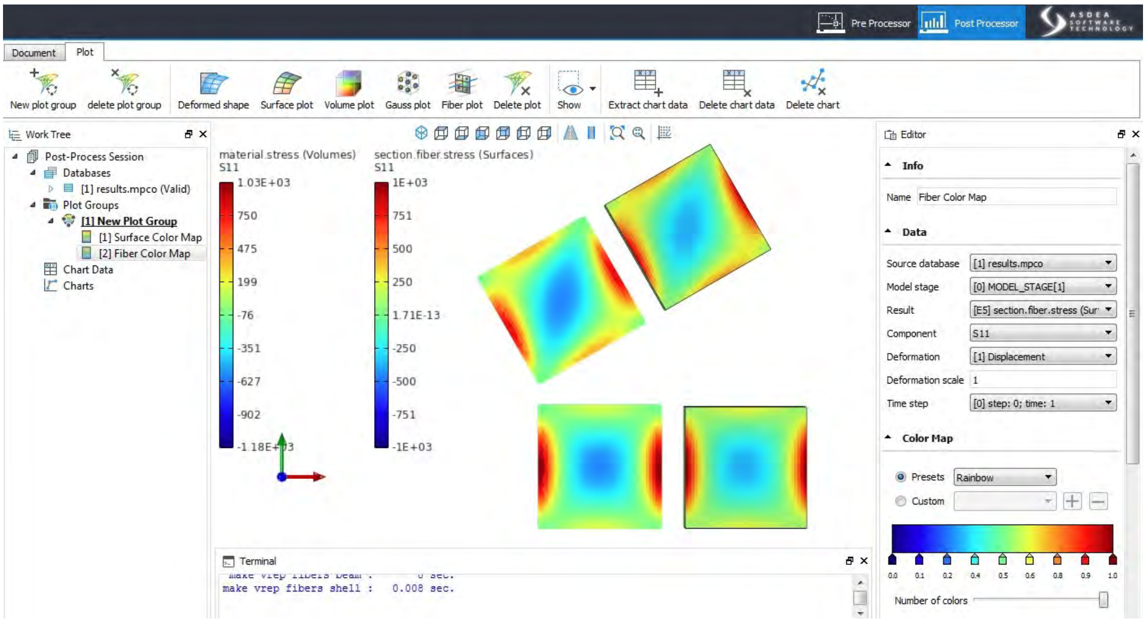


Figure 278. section.fiber.stress Color Map

Results will remain unchanged only for invariants.

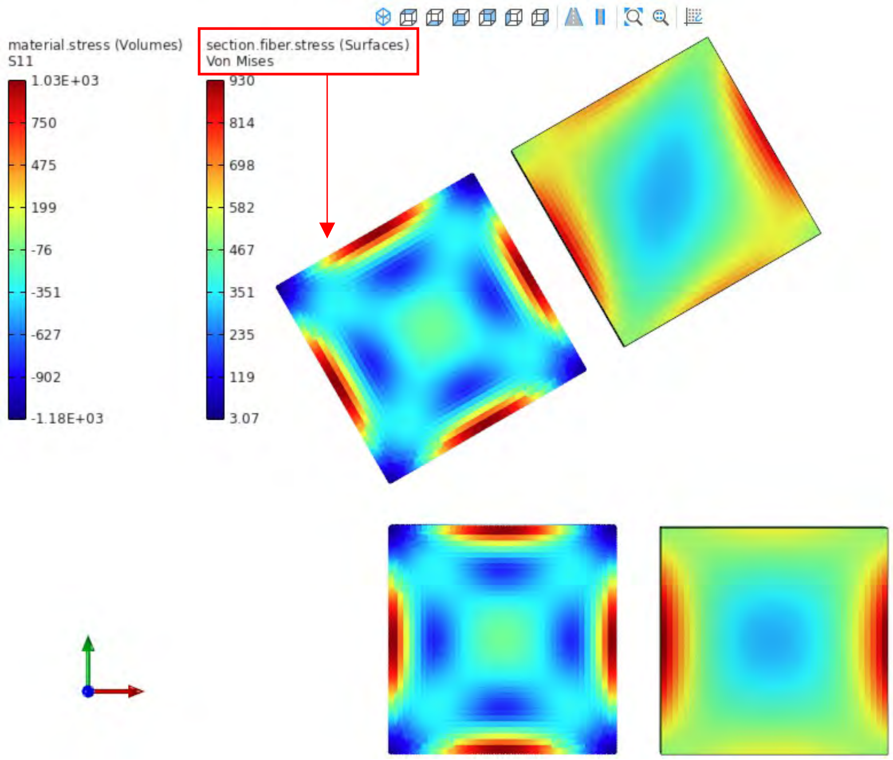


Figure 279. Invariant Results, e.g. Von Mises Stresses

Using local axes, the results are as follows:

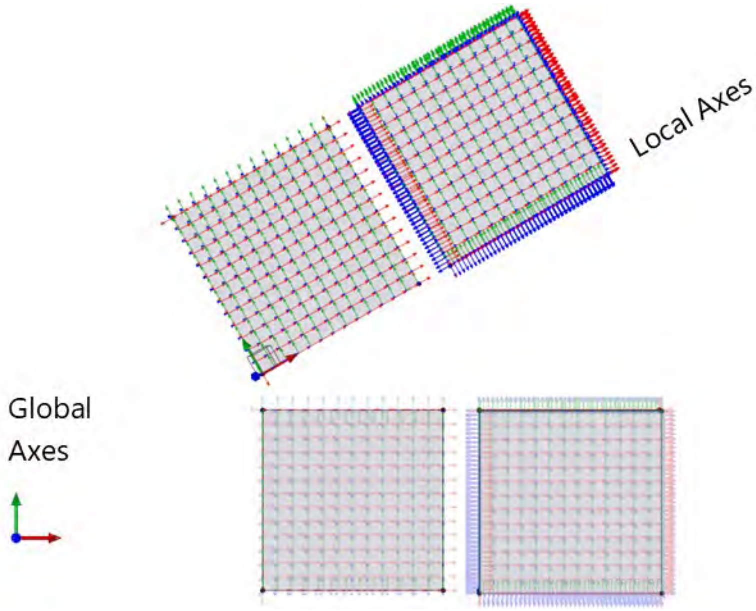


Figure 280. Local Axes

The **STKO Post-Processor** will show that there are no differences in the results:

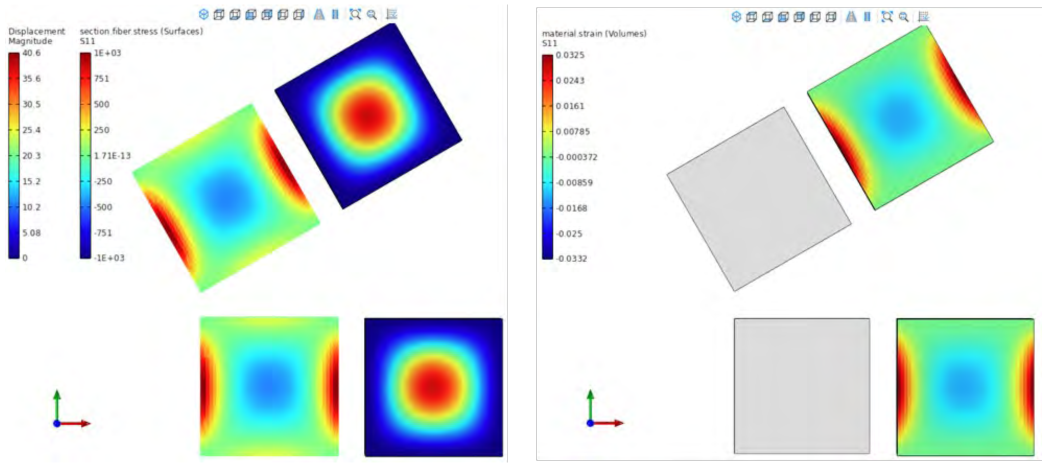


Figure 281. Local Axes Results

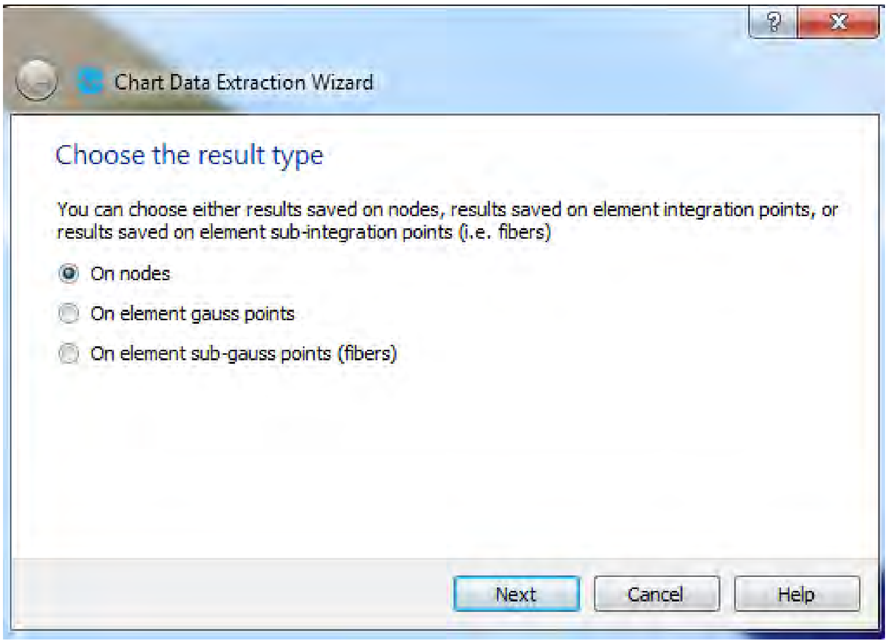
3.3 Extracting Chart Data

Chart Data is another command used to create charts.

3.3.1 Extracting Results on Nodes

Right-click **Chart Data** > **Extract Chart Data** on the Work Tree panel, or from the Toolbar *Click* > **Plot** > **Extract chart data**. A **Chart Data Extraction Wizard** will appear. It allows the user to choose the result type (1) and then the result to extract (2). The result to extract will depend on the result type chosen (i.e. on nodes, on gauss or on fibers).

Choose the result type, for this example select **On nodes** and *Click* **Next**.



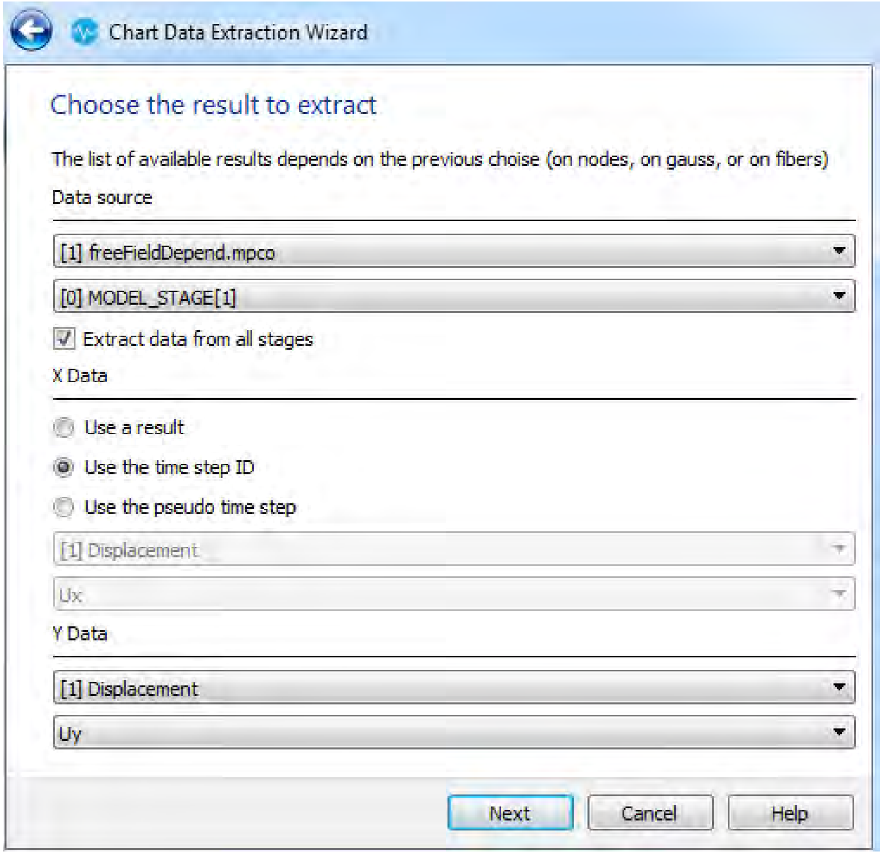


Figure 282. Chart Data Extraction Wizard

After choosing the results to extract, *click* **Next** and **Select** the locations (nodes) for chart data extraction. The selected nodes will appear on the list in the Extraction Wizard.

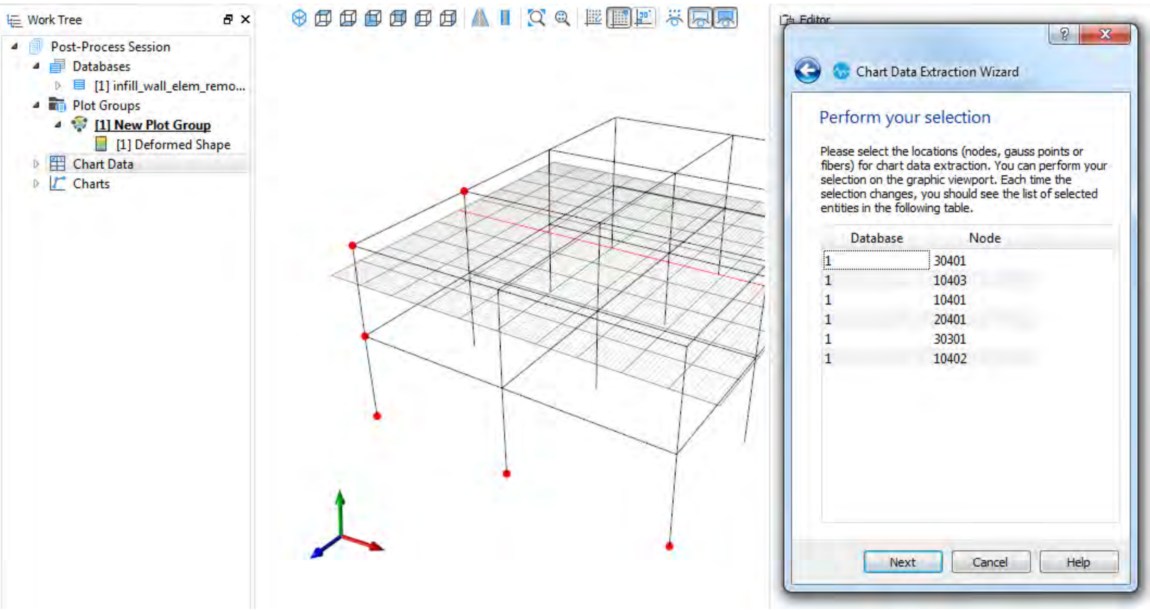


Figure 283. Selction of the Nodes

Click **Next** to assign a name to the Chart Data and set how the wizard should handle multiple items; then *Click* **Finish**.

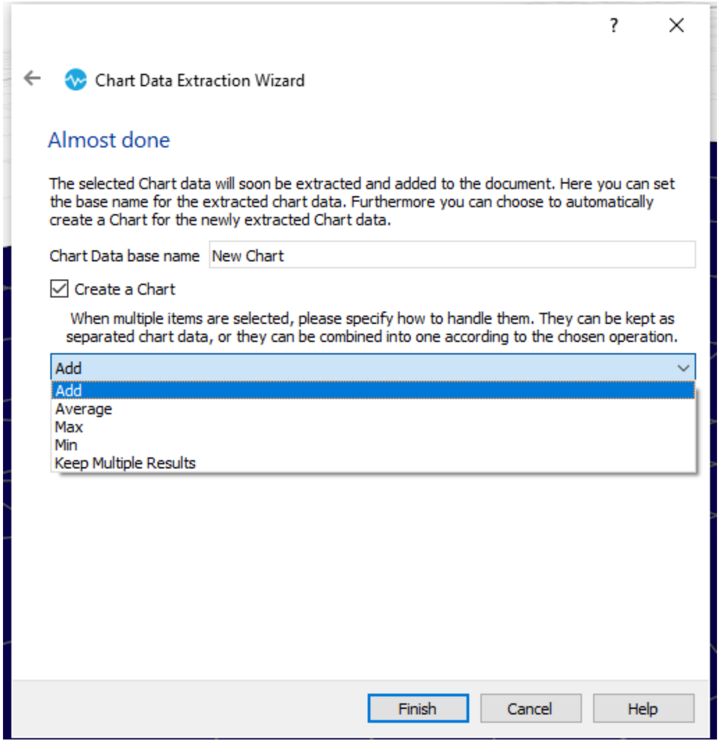


Figure 284. Chart Data Extraction Wizard

The new Chart will be visible in the Editor Panel and in the Work Tree.

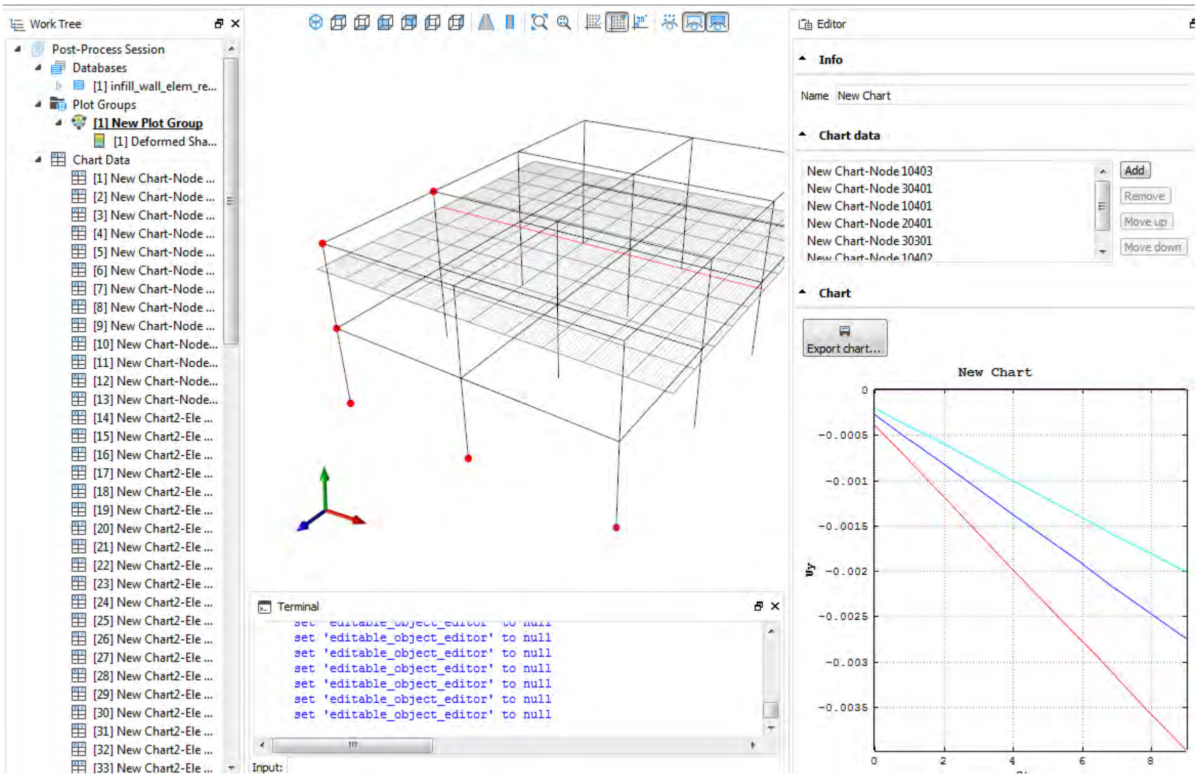
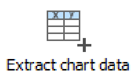


Figure 285. Plot Charts

The **Chart Data Tab** can display multiple charts, which are shown on the Work Tree will also show a list of plot data.
The Chart graphically displays plot data which can be exported by the user. *Click* on **Export Chart** on the Editor Panel and choose the available format: *PDF, SVG, Postscript Documents, or Image file types*.

3.3.2 Extracting Results on Gauss Points

After creating a Gauss Point Plot, the user can extract results on **Gauss Points**.
Right-click on **Chart Data** > **Extract Chart Data** from the Work Tree panel or from Toolbar > **Plot** > **Extract chart data**.



Select On element gauss points on the Extraction Wizard and *Click* Next.

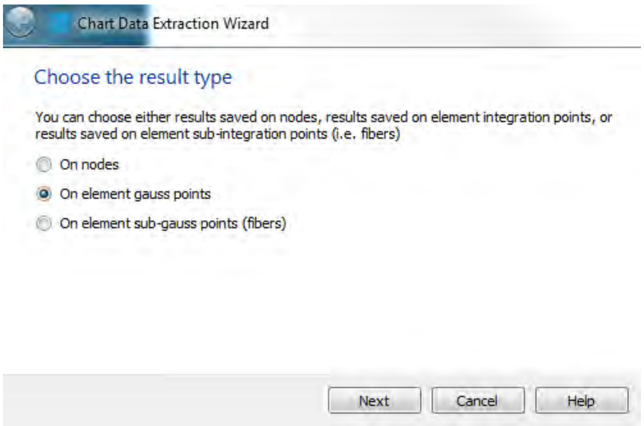


Figure 286. Choosing Result Type

Choose the results to extract and *Click* **Next**. Select the desired Gauss points on the model. *Click* **Next** to assign a name to the Chart Data and set how the wizard should handle multiple items, then *Click* **Finish**.

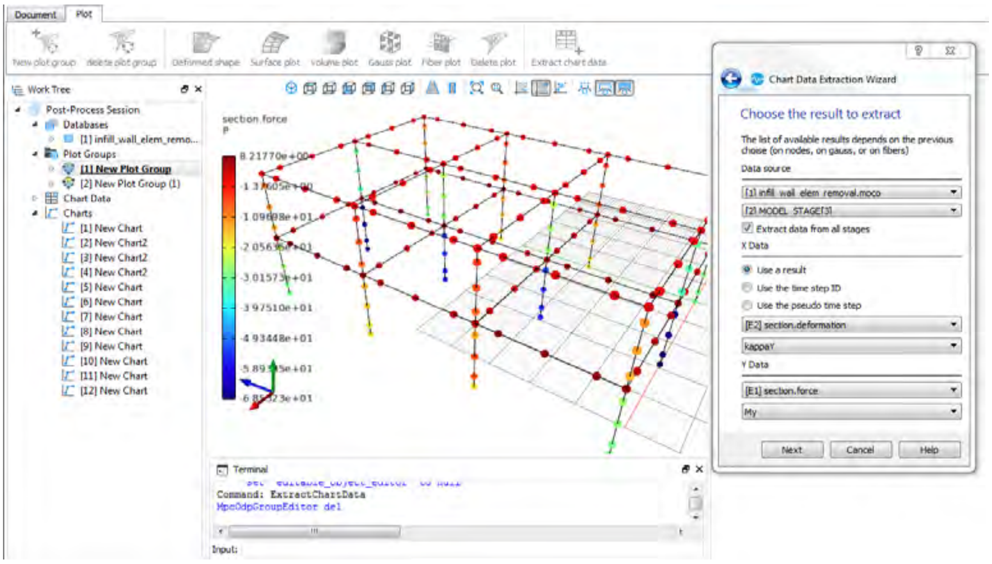


Figure 287. Gauss point plots

The new Chart will be visible in the Editor Panel and in the Work Tree.

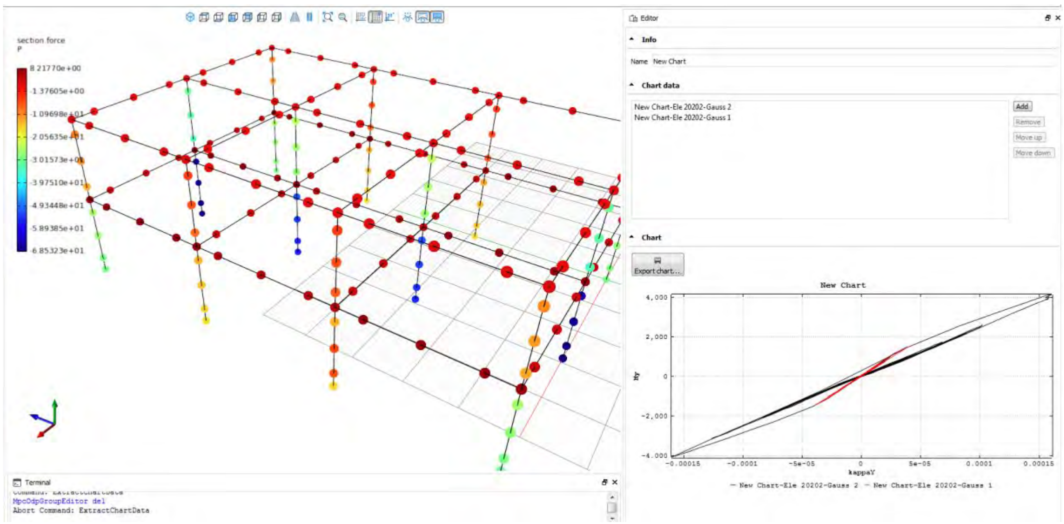


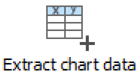
Figure 288. Gauss point chart

The Chart graphically displays plot data which can be exported by the user. *Click* **Export Chart** on the Editor Panel and choose the available format: *PDF, SVG, Postscript Documents, or Image file types*.

