

# **Grid*Pro* GUI Manual**

**Version 9.0**



Program Development Company LLC

1010 Brickell Avenue, Unit 2704

Miami, Florida, USA 33131.

Tel: +1 (914) 761-1732; Fax: +1 (914) 761-1735

Web: [www.gridpro.com](http://www.gridpro.com)

E-Mail: [gridpro@gridpro.com](mailto:gridpro@gridpro.com)

**GUI Version 9.0**

February 28, 2024

© Copyright Program Development Company, 1992-2024 – Licensed Materials, All Rights Reserved. This document contains proprietary and confidential information of PDC. The contents of this document may not be disclosed to third parties, copied, or duplicated in any form, in whole or in part, without the prior permission of PDC. The contents of this document are subject to change without notice and do not represent a warranty on the part of PDC.

# CONTENTS:

<b>Chapter 1 – Introduction</b>	<b>1</b>
1.1    What is <i>GridPro</i> ?	2
1.2    How <i>GridPro</i> works?	3
<b>Chapter 2 – Getting Started With GUI</b>	<b>6</b>
2.1    Introduction to GUI	7
2.2    Top Menu Bar	11
2.3    Global Dock	23
2.4    Workplane Toolbar	51
2.5    Toggle Surface Section	59
2.6    Toggle Display Section	61
<b>Chapter 3 – Surface</b>	<b>67</b>
3.1    Surface Types	68
3.2    Surface Orientation	83
3.3    Surface Probing Utilities	85
3.4    Surface Reconstruction Utilities	88
3.5    Right Click Menu	103
<b>Chapter 4 – Topology</b>	<b>104</b>
4.1    Topology Building	105
4.2    Topology Creation Utilities	110
4.3    Automatic topology creation utilities	126
4.4    Topology Grouping Utilities	136
4.5    Topology Modifying Utilities	147
4.6    Topology Transformation Utilities	158
4.7    Topology Enhancing Utilities	167



4.8	Corner Reposition Utilities	179
4.9	Topology Optimization Utilities	185
4.10	Surface Assignment Utilities	196
4.11	Surface Building Utilities	203
4.12	Gridding Process	219
<b>Chapter 5 – Grid</b>		<b>223</b>
5.1	Grid Viewing Utilities	224
5.2	Preprocessing Utilities	232
5.3	Post Processing Utilities	233
5.4	Grid Probing Utility	250
5.5	Quality Check Parameters	253
5.6	Grid Toggling Section	255
5.7	Grid Trimming Utilities	258
<b>Chapter 6 – Properties</b>		<b>263</b>
6.1	Properties Section	264
6.2	Solver Section	269
6.3	File Format	272
<b>Chapter 7 – CAD</b>		<b>274</b>
7.1	Geometry Faces Section	275
7.2	Tools Section	277
7.3	Right Click Menu	289
<b>Appendix A</b>		<b>290</b>
<b>Appendix B</b>		<b>294</b>
<b>Appendix C</b>		<b>300</b>



# CHAPTER 1

## INTRODUCTION

A good general-purpose grid (mesh) generator should be at least good on two accounts:

- 1) Quick and easy to set up typical complex gridding problems: The considerations include the first gridding turn-around time, the subsequent parametric design turn-around time, the modular parametric design (adding and subtracting features) turn-around time and the clustering capabilities (for CFD use).
- 2) Good grid quality includes grid smoothness, orthogonality, desired grid distribution and surface fidelity. Both accounts are better served through automation with different levels of user-selectable controls.

For a multi-block structured grid generator, automation can be classified into four areas:

- 1) Surface restructuring & repair
- 2) Topology generation
- 3) Bookkeeping of topological information
- 4) Optimum distribution of high-quality grid.

To this end, *GridPro* is a general-purpose, 3-dimensional, multi-block structured grid (mesh) generator using an advanced smoothing scheme that incorporates many automatic features.

## 1.1 What is GridPro?

GridPro is an automatic, object-oriented, multi-block grid generator with a highly efficient, interactive capability to build topology, edit surfaces, set CFD boundary conditions and view grids. GridPro is unique; it minimizes your work and maximizes your efficiency; it has full automation in high-quality grid distribution and the bookkeeping of topological information.

Relying heavily on automation and lightly on user input, GridPro generates multi-block, structured grids with relative ease and quickness. Input the surfaces and wireframe topology; GridPro does the rest. Users interact with GridPro through the GUI, an easy-to-use, dynamic 3D environment that combines a powerful scripting language with a user-friendly GUI front end. Automated processes minimize the amount of manual manipulation required to generate a grid, reducing errors in grid specification. GridPro automatically creates templates of the grids to permit fast optimization to meet specific criteria. The software engine incorporates a unique algorithm using advanced mathematical techniques to generate the highest quality grids. The algorithm optimizes each grid so that every element is smooth and orthogonal throughout the volumetric region under consideration. GridPro runs on all major operating systems and hardware.

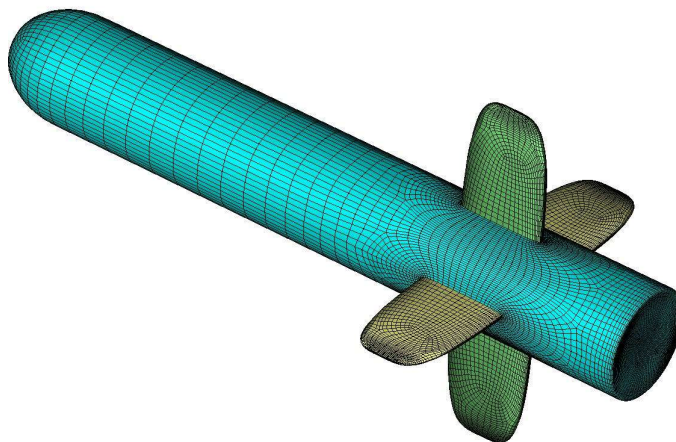


Figure 1.1: Missile configuration

The design of GridPro has followed the principles:

- 1) Minimizing the user input with a strong emphasis on a topological template, construction capability and reusability
- 2) Maximizing the grid quality
- 3) Optimizing the grid distribution.

The first principle cuts down the initial setup time and the subsequent setup time for configuration modifications more drastically. The second translates into a higher solution accuracy and faster convergence for the CFD solvers. The third reduces the demand for computer resources in CPU time and RAM usage.

The mathematical underpinnings that reduce the user effort also account for *GridPro*'s substantially high grid quality. This quality comes in near orthogonality, smoothness, low warpage, and curvature clustering for both concave and convex boundaries.

*GridPro* can be used for multiple applications in multiple ways. The program will grid simple or complex cases in two or three dimensions. By producing grids on objects as diverse as a nuclear test missile or a brain aneurysm, *GridPro* showcases its flexibility as well as its depth. Obtaining realistic configurations is an essential industrial need for engineering analysis. These configurations are often complex and typically require a large amount of human time to produce a grid. *GridPro*'s automated process handles the massive complexity that requires a large number of blocks very efficiently (with a large number of blocks). It can take hundreds or thousands of blocks. There is no limit to the number of blocks.

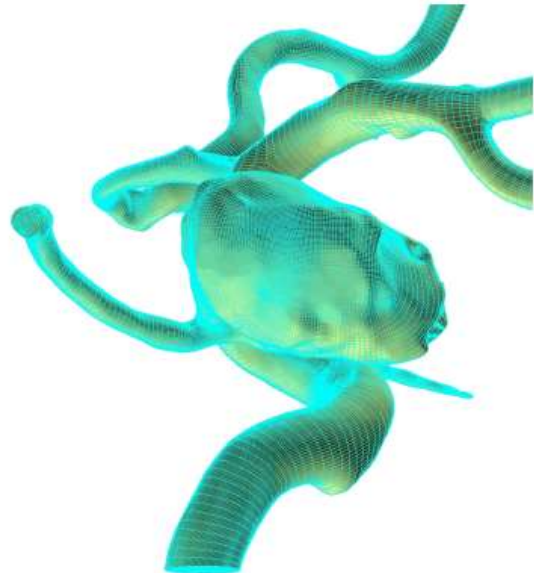


Figure 1.2: Brain aneurysm

## 1.2 How *GridPro* Works?

*GridPro* is an 'automatic' grid generator, which accesses an automatic process earlier than any other 'automatic' grid generator. *GridPro*'s automation includes surface grid generation, zone construction, and the intersection between surfaces.

*GridPro* essentially reduces grid generation to topology generation. The primary user input is a pattern of points referred to as the grid topology. The language that organizes this pattern of points is called 'Topology Input Language' (TIL). You can input these points directly in the GUI. As you input points in GUI, TIL code is automatically generated. Users have choices of using GUI or TIL code.

The grid generation process in *GridPro* primarily consists of four sequential processes; they are

1. Surface preparation
2. Topology generation
3. Grid generation
4. Property setting.

GridPro contains built-in surfaces, which are ellipsoids and planes that, although simple, are helpful in many practical circumstances. However, because real-life objects rarely mimic symmetrical regions, surfaces from other design programs such as CAD (Computer Aided Design)

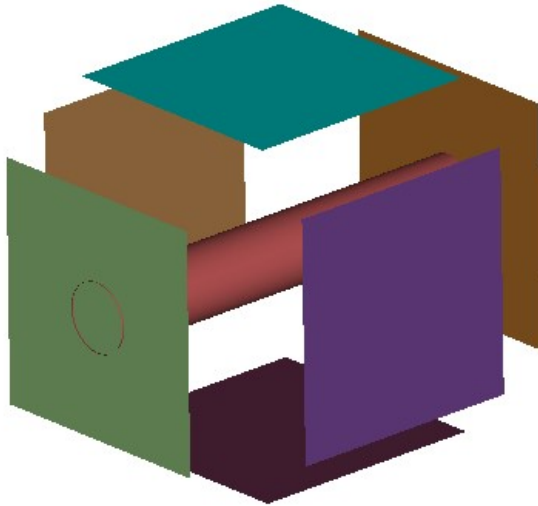


Figure 1.3: A collection of surfaces bounding a region to be gridded

can also be imported. It also has an automatic surface restructuring and repair capability, such as auto-stitching of surface gaps between surface patches and implicit surface trimming and intersection capturing. The first step in the gridding process is to define the region to be gridded, as shown in figure 1.3. After inputting surfaces, whether or not they are built-in from GridPro or CAD, a topology for the region bounded by those surfaces must be created. The majority of GridPro operations deal with creating and modifying topology. GridPro's unique methodology significantly increases productivity by minimizing the amount of user interaction and individual input. It partially automates topology

generation by reducing the user task to generate a coarse wireframe of the topology in which only imprecise corner and edge information are required. In contrast, the blocks and block faces are automatically generated from the wireframe.

Topology generation contains two parts: (a) wireframe construction and (b) surface assignments. A topological framework (or wireframe topology - see figure 1.4) is appropriately designed according to the multi-block scheme GridPro utilizes. Upon inspection, the user can note that the wireframe defines a pattern of grid points. To design a wireframe correctly, define the wireframe by creating a coarse unstructured quadrilateral (2D) or hexahedral (3D) grid about the region. Unlike the frames of some other grid generation systems, the pattern defining points do not need to be precisely positioned.

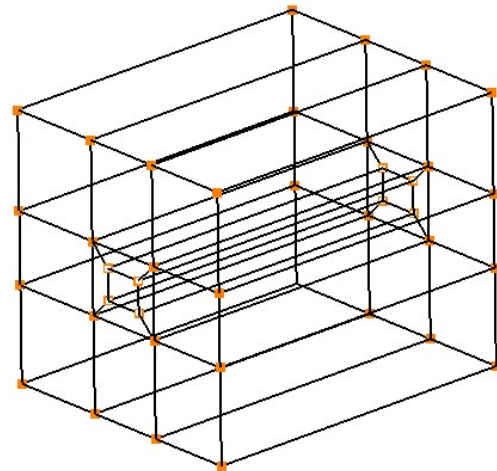


Figure 1.4: Wireframe for the region enclosed by the surfaces in Figure 1.3

The two functions, which help build a wireframe or can be said to be the heart of creating the topology, are *groups* and the *workplane*. Firstly, the workplane is a tool, which guides the cursor in three-dimensional UI. The workplane is a plane in space used for interactive topology generation and graphical viewing. Second, the groups are the most sophisticated tool to use while working on complex geometries. Groups are collections of or set of selected corners and edges. Groups allow the user to create topologies many times faster than sequential corner-by-corner,

edge-by-edge constructions on the workplane. These are the two functions that help the user to generate topology effectively and efficiently. Apart from these tools, an important term will often be used in the manual, i.e. the *current surface*. The current surface is a surface on which the function can be executed or applied. The current surface can be identified in two ways; one depends on the surface's colour. It is always highlighted in sea blue colour; the second method uses the surface id displayed in the GUI, which will be discussed later.

After completing the wireframe, an association between wireframe and surfaces is required, which is done by assigning appropriate wireframe corners to the respective surfaces. This is called *assignments* or *association* in *GridPro*. *GridPro*'s mathematical engine relieves much of the tedious work of constructing the various parts of a grid and then properly assembling them. With this technology, users are not required to generate surface grids or build items such as block edges and their point-wise distributions. The user gets both surface and volume grids simultaneously and automatically.

*GridPro* employs a powerful proprietary Dynamic Boundary Conforming (DBC) technology, which automatically morphs the topology wireframe grid into conforming the bounding geometry, generating a multi-block grid. The process is driven by the algorithms that ensure the highest possible grid quality level available under the constraints defined by the user. This quality level is difficult, if not impossible, to attain by traditional methods of multi-block grid generation. DBC technology enables high-quality design optimization to proceed automatically. As users change region boundaries to achieve the desired physical attributes, *GridPro* dynamically tracks the evolving boundary shapes. Traditional approaches to grid generation are labour-intensive and highly dependent upon user input, whereas, in *GridPro*, it's vice versa.

# CHAPTER 2

# GETTING STARTED WITH GUI

*GridPro* takes the grunt work out of grid generation. The program has features that enable you to perform various basic and complex procedures without having to manually write TIL (Topology Input Language) code with its associated minutiae and rules. The benefit of TIL code, on the other hand, is the versatility of being able to handle all possible scenarios. The topology codes are written to TIL code files. These files can be created by either interactive means or manual text edit and can be reused, edited, or combined.

Being the only program boasting a fully compiled language, *GridPro* provides you with the option of working in the TIL code or the GUI. The GUI provides an interactive environment for the gridding process, allowing the user to see and manipulate the object being gridded. Also, the GUI is the underlying system that enables *GridPro* to transform topologies into grids. The resulting grids are structured multi-block grids. In this manual, the effective usage of the GUI is thoroughly presented.



## 2.1 Introduction to GUI

The GUI window is designed in such a way to guide through the *grid generation process* required in GridPro to get a good quality grid. The main window consists of five different tabs such as 'Surface', 'Topology', 'Grid', 'Property' and 'CAD', which helps the user to work on different tabs simultaneously. Each tab has a drawing area, top menu bar, global dock, workplane toolbar, right side panel, and a bottom bar.

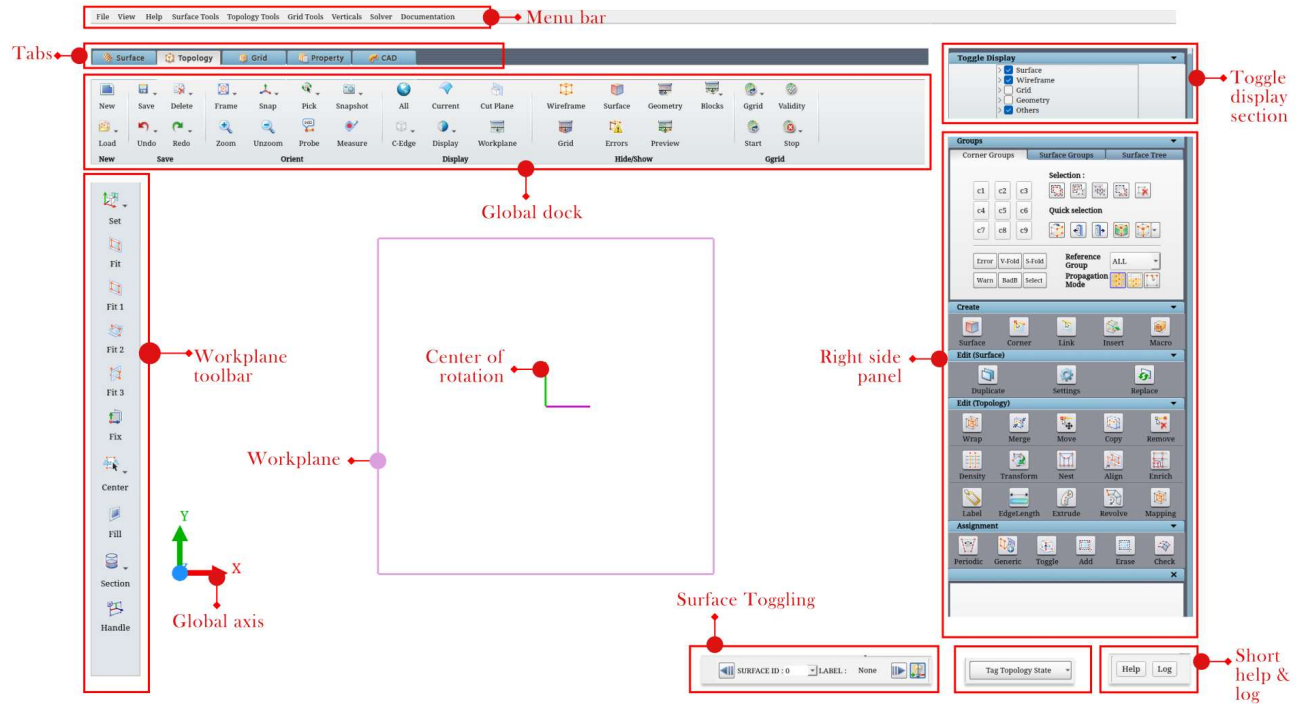


Figure 2.1: GridPro GUI

The empty space in GUI is the **drawing area**, which displays all the surfaces, topology and grids, depending upon the tab you are working on. You perform actions using the tools available in the GUI. At the top of the GUI, a **menu bar** is located that has nine menu's i.e. **File, View, Help, Surface Tools, Topology Tools, Grid Tools, Verticals, Solver and Documentation**. Then below the menu bar, the **global dock** is located. The global dock gives access to all the tools that are common to all the tabs. Frequently used tools are also added to this dock for convenience. In addition to this, the **toggle display** section is located on the right-hand side of the global dock. All the tools related to each tab are located on the right-side panel of each tab. All tools are categorized based on their use. Below the tools section, there is a 'pop-ups' section, where the dialog box of each tool opens. On the left side, the **workplane toolbar** is located. The bottom bar shows status messages, details of the grid, options to toggle through surfaces and short help. The messages related to the operations you perform in the GUI and surface information can be seen in the **status bar**.

Each tab has its own importance that helps in guiding the gridding process. In the gridding process, you first deal with surfaces that lead us to our first tab i.e. '**Surface**' tab.

The **Surface** tab contains operations, which allow you to edit and repair surfaces for use in GridPro. Tools that allow editing and repairing the loaded surfaces are also located on the right-side panel. Along with these, surface grouping tools are also located under the Groups section. All these tools will be discussed later.

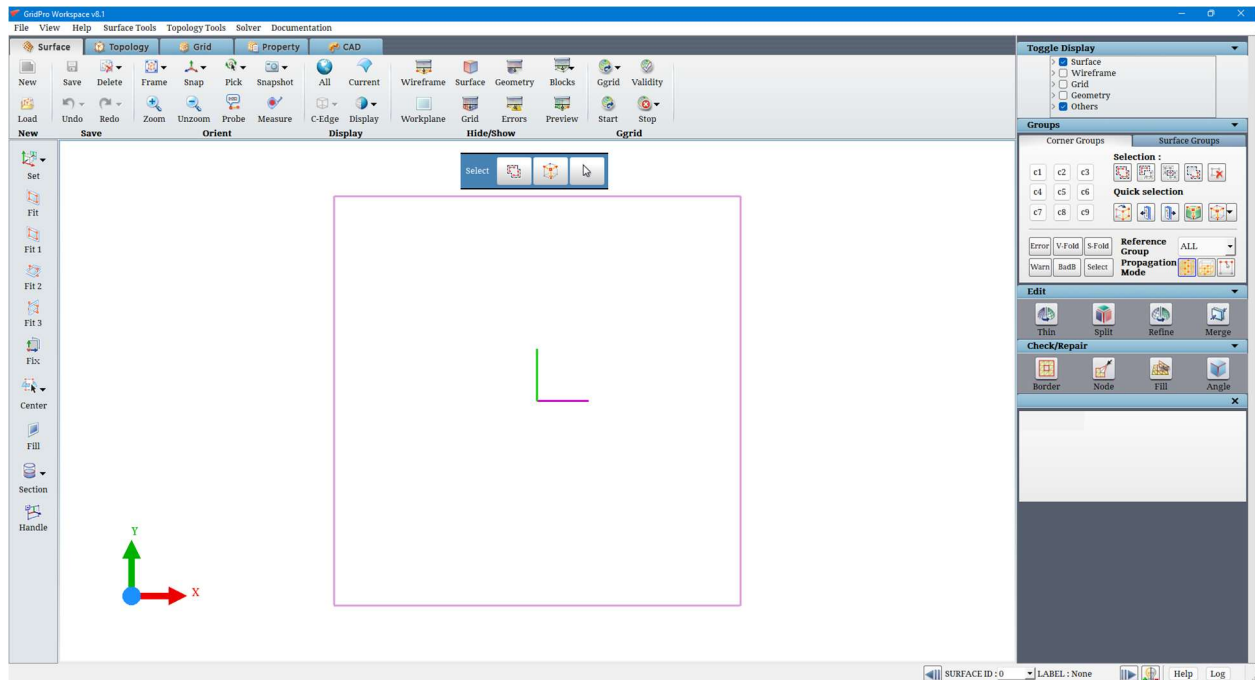


Figure 2.2: Surface tab

You do not want to view all the entities in the GUI every time. So all the entities you are working on in GUI can be switched on or off for viewing any time from any tab from the ***Toggle Display section*** located in the global dock.

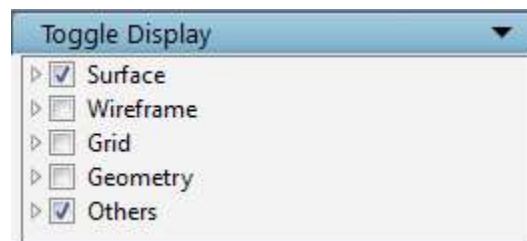


Figure 2.3: Toggle Display section

You can get an idea about each tool from the ***help button*** located in the bottom bar. Once clicked, a window opens, as shown in Figure 2.4. Once it is open, hover the cursor on the button for which you want to view the help.

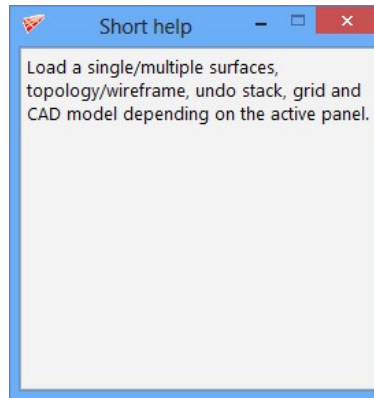


Figure 2.4: Short help window

After working on surfaces, you will create a topology for them, which can be done in the 'Topology' tab. In addition to corner grouping and surface grouping tools, the topology tab provides you with **Create**, **Edit (surfaces)**, **Edit (Topology)** and **Assignment** tools that help you build wireframe about surfaces.

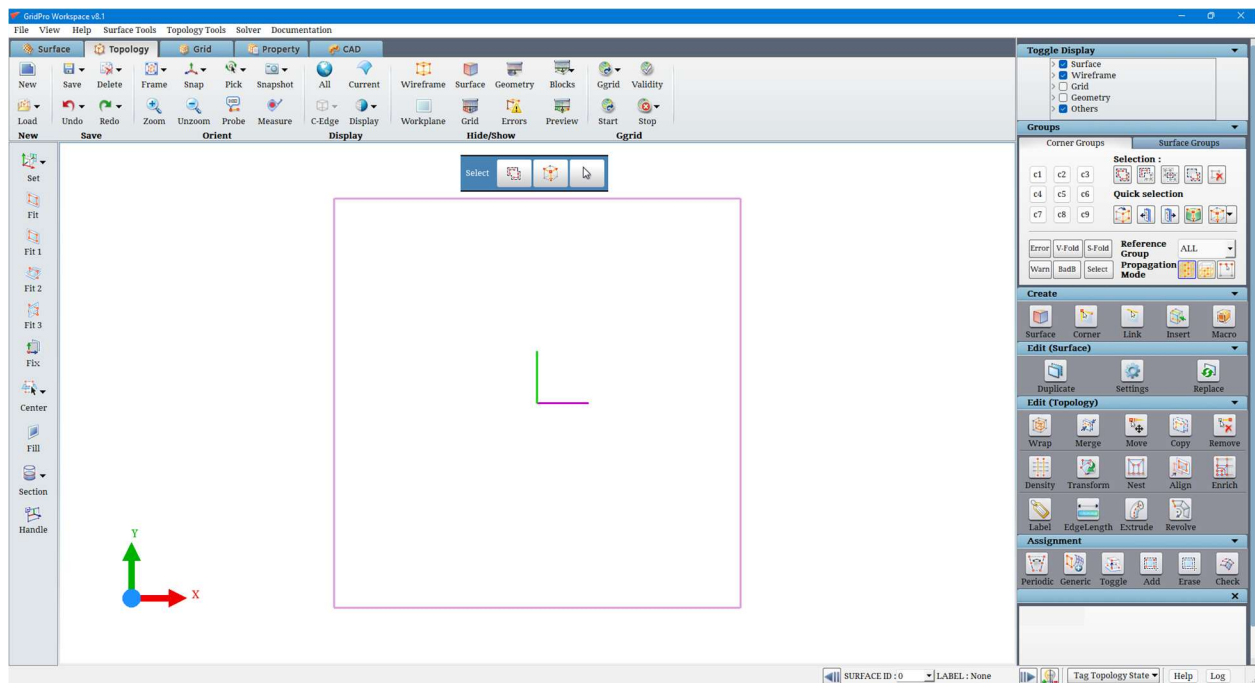


Figure 2.5: Topology tab

Once the topology building is complete, the **grid generator** is started, which can be done using the icons located in the global dock. Next is grid visualization, which is done in the 'Grid' tab. The Grid tab provides you with the tool to view the grid, Edit the grid and check the Quality of the grid.

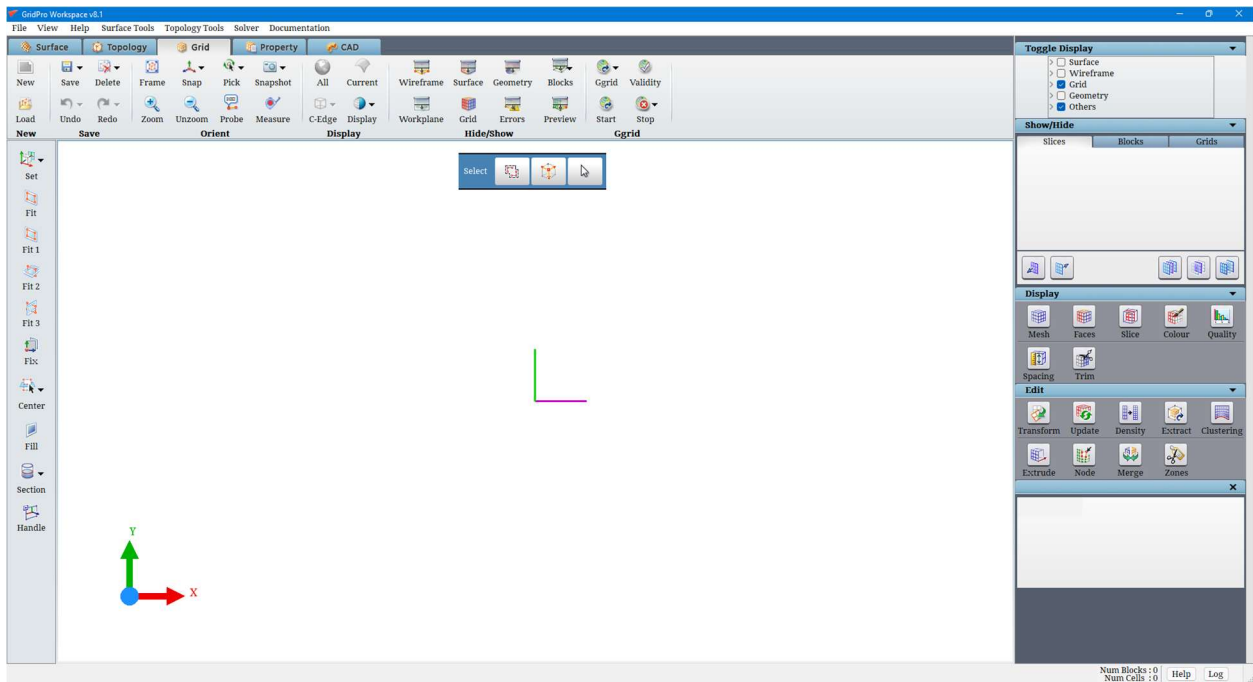


Figure 2.6: Grid tab

The fourth tab is the '**Property**' tab, which allows you to assign properties to the grid. The tools in the right-side panel allow you to assign properties to grids. All these tools will be discussed separately in the following sections.

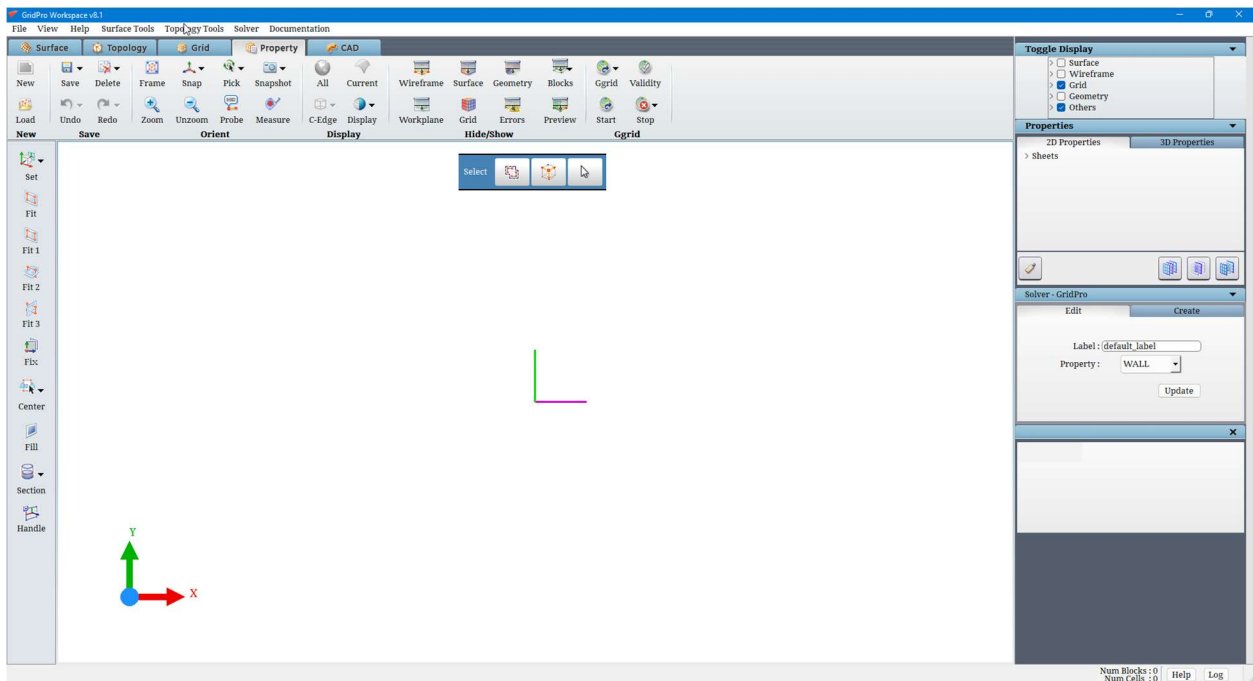


Figure 2.7: Property tab

The fifth tab is the '**CAD**' tab, which allows you to view and repair the CAD geometries and generate surface meshes. You can also save the repaired CAD files in other geometry formats.

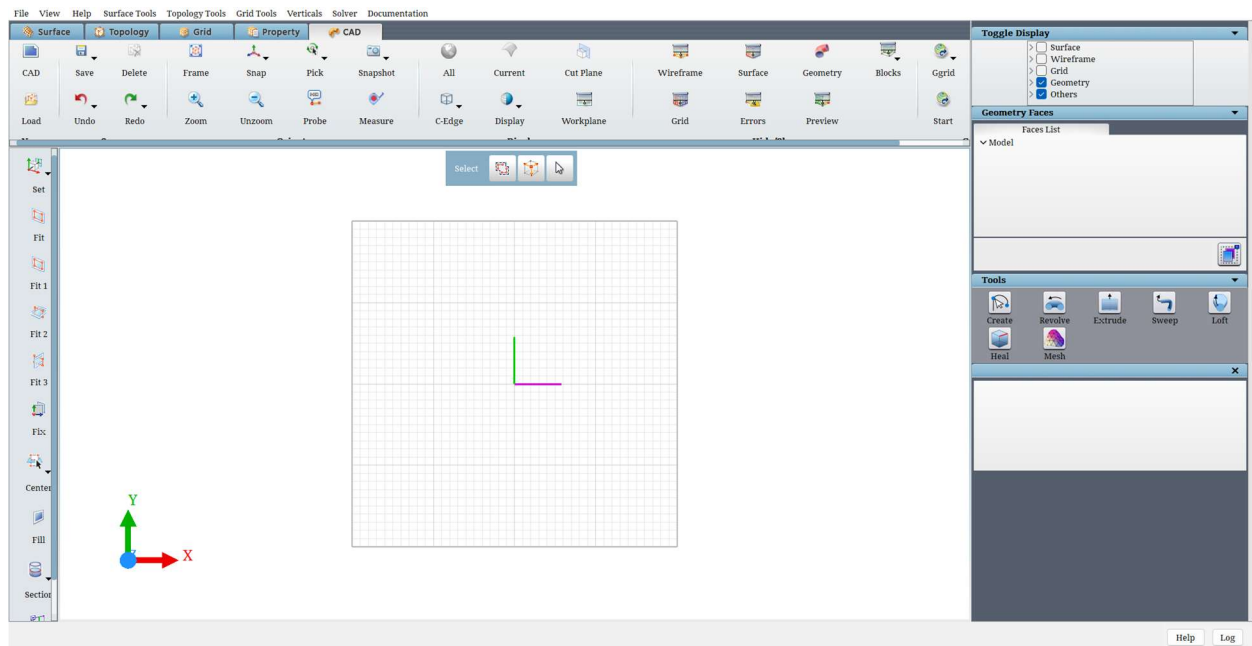


Figure 2.8: CAD tab

## 2.2 Top Menu Bar

The top menu bar of the GUI consists of nine menus such as **File**, **View**, **Help**, **Surface Tools**, **Topology Tools**, **Grid Tools**, **Verticals**, **Solver**. All the menus are common to all the tabs.

### 2.2.1 File menu

File menu has seven options, as shown in Figure 2.9.

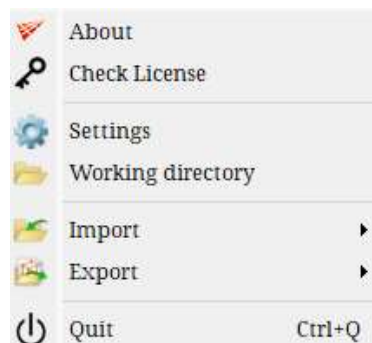


Figure 2.9: File Menu

**About** option displays the software information as shown in Figure 2.10.



Figure 2.10: About option displaying software information

**Check License** option displays the license information. If a valid license file exists in the \$GRIDPRO/lc\_mngr directory, a message pops out, as shown in Figure 2.11.

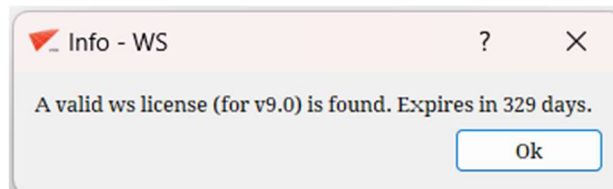


Figure 2.11: License expiry status

But if a valid license is not found, a message as shown in Figure 2.12 will be displayed. If a valid license is not available, contact GridPro support team at [license@gridpro.com](mailto:license@gridpro.com). The same message is also displayed if no valid license is found while opening the GUI.



Figure 2.12: License status

From the **Settings** option, you can change the predefined keyboard shortcuts, font size, background colour, and autosave frequency. In GUI, often-used functions are defined with keyboard shortcuts for easy usage that can be modified. Figure 2.13 shows the default keyboard shortcut keys. These shortcuts can be modified using the dropdown menu associated with each shortcut. The modifications will work across the sessions except for the background colour. To retrieve the default settings click the **Default** button in the pop-up window, as shown in Figure 2.13. There are also shortcut keys that can only be used and are not allowed for modifications. Hence it is shown separately as a different section.

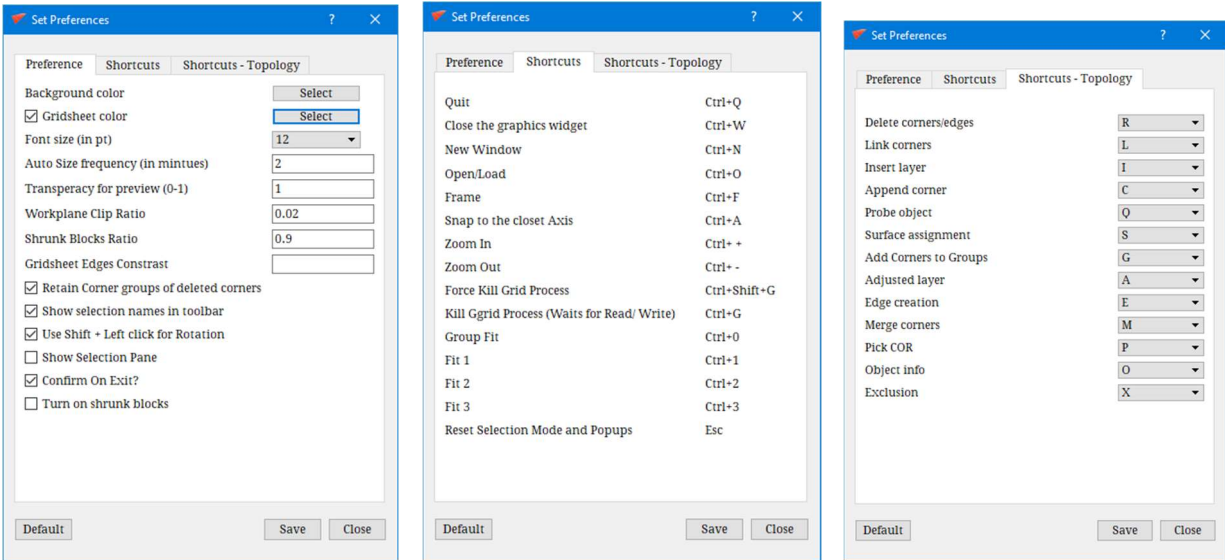


Figure 2.13: Default Keyboard Shortcuts

Font size can be changed by defining the desired value in the field next to the font size. It will change the size of the font displayed in the UI.

Background colour can be changed by clicking on the push button, 'Select', which opens a dialog box, as shown in Figure 2.14. The RGB value of the desired colour has to be given in the respective fields. In addition to that, a few standard colours can be chosen from the **Sample colours** dropdown list, as shown in Figure 2.14.

You can change the background colour to white by changing the RGB value of each colour to 100 either using the slider or else entering the values.

You can also set the transparency value for the preview button on the global dock. This helps in viewing the grid preview along with the underlying surfaces and topology. For more information on Grid Preview, please refer to section 5.1.6.