3.3.3 Extracting Results on Fibers

After creating a Shell Fiber Color Map Plot, user can extract results on **sub-gauss points (fibers)**.

Right-click on **Chart Data** > **Extract Chart Data** from the Work Tree panel or from the Toolbar *Click* > **Plot** > **Extract chart data**.



Select On element sub-gauss points (fibers) on the Extraction Wizard and *Click* Next.

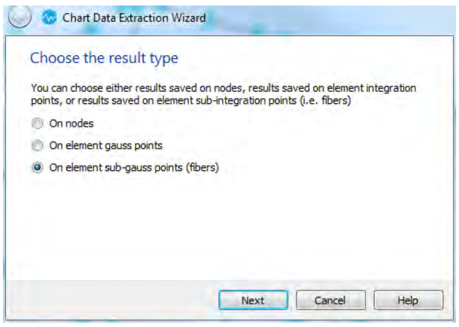


Figure 289. Using the Extraction Wizard

Choose the results to extract, and *Click* **Next**. Select the desired fibers on the model. *Click* **Next** to assign a name to the Chart Data and set how the wizard should handle multiple items, and *Click* **Finish**.

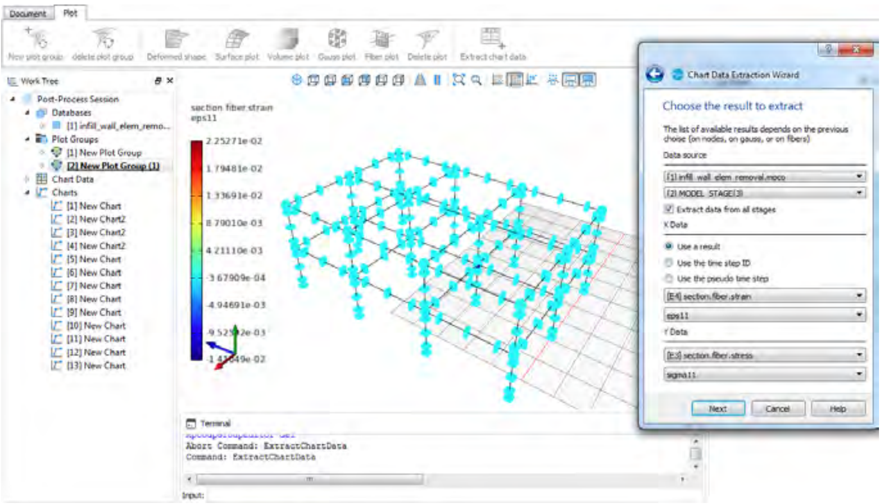


Figure 290. Extract fiber results

The new Chart will be visible in the Editor Panel and in the Work Tree.

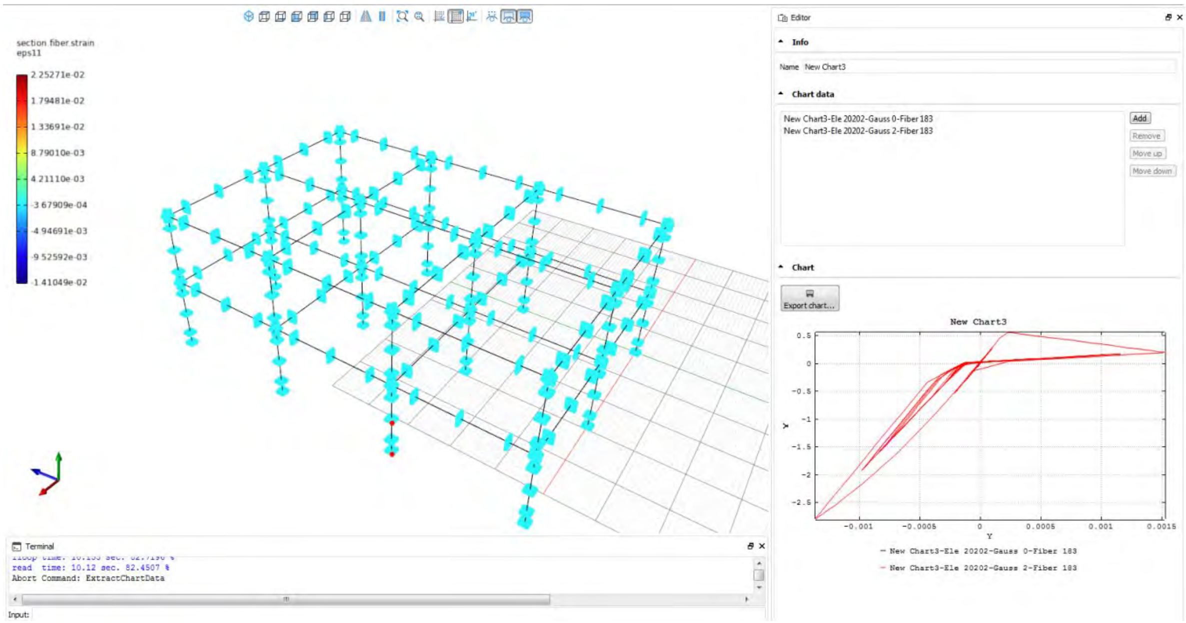


Figure 291. Fiber plot

The Chart graphically displays plot data which can be exported by the user. *Click* **Export Chart** on the Editor Panel and choose the available format: *PDF, SVG, Postscript Documents, or Image file types*.

The Chart Data Editor window gives users the option to link the chart data and the database together. This means, when the database is reloaded the graphic will also refresh and update itself according to any changes made.

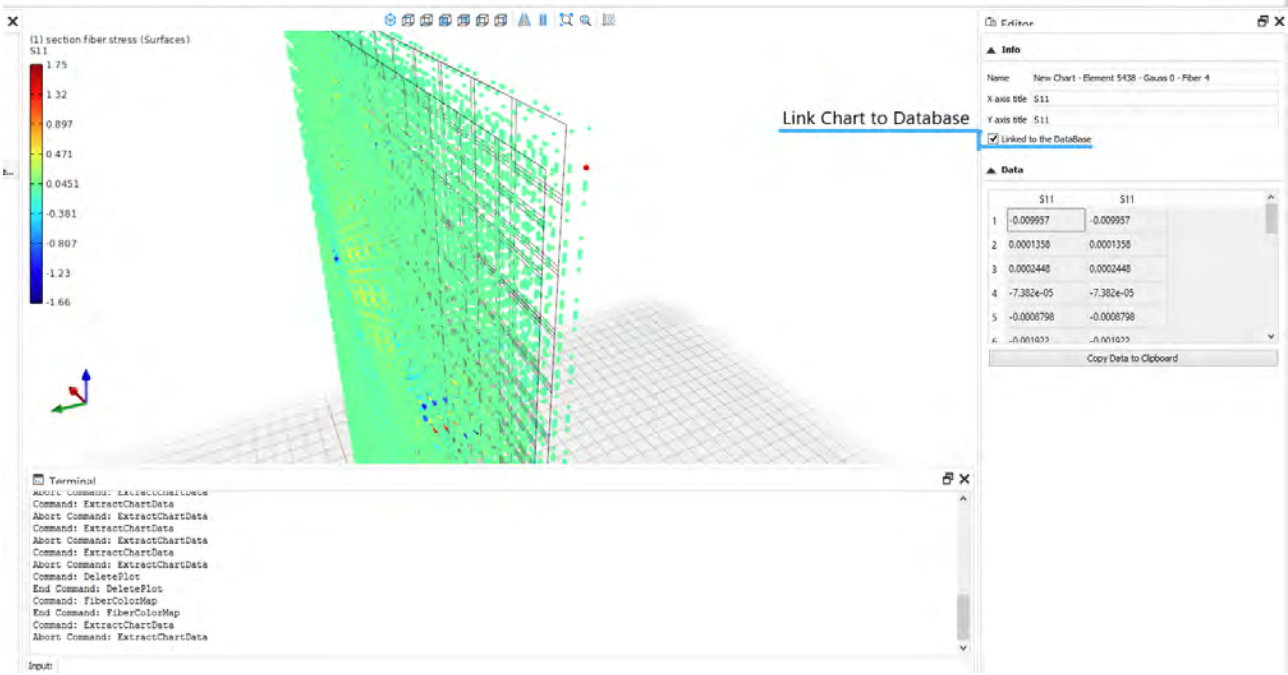


Figure 292. Link Chart to Database

4 PYTHON INTERFACE

Python is an interpreted, high-level, general-purpose programming language. The Python-based interface in STKO extends the capabilities of the pre and postprocessor and allowing the user to create and edit user interface objects for the input and for the output data.

Version 2.0 saw the addition of the long-awaited Scripting Framework for interacting with the pre and postprocessor documents. There is a new Dock Widget for scripting.

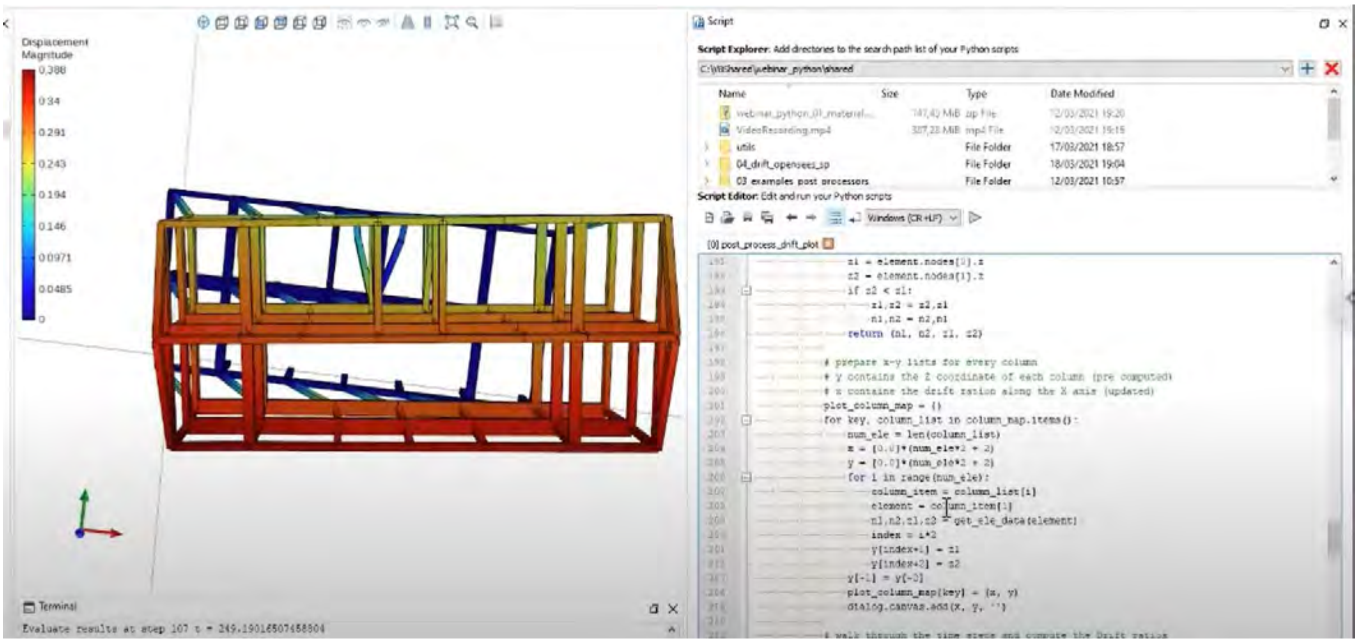


Figure 293. Python Scripting Interface

Using Python from a GUI (Graphic User Interface) presents differences compared to the normal usage that it is done of the programming language from a simple console application, and it requires a series of considerations. Therefore, we developed the PyMpc package specifically for **STKO's Python scripting interface of STKO**. The MPC part of the name stems from the original name of the software Multi-Purpose Cae. The classes, methods and functions included in the PyMpc package have been exported from STKO C++ source code to allow users to interact directly with the software by means of their own Python scripts.

With the Python scripting interface in the preprocessor, users will be able to design geometries directly by coding them and attribute elements and physical properties from the interface.

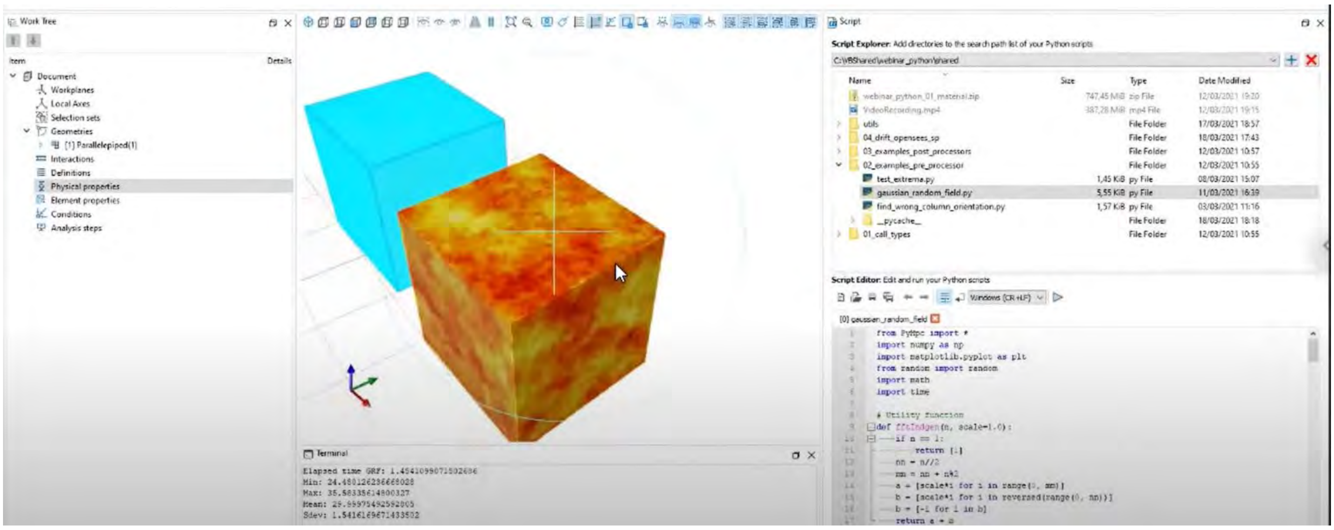


Figure 294. Python Scripting Interface in the Preprocessor

In the postprocessor, users will now be able to interact in real-time with the results or query the existing results. The interface means STKdOers will even be able to create additional customized derived results as a function of the existing results.

Another particularly exciting feature of the interface is the possibility to generate custom graphs using Python's libraries.

As this new feature involves its own package, rather than including the documentation here, it has its own documentation webpage accessible at the following link: https://asdeasoft.net/stko-wiki/

5. STKO MODEL EXAMPLES

5.1. Elastic Portal Frame

Create a simple model of a 2D Portal on the X-Y plane, like in the example below.

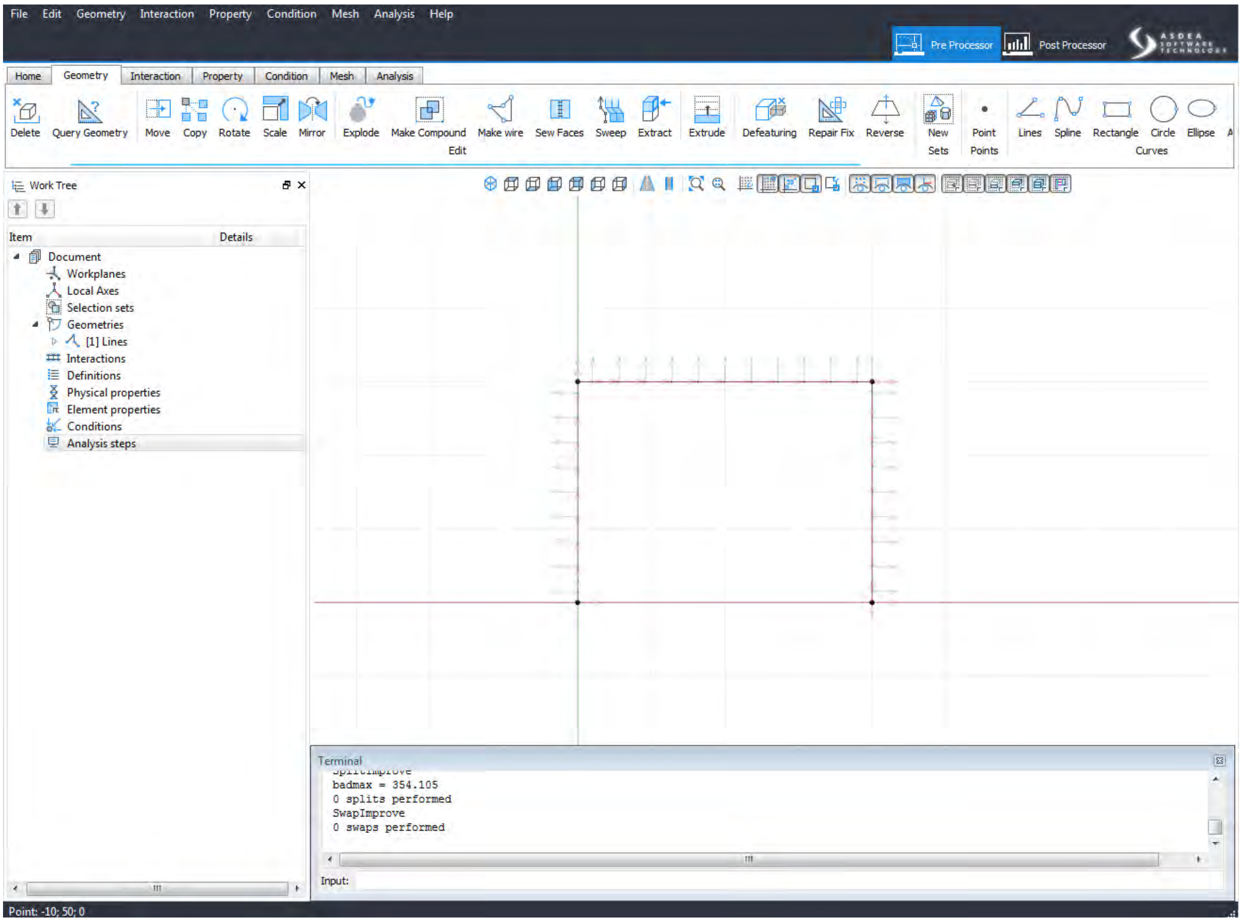


Figure 295. Frame

Then set the **Time Series** using the **Definition** Command.

Select **Property** > **New Definition** from the main Toolbar, or *Right-click* **Definition** > **Add** from the Work Tree panel. Then, using the drop-down menu select **Model** > **timeSeries** > **Linear.**

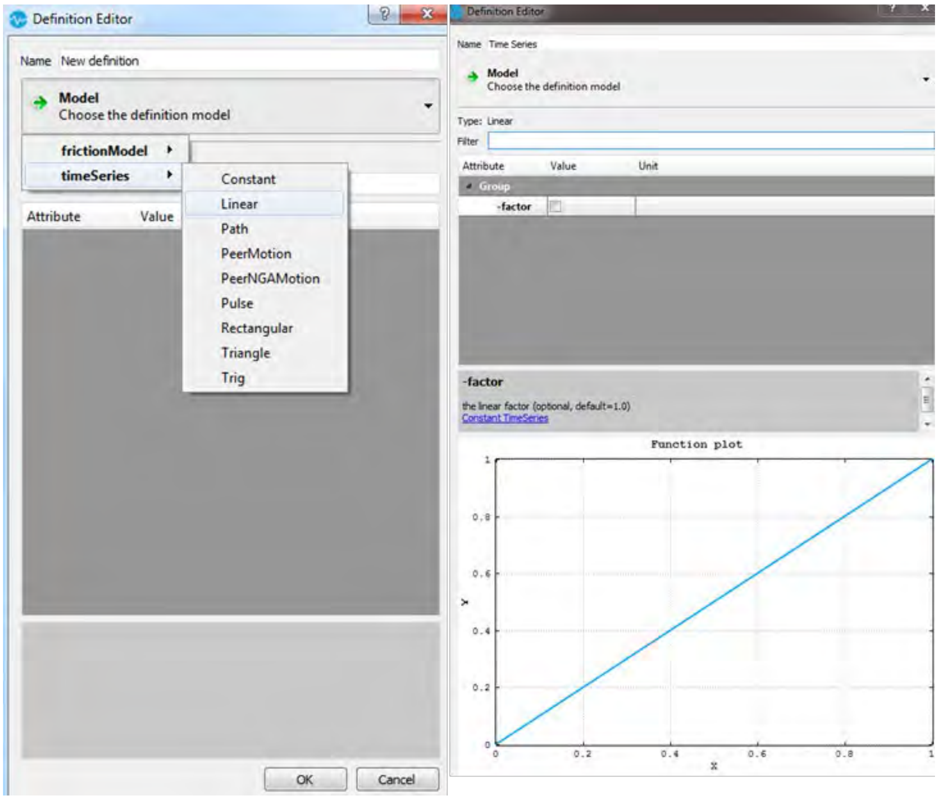


Figure 296. Linear time series

Click **OK** to confirm the settings.

Then, assign a **New Physical Property**. Select **Property** > **New Physical Property** from the Toolbar or *Right-click* **Physical Property** > **Add** from the Work Tree Panel. Using the drop-down menu, select **Model** > **Sections** > **Elastic**.

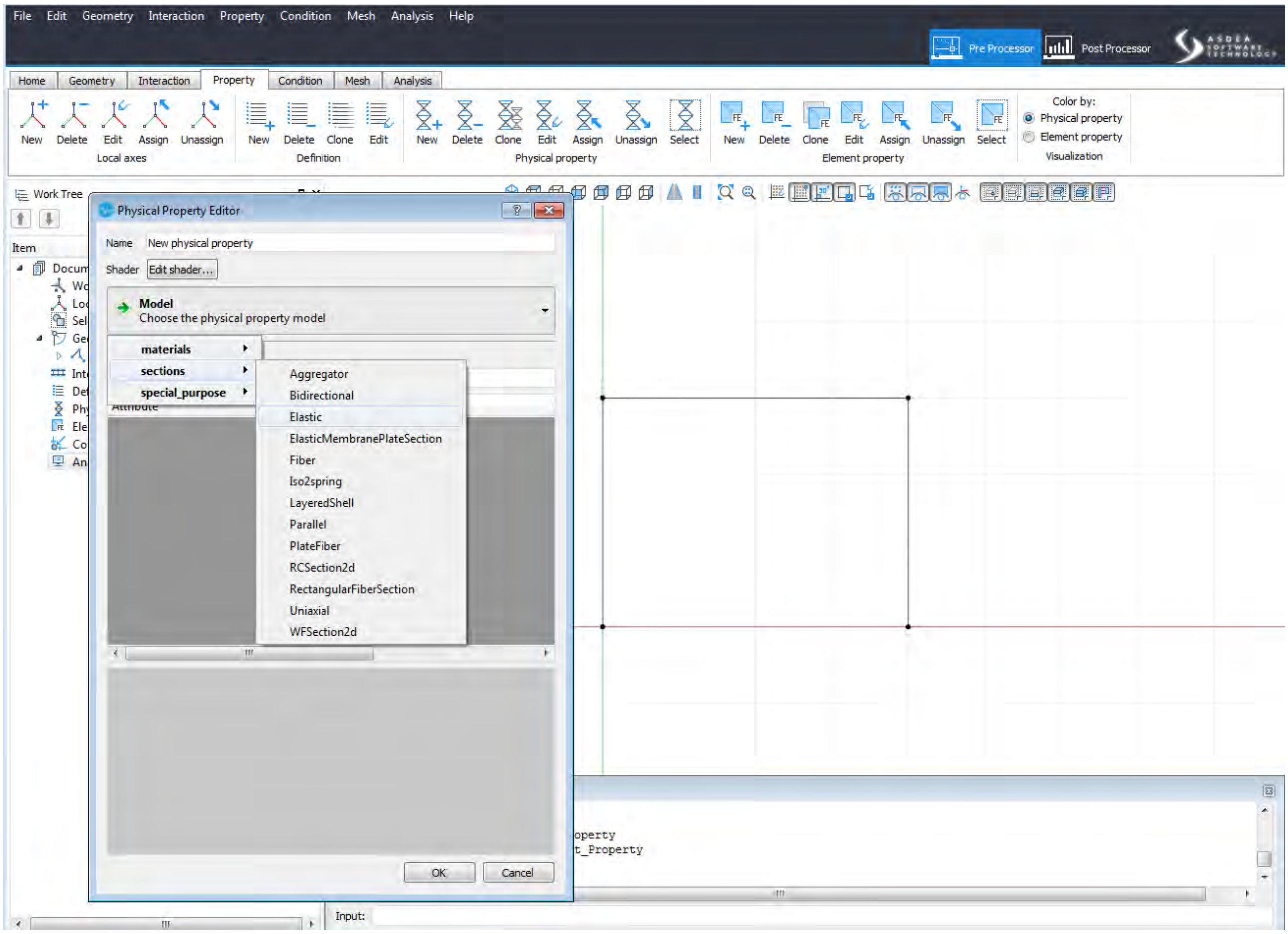


Figure 297. Assign elastic section

Define the type of 2D section with the **Beam Section Editor** window.