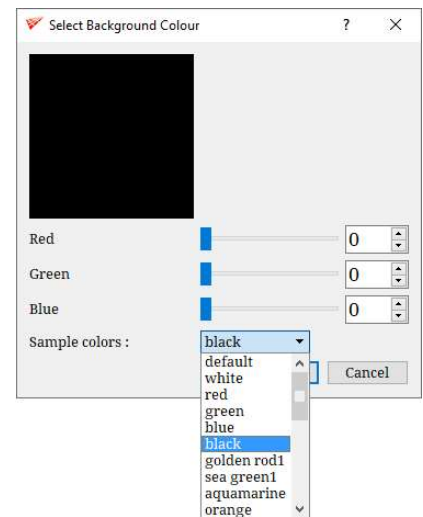


Figure 2.14: Background colour dialog box

The next option is the workplane clip ratio; it ranges from 0-1, which determines the distance of the surface that should be displayed on the clipped side of the workplane. For e.g., if the ratio is 0, no part of the surface will be displayed on the clipped side, but if it is set to 1, almost the entire surface will be displayed as if the workplane is not clipped as shown in Figure 2.15.



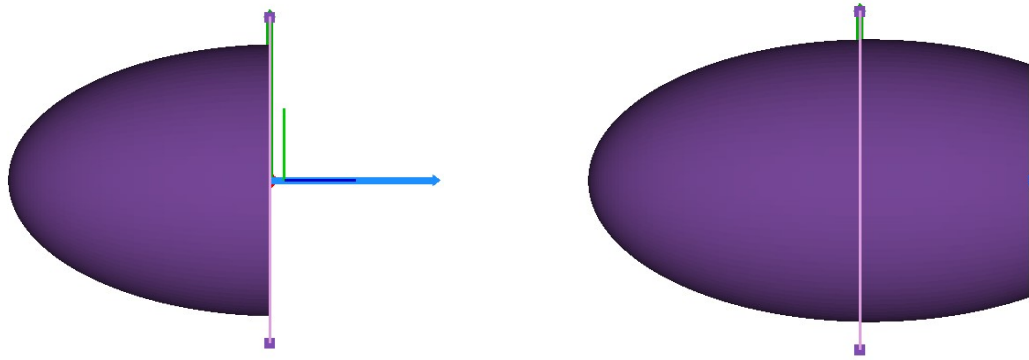


Figure 2.15: Workplane with Clip ratio 0 and 1

GUI saves the topology automatically with `_auto.fra` name based upon Auto-Save Frequency. By default, Auto-Save Frequency is 2; hence GUI will save the topology every 2 minutes with name `_auto.fra`. This ensures that even if the application crashes unexpectedly, the topology data is not lost. You can modify the frequency by defining the desired value in the field next to **Auto Save Frequency** in the **Settings** window of the **File Menu** bar. In addition to `_auto.fra` file, GUI also saves a file named `_auto.undo` at every auto-save frequency interval, which contains all the operations executed on the topology in the order. This file is saved so that the undo/redo operations can be done in the saved topology file in the same order as it is done while creating it.

Whenever the GUI is started, it prompts you to define the **Current Working Directory**. A current working directory is a directory where all the surface files, topology files, and grid files will be stored by default until you change the location. It always opens the Current Working Directory to load the files. All the file dialog boxes use this current working directory to load/save files. This directory is also used by the other functions in the GUI; for e.g. when Ggrid is started, the output files are written to the current working directory. While starting a new session, it is advisable to redefine your working directory as all autosave files (`_auto.fra`, `_az.fra`, `_undo.stack`) and the grid files will be saved to that folder which can be accessed easily when needed later. Hence, it is very important to reset the working directory. You can modify the Current Working Directory from the **Working Directory** option under the **File Menu**, as shown in Figure 2.9. A file selection dialog box is popped up to select the new working directory, as shown in Figure 2.16. The dialog box allows you to create a new folder/directory and select them or select the existing directory.



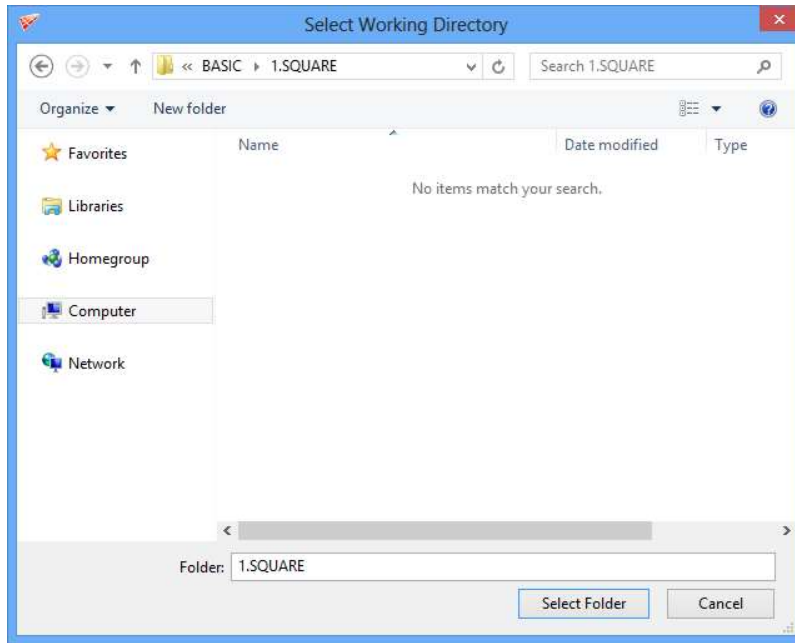


Figure 2.16: File dialog box for the working directory

When the topology or grid is loaded from a different directory other than the current working directory, the working directory will be updated to the recently used directory for loading the topology, which will be saved across the sessions. This is not applicable if the surface is loaded from a different directory. The selected directory should have read/write permissions. If a working directory without read/write permissions is selected, it re-prompts the user to select a different working directory as shown in Figure 2.17 until a valid directory is chosen.

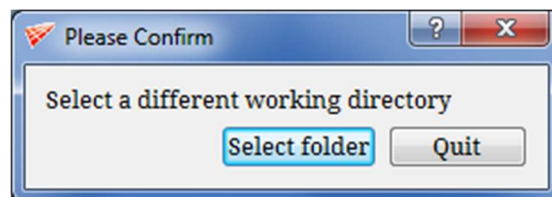


Figure 2.17: Working directory read/write permissions denied

When you quit the GUI, by default, it opens a confirmation dialog box to confirm the operation. If the '**confirm on exit**' checkbox is unchecked, it will not pop up the confirmation dialog box next time. Make sure the settings are saved after you uncheck the box.

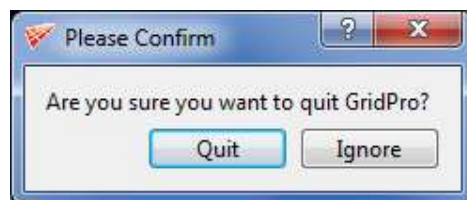


Figure 2.18: Confirmation dialog box

**Import** option allows you to import various geometry and grid formats into the GridPro UI. Once the appropriate file is selected, it converts the respective file to GridPro format and loads into the UI. Figure 2.19 shows various import geometry and grid formats available in the UI.

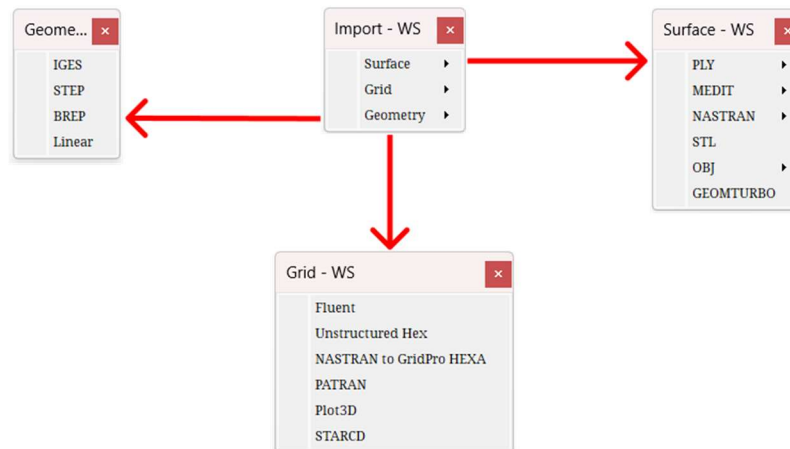


Figure 2.19: Import formats of geometry and grid

Upon selecting the formats, it opens a file dialog box in the working directory from which you can select the file name to be imported. The file dialog box would be filtered automatically for the selected file format.

**The export** option allows you to export various surface, grid and geometry formats from GridPro. Once the appropriate format is selected, it converts the file from GridPro format to the selected output format and saves it in the working directory. Figure 2.20 shows various export surface and grid formats available in the UI.

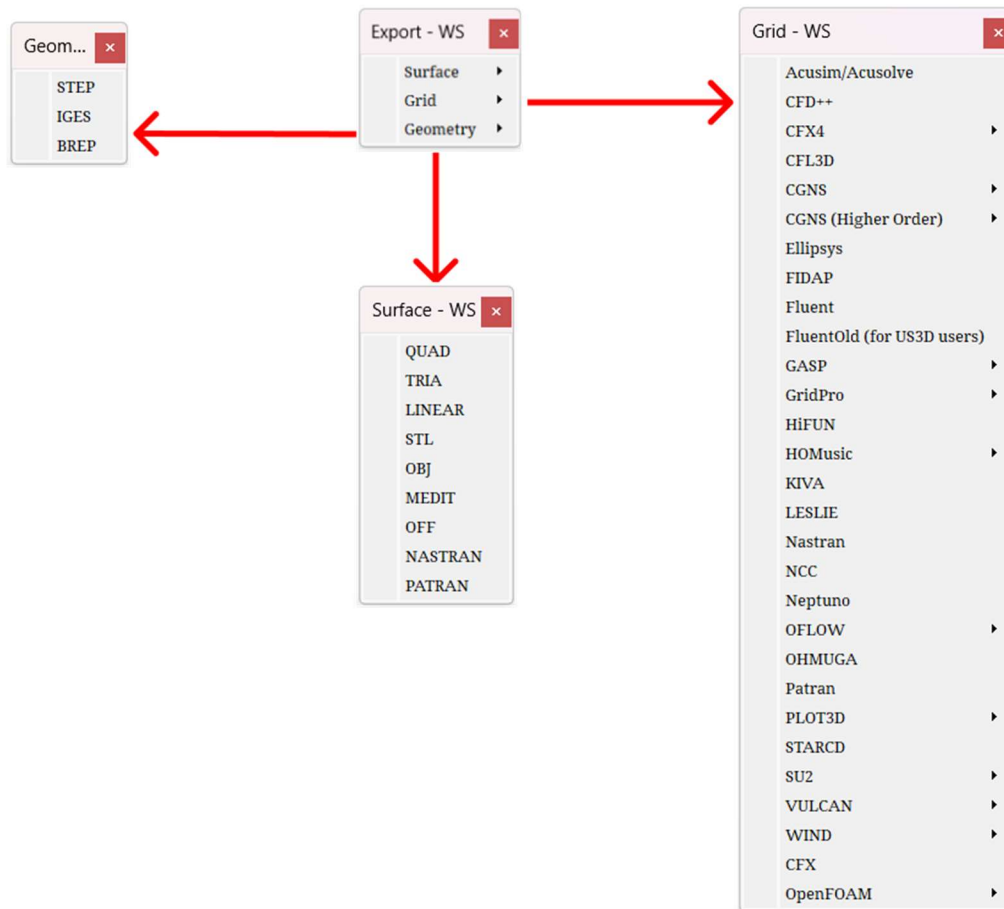


Figure 2.20: Export formats of surface and grid

\*Refer to Appendix A to know how to use your own export format.

You can also export the grid from superblock format to the respective solver format. For E.g., GridPro superblock format to PLOT3D, This will export the elementary block grid to superblock grid and then converts it to PLOT3D. Only a few solvers like CFX4, CGNS, GASP, OFLOW, PLOT3D, VULCAN, and WIND allows such conversion and the options for the same are available only for these solvers.

The **Quit** option defined under the File menu can quit the application. A confirmation dialog box, as shown in Figure 2.18 is popped up to ensure that it was not closed accidentally. The user has to ensure that all the modified data is saved before the application is terminated.

## 2.2.2 View menu

The **View** menu contains options to toggle the toolbars and the view screen mode, as shown in Figure 2.21.

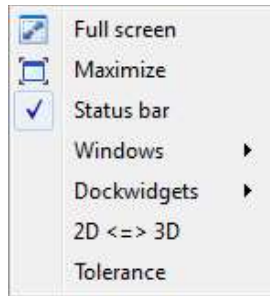


Figure 2.21: View Menu bar

To view the entire screen in Full-screen mode, select the **Full-screen** option under the view menu as shown in Figure 2.21. The full-screen mode is shown in Figure 2.22. This hides the desktop panel, system tray, and all other visible applications, but the top dock, left-side toolbar and right-side panels are not hidden. Click on the 'maximize' option to bring it back to the default size, which retrieves the default screen size.

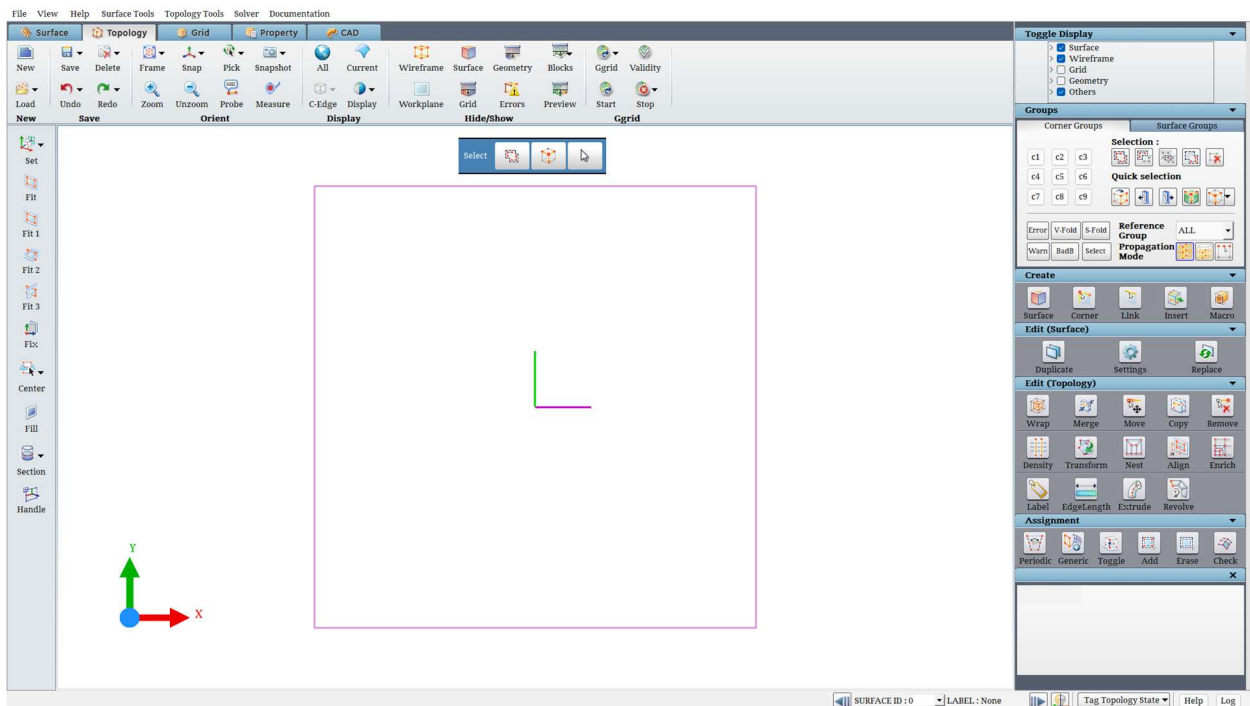


Figure 2.22: GUI in full-screen mode

The bottom Status bar that displays the information related to each icon and the status of each operation can be toggled off for display by unchecking the **Status bar** under View Menu.

Similar to the Status bar, the right-side panel, left side toolbar, and the top dock can be switched off for display using the options defined under **Dockwidgets** in the View menu, as shown in Figure 2.23. You can switch off all options at a time, as shown in Figure 2.24, or

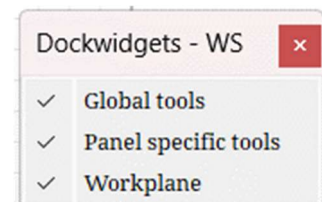


Figure 2.23: Dock widgets drop down list

you can manually switch off one option at a time, depending on your necessity. This is useful when you want to increase the size of the graphics window. Even though the size of the graphics might have been changed, the objects in the graphics window may appear disoriented as it needs to be updated by a simple left-click in the drawing window, which would redraw all the graphics objects and re-evaluate the view volume appropriately.

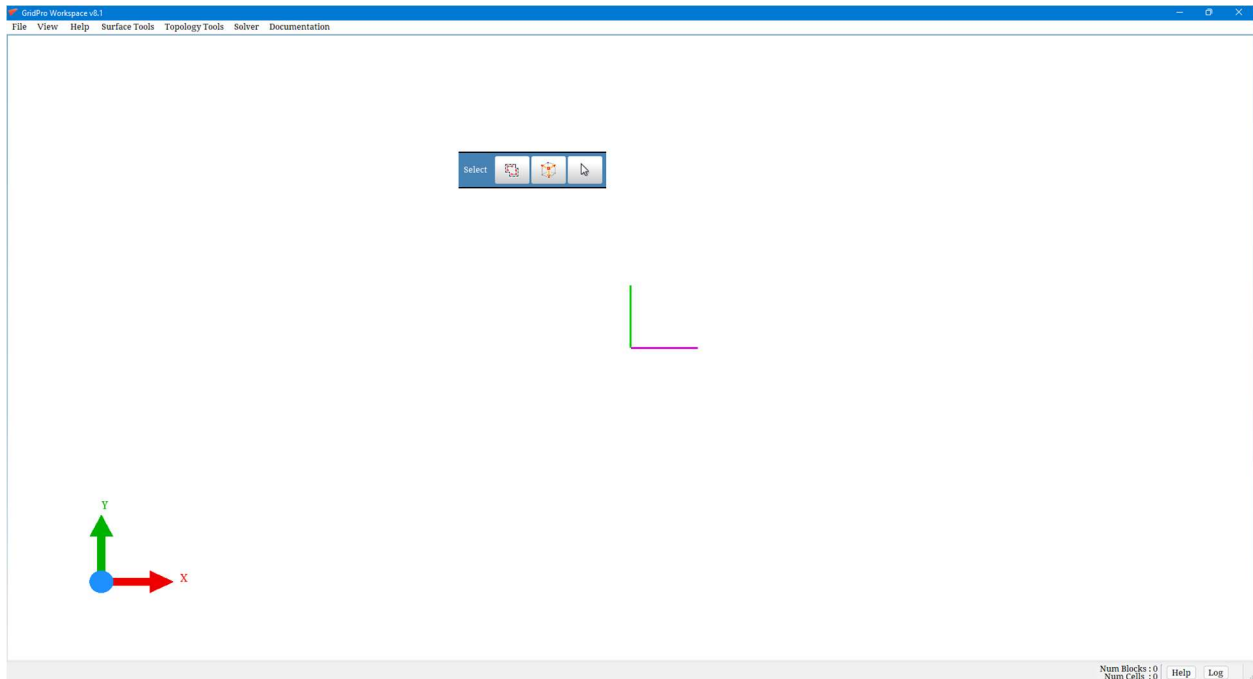


Figure 2.24: UI after switching off all the dock widgets

As you know, the GUI can support both 2D and 3D geometries, which can be chosen while opening a 'New' empty workspace. Suppose you have chosen a wrong dimension or want to load a 2D surface in a 3D workspace or vice versa, you can use the **2D↔3D** option located at the bottom of the View menu. The command automatically identifies the dimension of the existing workspace and change it to the other dimension. A confirmation dialog box will appear before changing the dimension, as shown in Figure 2.25.



Figure 2.25: 2d to 3d confirmation dialog box

Tolerance helps in viewing the edges to be merged in the CAD geometry. Once the geometry is loaded, modify the tolerance to view the edges which are merged. All the possible red coloured lines will be merged together and displayed in green colour based on the given tolerance value.