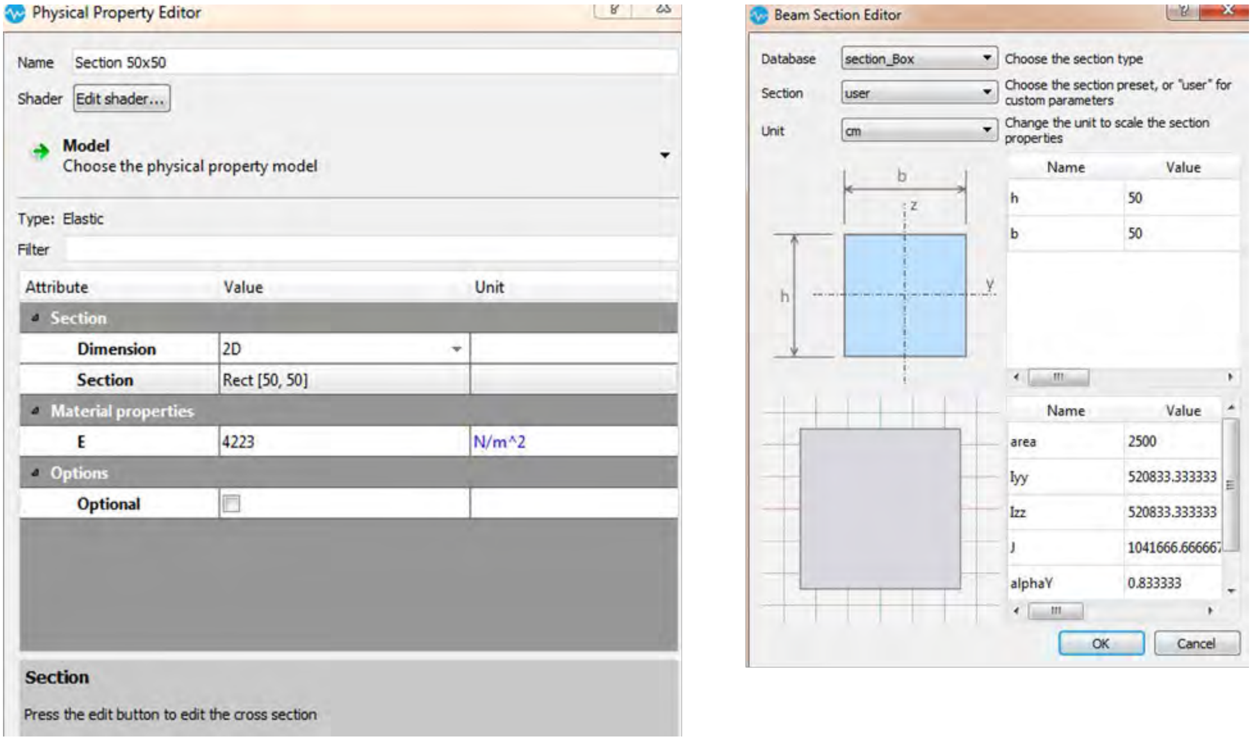


Figure 298. Define the section

Click **OK** to confirm the settings. Assign the **Beam Section** to the entire model.

<u>NOTE:</u> It is necessary to define a **Special Purpose** if the model contains elements with more physical properties. (See [§3.8.3 Special Purpose for Physical Properties])

Then select **Property** > **New element property** from the Toolbar**,** or *Right-click* **Element Property** > **Add** on the Work Tree. Using the drop-down menu, select **Model** > **beam_column_element** > **dispBeamColumn**.

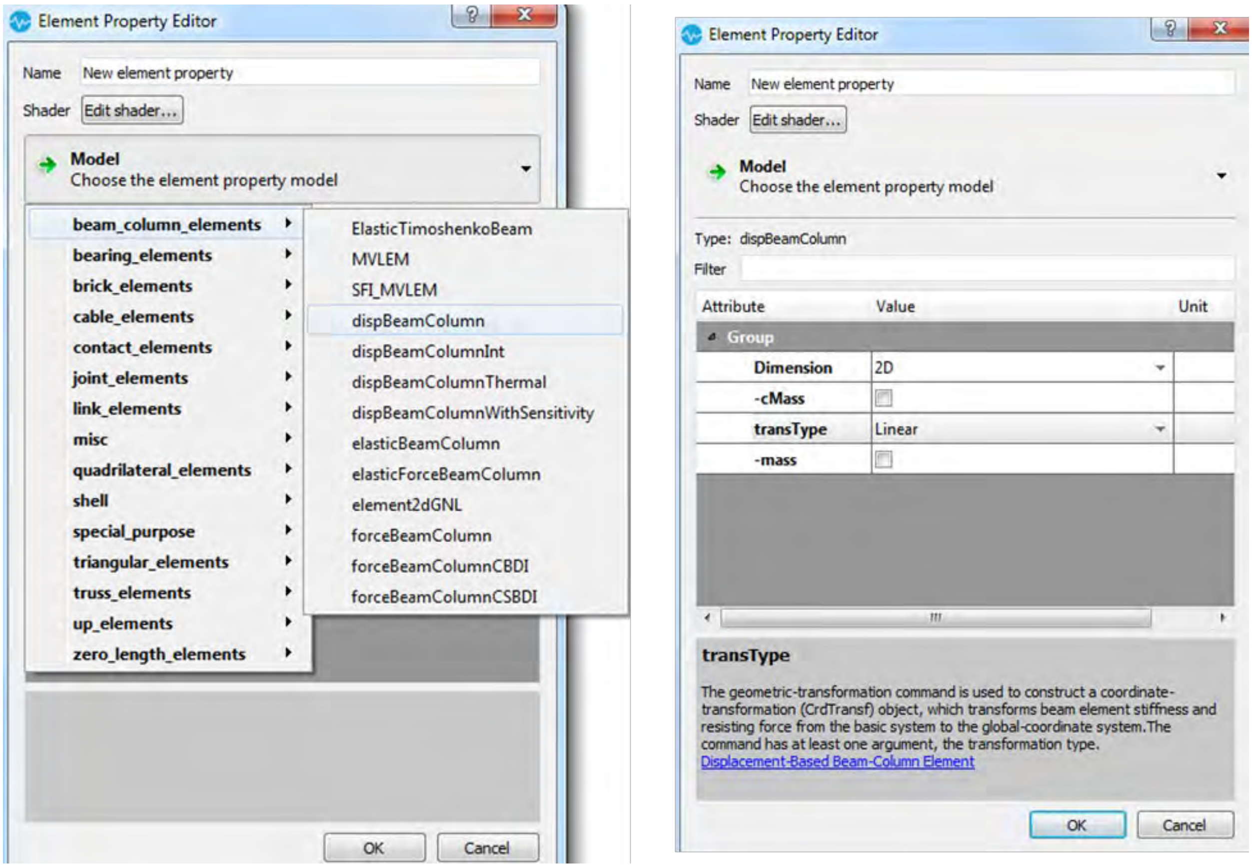


Figure 299. Assign the element

Click **OK** to confirm the settings and assign it to the entire model.

NOTE: It is necessary to define a **Special Purpose** if the model contains elements with more element properties. (See §2.5.1 Special Purpose for Element Properties)

To define constraints, select **Condition** > **New Condition** from the Toolbar, or *Right-click* **Conditions** > **Add** on the Work Tree Panel. Then using the drop-down menu, select **Model** and choose **Constraints** > **sp** > **fix**.

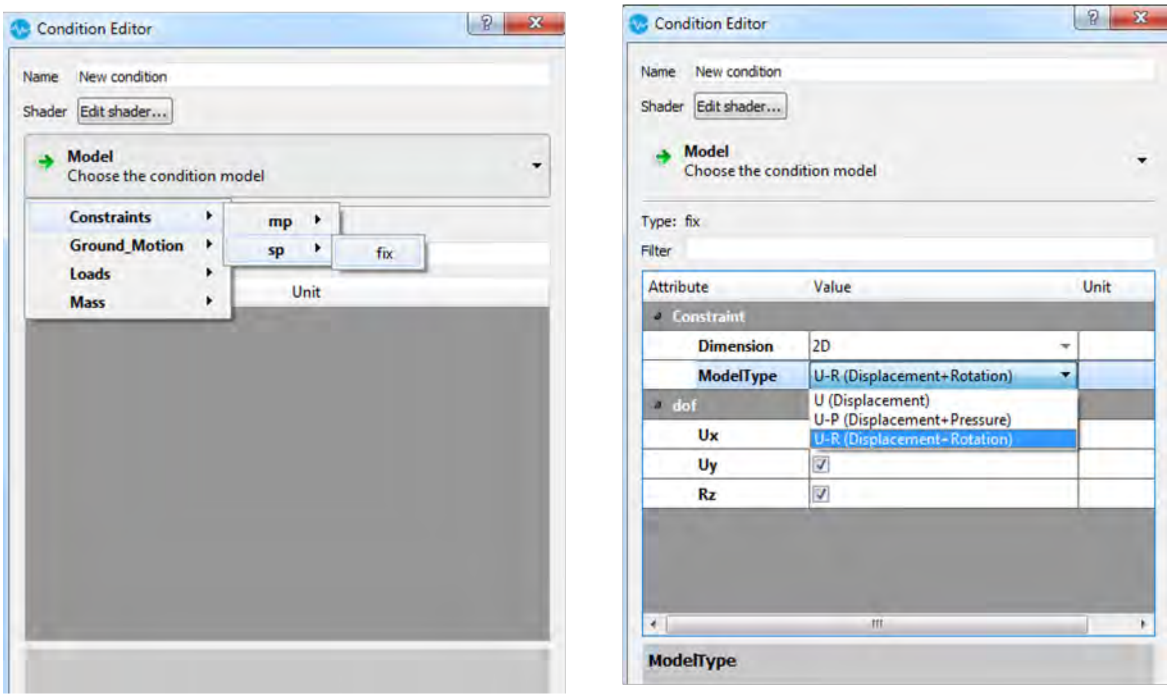


Figure 300. Assign constraints

Select the nodes on the base of the portal and *Click* **OK** to confirm the settings.

To apply a constant load to the beam, create a load by *clicking* **Condition** > **New condition** from the Toolbar, or *Right-click* **Conditions** > **Add** on the Work Tree Panel**.** Use the drop-down menu to **select Model** > **Loads** > **eleLoad** > **eleLoad_beamUniform**.

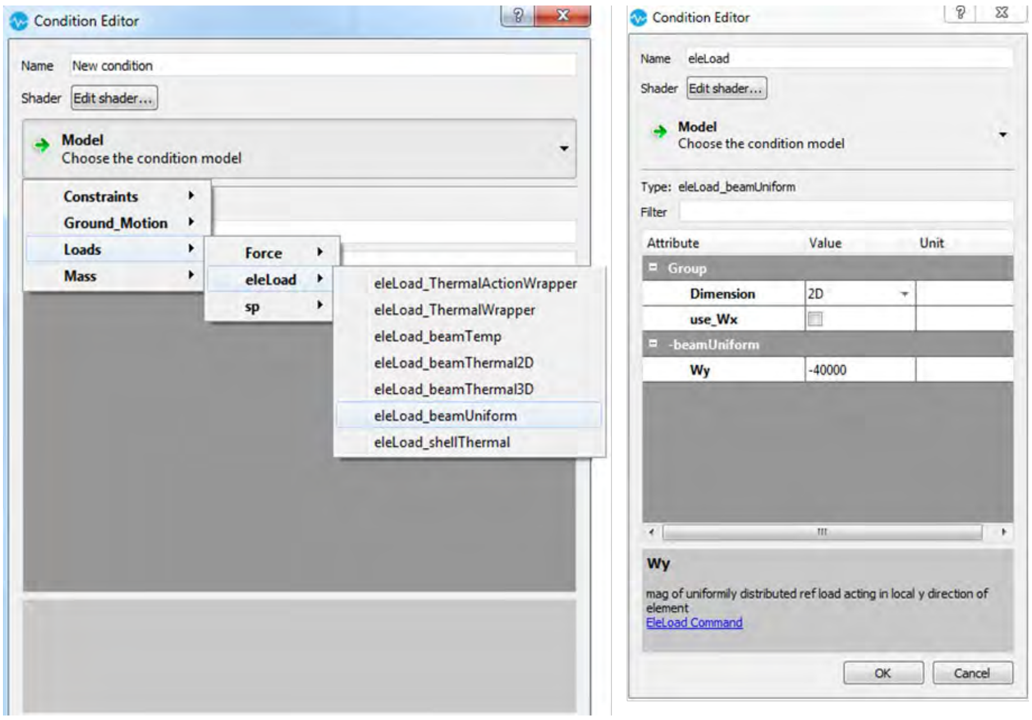


Figure 301. Define loads

Select the top edge of the portal and *click* **OK** to confirm the settings.

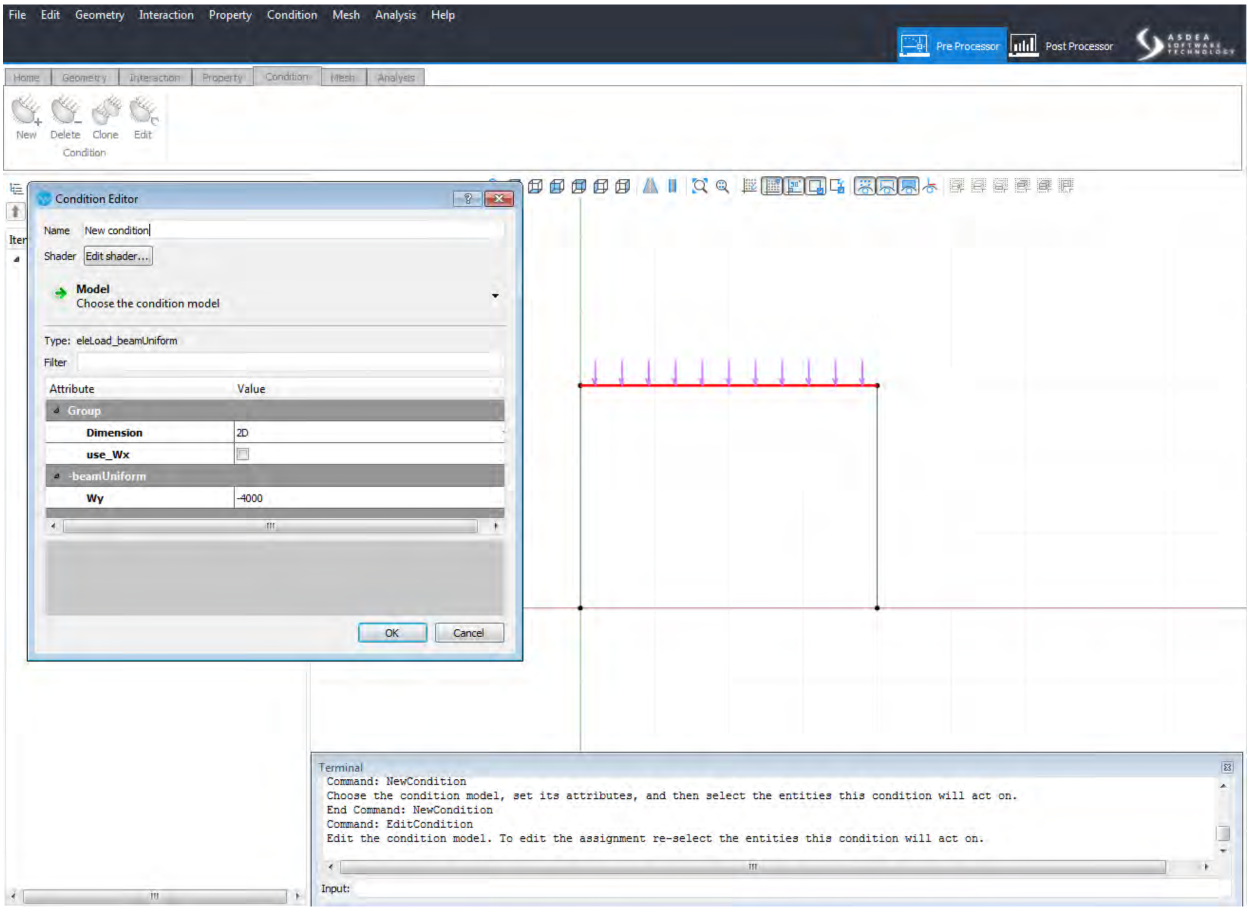


Figure 302. Assign distributed loads

After creating the model, define the mesh size along the edge. Select **Mesh** > **Global seed** from the Toolbar and set the global edge seed. The model is ready to be meshed. *Click* **Build Mesh** on the Toolbar.

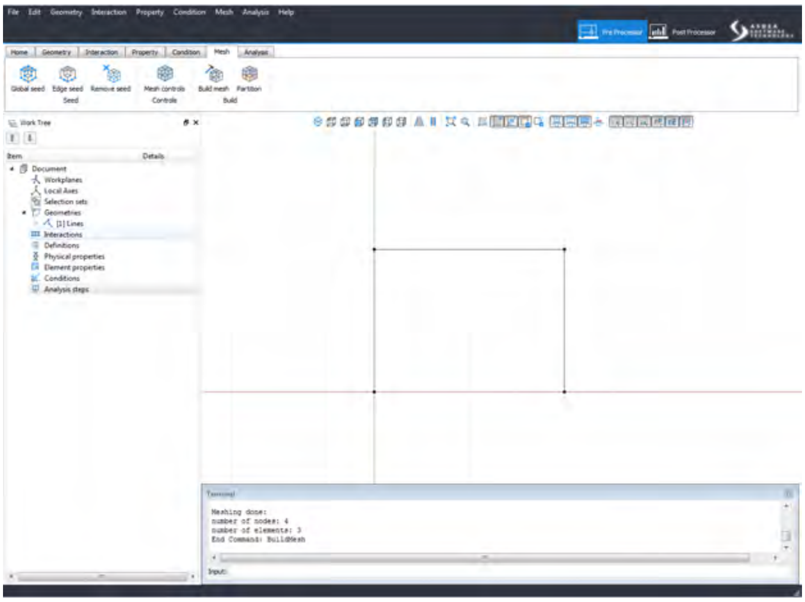


Figure 303. Build mesh

The next step is to define the analysis.

The first Analysis step is the **Recorder**. *Right-click* **Analysis Step > Add** on the Work Tree. Choose **Recorders** > **MPCORecorder** from the drop-down menus.

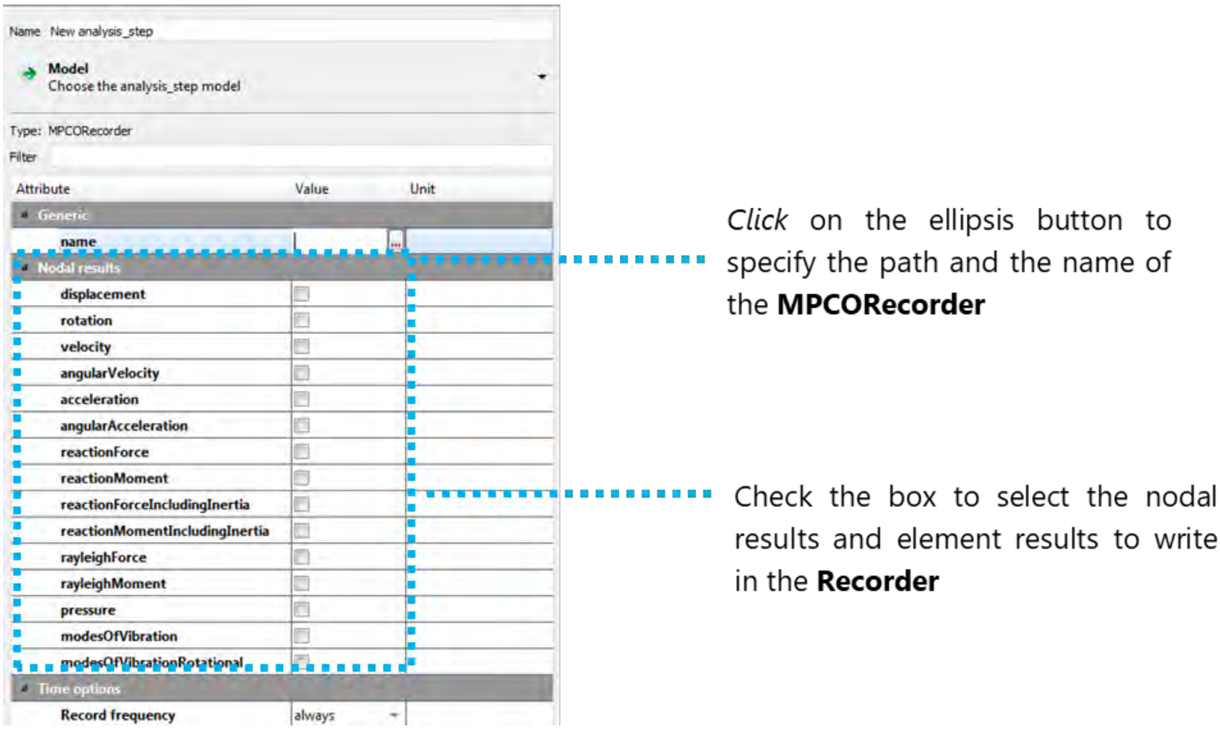


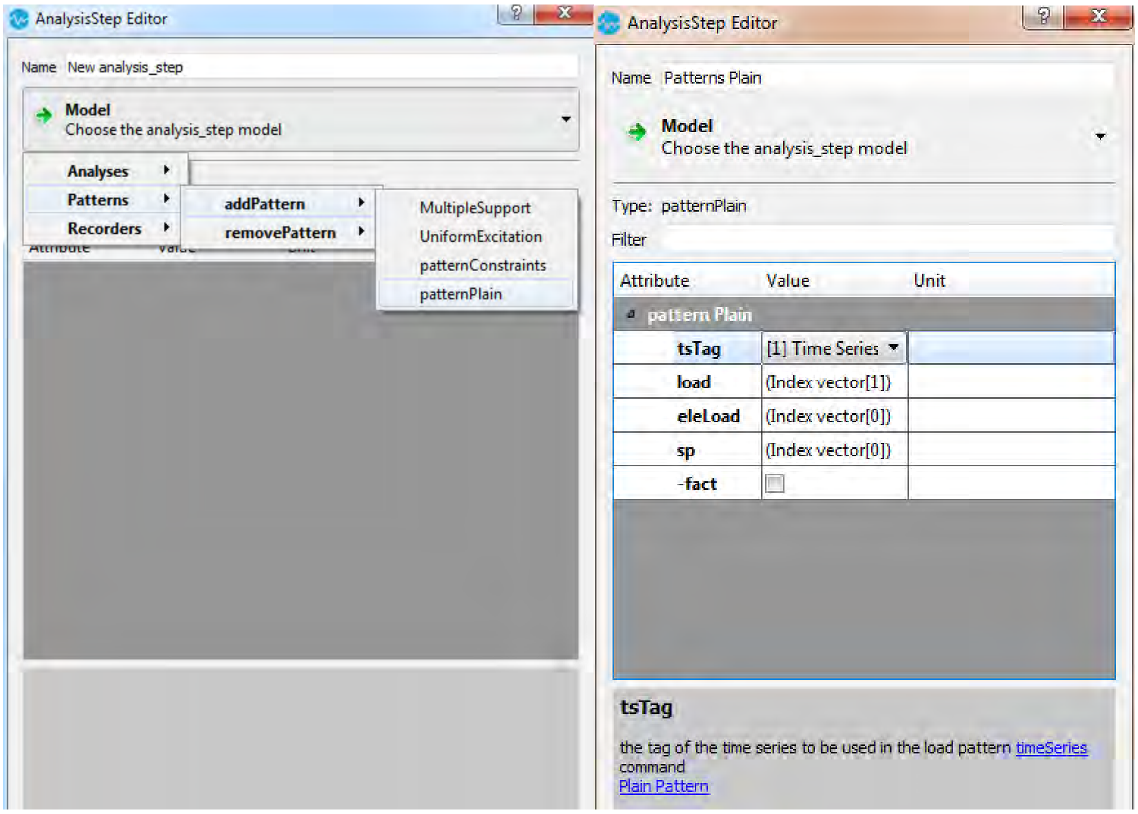
Figure 304. Select the output data to record

Click **OK** to confirm the settings.

NOTE: It is important to define **Patterns** and **Recorders** before the **Analyses**.

Next, define the Pattern and the Analysis steps.

Right-click **Analysis steps** > **Add** on the Work Tree.



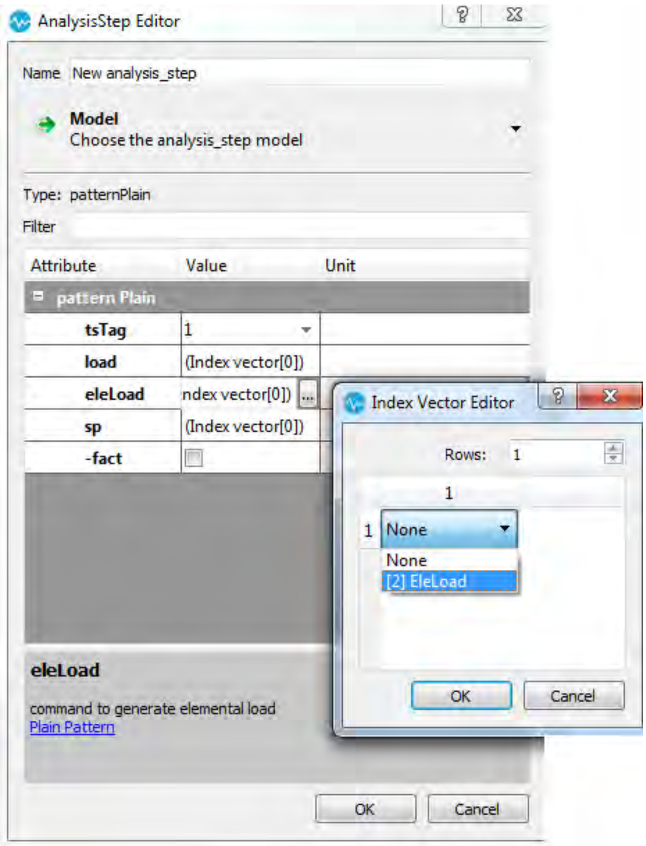


Figure 305. Define the load pattern

Click **OK** to confirm the patterns to analyse.

The last step is to add an Analysis. *Right-click* **Analysis Step > Add** on the Work Tree. Select **Model** > **Analyses** > **Analyses Command** from the drop-down menu and set all the attributes in the Analysis Step Editor:

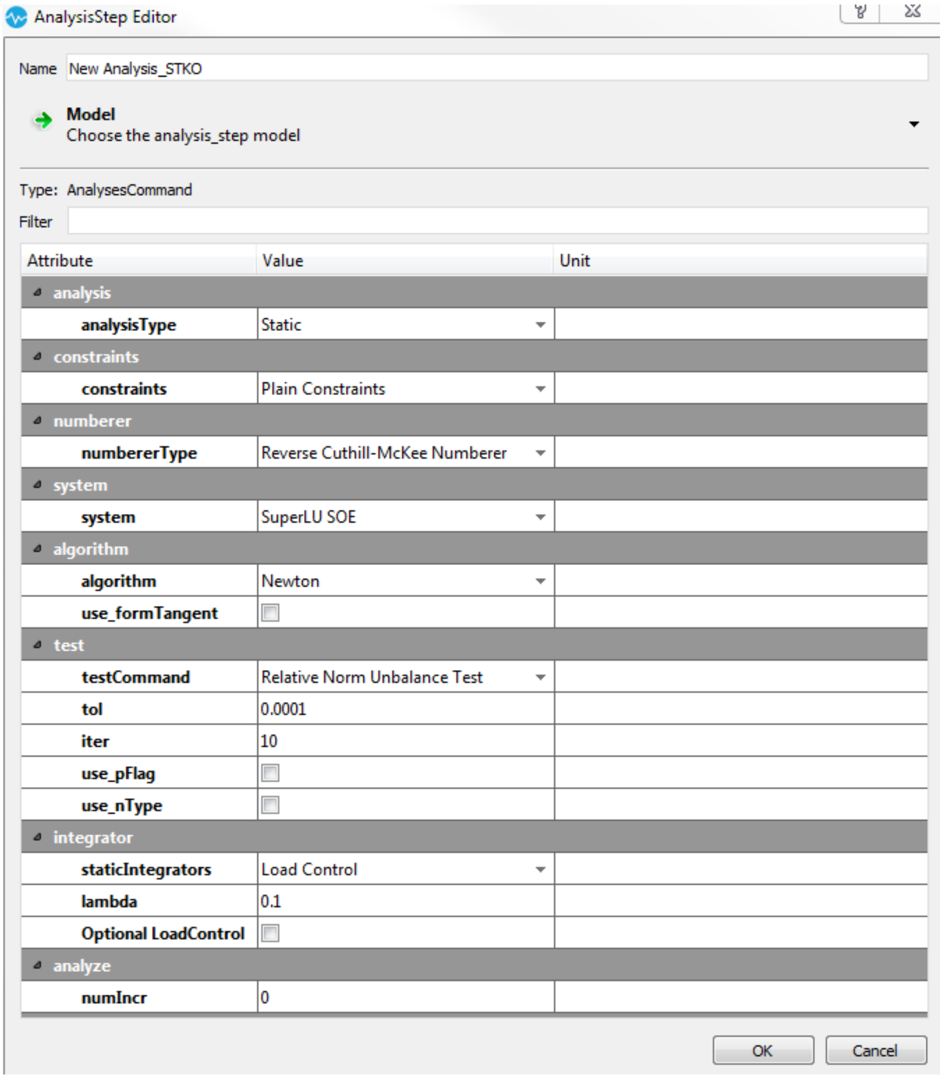


Figure 306. Set the analysis

Click **OK** to confirm the settings.

The model is complete and ready to be exported as **ticle files (.tcl)**.

Click **Analysis > Run Analysis** on the Toolbar. The software will connect to OpenSees to analyse all the inputs from STKO. It will generate an **.mpco** file to be analysed by the **STKO Postprocessor**.

Figure 307. Opensees output

5.2. Fiber Portal Frame

Draw a simple model of a 2D Portal on the X-Y plane, as shown in the following image.

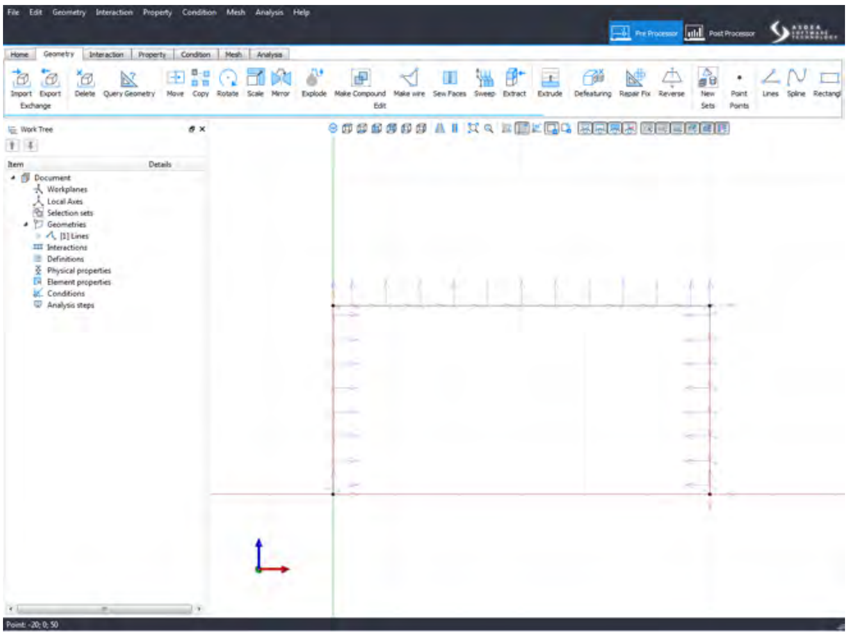


Figure 308. 2D fiber frame

To create a Physical Property, choose **Property** > **New Physical Property** from the toolbar, or *Right-click* **Physical Property** > **Add** from the Work Tree. Select **Model**, and choose a uniaxial material (for this example, **Concrete01**).

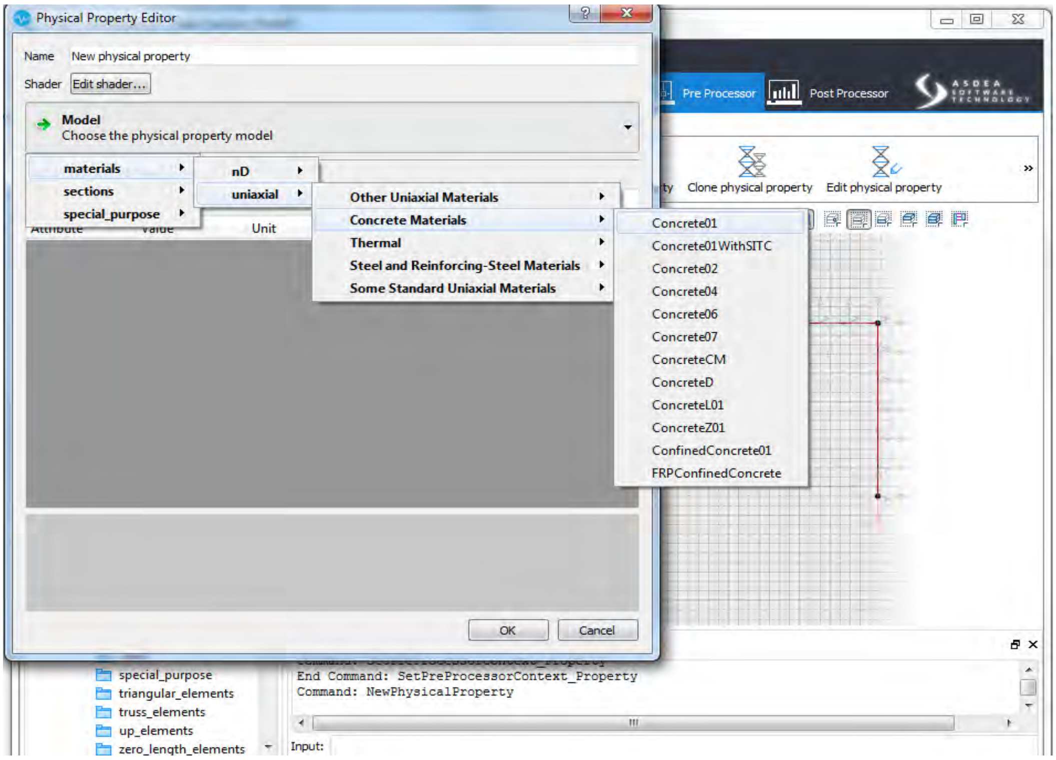


Figure 309. Create the uniaxial material concrete01

I nsert values to attribute to the **Concrete Core**. Each attribute links to a short explanatory description on the Opensees website.

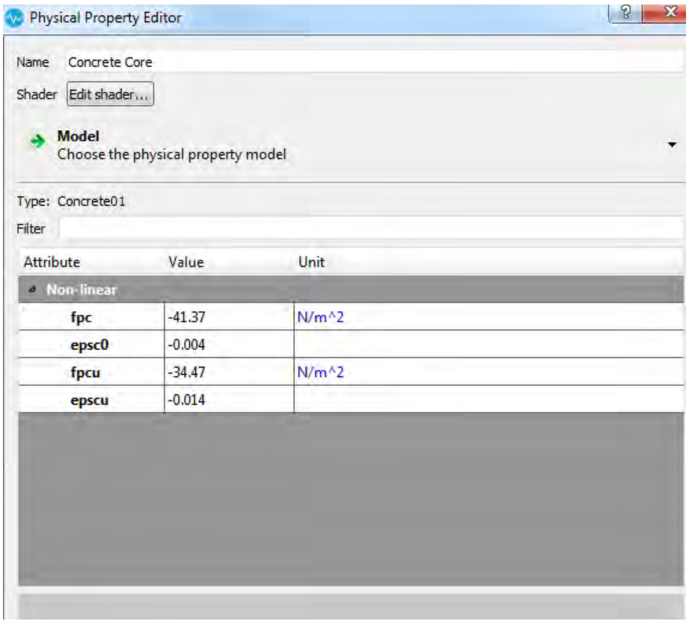


Figure 310. Define the uniaxial material concrete01 parameters

Click **OK** to confirm the settings.

Complete the same process to create a Concrete Cover, again selecting a uniaxial material (like **Concrete 01** for this example). Set and confirm all parameters.
The user should then define a third material, for this example, select **Model** > **uniaxial** > **Steel and Reinforcing-Steel Materials** > **Steel01**.

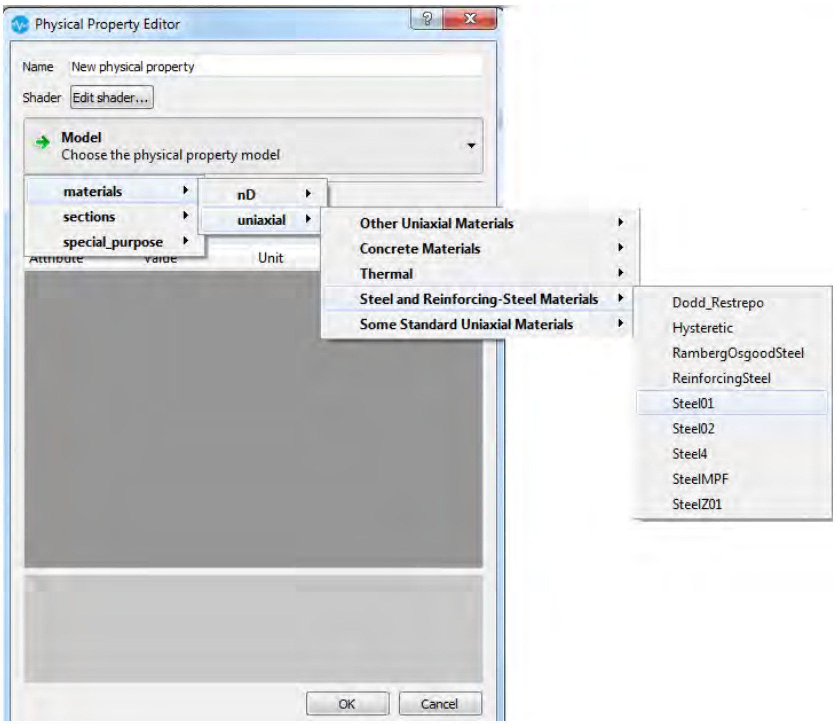


Figure 311. Create the uniaxial material Steel01 and assign the parameters

After defining the Materials, *Click* **Property** > **New physical property** from the Toolbar to create a **Fiber Beam Cross-Section** or *Right-click* **Physical properties** > **Add** from the Work Tree Panel. *Click* **Model** > **sections** > **Fiber** from the drop-down menu.

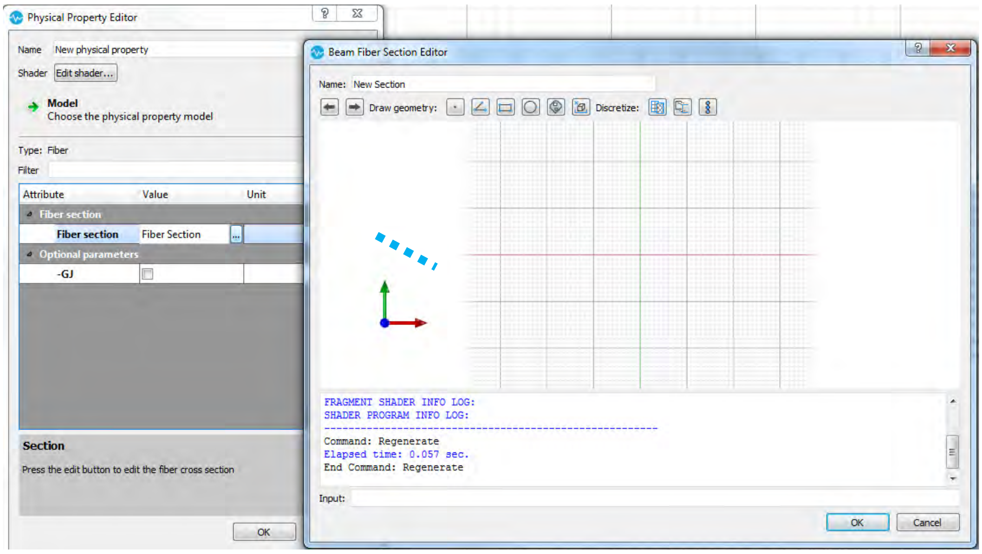


Figure 312. Create the fiber section

Draw a rectangular column (for example 600x380 mm) in the Beam Fiber Section Editor. Follow all the steps previously shown [§3.8.2.2. Fiber Beam Cross-Sections.] Assign the **Concrete Core**, **Concrete Cover**, and **Rebar** to the section, as modeled below.

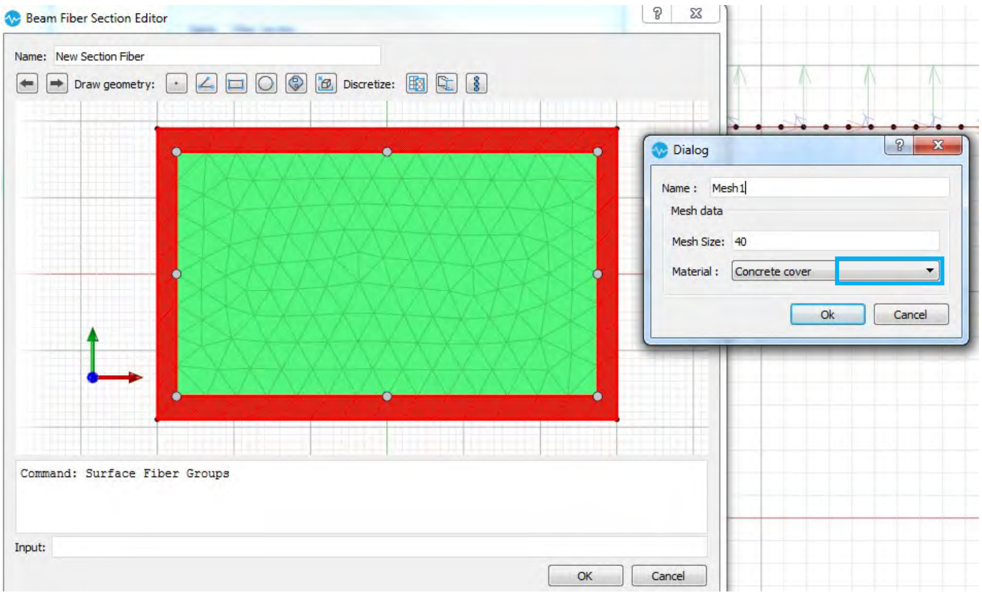


Figure 313. Draw the fiber section

Rename the Beam Fiber Section and confirm it.

Select the integration type and the **sectionTag** to attribute. *Click* **OK** to confirm the setting. After confirming the settings, assign it to the geometry by dragging it from the Work Tree Panel to the geometry.

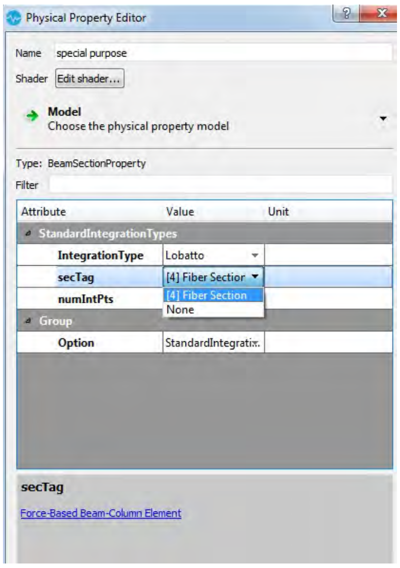


Figure 314. Assign the Fiber Section

Then create another **New Physical Property**. Select **Property** > **New Physical Property** from the Toolbar and choose > **Model** > **sections** > **Elastic** from the drop-drown menu.

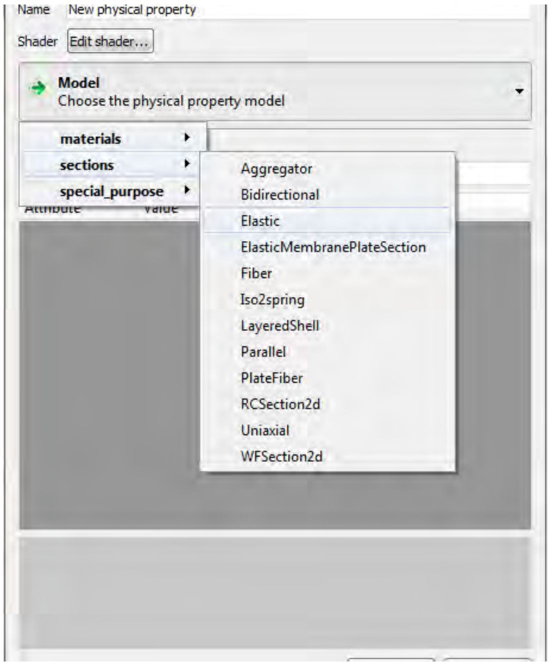


Figure 315. Create the elastic section

Define the type of 3D section with the **Beam Section Editor** window.

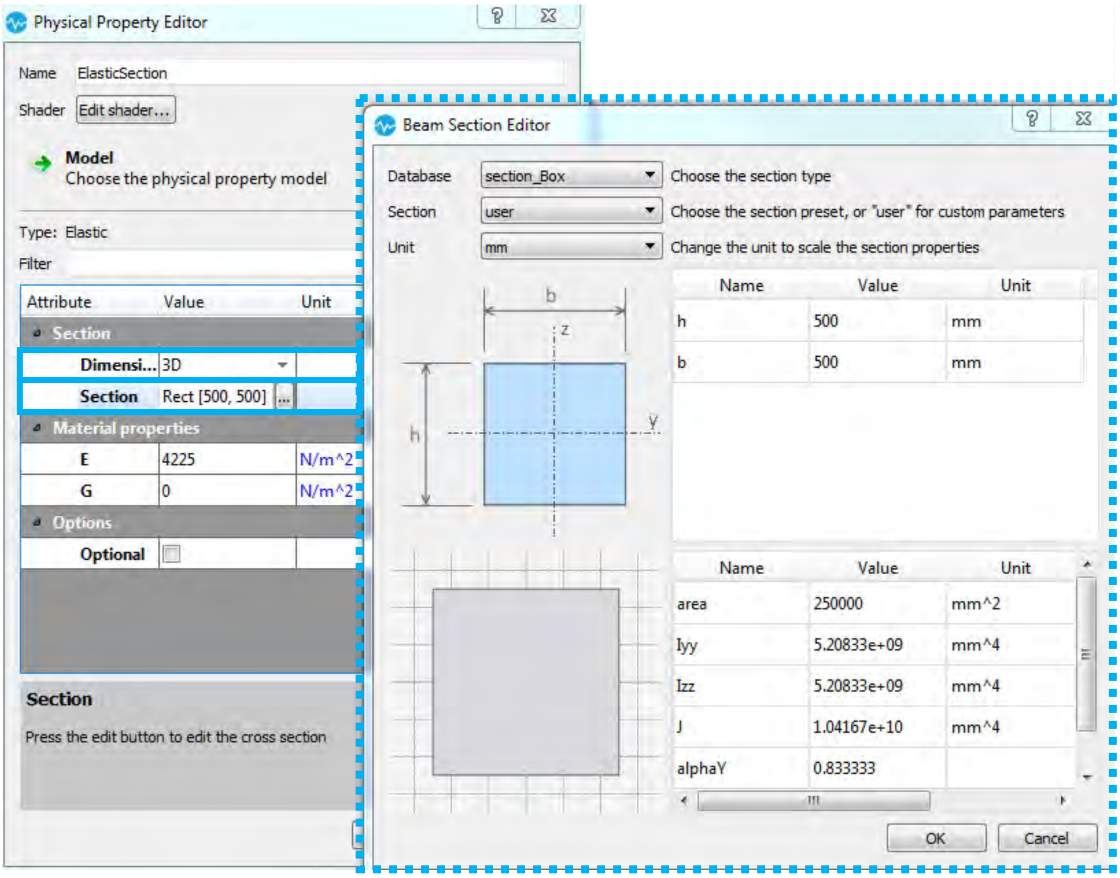


Figure 316. Create the section

Click **OK** to confirm the settings. It is now necessary to create new element properties. Choose **Property** > **New element property** from the Toolbar or *Right-click* **Element Properties** > **Add** on the Work Tree Panel. Select **Model** > **beam_column_elements** > **forceBeamColumn.**

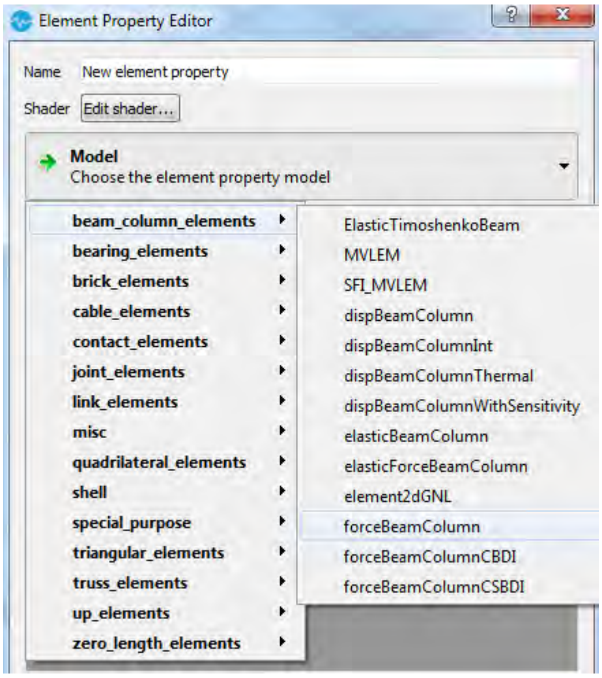


Figure 317. Add element type

After creating the **ForceBeamColumn** and the **Fiber Section**, assign them to the elements. Repeat the process to create and assign a new **elasticBeamColumn** element:

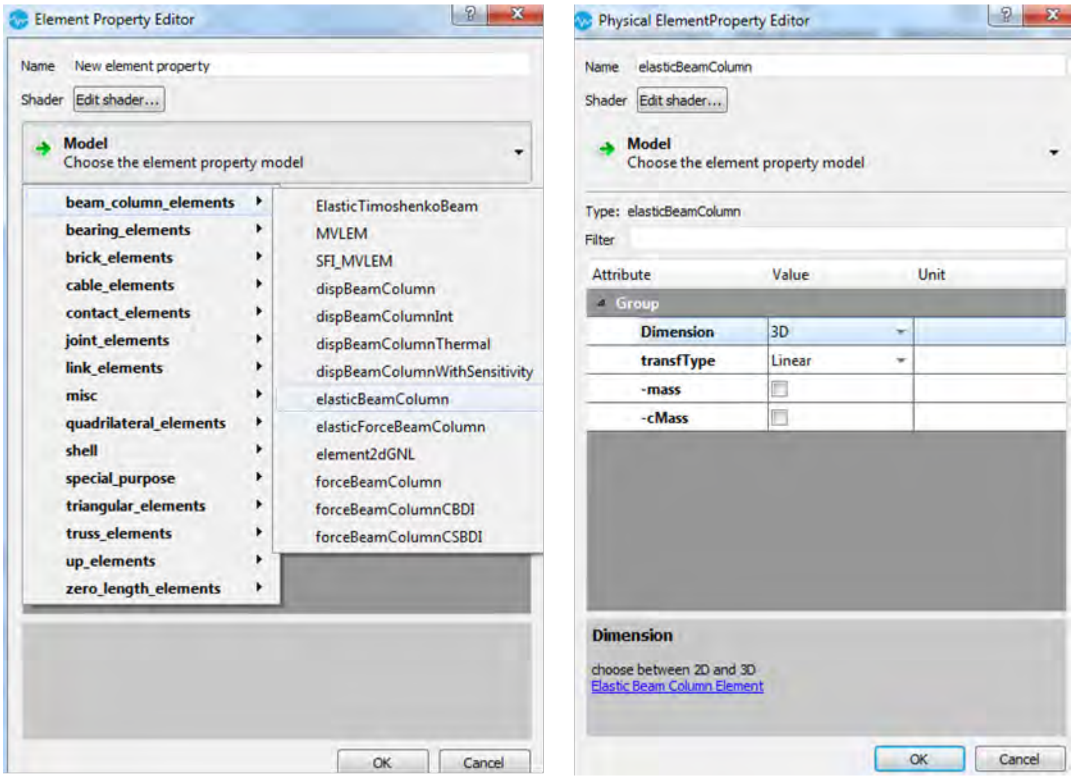


Figure 318. Define element type

Assign the elasticBeamColumn to the columns along with the Elastic Section.