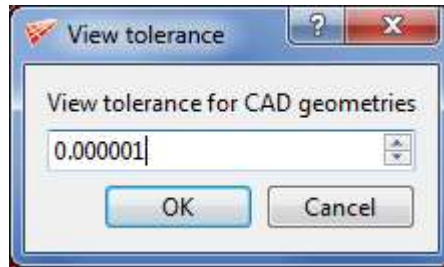


Figure 2.26: Tolerance dialog box

### 2.2.3 Help menu

To view this document, while working on UI, you can select the desired format under the Help menu. Though the short help is very useful to view the help content, sometimes the user may need to look at the entire Documentation to get a broad picture of the classification of various tools in the GUI. Thus, the offline help is packaged along with the software and can be accessed by you readily. The help content can be either viewed in HTML format using a web browser or as a PDF using the PDF reader. Figure 2.27 shows the options in the dropdown of the Help menu in the top menu bar.

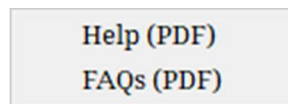


Figure 2.27: Help Menu

When the option *Help (PDF)* format is clicked, the default pdf reader is launched, and the documentation file (in pdf format) is opened using this application. Similarly, when the *Help (HTML)* is clicked, the default web browser is launched to open the HTML documentation files. *FAQs (PDF)* is clicked, FAQ document in the default pdf reader is opened.

### 2.2.4 Surface Tools

The **Surface Tools** menu has tools used for creating surfaces using a wireframe. Wireframe defines the shape of the surface. Figure 2.28 shows the dropdown list of the Surface Tools menu. All these tools except for convert (explained below) and Split disjoint surfaces (explained in *section 3.4.2*) are explained in *section 4.11*.

- **Convert**

Convert is a file format conversion tool similar to Import options where this is specific to *GridPro* formats. Figure 2.29 shows the available conversions.

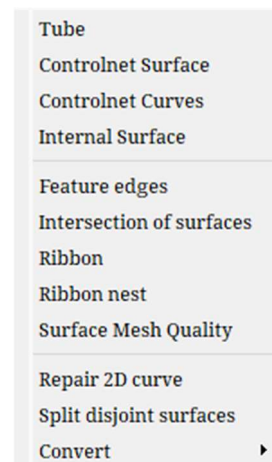


Figure 2.28: Surface Tools dropdown list

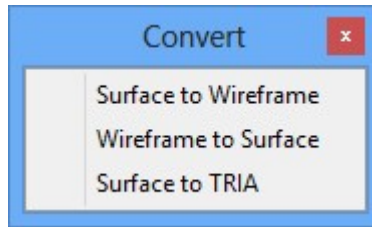


Figure 2.29: Convert dropdown list

Surface to Wireframe: Converts any surface (Supported formats\*: quad, tria, lin, dat) or surface groups which is loaded into the UI and saves it as a fra file in the working directory. Once the option is selected, it opens a dialog box where a surface group\* containing the surfaces to be converted should be selected in the respective field, as shown in Figure 2.30.



Figure 2.30: Surface to Wireframe conversion

Wireframe to Surface: Converts a given corner group as a quad surface and loads it into the UI. Once it is selected, Wireframe to Surface dialog box opens, as shown in Figure 2.31, where a corner group and an output file name has to be provided.



Figure 2.31: Wireframe to Surface conversion

Surface to TRIA: Converts a given quad or tube surface to a TRIA surface. Once selected, open a file selection dialog box where the output file name should be given to save the tria surface.

\* Refer section 3.1 for more information on surface formats and section 4.4 on groups.

### 2.2.5 Topology Tools

the advanced topology related tools are moved from the Edit (Topology) section to the **Topology tools** menu. Figure 2.32 shows the dropdown list of the Topology tools menu. All these tools are explained in different sections of Chapter 4.



Figure 2.32: Topology tools menu

### 2.2.6 Solver menu

The **Solver** menu in the top menu bar defines the solver format globally. All the surface and grid properties will be assigned according to the selected solver format. The various available solver formats can be seen in Figure 2.33.

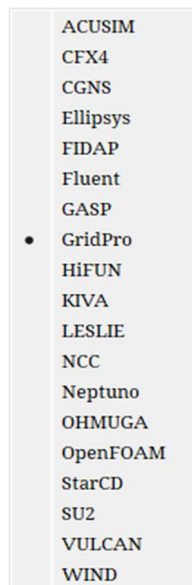


Figure 2.33: List of Solver formats

### 2.2.7 Documentation

Documentation menu in the top menu bar contains all the documents related to the GUI. It opens a dropdown list with two options, tutorials and manuals. Each one further opens another dropdown with the file names, as shown in Figure 2.34. Clicking on any file name opens the respective document in PDF format separately.



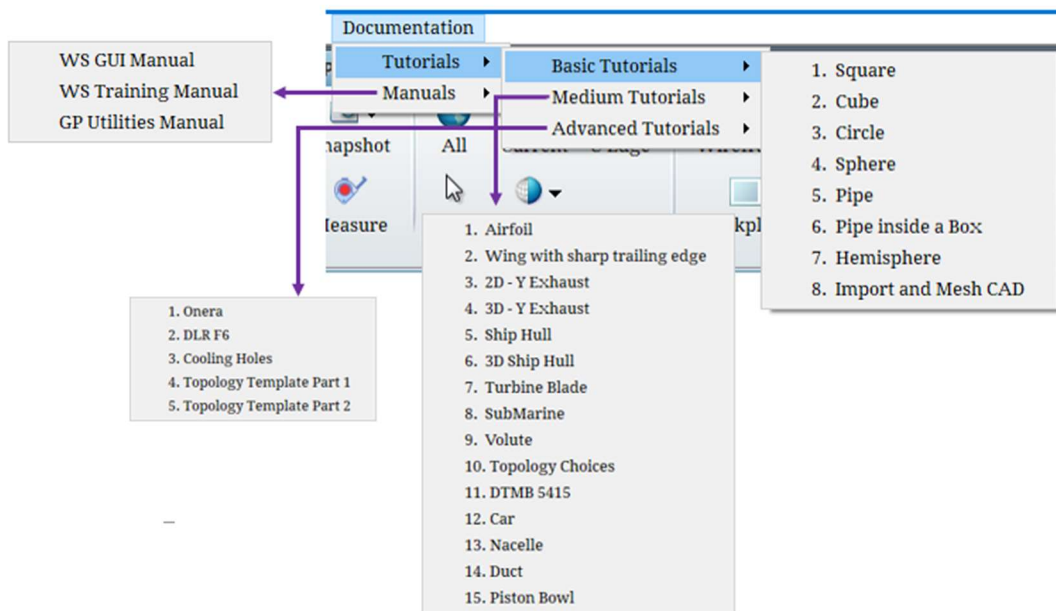



Figure 2.34: Documents Menu

## 2.3 Global Dock

Once you open the UI, all the buttons are disabled except for the **New** and **Load** button. All the icons in the global dock are discussed below.

### 2.3.1 New

When the UI is started, the Topology tab is opened as the default tab. Normally, you start with an empty workspace which can be done by clicking on the **New** button  which is the first option located in the global dock. If clicked, a window pops out asking for the dimension of the workspace. You can choose either the 2D workspace or the 3D workspace. It will open an empty workspace with the specified dimension. *The new* button is only active in the topology tab as topology can only be defined in the Topology tab.

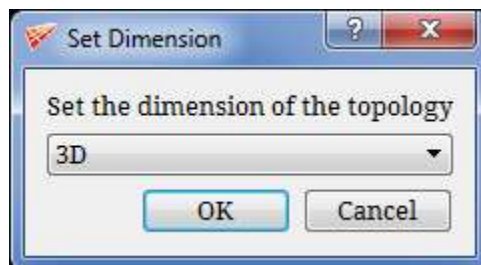




Figure 2.35: Set dimension dialog box

### 2.3.2 Load

The Load button helps to load a topology, surface, grid or CAD file which you have created or you were working on it previously. To load surfaces into UI, use the **Load** button  in the **Surface** tab located below the 'New' button. You can load the surfaces from the **Topology** tab even. **Load → surfaces**  option allows you to load single or multiple surface files from the working directory in one of the GridPro formats. The supported surface formats are quad, tria, tube, lin, dat and tmp. When you load many surfaces into UI, each surface is assigned an id (a numeric value starting from 0) based on the order it is loaded into the UI.

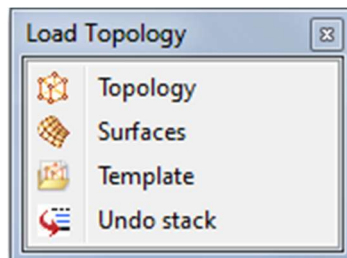




Figure 2.36: Load dropdown list in topology tab

Topologies that are saved in \*.fra format can be loaded into the UI from **Load → Topology**  option in the **Topology** tab. **Load → Topology** allows you to retrieve topologies in \*.fra format. All user-defined topology files must have the extension '\*.fra'. Figure 2.36 shows the dropdown list of the Load button in the global dock of the topology tab.

In addition to predefined templates, a user-defined topology can be loaded into the current graphics window. You can load either 2D or 3D user-defined topology respective to the modes (2D or 3D) using **Load → Template**  option. Load Topology Template dialog box will pop out as shown in Figure 2.37 if **Load → Template** is selected. Select the file you would like to input by pressing the *Select file* push button. You can load a particular corner group from the chosen file if you want. Similarly, you can also load a particular surface group from the chosen file. You can change the position of the loaded topology by using the transformation options. The Load Template options are available solely to maximize your efficiency in using GridPro. It acts as a template for the geometry of similar shapes. It's mainly helpful while doing design optimization, where you modify the surfaces every now and then to get an optimum shape; in such cases, building the wireframe for every modification becomes a painful and time-consuming process. It also helps in loading the previously used wireframe and modifies it according to the surface modification.

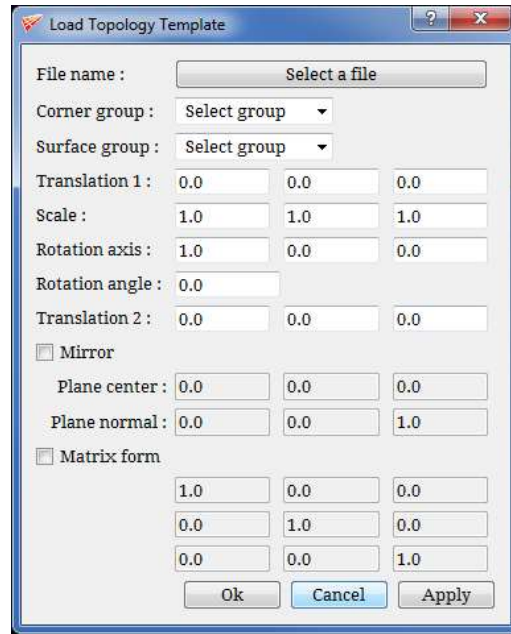




Figure 2.37: Load Topology Template dialog box

While working on any topology in GUI, your topology is saved by default every two minutes with `_auto.fra` name. You can change the default frequency of saving the topology from the **Settings** option in the **File** menu bar. At the same time, it also saves a file automatically with `_auto.undo`. The topology file is in the state before which you did any undo or redo operations. This file can retain a session even after closing a session or if the current session has crashed. This file is overwritten every time you open the UI in the same working directory. You can access `_auto.undo` stack file in **Topology** tab from **Load→Undo stack**  option.

Once a grid is generated, or you want to access the generated grid, you can load that in the **Grid** tab or **Property** tab from the **Load→grid** option. You can load multiple grids in the GUI in one go by dragging and selecting the grid files to be opened, and then clicking on **Open**.

If you want to repair/fix a geometry using *GridPro* tools, you can load the geometry files from the CAD tab in IGES and STEP formats and use the *GridPro* geometry fixing tools.

### 2.3.3 Save

When you are working on any topology in GUI, at any point of time you can save your work using Save button . Using **Save→Topology→All** option under **Save** menu in the **Topology** tab you can save your topology. When you save a topology, you save information related to surface files, their location and their association to wireframe in addition to the wireframe. It allows you to save your specified file name with extension `*.fra`. You can also save a part of your topology which is kept in a particular group [for more information on groups, refer to *section 4.4*];

to perform that you can use **Group** under *Save Topology* in the **Topology** tab which prompts you with a dialog box asking a particular group and the file name with which you want to save. Save a corner group saves all the surfaces in addition to the specified corner group.

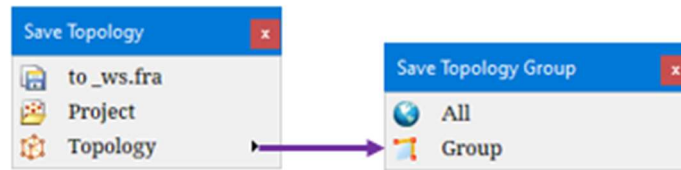




Figure 2.38: Save dropdown list in the topology tab

Another option, **Save to → \_ws.fra** option in the **Topology** tab, allows you to save a topology under the file name '\_ws.fra' until you overwrite the file. This is the file that is used by the grid generation process, and each time you start the grid generation, GUI automatically saves the file to \_ws.fra name. The '\_ws.fra' file has four backups, '\_ws.fra.~1~', '\_ws.fra.~2~', '\_ws.fra.~3~', and '\_ws.fra.~4~'. When you save a new file to '\_ws.fra', the last file saved under '\_ws.fra' becomes a backup file, and the oldest backup file is erased. In essence, GUI saves five files under the '\_ws. fra' name, then erases them as new '\_ws. fra' files are saved.

It is possible to save the existing topology as a project to another directory by using **Save → Project**  in Topology tab, which allows you to save the topology to another directory, i.e. the wireframe and the associated surface files. It helps extract files related to your topology and copy them to another directory.

GridPro saves grids to the default file name as 'blk.tmp'. To assign a user-defined name to a grid, use **Save → Grid → Elementary grid**  in the **Grid** tab. By altering the file name, you will be able to retrieve your file whenever you wish because all other grids generated in the working directory will be saved under 'blk.tmp'.

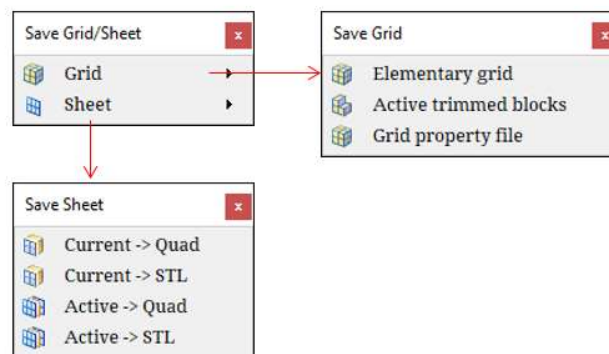


Figure 2.39: Save dropdown list in the grid tab

GUI also enables you to save certain blocks and sheets just as it allows you to save grids. To save a set of grid blocks, select the grid as *current*, activate the set of blocks, then select the **Save → Grid → Active trimmed blocks** option in the **Grid** tab. You can also save the property file in a different name by selecting **Save → Grid → Grid property file**. Similarly, to save the *current* grid sheet, select the grid as *current*, select the sheet you wish to save as *current* sheet, and select **Save → Sheet → Current → Quad/STL** in the **Grid** tab. This grid sheet would be saved as quad format, which can be loaded into the GUI using the **Load → Surfaces** option. You can also save all the visible grid sheets of the *current* grid by selecting the **Save → Sheet → Active → Quad/STL** option. All the options related to the saving grid and saving sheet are also available in the Property tab.



Figure 2.40: Save drop down list

In the CAD tab, at any point in time, a CAD geometry can be saved into the native **GPCAD** format. **Save → GPCAD** option in CAD tab can be used for saving the geometry in .gpcad format.

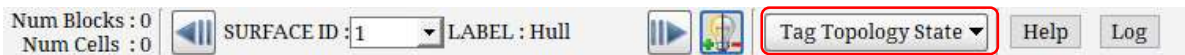


Figure 2.41: Tag Topology State

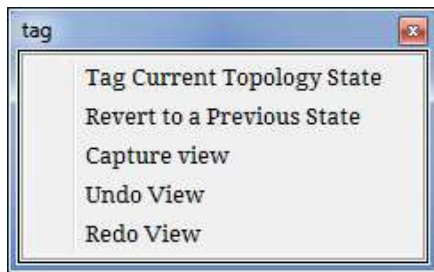


Figure 2.42: Tag topology state drop down list

Apart from these save options, you are provided with one more save option called **Tag Topology State**, which can be located at the bottom on the right in the status bar, as shown in Figure 2.41. It can be used to tag a state, in other words, tag before/after an operation in a file.

This will not be saved as a separate file in the working directory but saved inside the .fra file. These tags can be reverted to previous operations, which is similar to undo while working, where undo cannot be done after closing the session unless you load an undo stack file. Still, the tag topology state can be reverted to a previous tag, and the topology can be modified. Once it is reverted to a previous state, all the tags after that will be removed. Clicking on the button will show the dropdown, as shown in Figure 2.42. To tag a state, select **Tag Current Topology State**, where you will be prompted with a pop-up where you need to input a name for tagging the state. To revert to a previous state, select **Revert to a Previous State**, and a window will pop up as shown in Figure 2.44, which contains all the states that were tagged in the current \*.fra file, which is loaded. Selecting a state and clicking on OK will revert the existing topology to the selected state, and the tags after that will be removed from the file. From v8.0 onwards, a few of the major operations such as



Figure 2.43: Tag current topology state dialog box

load topology, wrap, merging etc., are all automatically tagged, which can be used to revert whenever needed.

Next 3 sets of options are used to undo/redo the screen views. It is primarily helpful when you want to capture a series of topology/grid images with the same screen view. You often tend to change the screen view by mistake or change itself while loading a new grid/topology. In such cases, it is better to position the screen first and click on capture view to save the screen view to the undo stack. Once it is saved, it will be there till you close the UI instance. A maximum of 32 no. of views can be stored at any time. When you want to go back to the previous view, click on Undo view option under the dropdown, which will go back to the last captured view; however you have changed the screen afterwards. Suppose if you have gone back multiple views, redo view will help in bringing you back to the recently saved view. This is also helpful while working on a certain geometry detail in a complex case where you can save time by avoiding multiple screen rotations/zooming to get to that desired view.

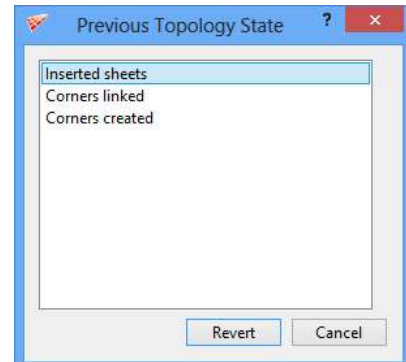



Figure 2.44: Revert to previous state dialog box

## 2.3.4 Delete

You can delete any surface, topological entity or entity related to the grid using the **Delete** button  in all the tabs located next to the **Save** option in the Global Dock. A **current** surface can be deleted using **Delete → Current surface** in the **Surface** tab or **Delete → Surface → Current** in **Topology** tab. You have the facility to display a few surfaces in GUI and hide a few surfaces. You can delete all the hidden surfaces at a time by using the options **Delete → Hidden surfaces** in the **Surface** tab or using **Delete → Surface → Hidden** in **Topology** tab. Similarly, you can delete all the visible surfaces using **Delete → Surface → Visible**. You can delete all the surfaces by using **Delete → All surfaces** in the **Surface** tab or using **Delete → Surface → All** in **Topology** tab. There is also an option available to different group surfaces. You can access surface groups in different ways, explained in *section 4.4*. You can delete a specific surface group using the option **Delete → Surface group** in the **Surface** tab or **Delete → Surface → Group** in **Topology** tab. You can delete all surfaces either visible or hidden in GUI by using **Delete → All surfaces** option in the **Surface** tab or using **Delete → Surface → Group** in **Topology** tab. You can also delete the unassigned surfaces from the GUI using **Delete → Surface → Unassigned**. All the topological entities (wireframe + surfaces) in the GUI can be deleted using **Delete → Topology → All** option in **Topology** tab. It can be retrieved back into the GUI by using **Undo** option. Similarly, a current corner group can be deleted by selecting **Delete → Topology → Group** option.