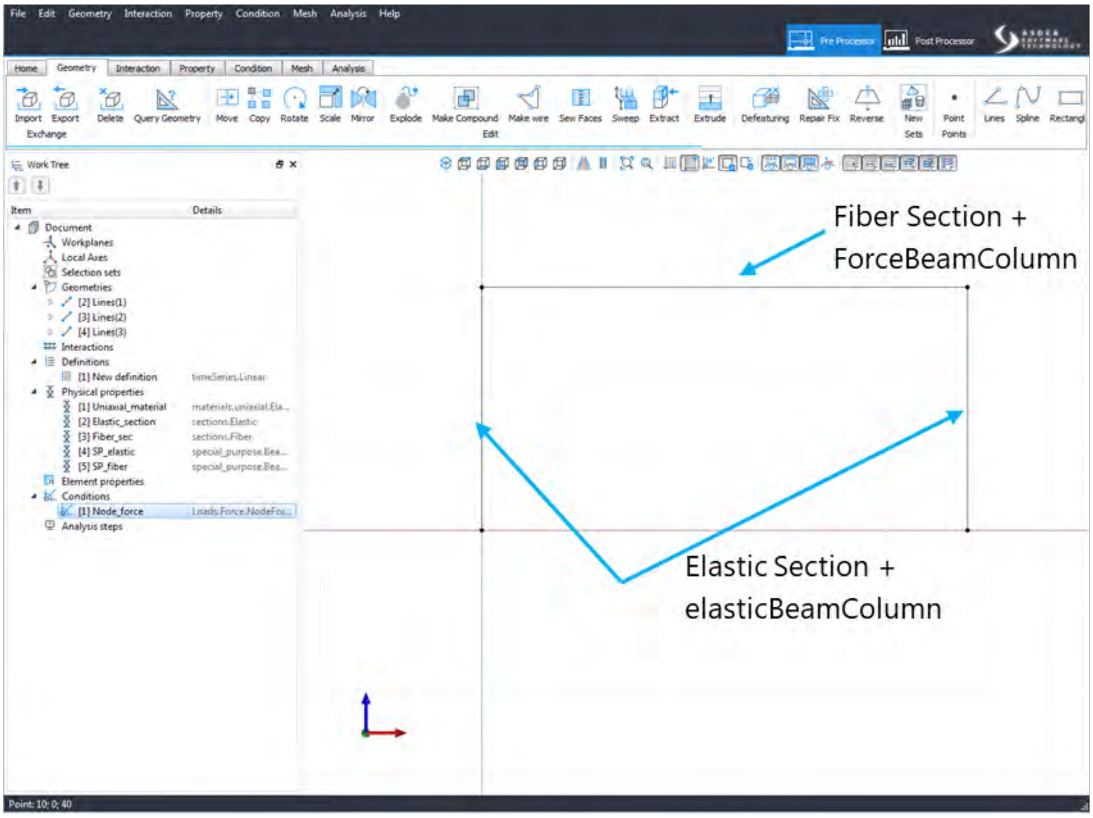


Figure 319. Assign element type

After that, create a constraint to apply to the base (**fix constraint**). Choose **Condition** > **New** from the main Toolbar, or *Right-click* **Condition** > **Add** on the Work Tree.

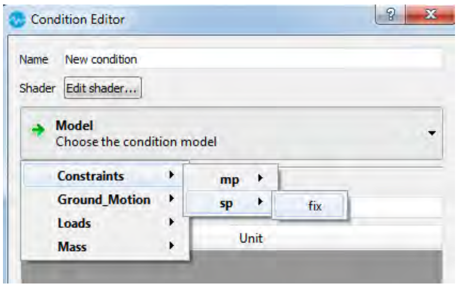
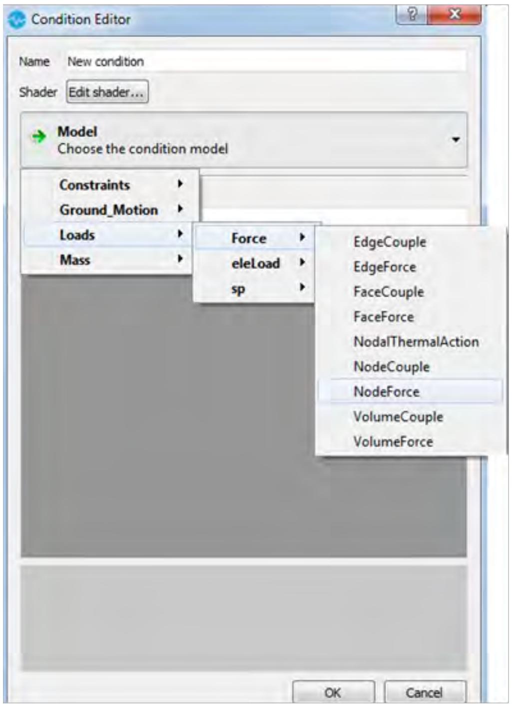


Figure 320. Add single point constraint

Before confirming the Condition Editor settings, select the nodes at the base of the geometry. Then, *Click* **OK** to assign the settings.

Then add a force to the nodes of the beam (**NodeForce**) by selecting **Condition** > **New** from the main Toolbar, or *Right-click* **Condition** > **Add** on the Work Tree
Select **Model** > **Loads** > **Force** > **NodeForce** from the drop-down menu.



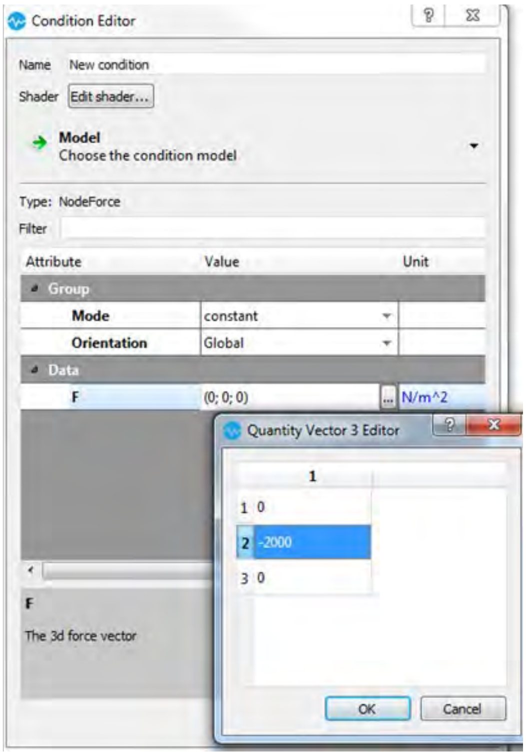


Figure 321. Add nodal force

Before confirming the Condition Editor, select the nodes of the beam, and *Click* **OK** to assign the Condition.

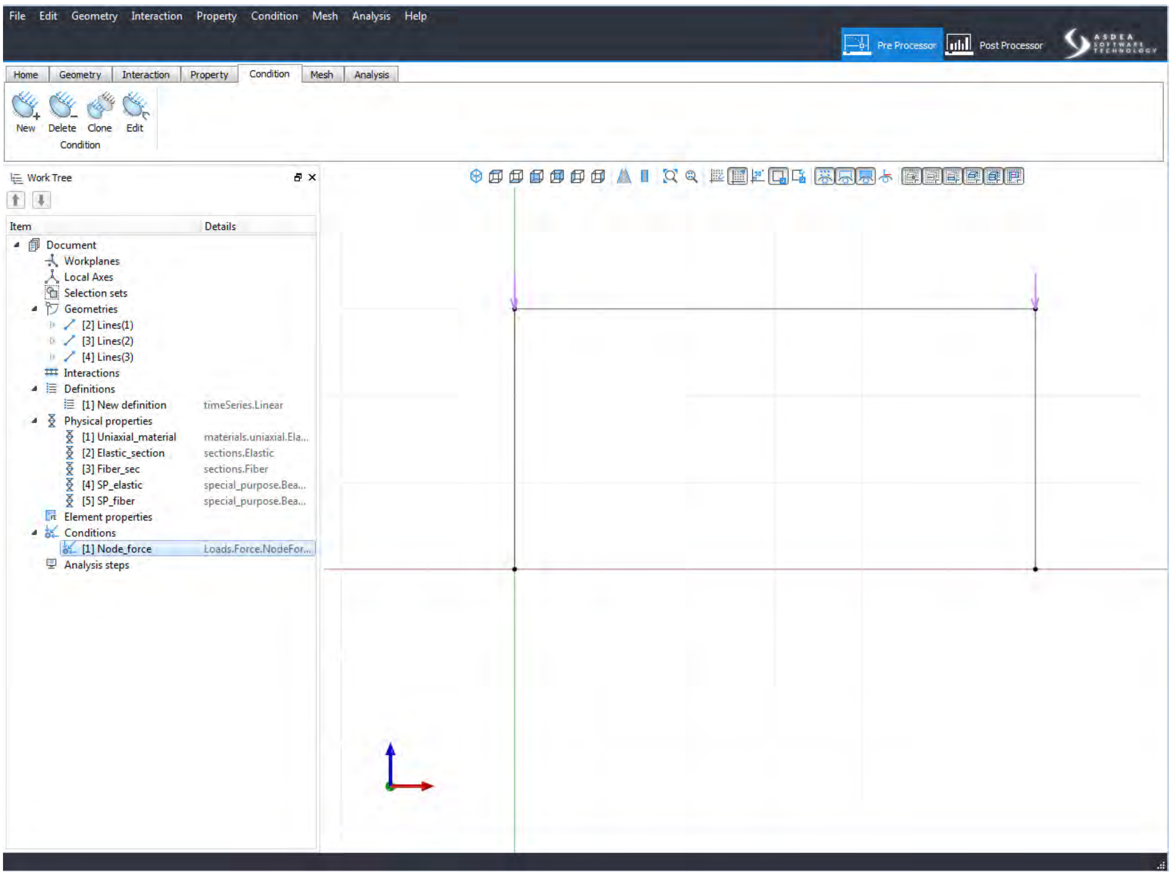


Figure 322. Assign nodal forces

It is now possible to set the **Time Series** using the **Definition** Command.

Select **Property** > **New Definition,** then **Model** > **timeSeries** > **Linear** from the drop-down menu.

After the user has defined the physical and element properties, loads, and constraints, they may create **Patterns** that make up the Analysis steps. *Right-click* **Analysis steps** > **Add** from the Work Tree. Then select **Model** > **addPattern** > **loadpattern**.

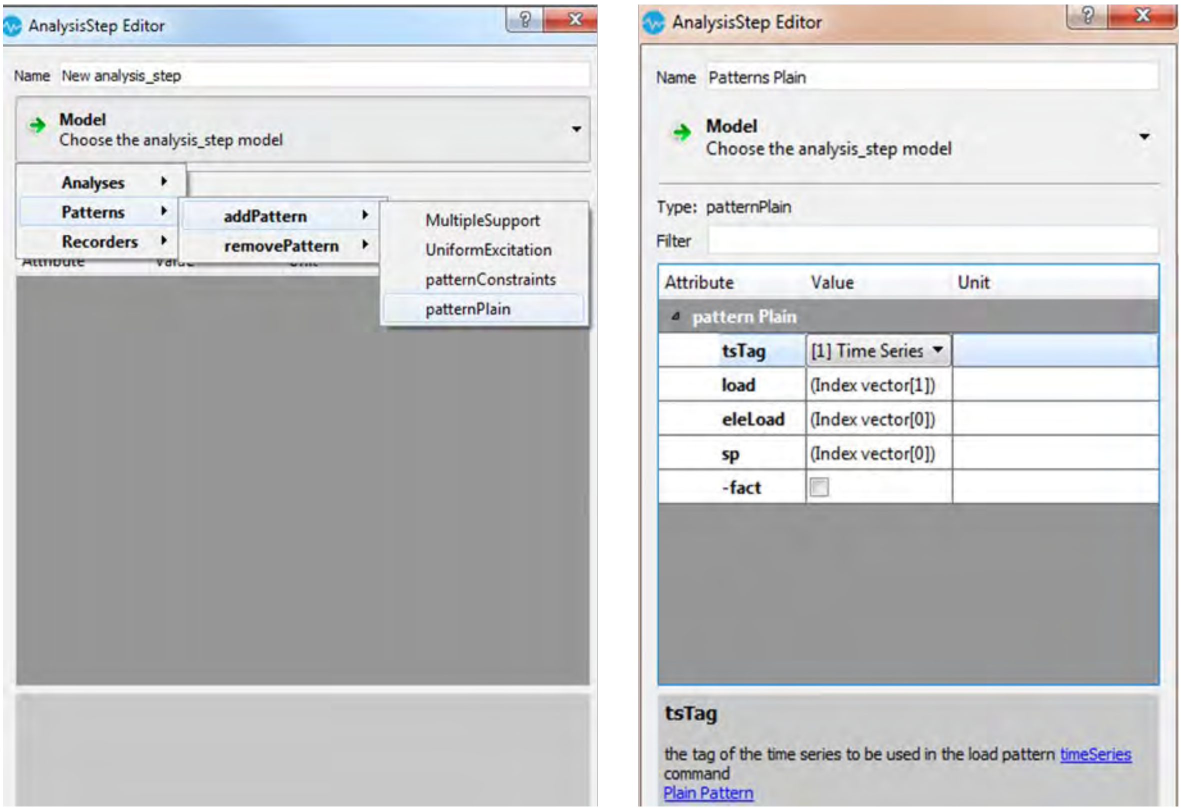


Figure 323. Define load pattern

Then, define the mesh size along the edge. Select **Mesh** > **Global seed** from the Toolbar and set the global edge seed type and size. The model is ready to be meshed.

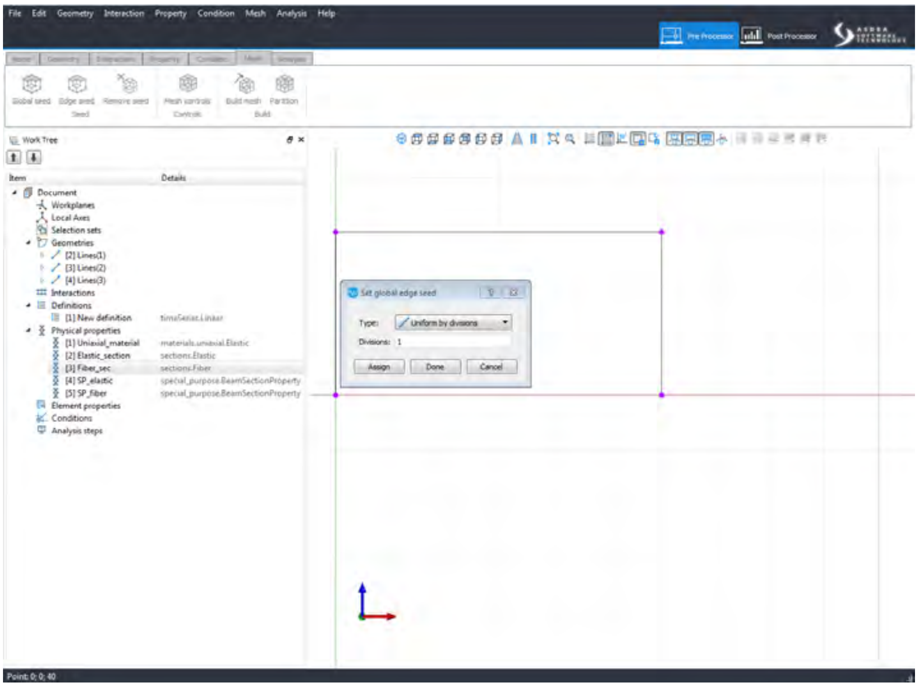


Figure 324. Mesh the geometry

After setting all the parameters need to be analysed. *Click* on **OK** to confirm them.

The second Analysis step is to create a **Recorder**. *Right-click* **Analysis Step > Add** on the Work Tree. Choose **Model** > **Recorders** > **MPCORecorder** from the drop-down menu.

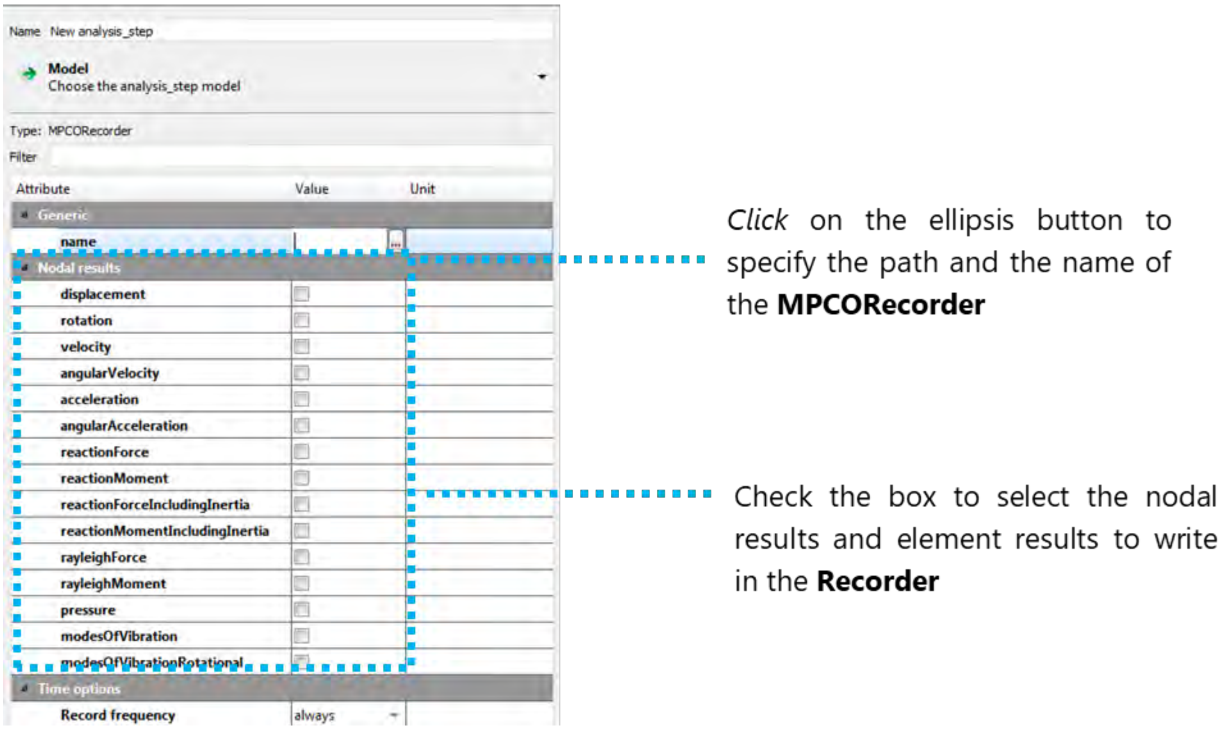


Figure 325. Set the MPCO Recorder

Click **OK** to confirm the settings.

NOTE: It is important to define **Patterns** and **Recorders** before the **Analyses**.

The last step is to add an Analysis. *Right-click* **Analysis Step > Add** on the Work Tree. Choose **Model** > **Analyses** > **Analyses Command** from the drop-down menu and set all attributes using the Analysis Step Editor, as shown in the following image.

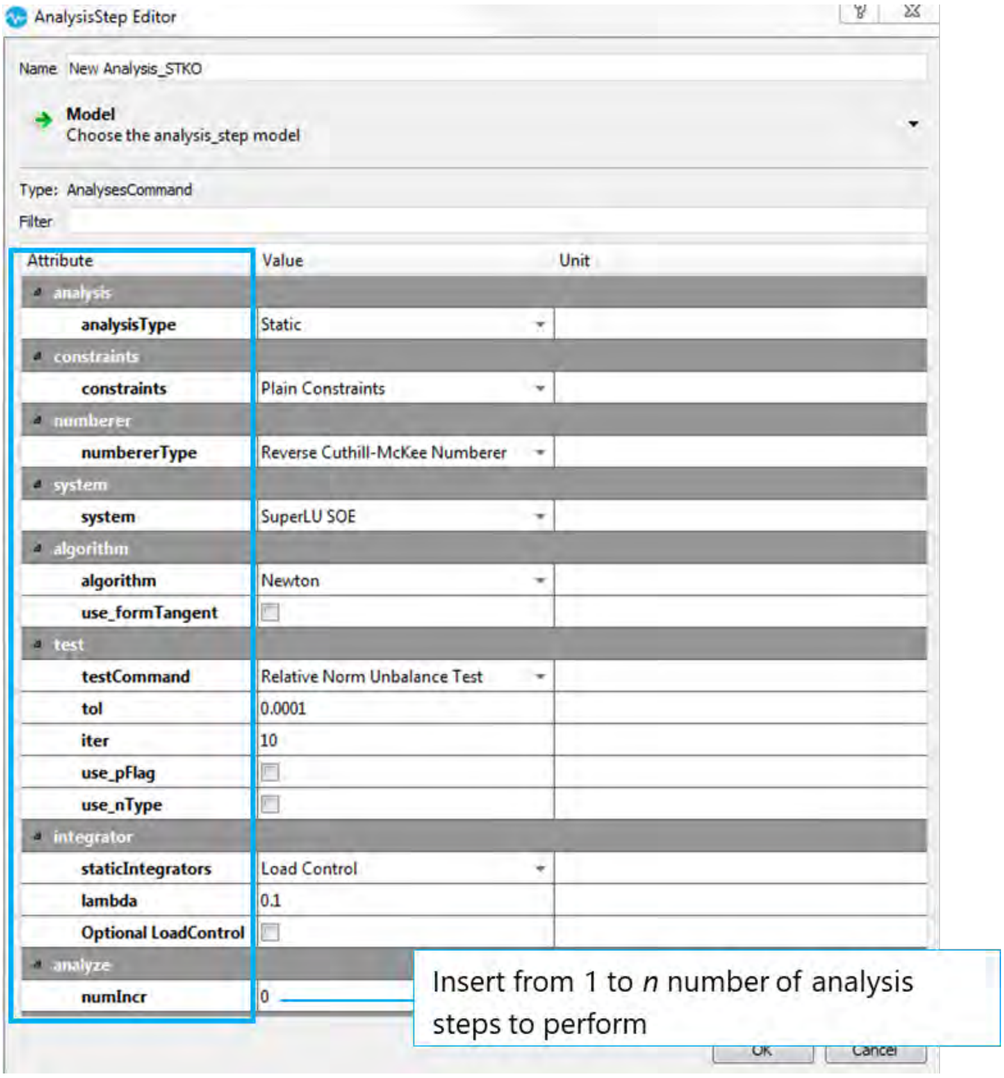


Figure 326. Add the Analysis Step

Click **OK** to confirm the settings.

All the Analyses steps are completed and ready to be exported as **ticol files (.tcl)**.

Click **Analysis** > **Run Analysis** on the main Toolbar. The software will automatically connect to OpenSees and analyse all the inputs from STKO. It will generate an **.mpco** file to be analysed in the **STKO Postprocessor**.