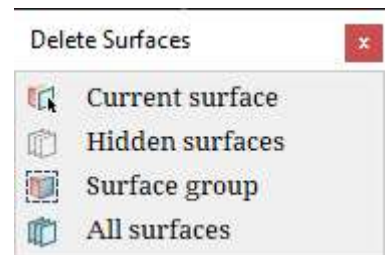


Figure 2.45: Delete drop down list in surface tab



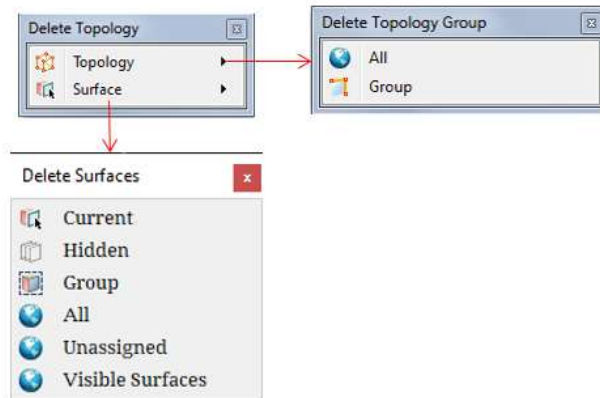


Figure 2.46: Delete dropdown list in the topology tab

As discussed earlier, you can load multiple grids in the GUI. The grid you are manipulating or working on is called the *current grid*. In the **Grid** tab, the **Delete → Grid → Current** operation deletes the current grid and the **Delete → Grid → All** operation deletes all the grids loaded in the **UI**. Similar to the topological entities, corners, and edges, the grid also has sheets as an entity that can be manipulated. All sheets can be deleted using **Delete → Sheet → All** option in the **Grid** tab and the Current sheet can be deleted using **Delete → Sheet → Current** option. You can hide the unwanted sheets in the **Grid** tab to avoid clutter, and hidden sheets can be deleted using **Delete → Sheet → Hidden** in **Grid** tab. All the options related to deleting the grid is also available in the Property tab.

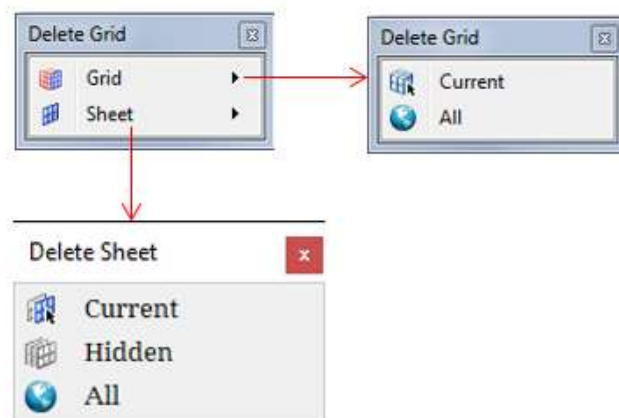

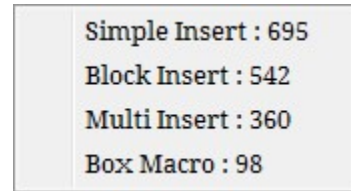


Figure 2.47: Delete dropdown list in Grid tab


### 2.3.5 Undo/Redo

When you input corners and edges in the GUI, you will often make mistakes. GridPro remembers every topological operation you perform and enables you to go back and scroll through every operation using **Undo** button  in the global dock. Therefore, you can erase all topologies you have created at any point in the process.



Simple Insert	: 695
Block Insert	: 542
Multi Insert	: 360
Box Macro	: 98

Figure 2.48: Undo/Redo drop down list

You can also scroll forward using **the Redo** button ; when you realize that you do not change anything. Both buttons are disabled in all tabs except the topology tab and CAD tab because you cannot undo any other surface, grid, properties related modifications. However, adding and deleting surfaces can be undone using the undo button in the Topology tab.


You can only undo/redo up to 32 times. Also, from v6.7, you can directly go back to the desired operation instead of undoing one by one. i.e., all the major operations are saved with their names under the undo/redo dropdown list. So you can directly undo from the 10<sup>th</sup> step to the 5<sup>th</sup> step with a click of a button by selecting the desired operation from the dropdown. Though you can undo/redo up to 32 times, only 10 operations would be shown in the dropdown at any point in time. All the minor operations which are not displayed under the undo/redo dropdown still exist in the background. It can be undo/redo by simply clicking on the respective button.

### 2.3.6 Frame

When you are manoeuvring any object in the UI, it tends to happen that you moved away really far or you zoomed in too much, or you zoomed out beyond a limit, so to position the object back to the screen display, you can use the **Frame**



Figure 2.49: Frame drop down list in surface tab

button  located in the global dock. **Frame → Current surface** option will change the view to the current surface. The current surface will fill up more than 80% of the screen display. Also, the centre of rotation is set to the centre of the surface. It is useful when your *current* surface is relatively smaller than the whole topology, and you can locate it easily using this option.

Similarly, you can frame to the current group of surfaces using **Frame → Current surface group**. Similarly, the frame can be used for **Frame → Entire topology** or **Corner Group** in **Topology** tab. The dropdown list of frame button in surface and topology tabs are shown in Figure 2.49 & Figure 2.50, respectively.



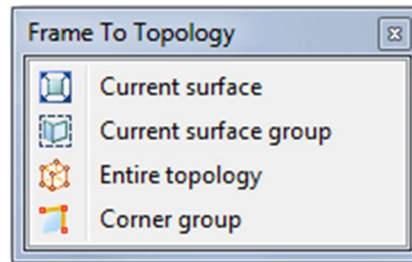


Figure 2.50: Frame drop down list in topology tab

Figure 2.51 shows how the frame option behaves. Figure 2.51 shows the zoom-out view of topology with an active surface and corner groups. Figure 2.52a shows the frame to the Current surface view. Figure 2.52b shows the frame to the Current Surface Group view. Figure 2.52c shows frame to Topology view. Figure 2.52d shows frame to an Active Corner Group view.

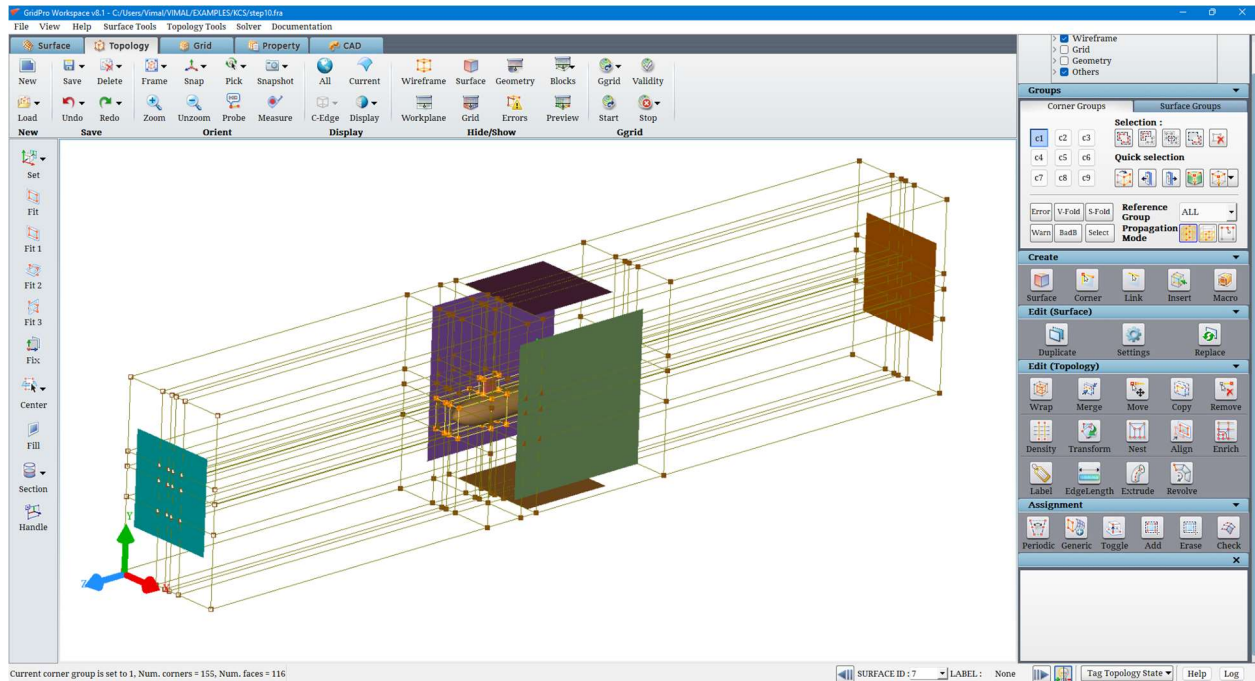


Figure 2.51: Original View Position

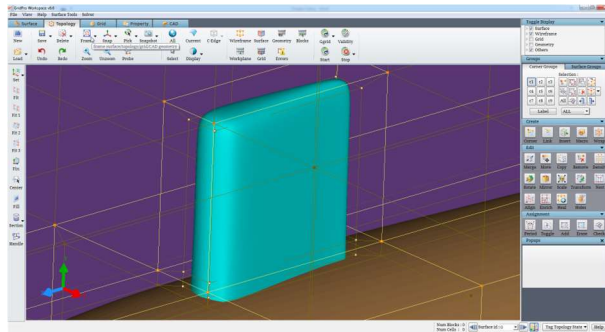


Figure 2.52a: Frame to Current Surface

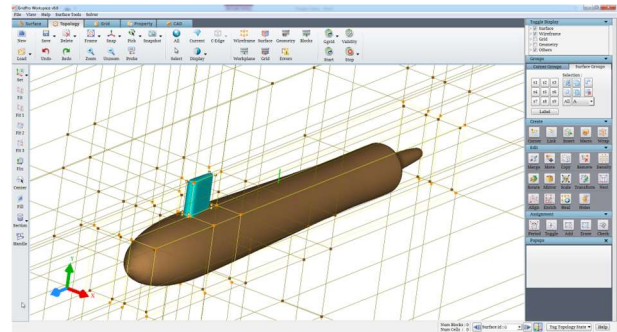


Figure 2.52b: Frame to Current Surface Group

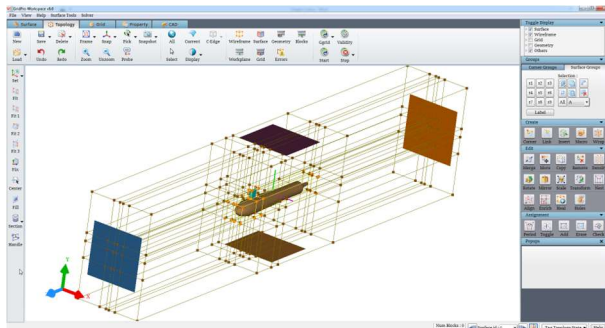


Figure 2.52c: Frame to Topology

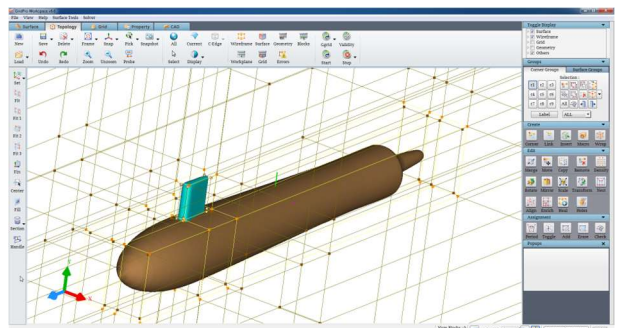


Figure 2.52d: Frame to Active Corner Group

In **Grid**, **Properties** and **CAD** tab, the **Frame** button can be used for framing a grid and CAD geometry, respectively.

## 2.3.7 Snap

The GUI provides the option to align world axes or workplane axes to the screen axes. The screen axes are fixed to the screen; the screen X-axis travels left to right, the screen Y-axis travels bottom to top, and the screen Z-axis travels out of the screen. The world axes are fixed to the model. They can be translated and rotated relative to the screen axes. The **Snap** button



located in the global dock enables us to perform a transformation on the world axes so that they correspond to the screen axes. In the **Snap** option, the **Closest** operation rotates the model such that the world axes go to the closest possible screen axes. For example, **Closest** might align the world Y-axis to the screen Z-axis, the world X-axis to the screen Y-axis, and the world Z-axis to the screen X-axis because that transformation requires the least rotation. **Snap** can be used for snapping to **XY**, **YZ**, **ZX**, **YX**, **ZY** and **XZ** planes. Similarly, workplane axes can also be aligned to screen axes. The option **Closest (Workplane)** also

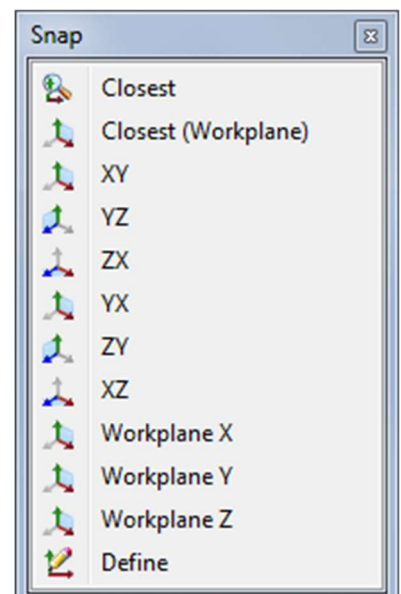


Figure 2.53: Snap drop down list

**Snap** can be used for snapping to **XY**, **YZ**, **ZX**, **YX**, **ZY** and **XZ** planes. Similarly, workplane axes can also be aligned to screen axes. The option **Closest (Workplane)** also

works similar to the **Closest** option, wherein here, the workplane axis is aligned to the closest screen axis.

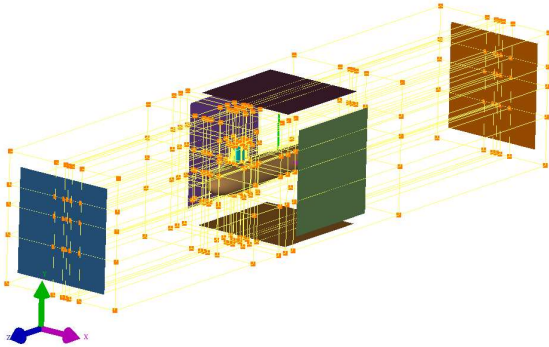


Figure 2.54a: Rotated View

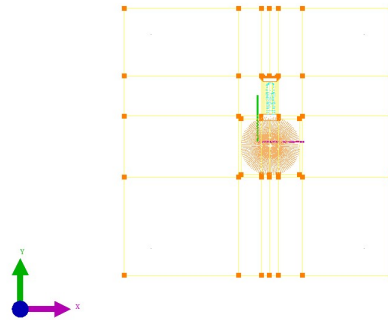


Figure 2.54b: Snap to the closet View i.e. XY plane

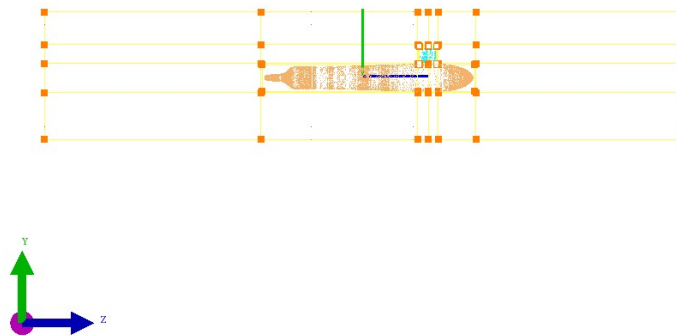



Figure 2.54c: Snap to the ZY plane

You can also snap screen axes to user-defined axes using the **Define** option  under the **Snap** menu. If selected, a dialog box will pop out. For that, you need to input the orientation and Up-direction, where the orientation axis is pointing towards the screen, and Up-direction is the axis pointing upwards. The third direction is automatically calculated.

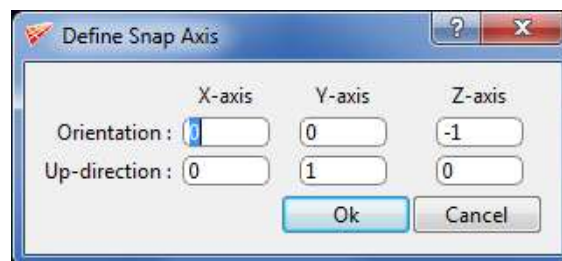


Figure 2.55: User-defined snap axis dialog box

### 2.3.8 Zoom/Unzoom



Zoom



Unzoom

Zoom and un-zoom help in zooming/un-zooming a portion of the screen with respect to the centre of the screen. The right mouse button can also be used to zoom into a portion of the screen to dilate that portion for a more detailed view. You form a white zooming rectangle by clicking and dragging a box using the right mouse button. The point you click on is one vertex of this zooming rectangle. As you drag the mouse, the rectangle expands so that the line between the mouse's original position and the current position is the diagonal of the rectangle. When you release the button, the screen automatically zooms in on the specified rectangle. If you have started to drag the zooming rectangle and then decide to eliminate it, resize the rectangle as small as possible. Most likely, the rectangle will then disappear. You can also use the mouse scroll to zoom in and zoom out, which when you scroll out, it zooms out, and when you scroll in, it zooms in. When the zoom or un-zoom is done using mouse scroll, the centre of the screen is assumed as zoom/un-zoom centre.

### 2.3.9 Pick

GUI gives you an option to change the centre of rotation based on your convenience. The Centre of Rotation (COR) is the point in 3D about which the camera is rotated in the visualization process. Centre of rotation is seen as small three axes represented in pink, green and blue colour in the graphics window. Note that this centre of rotation is not used when zooming in/out. The





Pick

centre of rotation can be picked using **Pick** option located in the global dock. When the button **Pick** is selected, any entity (corner, edge, surface, grid block edge, grid sheet or geometry) can be picked using the left button of the mouse to set that point as the centre of rotation (COR). Note that when there is no object under the mouse, the mouse is still under the selection mode, and the next mouse-click is used for the COR selection. Also, note that the workplane is not treated as an object and cannot be selected using this option.



Figure 2.56: Pick dropdown list

The centre of rotation can also be picked using a custom right-click menu of the entities in the graphics window. By holding the keyboard shortcut 'P' key and clicking on any point on any entity, you can pick the centre of rotation. Instead of picking, you can fix the centre of rotation to

the centre of view volume or centre of the screen by using **Reset**  in **Pick** dropdown list. By selecting **Origin** , you can fix the centre of rotation at the origin of the coordinate system. You can also fix the centre of rotation to the mass centre of the model by selecting the **Model** option in the **Pick** menu. Selecting **ALL** under **Pick** will set the centre of topology as the centre of rotation. Else you can define the x, y, and z coordinates for the centre of rotation by selecting **Define** in the **Pick** menu. Figure 2.57 shows the dialog box of the Define COR option.

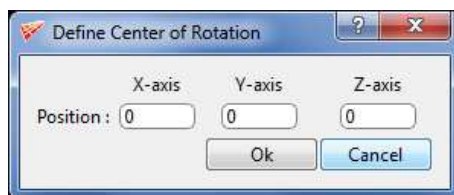


Figure 2.57: Define COR dialog box

### 2.3.10 Probe

You can get information about topological, surface and grid entities by using the **Probe**



button **Probe** located in the global dock. When you click on it, the mouse changes to the object selection mode, and you can click on the corner/edge/surface of the topology or grid sheets/block edges of the grids in the graphics window. The information regarding the selected object is displayed in a dialog box that pops up after the selection. When this button is clicked, it is pressed down until an object is selected from the graphics.

When a corner is selected, the corner data is shown in tabular form as shown in Figure 2.58. The corner index, its position vector, the list of corners linked to it, the surfaces to which it is assigned and the corner to which it is paired (if the topology is periodic) are all listed in the table. Note that this information can also be obtained from the topology (fra) file.