5.3. Shear Frame

Draw a simple model of a 2D Portal on the X-Y plane and create a **Node to Node link Interaction**, like in the following example.

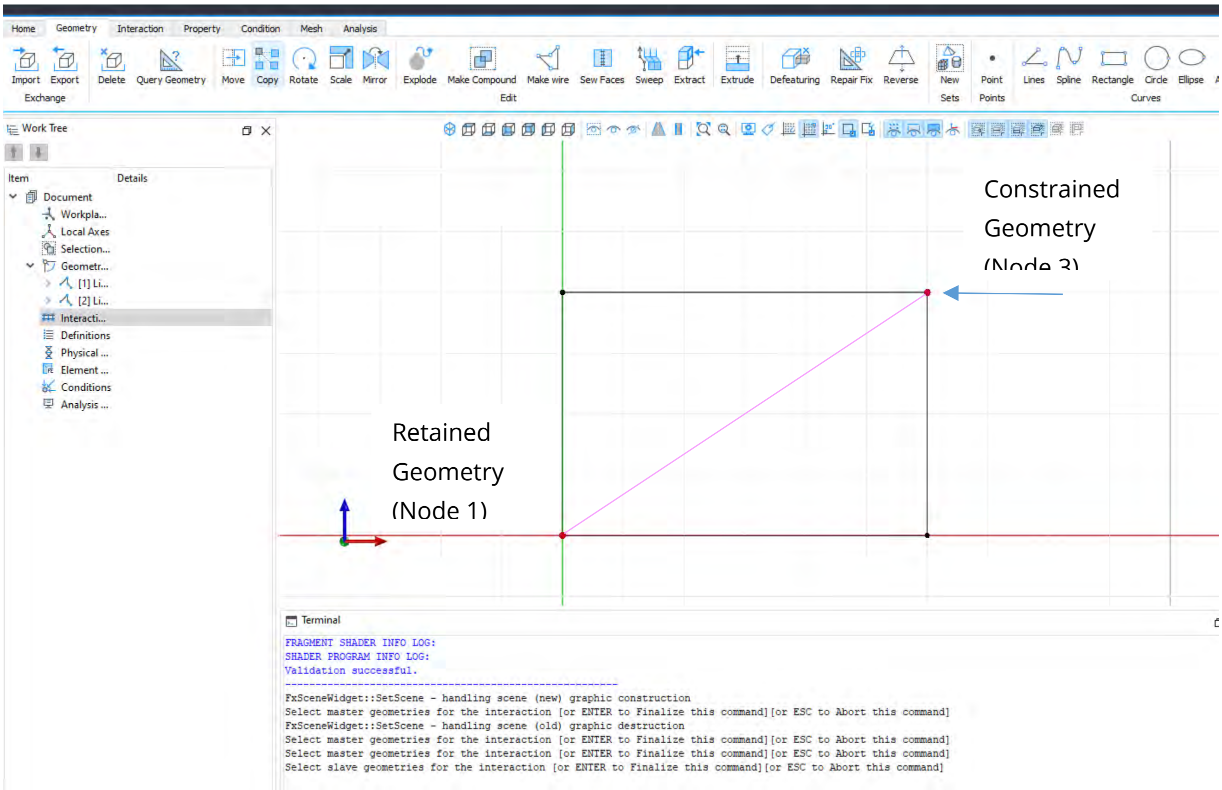


Figure 327. Setting an Interaction

To create an **Interaction**, under **Geometry** on the **Work Tree**, *Right-click* **Interactions** > **Add**. Select **Node to Node links**. *Click* **OK** to confirm the Interaction, then select the retrained and constained geometries to apply the interaction to (in this case Node 1 and Node 3, respectively.) Select **Condition** > **New** from the main Toolbar, the select **Model** > **Constraints** > **sp** > **fix.** Using the Condition Editor, apply the settings shown in the following image.

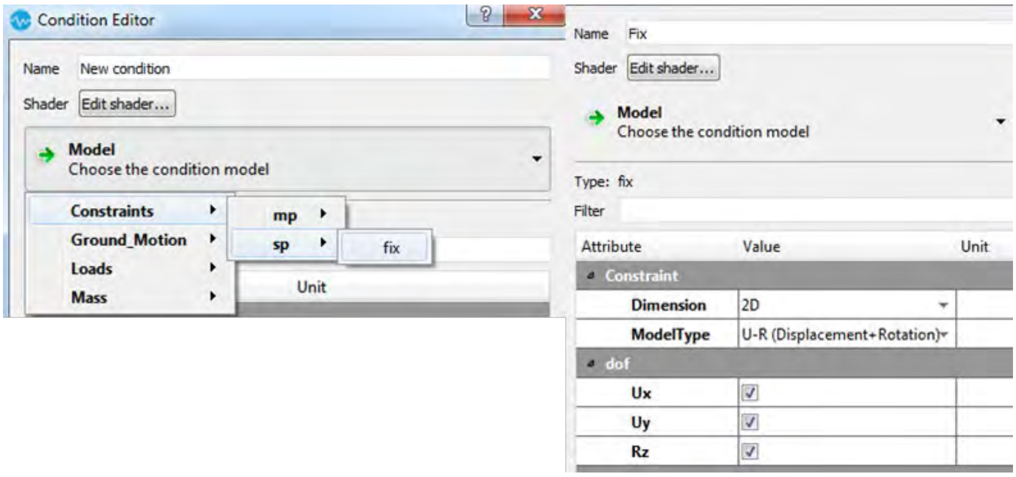


Figure 328. Add an Interaction

Select the node at the base, then *click* **OK** to confirm the settings.
Create another **New Condition** by selecting **Condition** > **New** from the main Toolbar. Use the drop-down menu to select **Model** > **Constraints** > **mp** > **equalDOF**. Using the Condition Editor, apply the settings shown in the following image.

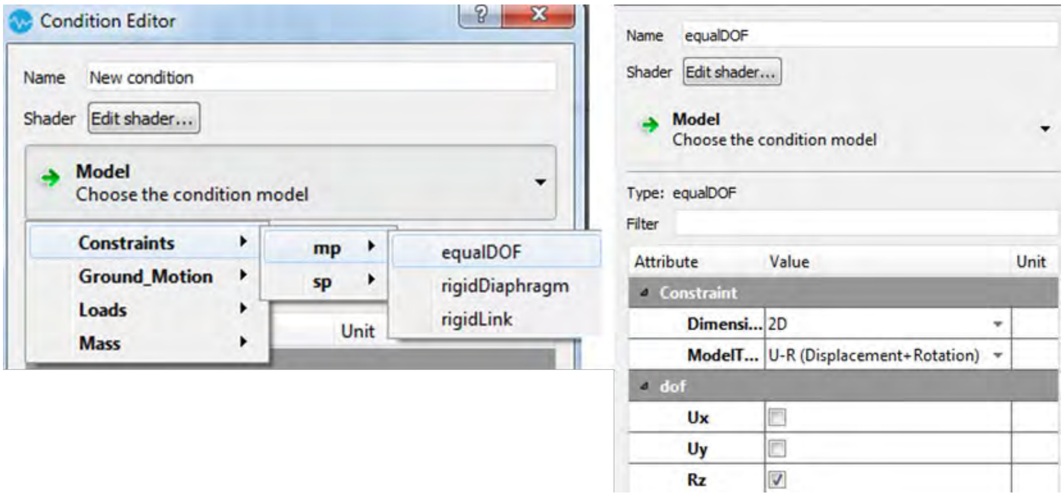


Figure 329. equalDOF

Select the **Interaction** to assign the **equalDOF** to the brace, *Click* **OK** to confirm.
Create the last Condition by selecting **Condition** > **New** from the main Toolbar. Use the drop-down menu to select **Model** > **Mass** > **EdgeMass**, and apply the settings shown in the following image using the Condition Editor.

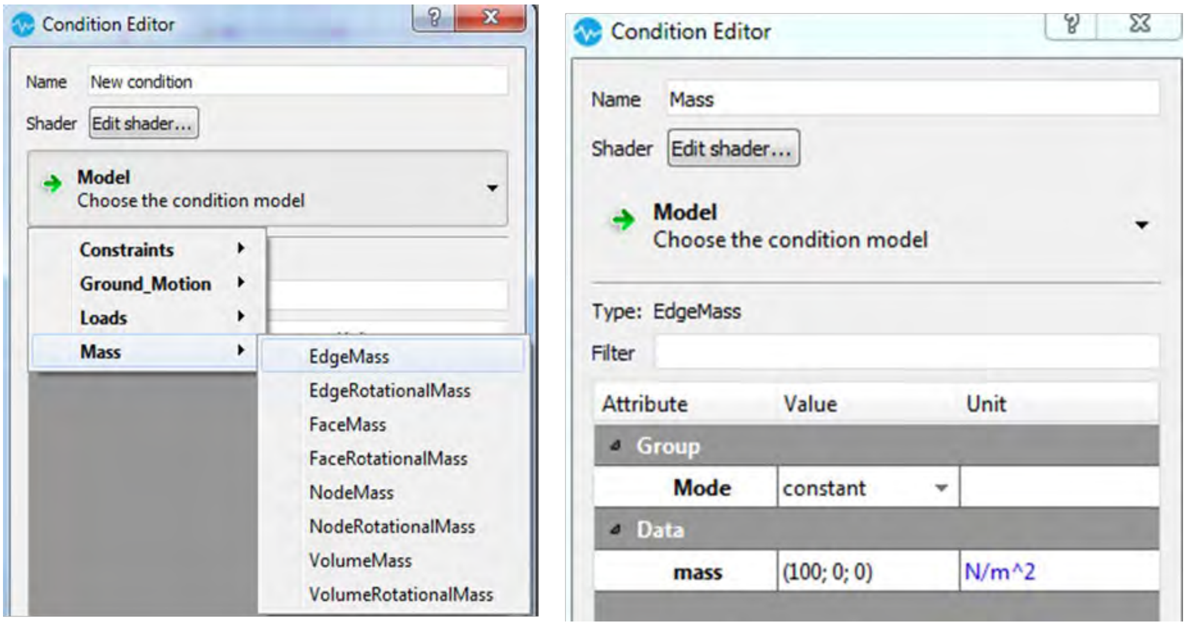


Figure 330. Add EdgeMass

Select the Portal to assign the **EdgeMass** to it. *Click* **OK** to confirm.

Next, define the **Physical Property** of the model. Select **Property** > **New Physical Property** from the main Toolbar, or *Right-click* **Property** > **Add** on the Work Tree. Use the drop-down menu to select **Model** > **sections** > **Elastic** and insert the following settings.

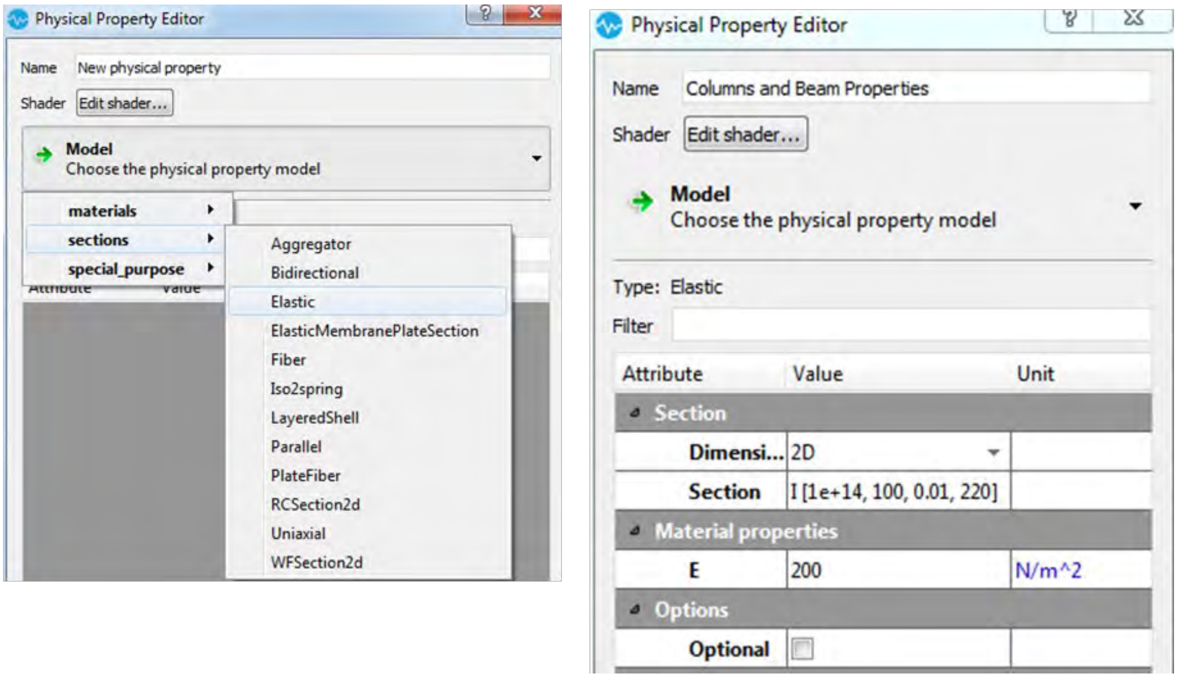


Figure 331. Define the Physical Property

Select **OK** to confirm the settings. Assign the Elastic Section to the Portal by dragging it from the Work Tree panel to the geometries.

Create another Physical Property for the brace. Select **Property** > **New Physical Property** from the Toolbar and select **materials** > **uniaxial** > **Other Uniaxial Materials** > **ViscousDamper.** Insert the following settings into the Physical Property Editor.

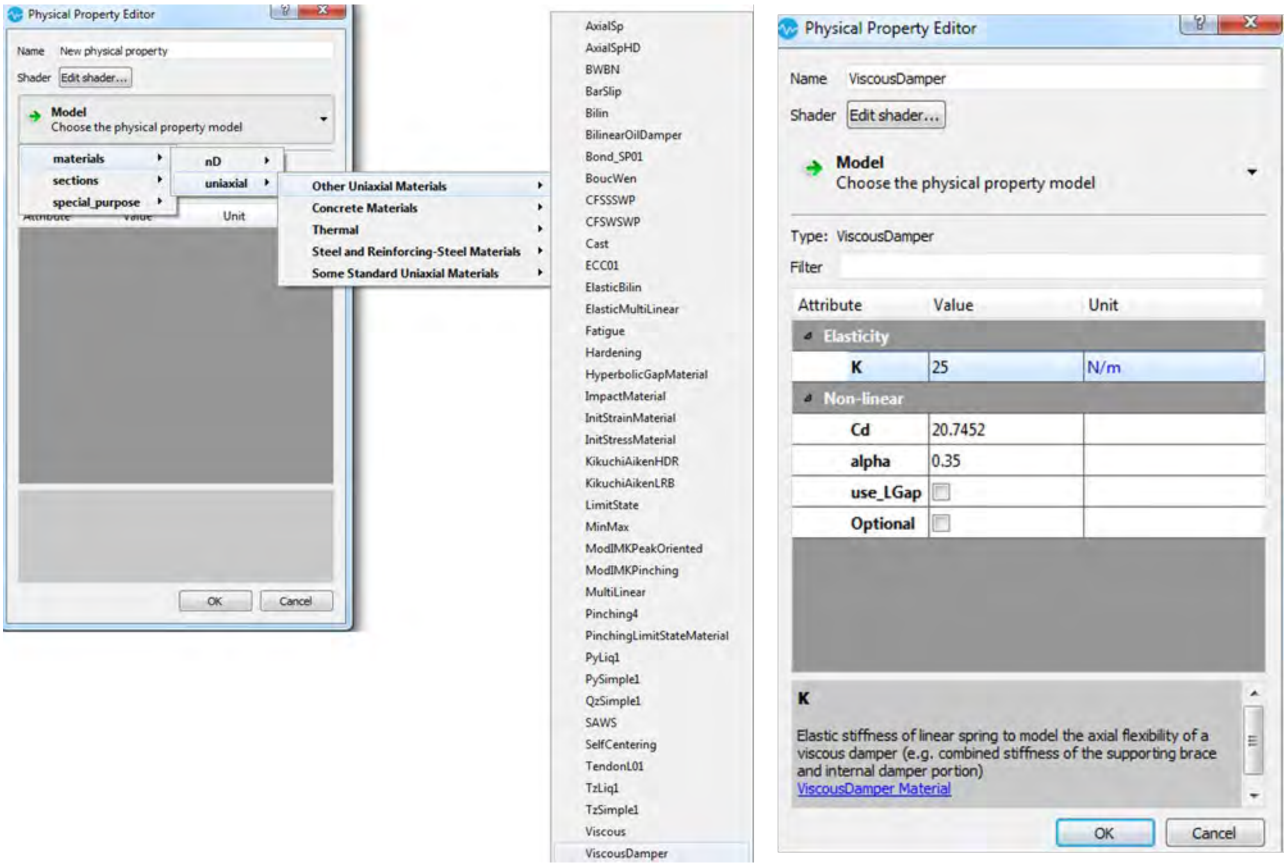


Figure 332. Define the ViscousDamper

Click **OK** to confirm the settings. After creating a ViscousDamper, it is important to define a **Special Purpose** as shown in the following image.

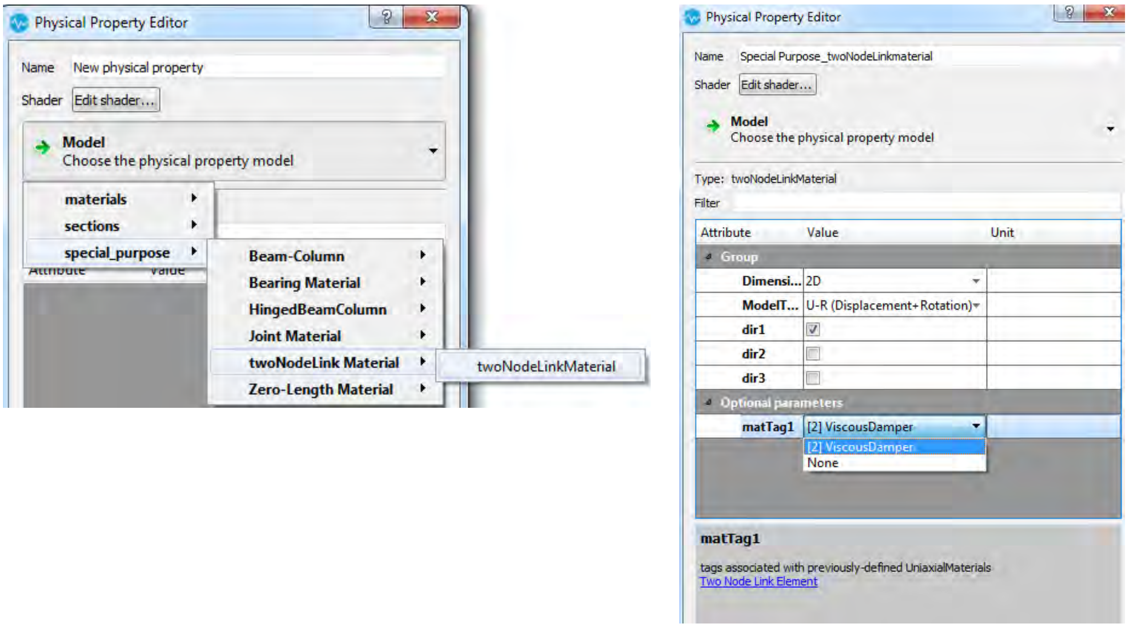


Figure 333. Define a Special Purpose

Click **OK** to confirm the settings. Use the Quick Access Toolbar to view only the interaction, allowing the user to easily assign the special purpose to the interaction:

Figure 334. Using the Quick Access Toolbar

NOTE: To view all physical properties with different colors, use the **Edit Shader** command in the Property Editor to set the colors.

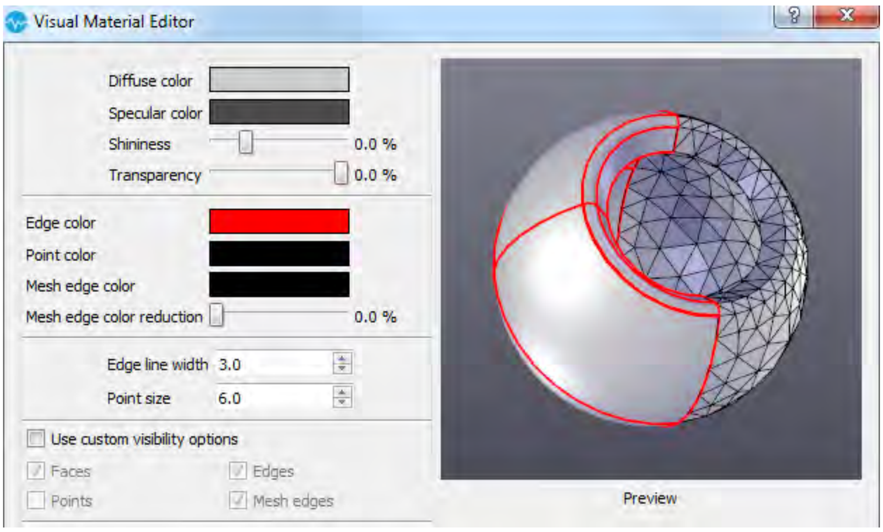


Figure 335. Using Edit Shader

Choosing different colors for each property allows the user to easily see if the properties have been correctly assigned.

After that, it is necessary to define New Element Properties. Select Property > New element property from the Toolbar, and then *Click* Model > beam_column_element > elasticBeamColumn.

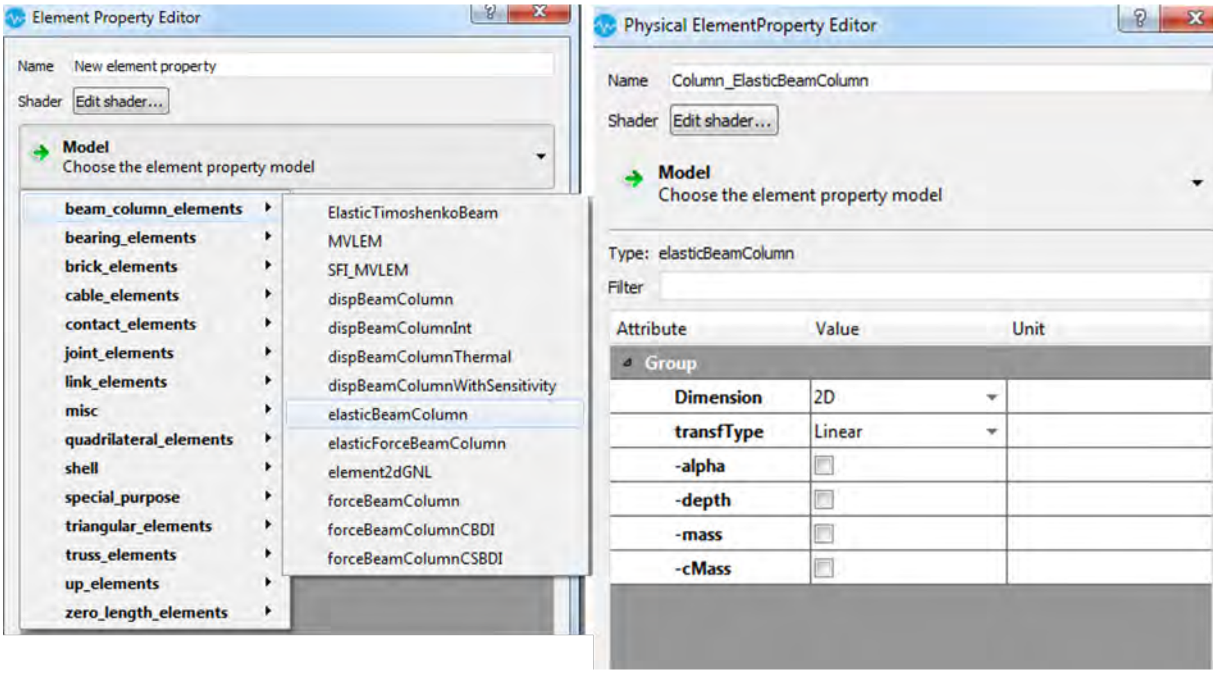


Figure 336. Selecting elasticBeamColumn

Click **OK** to confirm the Physical Element Property Editor settings. Assign it to the Portal by dragging it to the Portal from the Work Tree.

Next, define a link_element, select Property > New element property from the main Toolbar. Then select Model > link_element > twoNodeLink.