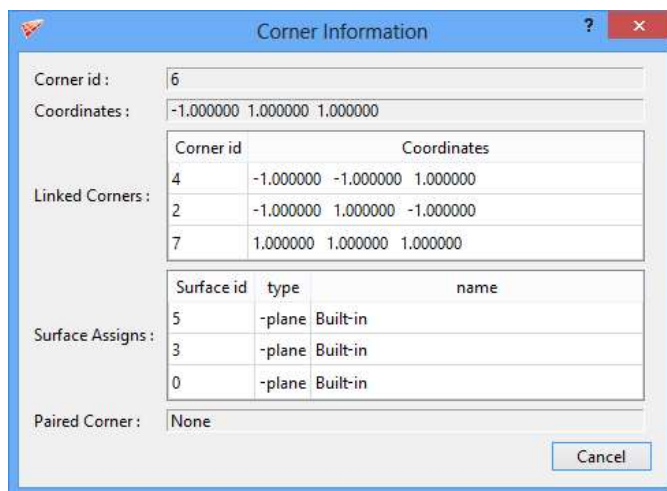


Figure 2.58: Corner information



When an edge is selected, the edge data is displayed in a dialog box, as shown in Figure 2.59. The edge index, its end vertices, its length, and the edge density are all displayed in the dialog box. The length of the edge is the distance between its terminal corners. Once the corners, edges or surfaces are created, their indices cannot be changed later.

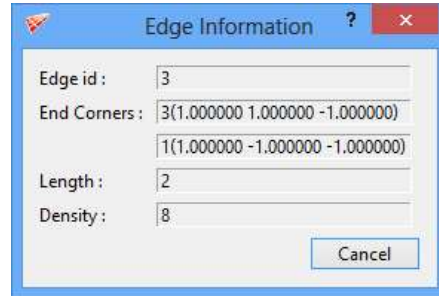


Figure 2.59: Edge information

When the surface is selected, the surface data is displayed in the dialog box, as shown in Figure 2.60. It includes its index, the type of the surface (can be one of the plane/ellip/xpolar/xyz/tria/quad/linear/tube), surface label (note that the surface property is a special kind of label), the complete surface information, the surface groups to which the surface is added.

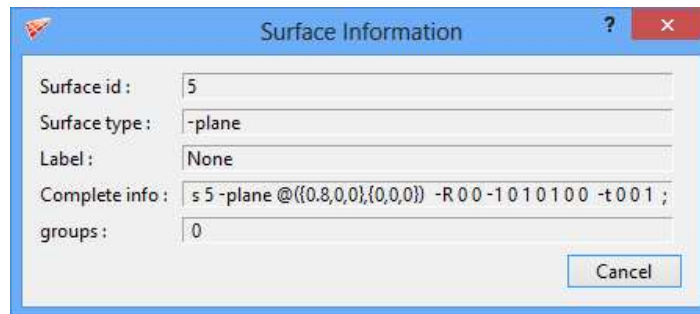


Figure 2.60: Surface information

When a grid block edge is selected, its information is displayed, as shown in Figure 2.61. The block edge information consists of the block index, the edge index within the block, and the edge density.

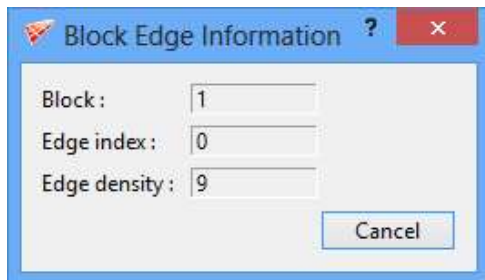


Figure 2.61: Grid block edge information

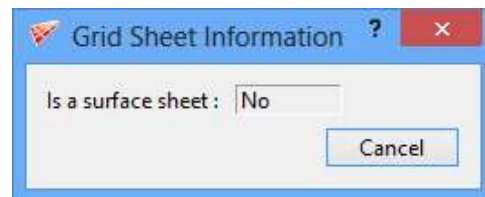


Figure 2.62: Grid sheet information

The grid sheet information is very limited because the grid sheets pass through various blocks of the grid, and the list of these blocks may be huge depending on the size of the grid. A grid sheet may not be linear, and hence it does not just contain  $x * y$  number of points on it to display its size. Hence, the only useful information displayed is whether a grid sheet is a surface sheet or not.

### 2.3.11 Measure

The distance between any two selected entities such as corners, edges, surfaces, grid blocks, grid sheets and geometry faces can be found using the measure button. Once the **Measure**



button is switched ON, you can pick any entities in the workspace. When you pick an entity, a white dot appears at the picked location, and once the second entity is picked, both the dots are connected by a green line with the distance highlighted over the line, as shown in Figure 2.63.

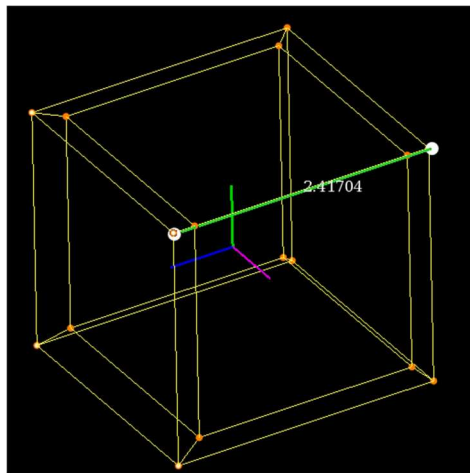


Figure 2.63: Distance between two corners

### 2.3.12 Snapshot

**Snapshot** button is used for capturing the GUI region. It has two options, i.e. *Entire UI* and *Graphics window*, as shown in Figure 2.64. By default, if clicked, it captures the GUI region, as shown in Figure 2.65. Entire UI option captures the whole GUI, and the Graphics window option captures the graphics window of the GUI, as shown in Figure 2.66. Once clicked, a file dialog box with a default name shows up. The default name for the snapshot file is "Snapshot-ws-year-mm-dd-hh-min-sec.png". The default path for the snapshot is the Current working directory. You can change the path of the file. You can save the file in .png and .bmp format.



Figure 2.64: Snapshot dropdown list

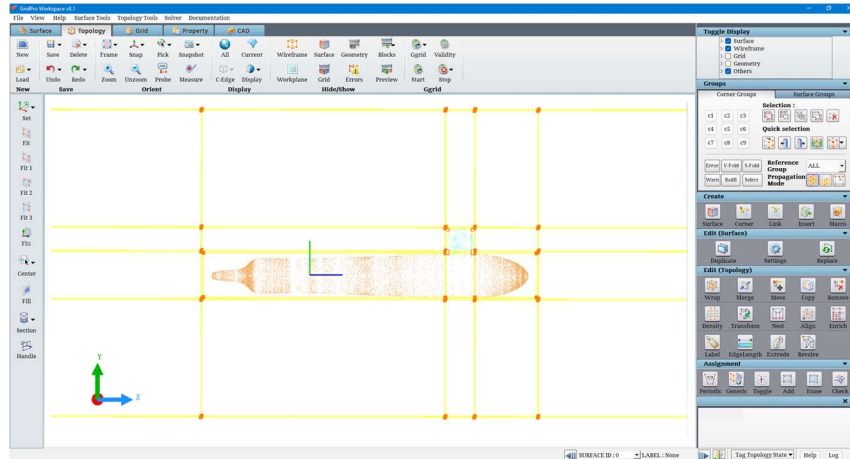


Figure 2.65: Snapshot of Entire UI

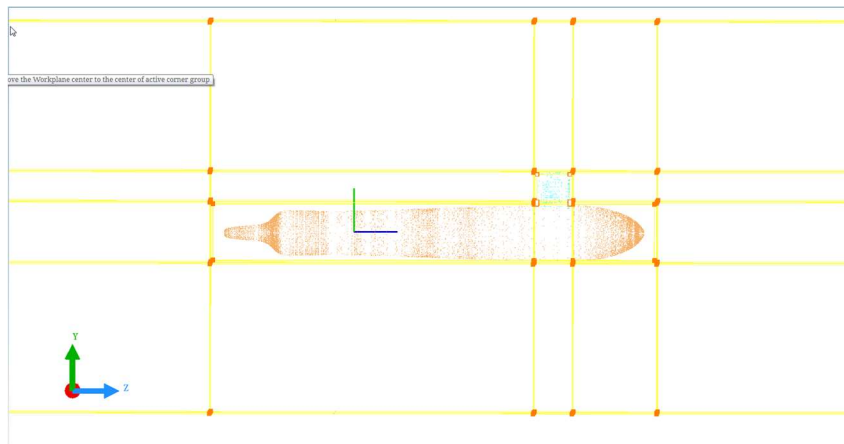


Figure 2.66: Snapshot of Graphics Window

### 2.3.13 All

You can turn on the visibility flags of all the surfaces of the topology by using **ALL** button




. Note that the surface groups act as a final layer of filters, and only the surfaces in the current/reference surface group are visible. The surfaces which are hidden using the right-click surface menu are also visible after this operation.

### 2.3.14 Current



Similar to turning the visibility flag on for all surfaces, you can turn the visibility flag off

for all surfaces (excluding the current surface) by using the **CURRENT**  button. But the current surface is always displayed using a special flag. Hence, all the surfaces (except the current surface) are invisible. When the *current* surface is scrolled, the new *current* surface alone is displayed, and all other surfaces are hidden including the previous *current* surface.

### 2.3.15 Display

Surfaces, geometries, and grids can be viewed in various modes for better visualization and a clear picture of these objects. The modes in which it can be displayed are:

1. Shade with HLR
2. Shade
3. HLR
4. Lines
5. Points
6. 1/10th points

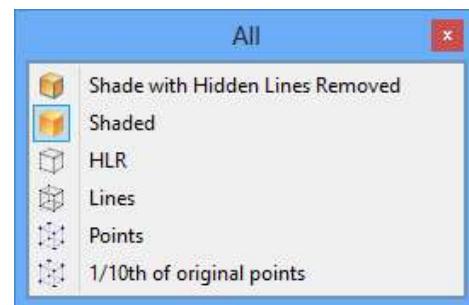







Figure 2.67: Different display modes

When an object is set to **Shade with HLR**  visualization mode, the polygon faces of the object are drawn using the object's own colour, but the polygon boundaries (edges) are drawn in white. Figure 2.69 shows a surface in this visualization mode. Note that since the polygon faces are drawn and are opaque, the objects on the other side of the object cannot be seen.

In **Shade**  visualization mode, the polygon faces of the object are drawn using the object's own colour. The polygon boundaries (edges) are not drawn at all. The objects on the other side cannot be seen because of opacity.

In **HLR** (Hidden Line Removal)  visualization mode, the polygon faces of the object are drawn using the background colour, and the polygon boundaries (edges) are drawn using the object's own colour. Note that the objects on the other side of the object cannot be seen.

When the display mode is set to **Lines** , the polygon faces of the object are not drawn, and the polygon boundaries (edges) are drawn using the object's own colour. In this display mode, objects on the other side are visible.

When the display mode is **Points** , only the polygon vertices are displayed, and all the polygon edges and faces are not drawn at all. This mode is suggested for large surfaces.

Similarly, display mode 1/10 points can be useful when working on large surfaces. It displays only 1/10th of all the polygon vertices.

By default, all surfaces are displayed as shaded. The visualization of the surfaces can be changed based on the display type. There are three display types given in the dropdown list: All, Current, and *Moving*. The visualization for all surfaces can be changed by specifying the display type under the *All* dropdown menu. The visualization mode of the current object (surface or grid) can be set independent of the remaining objects. For example, when all the surfaces are set to **HLR** mode and the current surface can be set to **Shade** mode, it can be visualized better. This can be done by

specifying the display type for the Current object under the *Current* dropdown menu. The visualization mode of the moving objects (while panning, rotating or zooming) can be set separately by specifying the display type *Moving* dropdown; it defines the outlines of the object when they are moving. By default, the visualization mode of the moving objects is set to "1/10<sup>th</sup> points" because they can be moved very easily and smoothly. Display mode while moving an object should be either points or 1/10th points. Under *Current* and *Moving* dropdown, another option is available, i.e. **Follow all surfaces display type**; it copies the display type of all surfaces to the current surface or while moving surfaces for whichever option is selected.

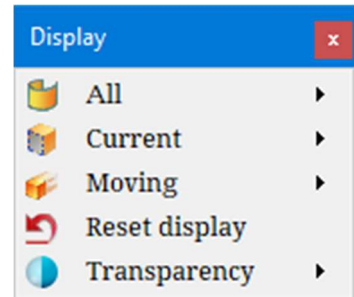


Figure 2.68: Display drop-down list

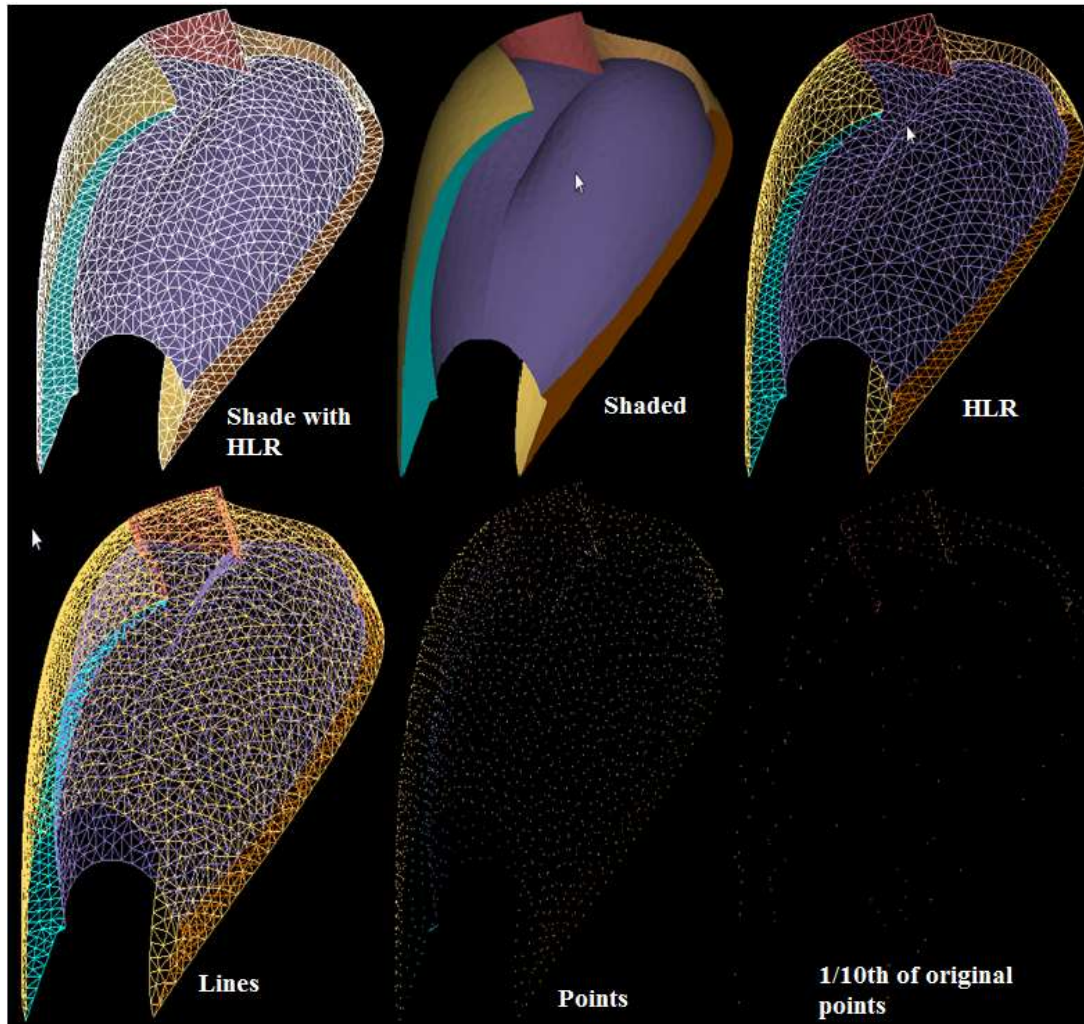


Figure 2.69: Surface with different display modes

Note that the visualization type can also be changed/set by right-clicking on the object and selecting the appropriate visualization mode (will be explained later). In this case, the mode which is set manually from the right-click custom menu overrides the mode selected through this display menu.

You can reset the manually set visualization modes of all the objects (using right-click custom menu) in the graphics by using ***Reset Display***. When the visualization mode is reset, the display mode of the surfaces uses the mode selected using the display menu. For example, when the all surfaces display mode is set to HLR and a surface is set to Shade using the right-click menu, the surface display mode is reset to the HLR when the reset display mode is clicked.

You can also set ***transparency*** for all surfaces. By default, all the surfaces are displayed with 0% transparency. As the value increases, the colour of the surface becomes lighter and transparent such that you can see the wireframe behind the surfaces. Clicking on the dropdown lists a few predefined values; however, you can also set a custom value using the *custom* option at the end of the list.

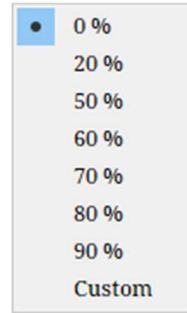


Figure 2.70: Surface transparency dropdown list

### 2.3.16 C-Edge

Clicking on this button will toggle the colour coded edge connectivity of the geometry loaded into GUI. If an edge is defined by a single geometry face, it is displayed in red colour. If an edge is defined by two geometry faces, it is displayed in bright green colour, as shown in Figure 2.71. If an edge is defined by 3 faces and 4 faces, it is shown in yellow and bright blue colour respectively.

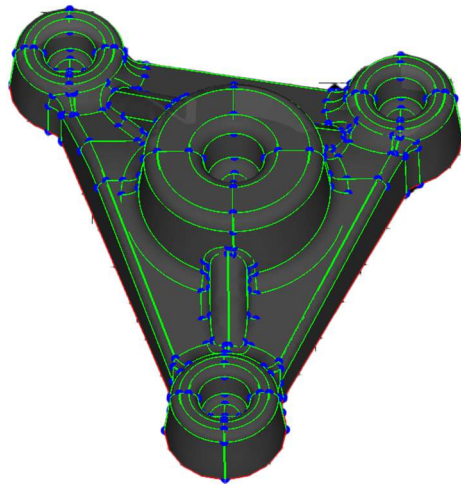


Figure 2.71: Geometry connectivity lines

### 2.3.17 Wireframe

Clicking on this button will toggle the entire wireframe and its related objects such as corner ID and errors if it is on.

### 2.3.18 Surface

Clicking on this button will toggle all the surfaces and related objects such as border, path selection, and feature edges.

### 2.3.19 Geometry

Clicking on this button will toggle the geometry and related entities such as edges and faces.

### 2.3.20 Workplane

Clicking on this button will toggle the workplane and its handle.

### 2.3.21 Error

Clicking on this button will toggle all the error highlights in the GUI.

### 2.3.22 Blocks

Clicking on this button will toggle the shrink blocks in the UI. Shrink blocks are small cubes that are shown for all valid block definitions. There are 4 kinds of shrink blocks and 3 kinds of shrink faces shown in different colours. First one is All Blocks, where all the valid block definitions are shown in blue coloured cubes. Second and third ones are based on validity. Once the validity button is clicked, the topology is evaluated, and the shrink blocks are displayed in green and red colour. The green colour represents that the topology block is valid for grid generation and the red colour represents that the block is valid but invalid for grid generation. i.e. the block definition is valid, but the block validity is invalid. The fourth one is excluded blocks which are shown in white dotted lines. Please refer to section 4.7.1 for more information on the face excluded block. Apart from these, the singular blocks are shown in different colours based on the severity of the issue.