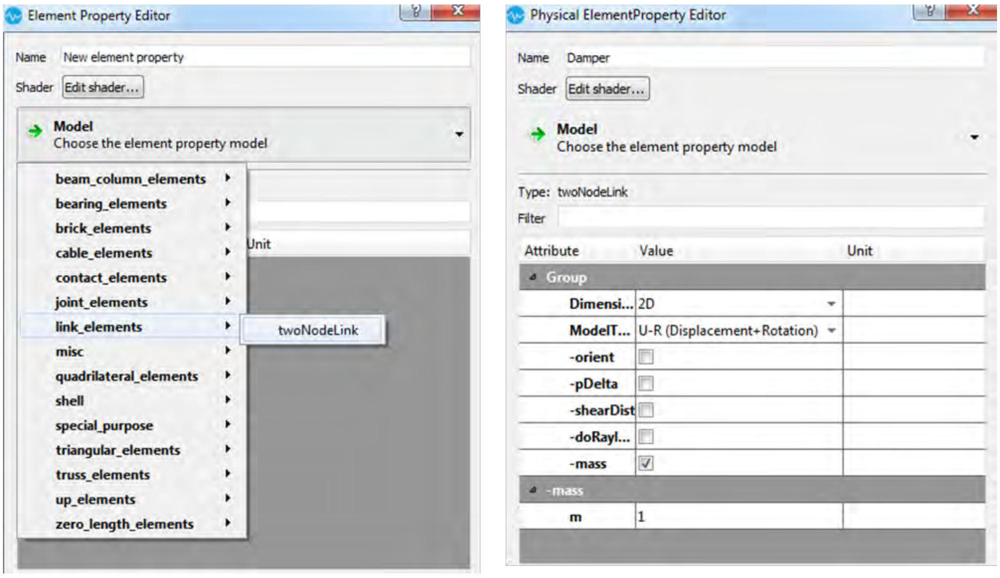


Figure 337. Creating a twoNodeLink

Click **OK** and assign it to the interaction by dragging and dropping it from the Work tree.
Before defining the Analysis step, the user must define a **Time Series**. Select **Property** > **New Definition** from the Toolbar, then select **Model** > **timeSeries** > **Path**.

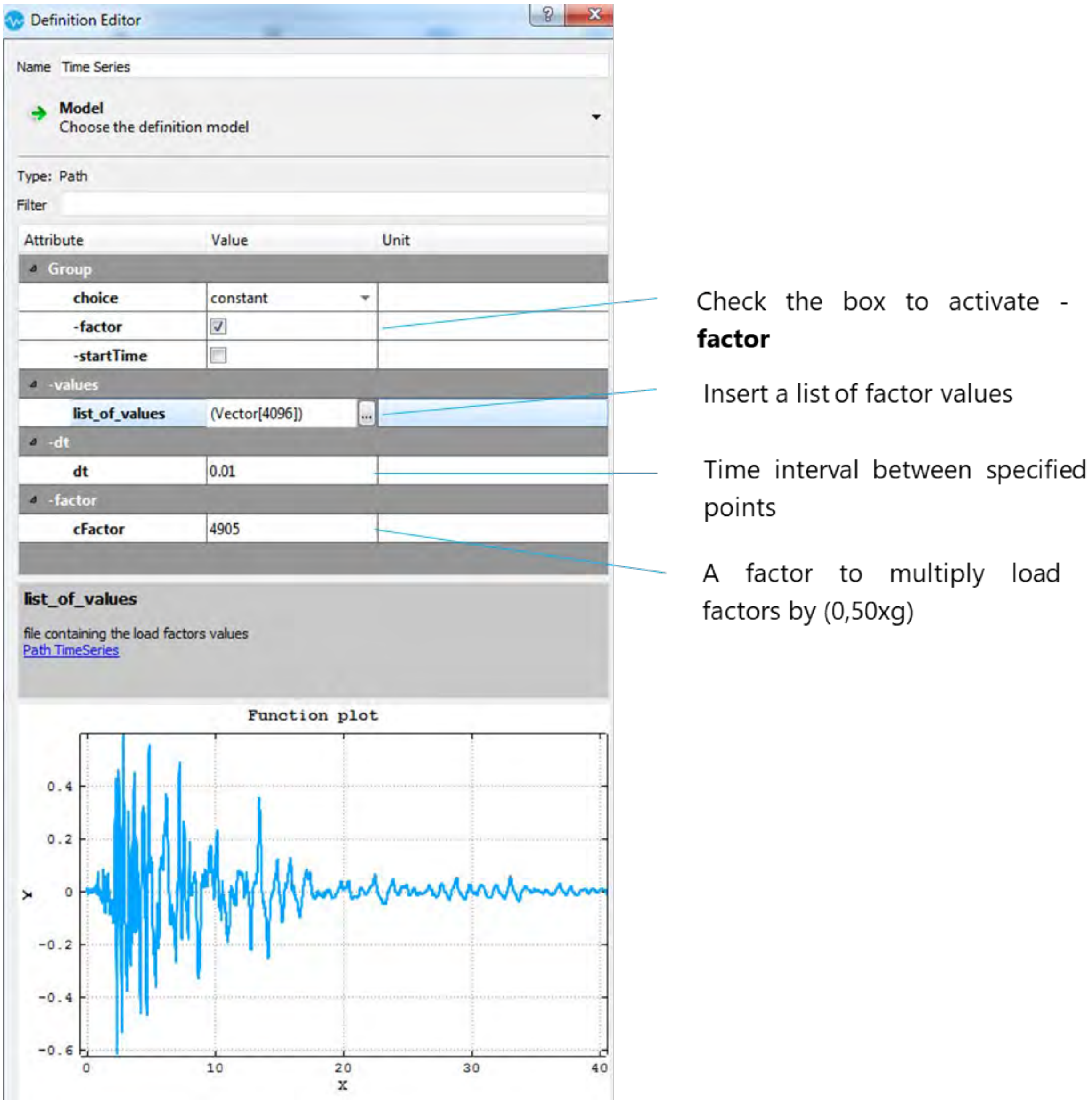


Figure 338. Setting timeSeries Path

Click **Ok** to confirm the settings.

Next, the user should create the **Patterns** that compose the Analysis steps. *Right-click* **Analysis steps** > **Add** on the Work Tree and choose **Model** > **Patterns** > **addPattern** > **constraintPattern.**

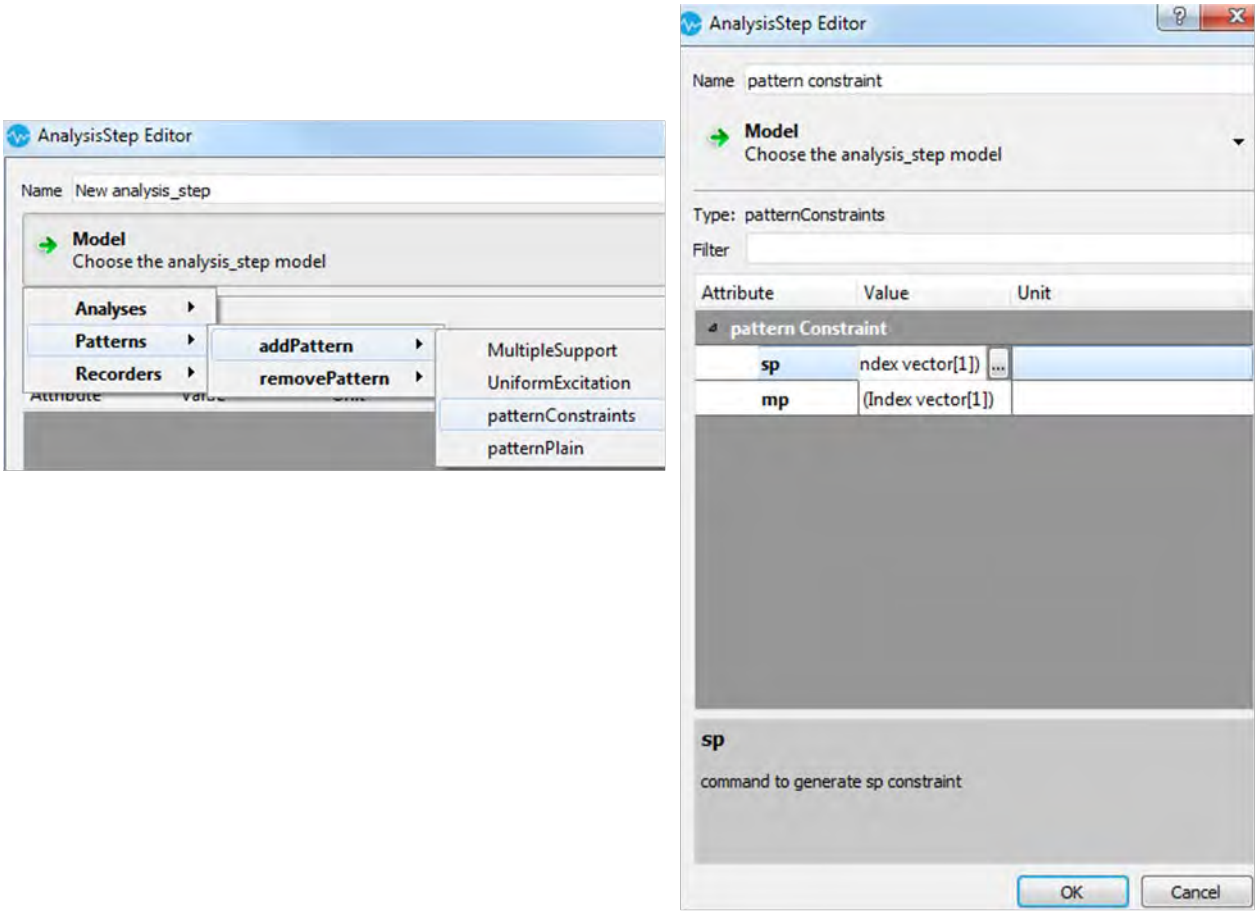


Figure 339. Setting the constraintPattern

Click **OK** to confirm the setting. Repeat the process to add a **UniformExcitation** with the following settings.

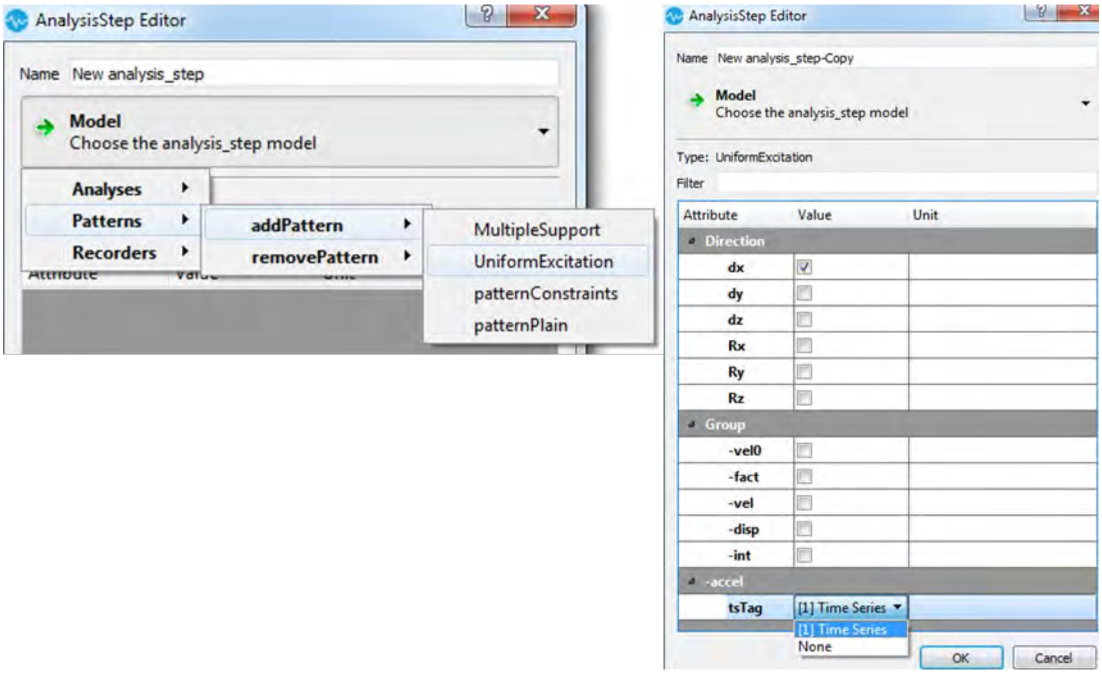


Figure 340. Setting the UniformExcitation

Click **OK** to confirm the settings. The user must now create a **Recorder** by *Right-clicking* **Analysis steps** > **Add** from the Work Tree, and then insert the following settings into the Analysis Step Editor.

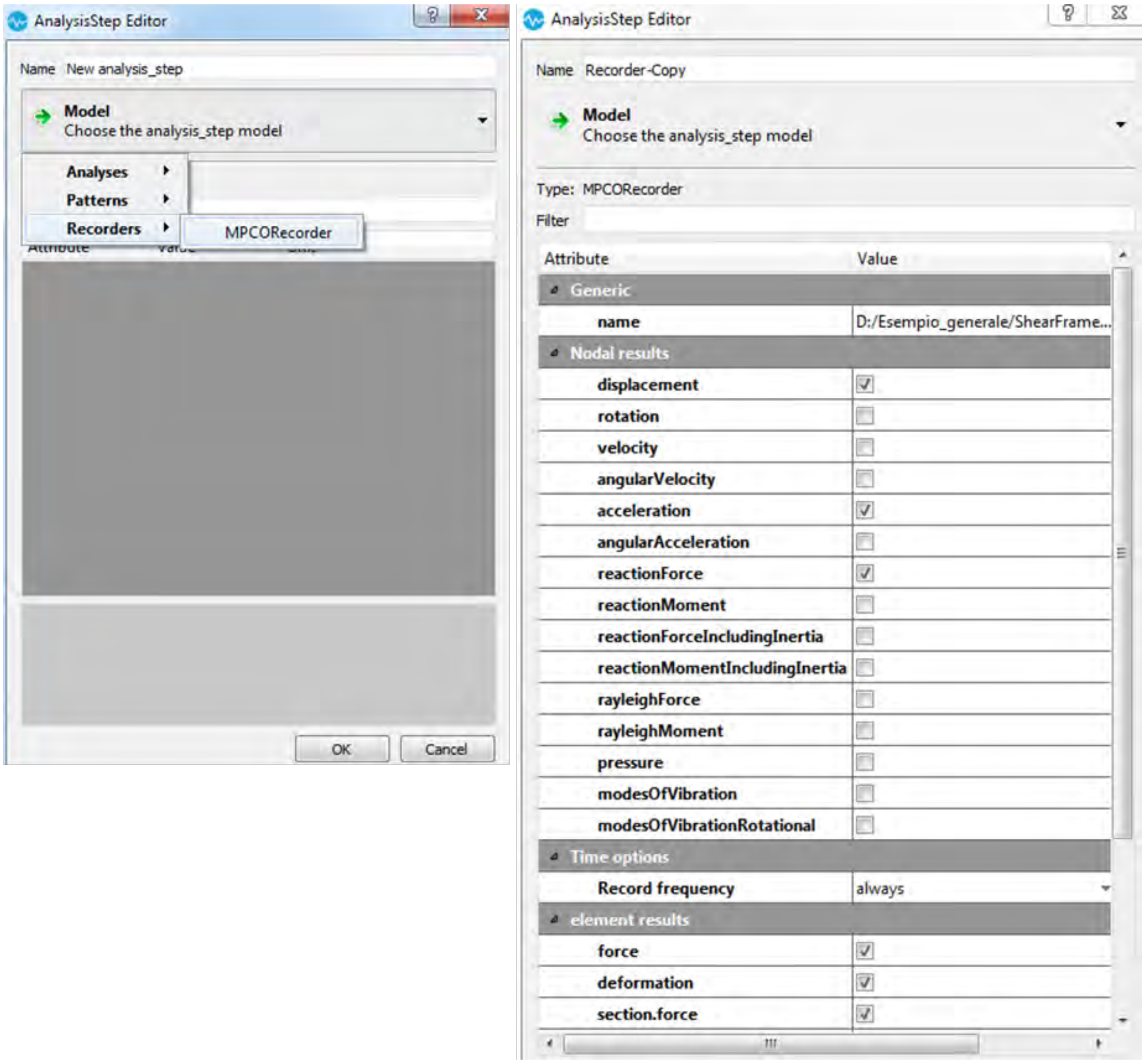


Figure 341. Setting the Recorder

The last step is to create an Analysis. Use the following image as a guide for this step.

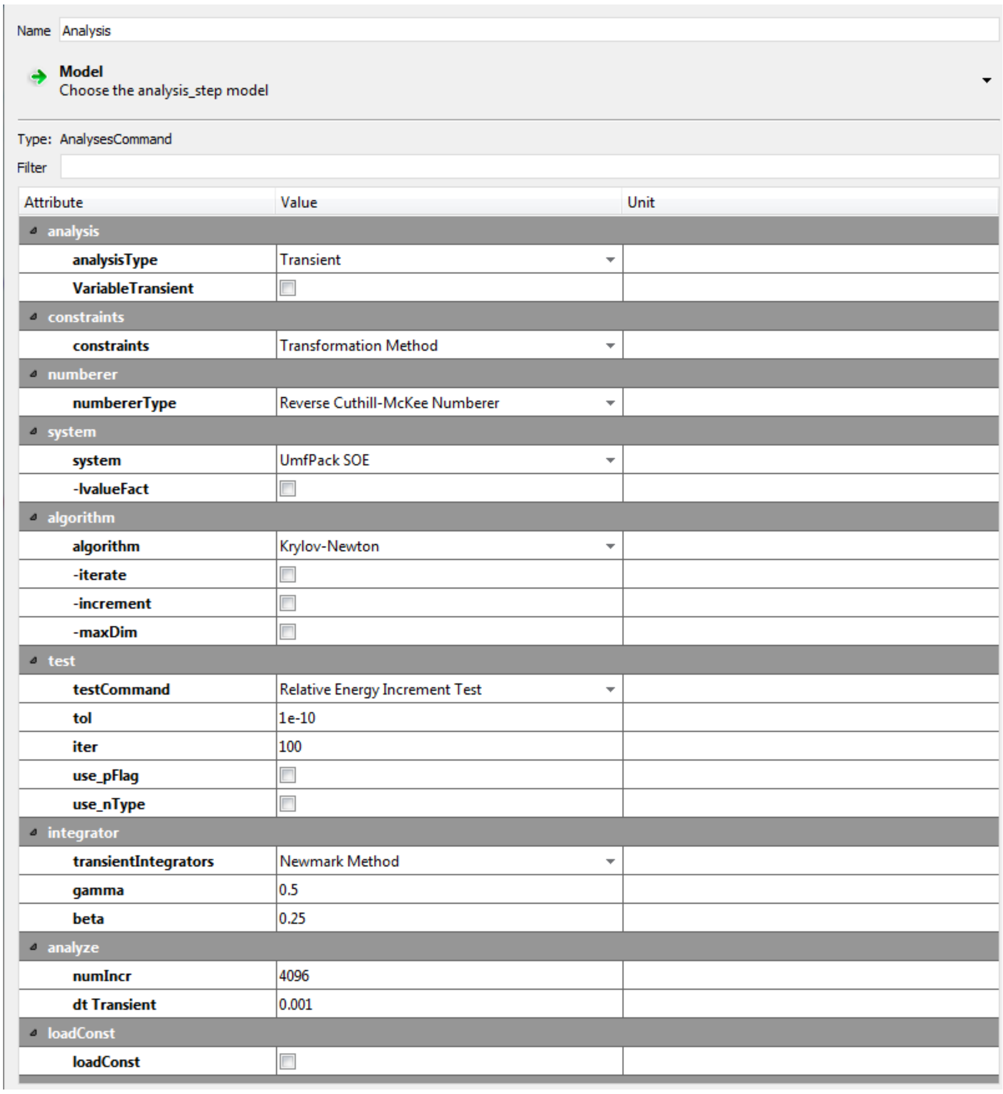


Figure 342. Create the Analysis

Click **OK** to confirm the settings.

Now the analysis is ready to be launched and exported as **ticol files (.tcl)**.

Click **Analysis > Run Analysis** on the main Toolbar. The software will automatically connect to OpenSees and analyse the inputs from STKO. It will generate an **.mpco** file to be analysed in the **STKO Postprocessor**.
Move to the **Postprocessor interface** to load the new database. Select **Open DB** from the main Toolbar or *right-click* to **Databases** > **Add.** Select the file **.mpco** previously created.

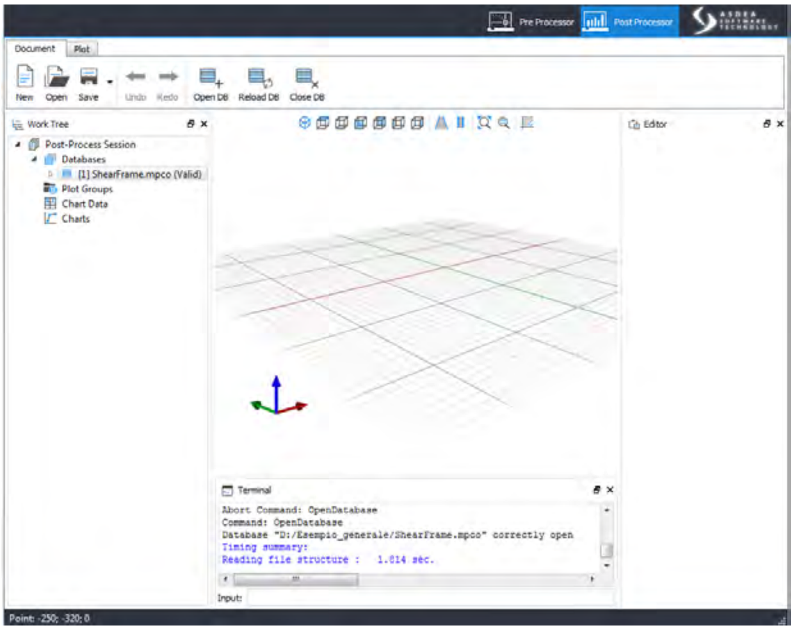


Figure 343. Open the Database

Right-click **Plot Groups** > **New Plot Group** on the Work Tree panel, otherwise directly select **Plot** > **New plot group** from the main Toolbar and choose the Plot to analyse (**Deformed Shape**, **Surface Color Map**, **Volume Color Map**, **Beam/Shell Fiber Color Map,** or **Gauss Point Plot**), for this example, choose **Deformed Shape**.

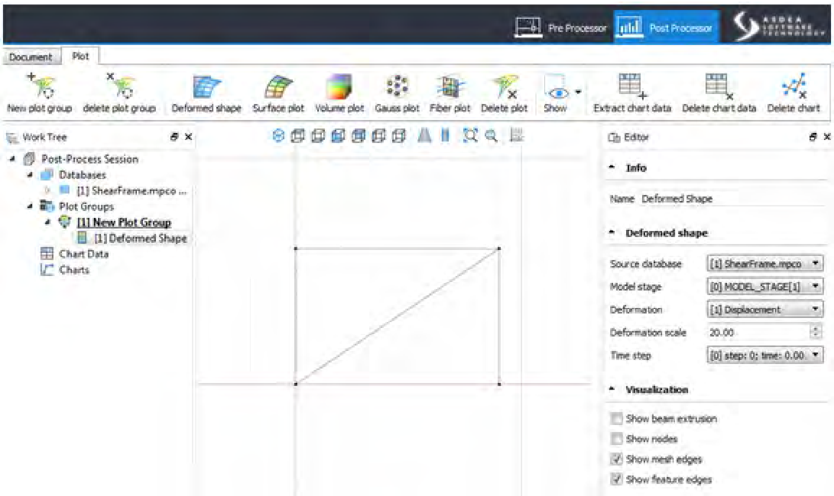


Figure 344. Create a Deformed Shape Plot

Set the type of **Deformation** (**Displacement**), the **Deformation scale** (the *default* is 1, but for this example set it to 20.00), and the **Time step** previously defined in the preprocessor.

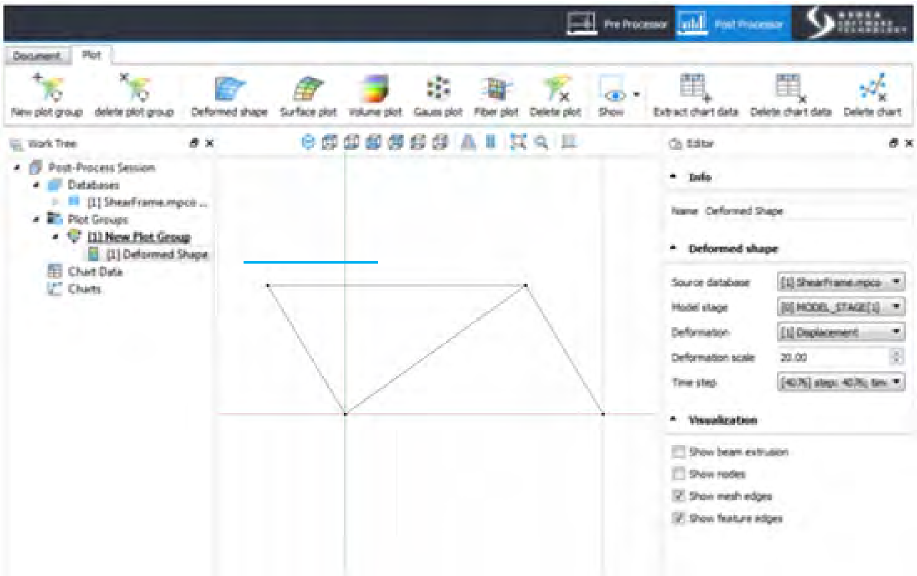


Figure 345. Adjusting the Deformed Shape Plot

To add new **Plots**, *right-click* on **New Plot Group** and select the new Plot. For this example, select **Surface Color Map**.

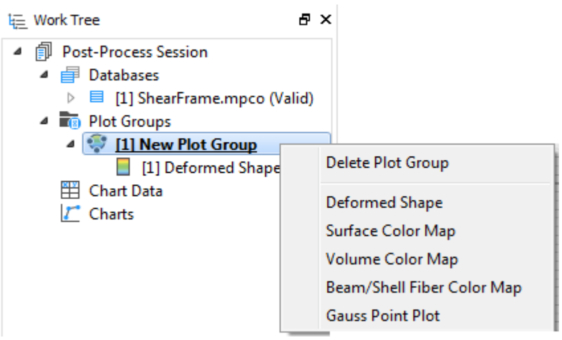


Figure 346. Adding a Surface Color Map

All customizable information for the **New Plot Group** appear in the Editor panel.

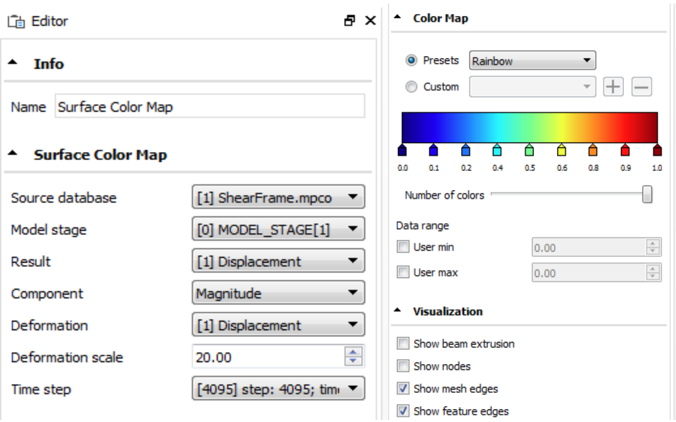


Figure 347. Surface Color Map Editor Panel

5.4. Advanced Modeling Examples

Bridge Modeling

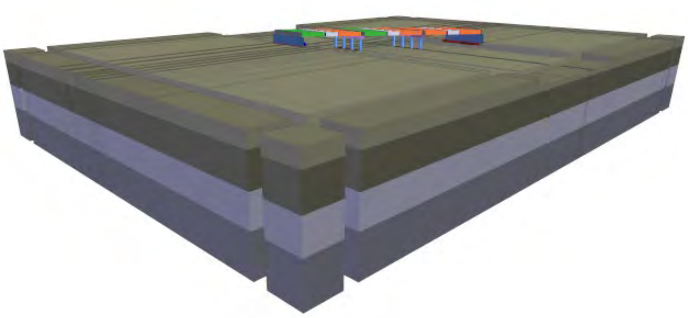


Figure 348. Example of bridge modeling

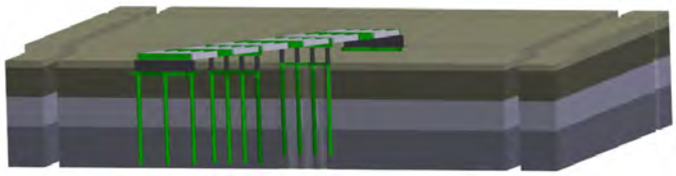


Figure 349. Partial model view

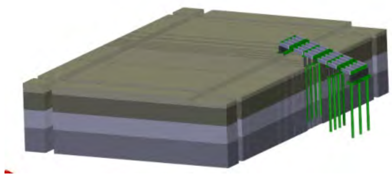
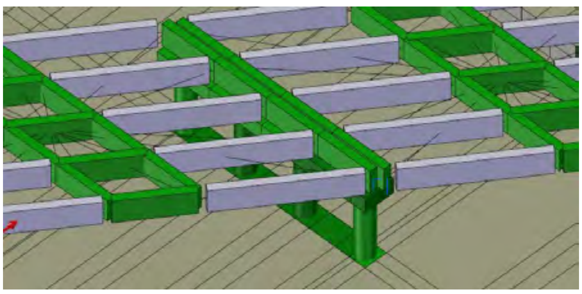


Figure 350. Partial model view



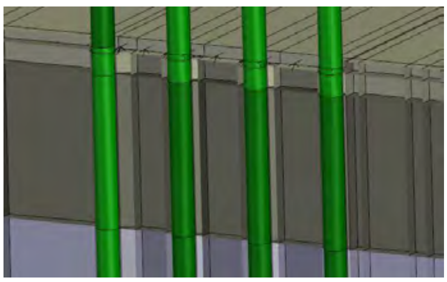


Figure 351. Earth interaction beamSolidCoupling

Figure 352. Pile

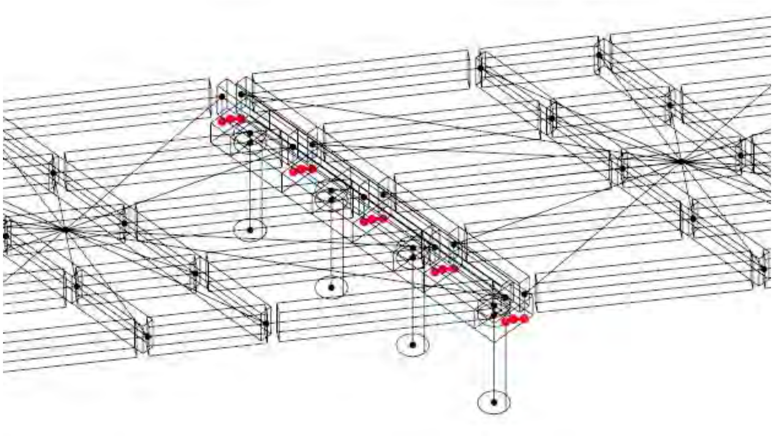


Figure 353. Rigid link application to pile

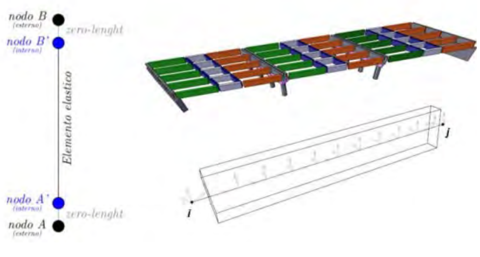


Figure 354. Hinged beam element application

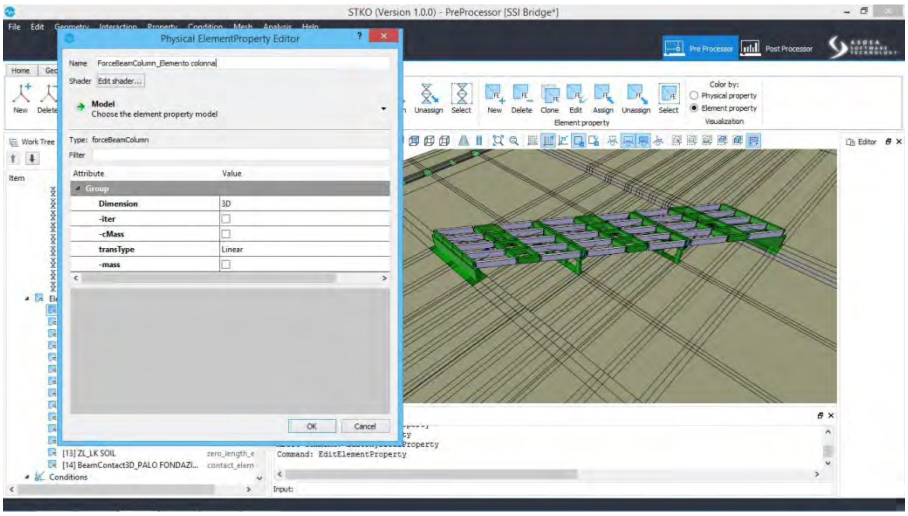


Figure 355. Example of column modeling

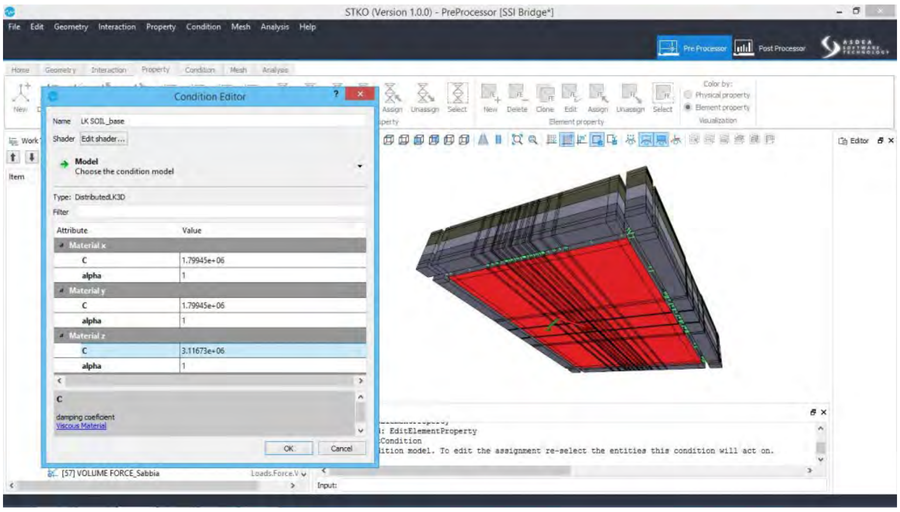


Figure 356. DistributedLK3D

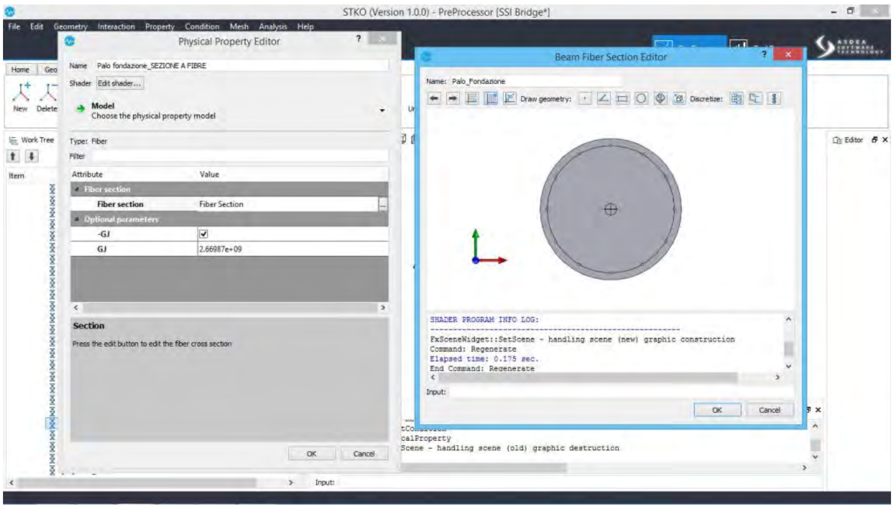


Figure 357. Pile fiber section

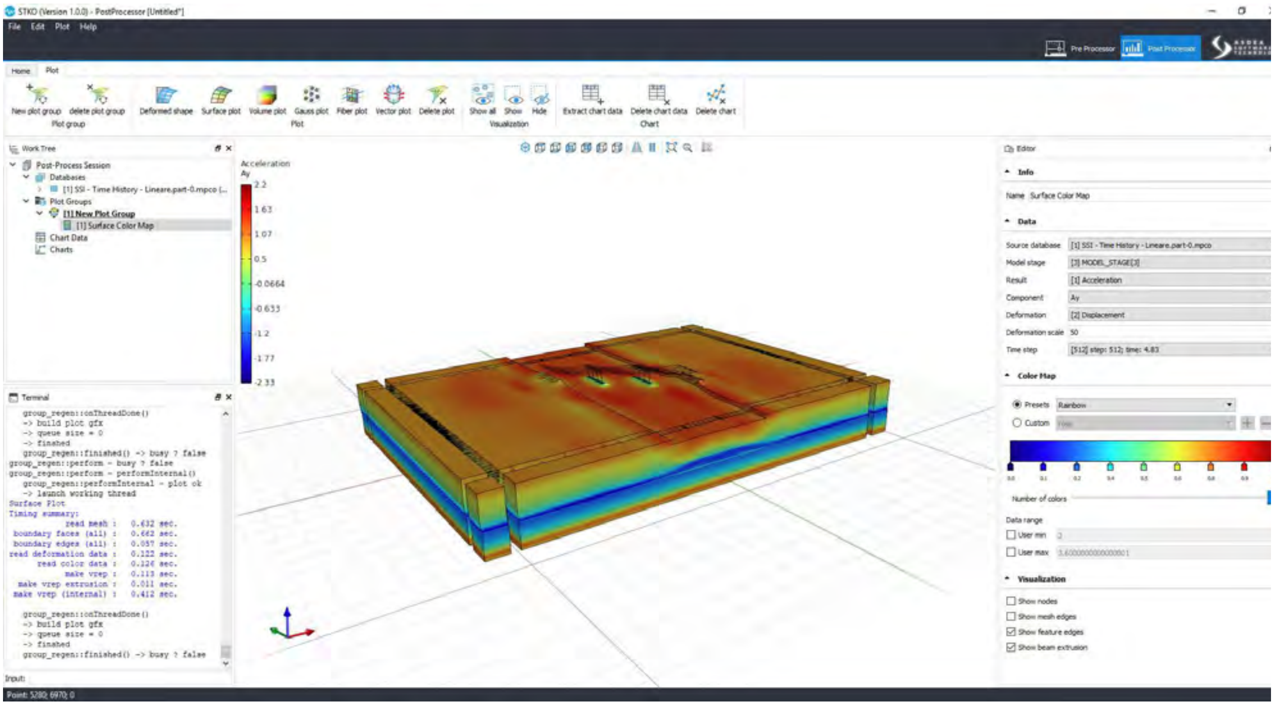


Figure 358. Post-processor analysis

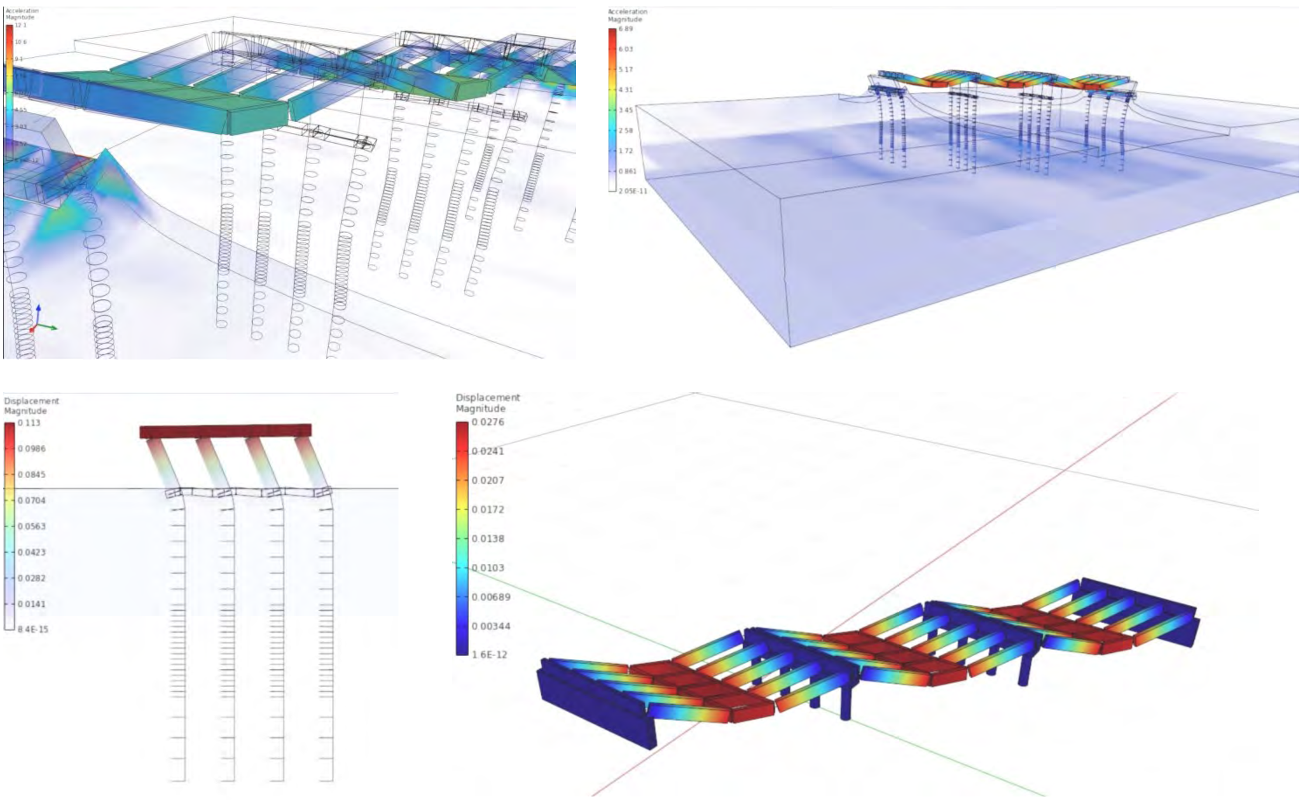


Figure 359. Post-processor various plots

Residential Tower Modeling

Seismically isolated tower.

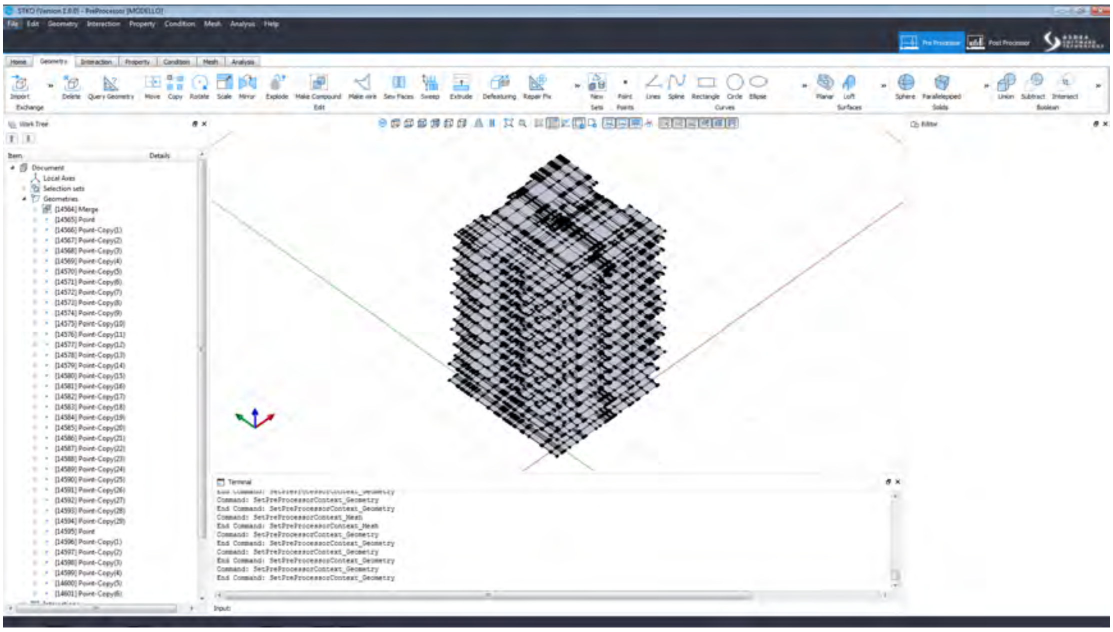


Figure 360. Construction of Geometries

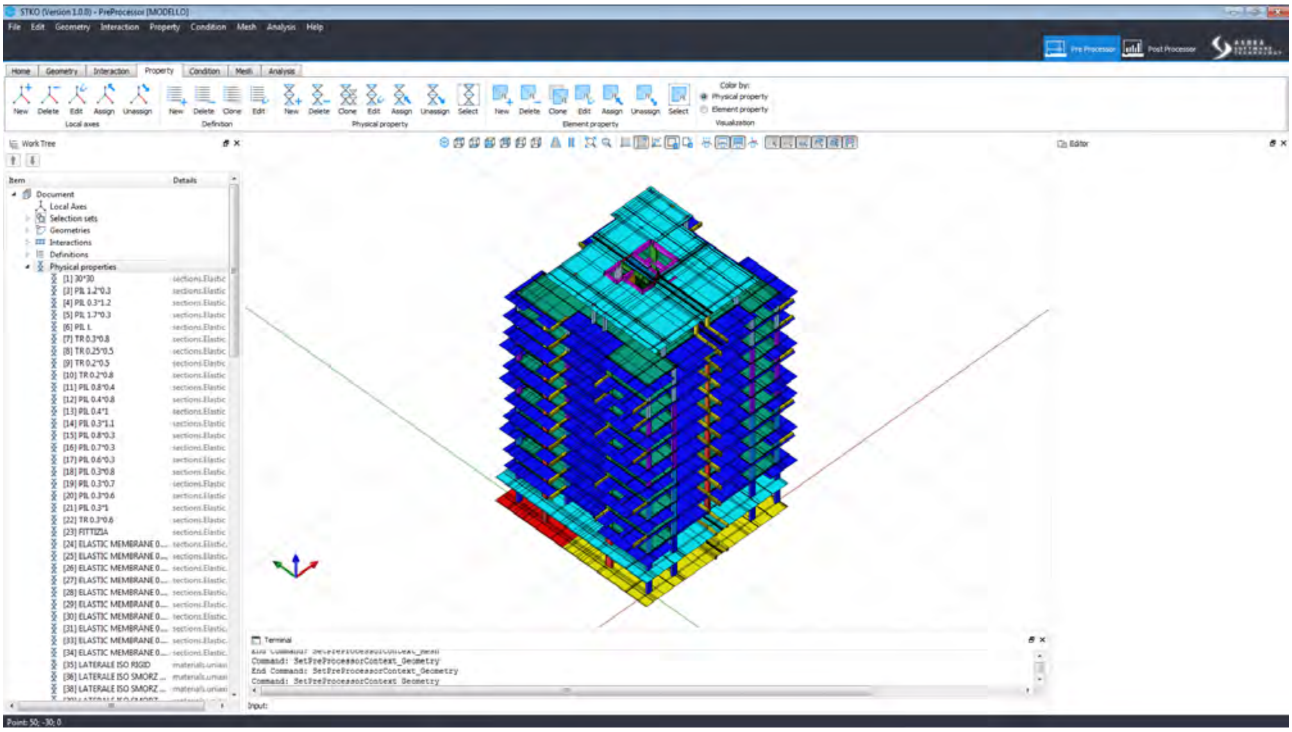


Figure 361. Physical properties attribution.

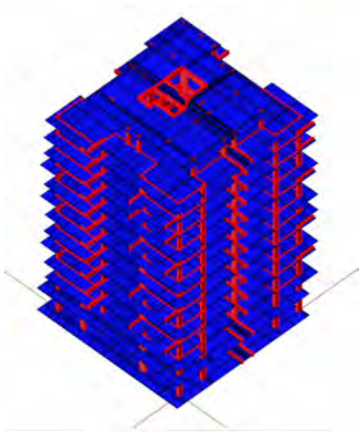


Figure 362. Element attribution: beam column, ShellMITC4, and ZeroLength

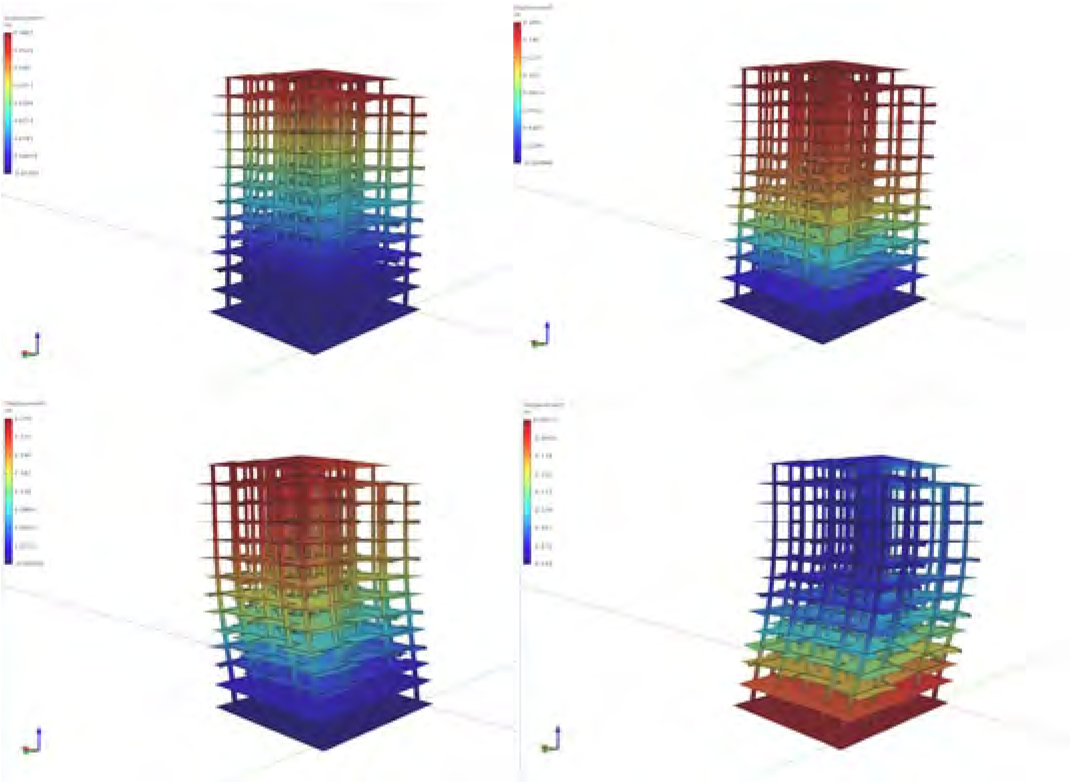


Figure 363. Post processor plot