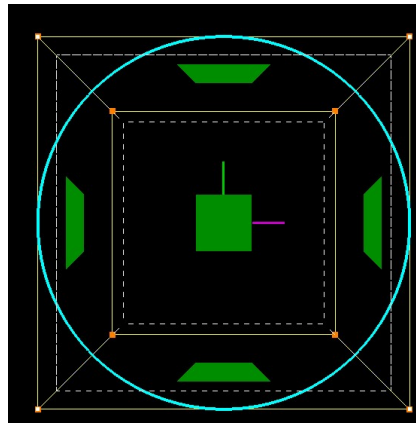
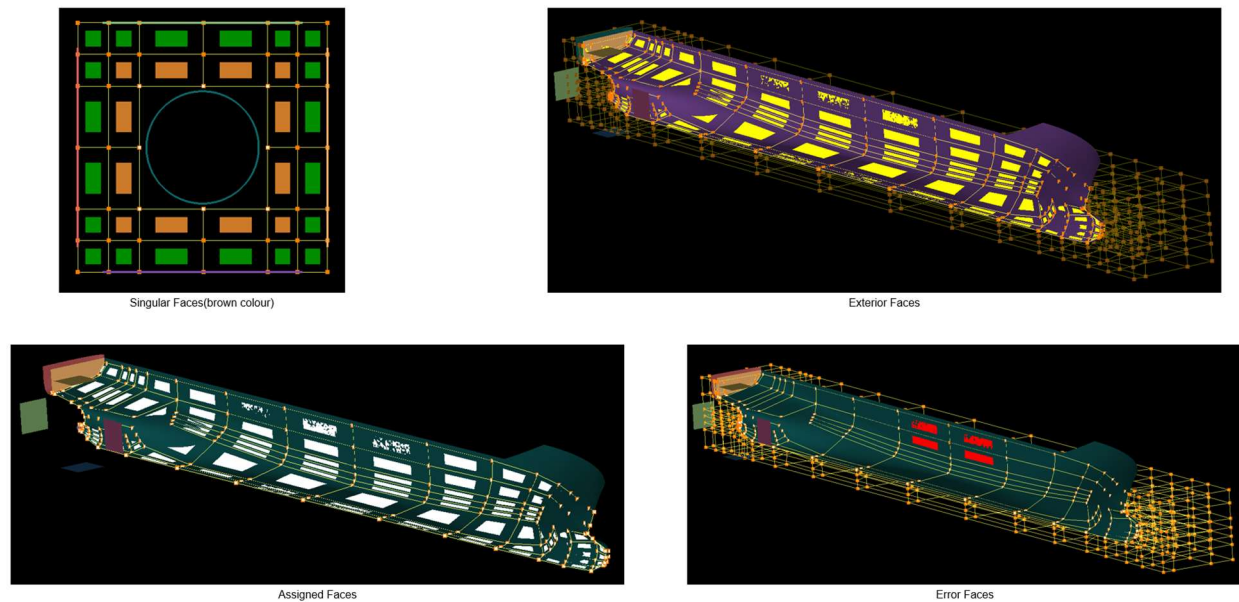


Figure 2.72: Shrink Block Display

Similar to blocks, the faces are also displayed as shrunk faces from v8.1. Found kinds of faces are displayed - one for exterior faces wherein all the boundary faces of the topology are displayed, second for assignment faces where all the assignments faces of the current surface are displayed, third for singular faces where all the faces corresponding to a singularity are displayed and lastly for error faces where all the error faces are displayed. Each one of them is displayed in different colours. Whenever a singular, assignment, or error face overlaps with an exterior face, they are displayed over the exterior face. i.e. singular, assignment and error faces are shown in place of exterior faces. These faces are automatically switched ON/OFF based on the scenario to make the workflow easier. For e.g. when the insert button is clicked, the exterior faces are switched ON automatically. Similarly, when the check button is clicked, the assignment faces are switched ON automatically.



*Figure 2.73: Shrunk Faces*

### 2.3.23 Grid

Clicking on this button will toggle all the grids loaded into the GUI and related objects such as blocks, sheets, block IDs, and orientation.

### 2.3.24 Preview

Clicking on this button will toggle the grid preview displayed in the UI. Please refer to section 5.1.6 for more information on Grid Preview.



### 2.3.25 Start Ggrid

In this global dock, commands related to starting and stopping **Ggrid** are added for convenience as it is commonly used at various stages of topology creation. Once a valid topology is built, the grid generation engine can be used to get the structured multi-block grid.

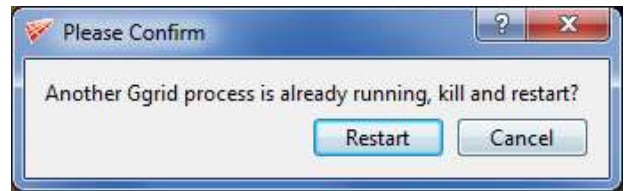


Figure 2.74: Ggrid error message



The grid generation engine Ggrid can be started using **Start** icon located in the global dock. Ggrid runs on the topology of the current graphics window of the GUI. If Ggrid is already running, an error message is displayed asking the user to kill the existing one and restart, as shown in Figure 2.74. If there is no topology, an error message pops up, as shown in Figure 2.75. GridPro

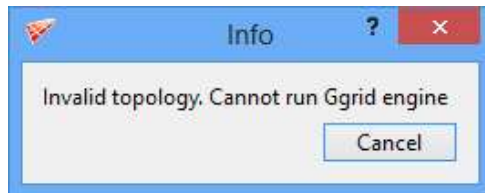



Figure 2.75: Ggrid error message

will also refuse to begin the gridding process if the topology is incorrect. The errors include not assigning corners to surfaces, not assigning corners to the right surface, and failing to create wireframe quadrilaterals or hexahedrons. Also, the topology itself could be implausible for a given region. If you make a mistake, GridPro will guide you to that location by automatically turning on the error and highlighting it, which serves to show you your mistake. Errors are displayed only when you attempt to start the gridding process of an incorrect topology. Concurrently, a more technical error message is also displayed in the command terminal.

- **Schedule**

If the topology is valid, a window will pop up to schedule the generation engine, as shown in Figure 2.76. The **Grid output** specifies the name of the grid file that is to be written. By default in GridPro all the grid files are written as 'blk.tmp'. The grid output name is editable, the extension should either be .tmp or .grd. Next to the text box is the **Settings** button , which opens up

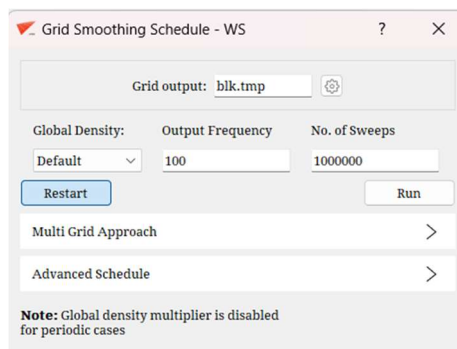


Figure 2.76: Grid schedule dialog box

additional settings. Here, the **Clustering** options, **Output File Format**, **Output Restart File**, the **Schedule File** name and **Input Groups** are located. The **dump frequency** is the sweep interval after which the output grid is rewritten. This is typically a multiple of hundred.

You can choose to **Comply** with the clustering parameters or **Ignore** them using the drop down. The **Output File Format** of the grid can be ASCII/Binary. Binary output format is preferable if the topology is huge, but an ASCII grid file is readily readable and can be

modified easily if needed. Binary grid cannot be reloaded into UI while running Ggrid because the grid loaded into the UI is converted from binary to ASCII. If you want to reload the binary grid, delete the existing grid and load it again. The name of the **Output Restart File** is set to 'dump.tmp' by default, which can be changed by editing the name. The checkbox next to the Output Restart File can be used to determine if the restart file is to be written or not during grid generation. If the user does not want to restart the grid generation process, this option can be disabled, which can save time while writing the grid file.

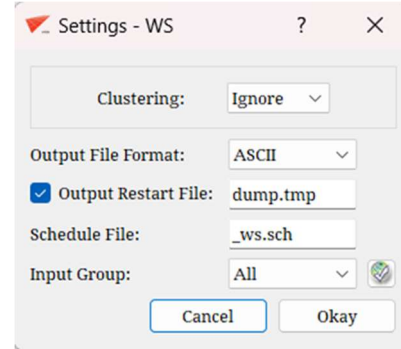


Figure 2.77: Grid schedule settings dialog

You can also save the schedule file parameters to a user-specified name using the **Schedule File** input box. The advantage of changing the schedule file name helps in running Ggrid from the command terminal. Ggrid can also be run on part of the topology. Group the respective wireframe to be run and input the same under **Input Group** option. Once the corner group is selected, the validity button next to it is enabled. Click on the validity button to check whether the given group is valid to run Ggrid or not. The corresponding error message will be shown similar to the 'Validity' button on the global dock. Note that the given corner group must have all its boundary faces assigned to either one-sided or two-sided surfaces. If the topology is disjoint, meaning that there are regions of the topology which are completely disconnected from one another, then Ggrid can be run on the individual topologies by specifying their respective corner groups in the **Input Group**.

Coming back to the **Grid Smoothing Schedule** dialog box as shown in Figure 2.76, let's now look at the **Global Density** option. All the edge densities of the topology can be doubled, quadrupled, halved or quartered. This can be set by changing the options in the associated dropdown. Density designates to cell count per edge. When the density of an edge is halved or quartered, the integer closest to the multiplied value is taken as the density of the edge. In the schedule file, density can be scaled by any multiplier (not just restricted to 2, 4, 0.5 and 0.25). The **Output Frequency** specifies the number of sweeps after which the grid file is rewritten. The **No. of Sweeps** is the total number of sweeps Ggrid will run for after which it stops automatically. The user can manually stop Ggrid before this sweep count is reached.

If you want to run a previous computation, click the **Restart** button and input the corresponding restart grid file, which is by default called 'dump.tmp'. Clicking **Run** gives a popup as shown in



Figure 2.78. Here, you can either overwrite or use the existing grid and dump file by selecting the appropriate option.

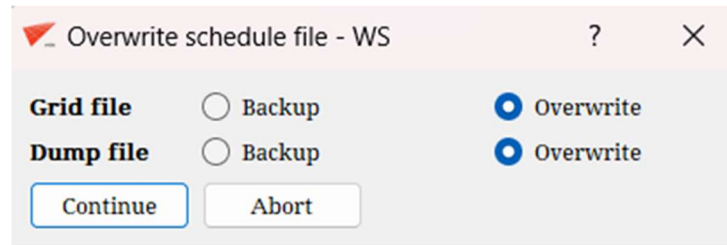


Figure 2.78: Overwrite Schedule File dialog box

Clicking on the **Multi Grid Approach** opens the Multi Grid Approach dialog box. Here, the user can specify the number of sweeps the grid smoother must run on for different Global Densities.

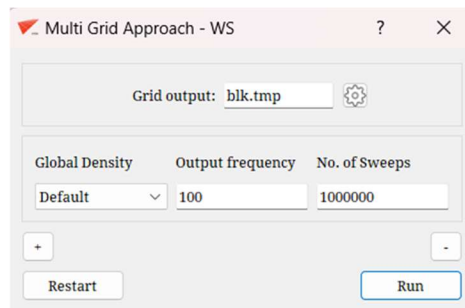




Figure 2.79: Multi Grid Approach dialog

For large topologies, it is helpful to first run the topology for a lower density, and then once the corners are approximately smoothed and placed, the smoother can be run for higher densities. This aids in faster grid convergence. For example, the figure shows how we can run the grid smoother at half density for a 1000 sweeps, then go to the default density for 1500 sweeps, and finally run at double the density for 500 sweeps. The smoothing computations are automatically restarted from the previous computations after each step. Additional

steps can be added using the  button and removed using the  button.

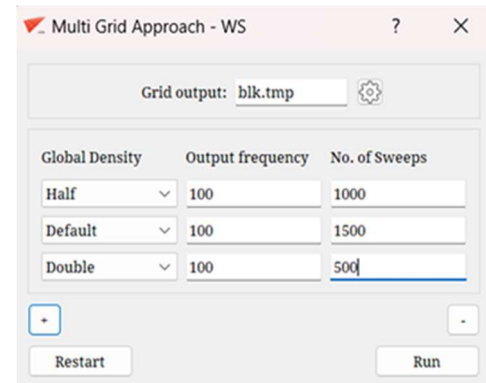


Figure 2.80: Multi Grid Approach example

You can also input Advanced Schedule options by clicking on the **Advance Schedule** button. Clicking on it will pop up a dialog box, as shown in. Loop count is nothing but the step numbers mentioned in the \*.sch files. The rest are all self-explanatory options.

Please refer to **Appendix C** for more information on editing schedule file. The tabular form is nothing but a UI version of the \*.sch file.

Loop Count (#)	Density (*)		Clustering							E-Wall			Acceleration		Sweep Frequency
	Value	Multiplier	Enable	Label	Multiplier	Growth Ratio	MultiGrid	Algorithm	Enable	Multiplier	No. of Layers	Label	No. of Sweeps		
1			<input type="checkbox"/>	ALL	1.0	1.2		slowest	<input type="checkbox"/>	1.0	3			100	
2			<input type="checkbox"/>	ALL	1.0	1.2		slowest	<input type="checkbox"/>	1.0	3			100	
3			<input type="checkbox"/>	ALL	1.0	1.2		slowest	<input type="checkbox"/>	1.0	3			100	
4			<input type="checkbox"/>	ALL	1.0	1.2		slowest	<input type="checkbox"/>	1.0	3			100	
5			<input type="checkbox"/>	ALL	1.0	1.2		slowest	<input type="checkbox"/>	1.0	3			100	
6			<input type="checkbox"/>	ALL	1.0	1.2		slowest	<input type="checkbox"/>	1.0	3			100	
7			<input type="checkbox"/>	ALL	1.0	1.2		slowest	<input type="checkbox"/>	1.0	3			100	
8			<input type="checkbox"/>	ALL	1.0	1.2		slowest	<input type="checkbox"/>	1.0	3			100	
9			<input type="checkbox"/>	ALL	1.0	1.2		slowest	<input type="checkbox"/>	1.0	3			100	

\* Do not provide both value and multiplier for density. Only one will be considered.  
# Only rows with step count are considered. Hence it cannot be empty.


Write Format : ASCII

Close Reset Ok

Figure 2.81: Advanced Schedule Dialog Box

### 2.3.26 Stop Ggrid

You have to stop the Ggrid engine manually once started. To stop the Ggrid, click on **Stop Ggrid**

icon  located in the global dock. If Ggrid is running, a confirmation dialog is pops-up, as shown in Figure 2.82. But if the grid generation engine is not running, an error message is displayed (see Figure 2.83). Sometimes you may encounter a situation where the UI may have crashed while running the Ggrid, or you may have stopped the Ggrid, but the process will still be running in the background. In such cases, you cannot run a new Ggrid process; then, you can click on **Kill Process** under the **Stop** button to resolve the issue in such scenarios. If Ggrid attempts to stop while it is writing the grid file, it will finish the writing first and then stop the process. In such a case, if you do not want it to write the grid file, the *Kill process* option can be used.

Please Confirm

Are you sure you want to stop Ggrid?

Ok Cancel

Figure 2.82: Confirmation dialog box

Info

Ggrid process is not running.

Ok

Figure 2.83: Error message when ggrid is not running

Ggrid engine also can be stopped using the keyboard shortcut, i.e. **CTRL+Shift+G**. It terminates the gridding process immediately and stops the grid writing. A confirmation dialog box does not pop up to confirm this process.

### 2.3.27 Ggrid

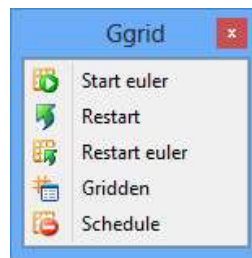



Figure 2.84: Ggrid dropdown list

- **Restart**

Once Ggrid is stopped, and after looking at the grid, you may have realized that it would have been better to have run it for a few more sweeps to get a good quality grid; then, you can use the **Restart** option  under **the Ggrid** button to resume the previous computation. The last saved dump file is used to restart the grid generation engine. If an instance of Ggrid is already running, an error message pops up, as shown in Figure 2.85, and if the topology is invalid, an error message is displayed as shown in Figure 2.74. If the topology is valid, a file dialog box, as shown in Figure 2.85, is displayed to select the appropriate dump file (having *.tmp* or *.grd* extension). By default, the dump.tmp file in the current working directory is selected.

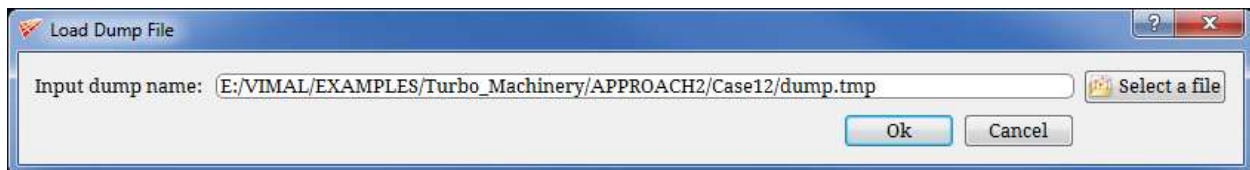




Figure 2.85: Restart Ggrid dialog box

Note that here it is assumed that a previous topology is modified and Ggrid is restarted. Hence, a scheduler is not launched here, and the previous schedule file, if any, is used. Also, note that Ggrid cannot be restarted on a disjoint topology because a single dump file is not generated when it has been started using multiple processes.


- **Start Euler**

There is one more option to start Ggrid, i.e. **Start Euler** , which ignores the surface clustering parameters, i.e., the off-wall spacing of the topology. Except for this, it behaves in the same way as the **Start Ggrid**.

- **Restart Euler**

Similar to **Restart Ggrid**, the UI provides the option of **Restart Euler** , which behaves in the same way as **Restart**. But it ignores all the clustering parameters and off-wall spacing of the topology.


- **Gridden**

UI also provides you with the option of changing the edge density while running the Ggrid. You can change the density and update the Ggrid engine by using the **Gridden** option  located in the global dock. The **Gridden** changes the edge density dynamically. It can also be done automatically by the density button. Refer to section 4.9.1 for more information.

- **Schedule**

The schedule file is a file that controls the gridding process, which can either be modified from UI or by editing it in notepad manually. Refer to *section 2.3.24* for more information on modifying the schedule file from UI or refer to *Appendix C* to edit the schedule file manually.

## 2.3.28 Validity

Before starting Ggrid, at any stage, the validity of the topology can be checked using the **Validity** button . If the topology is valid, a message will pop up, as shown in Figure 2.86 with the grid statistics. If the topology is invalid, an error message will pop up, as shown in Figure 2.83. If the topology is valid and it has singular edges, then a warning message will pop up, as shown in Figure 2.87, explaining the issue and possible solution.

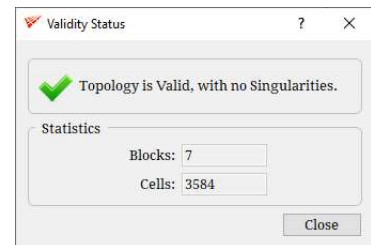


Figure 2.86: Valid topology status

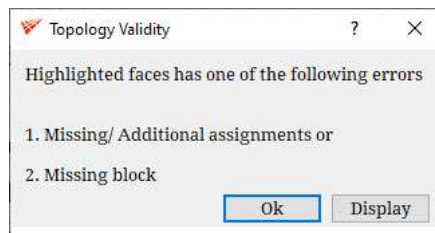


Figure 2.87: Invalid topology status

If it has a singularity, then the respective quality issue message would be shown along with the reason and the possible fix. Clicking on the image icon opens a pop-up box with an image depicting the issue and fix.

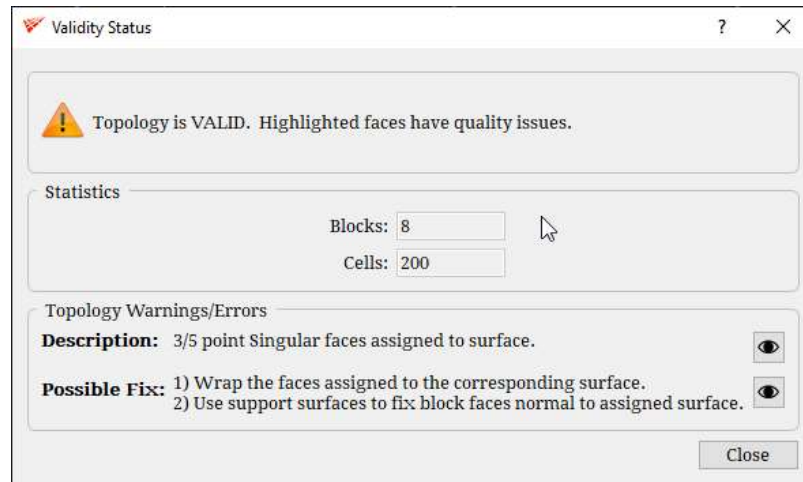


Figure 2.88: Mediumly Severe Singularity Message

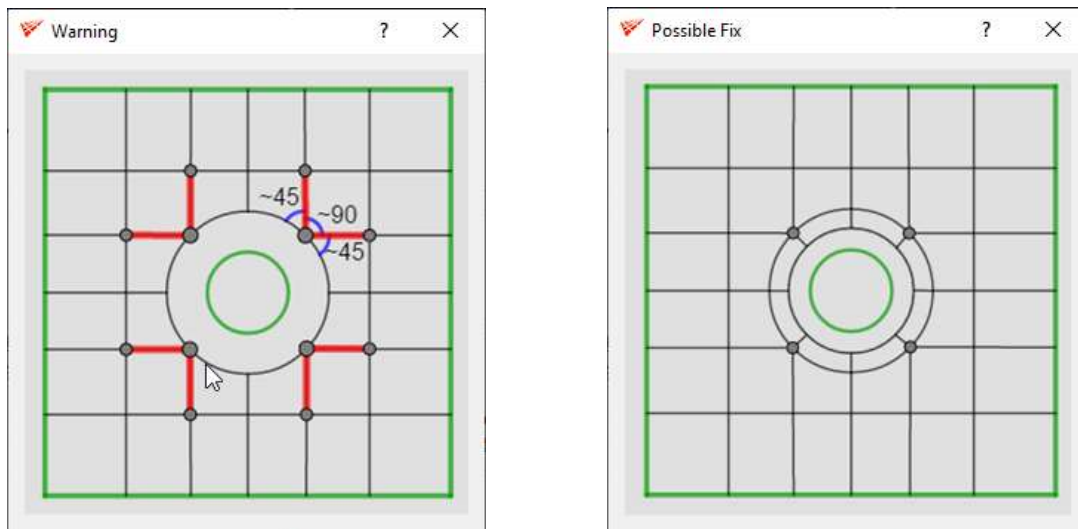


Figure 2.89: Images explaining the issue and possible fix

## 2.4 Workplane Toolbar

- Workplane

As discussed before, the workplane is the heart of creating and visualizing three-dimensional topology. It allows you to place topology around, behind, and below objects. For example, if you want to create a wireframe of a cube enclosing a sphere, you need a device that can move you around the six faces of the cube. The workplane allows you to place topology at any (x, y, z) coordinates in space. The plane can be rotated and translated to fit any position. The

workplane is infinitely long; however, to facilitate easy operational use, GridPro represents the workplane as a finite rectangle represented in plum colour, as shown in Figure 2.90. This rectangle is called the workplane rectangle. It can be shrunk or expanded to work on a global or local scale.

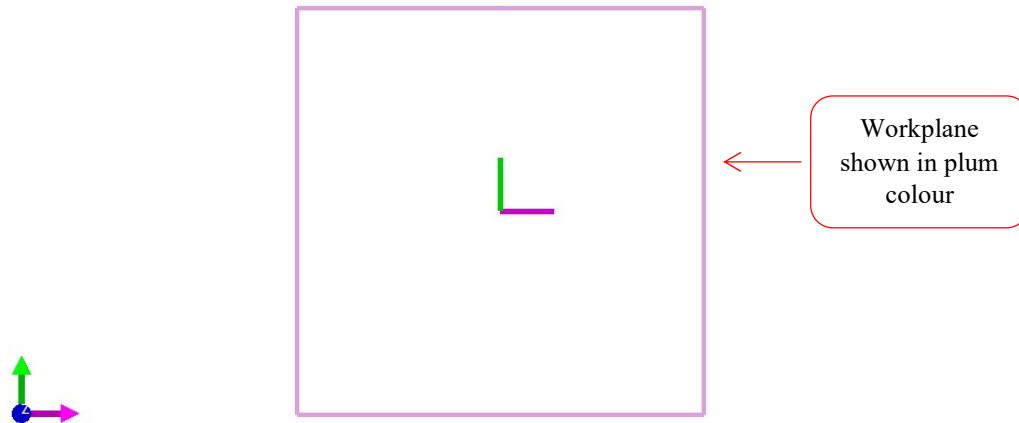



Figure 2.90: Workplane is shown in plum colour

The workplane should be thought of as a tool that guides your cursor. Cursor movement is always along the workplane. In other words, although you see the cursor move on a flat, two-dimensional screen, it is actually moving on the workplane. Essentially, the cursor is "attached" to the workplane regardless of the position or size of the workplane rectangle. Before you input or move topology, always think of where the workplane is positioned.

If the rectangle is too large, you will see nothing because you are looking inside the rectangle. In this case, zoom out the screen. By contrast, sometimes, the rectangle will appear as a white speck. In this case, zoom in on it.

All the tools related to the workplane are located in the left side toolbar as shown in Figure 2.91 and can be accessed from all four tabs.

- **Workplane Axes**

Similar to global axes, a set of workplane axes are represented in pink, green, and dark blue colour. Along with axes, there are four violet squares, each of which is placed on the vertex of the workplane rectangle, as shown in Figure 2.92. The pink axis is the workplane x-axis, the green axis is the workplane y-axis, and the dark blue axis is the workplane z-axis. When you bring the mouse cursor near an axis, the cursor will transform into two curved arrows . Make sure the cursor is changed into two curved arrows before you try to click on any axis.

Usually, workplane axes are referred to as handle as it helps you translate or rotate in the graphics window. By dragging an axis with the translating mouse button (the left mouse button), you can translate the workplane along that axis. You will use this imprecise but simple method of translation most often to position loosely the workplane. If you wish to rotate the workplane about one of its axes, click on the axis with the left mouse button. Notice that the specified axis becomes thin. Then, drag one of the two fat axes with the rotational mouse button (the middle mouse button) to rotate the workplane about the thin axis. You can change the size of the workplane rectangle by dragging one of the four violet vertexes either inward or outward. When you position the cursor on a vertex, the cursor becomes a blue square with a centre dot.

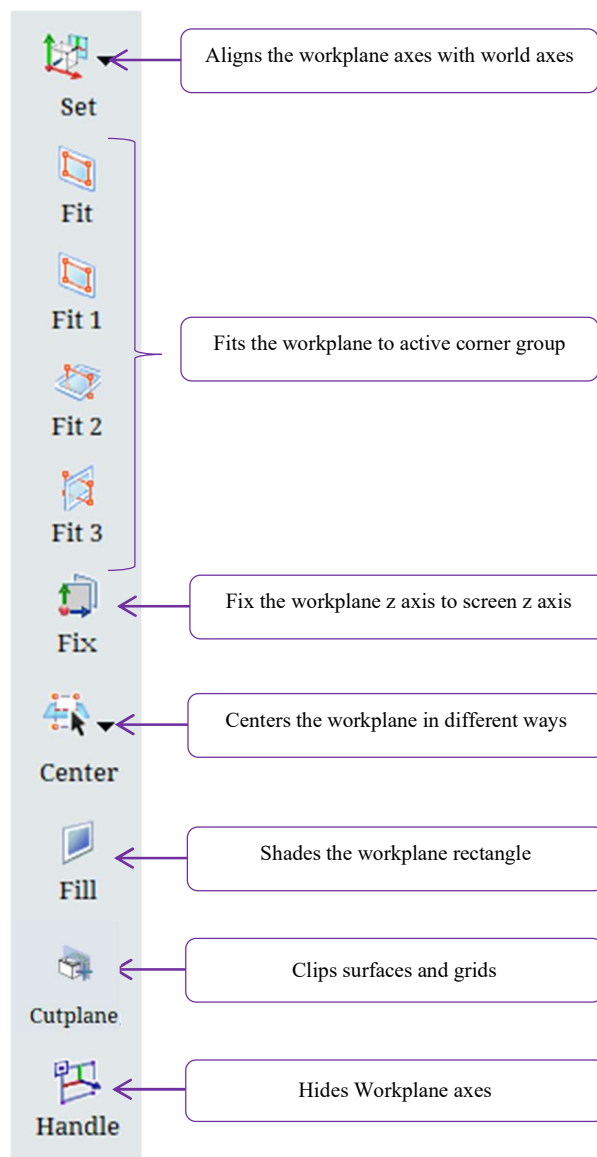


Figure 2.91: Workplane toolbar

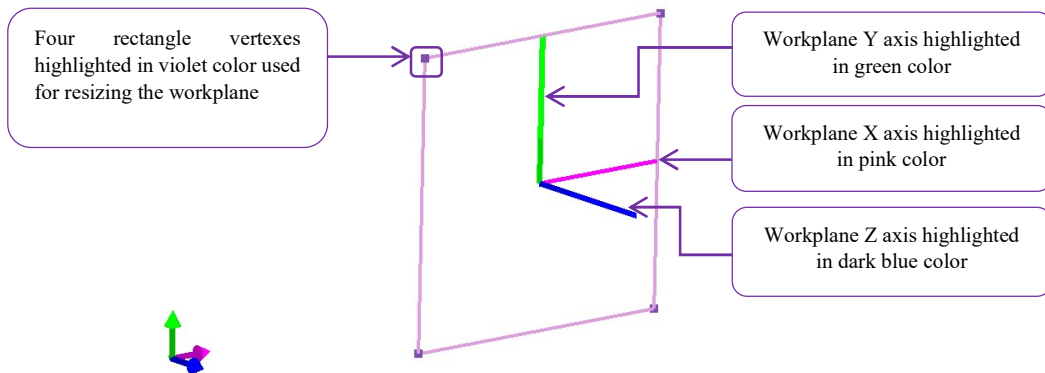


Figure 2.92: Workplane with Handles switched ON

### 2.4.1 Set

Topology creation in three dimensions is sufficiently complex; GUI provides you with all the required tools to help you create these topologies easily. The workplane allows you to place corners anywhere in space. Also, the workplane helps you orient yourself.


GUI provides you with the shortcuts to align the workplane axes to global axes and screen

axes using the **Set** button. The *set* menu contains eight options, as shown in Figure 2.93. Each option allows aligning workplane axes with world axes/screen axes by transforming the normal of the workplane. For example, if you select the **World X** operation, the normal of the workplane will align with the world X-axis. As a result, the workplane will snap to the yz plane. The **World Y** and **World Z** operations work similarly. The **Screen X**, **Screen Y** and **Screen Z** option aligns the centre of the workplane axis with the centre of the respective screen axis.





Figure 2.93: Set dropdown list

It can also be positioned anywhere you want if you have the centre and normal of the desired location using the **Define**  option under **Set** dropdown. For define option, the *normal* and *centre* fields, define the position of the workplane. Like any plane, the workplane can be positioned by inputting the coordinates of a normal vector to the plane and the coordinates of one point on the plane. The point on the workplane for which you provide coordinates becomes the centre of the workplane. You can input (x, y, z) coordinates of the normal vector in the *normal* field and input (x, y, z) coordinates of the point in the *centre* field. Like the position of the workplane, the size of the workplane can also be defined using the *size* field. By default, the size is defined as the current workplane size that is calculated automatically once the **Define** button is clicked, and it will retain the same unless the user changes it; when you finish inputting all these values, press **Ok** to implement your input as shown in Figure 2.94. The workplane should immediately conform to its new properties. The workplane centre can also be defined based on a corner ID. The corner ID can be found from the corner information pop-up. The selected corner ID's position acts as the centre of the workplane, and the user can define the normal and size of the workplane.

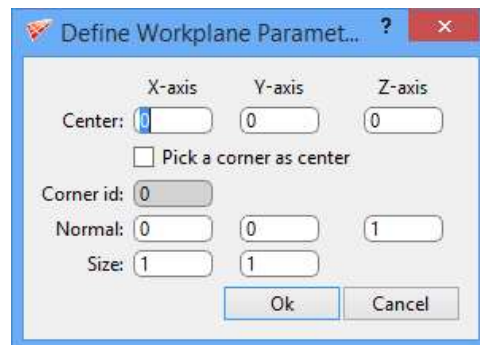


Figure 2.94: Define Workplane Parameters dialog box

## 2.4.2 Workplane Group fit

In three dimensions, you will almost surely use both corner groups and the workplane that is useful in creating a topology with ease. Therefore, *GridPro* provides you with operations that

integrate group and workplane features. Mastering these operations will help you manoeuvre quickly and efficiently in the GUI.

Often, you might want to align the workplane with a group of corners that can be used for copying or moving in all three directions. **Fit** operations are used for this purpose. The **Fit** operation aligns the centre of the workplane rectangle with the centre of mass of the active group.

The other fit buttons (*fit1*, *fit2*, *fit3*) centre the workplane in all three axes with respect to different reference frames. The **Fit 1** operation aligns the centre of the workplane with the centre of mass of the active group. Also, it aligns the workplane normal axis to the shortest major group axis and the other two workplane axes to the other two group axes. The **Fit 2** operation performs the same function, i.e. align the centre of the workplane with the centre of mass of the active group, but also aligns the workplane normal axis to the second shortest major group axis and the other two workplane axes to the other two group axes, whereas, the **Fit 3** operation aligns the workplane normal axis to the longest major group axis and the other two workplane axes to the other two group axes.

Figure 2.95 displays how the workplane changes position for different fit options for the same group of active corners.

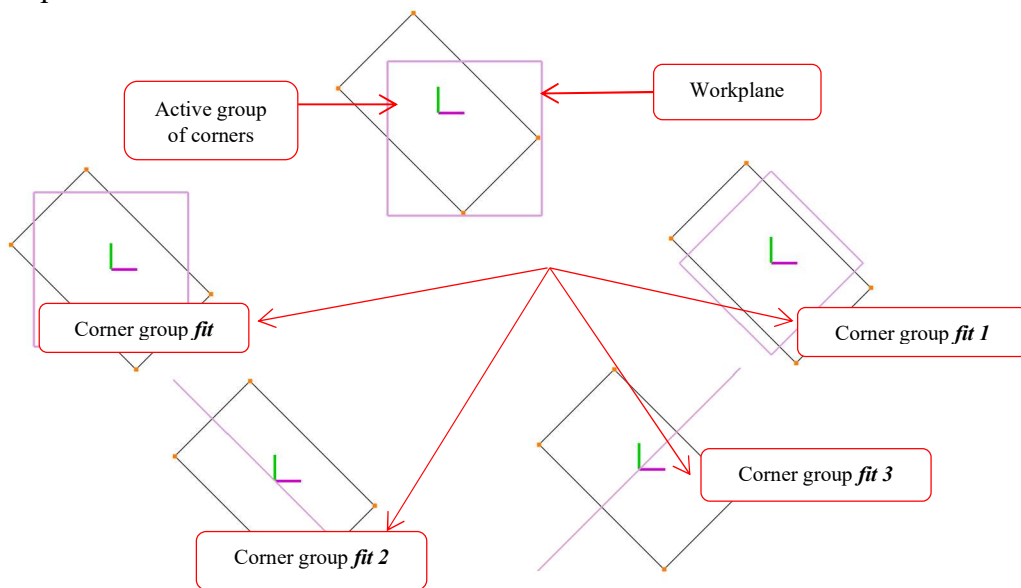


Figure 2.95: Group fit position for active corner groups

### 2.4.3 Fix

The workplane centre can be fixed to the centre of the screen, and it will update every time whether the model is rotated or translated. You can achieve that by switching on the **Fix** button in the workplane toolbar; however, when the screen is rotated, the workplane Z-axis is always aligned with the screen Z-axis.

## 2.4.4 Centre

You can position the workplane centre on any surface or any corner/edge on the wireframe or the grid sheet with just a click on it using the



**Centre** button in the workplane toolbar. Click on the **Centre** button and click on any surface or corner/edge to transform the centre of the workplane to the clicked position. You can also position the workplane centre on to any predefined options such as Origin, Center of the Model and Center of the Topology.

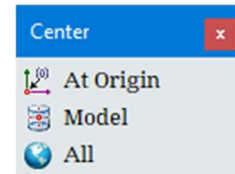


Figure 2.96: Center dropdown list

## 2.4.5 Workplane Fill

**Fill** operation of the workplane serves as a visual aid. Clicking on the **Fill** option will leave the workplane rectangle shaded in translucent white. Usually, the **Fill** operation is most useful when a side of the surface is clipped, and you want to better orient your eyes to the view screen. Essentially, the **Fill** operation locates the workplane for you and allows you to view better 2-dimensional slices of a surface, as seen in Figure 2.97.

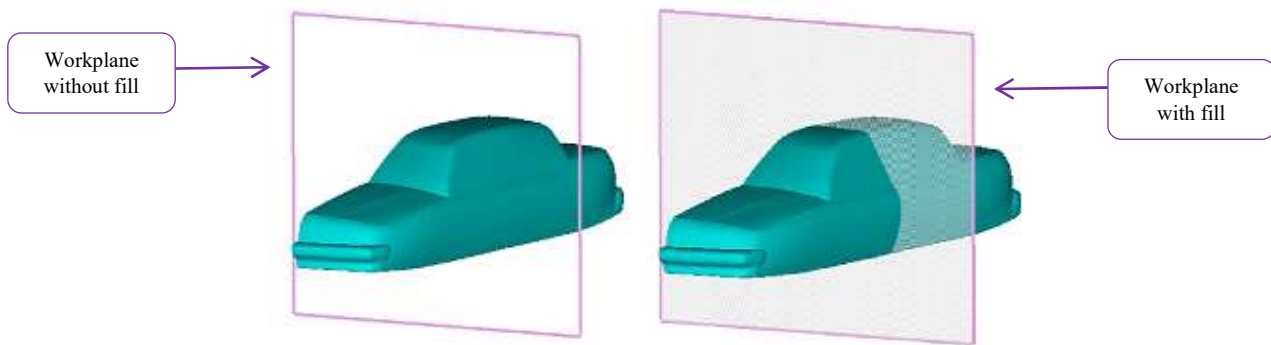


Figure 2.97: With and without fill for workplane respectively

## 2.4.6 Cutplane

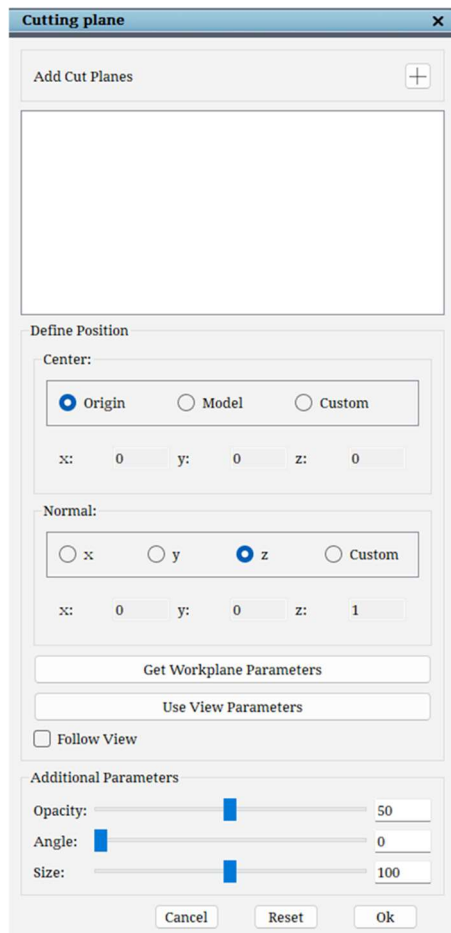




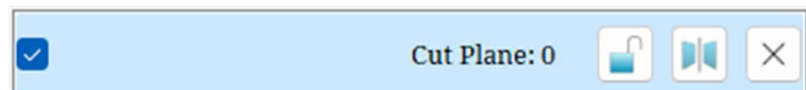


Figure 2.98: Cutplane dialog box

In the **Define Position** section, we can define the **Center** and **Normal** of the cutplane. The center can be defined using one of three options - **Origin**, **Model**, or **Custom**. Origin selects the origin defined by coordinates of 0,0,0. Model calculates and selects the center of the model, whereas using the Custom option the center can be defined manually. The Normal can be defined using one of four options - **x**, **y**, **z** and **Custom**. The first three options define the normal of the cutplane along the x, y and z axes respectively. Using the Custom option, the normal of the cutplane is defined by inputting the direction vector of the normal.

The **Get workplane parameters** option is used when the workplane is positioned at a location where the cutplane is required. This can be used when the normal and center is not known and is determined visually by positioning the workplane at the required location. **Use View Parameters** can be used to create a workplane which is parallel to the current screen orientation.

The cutplane is used to clip a part of a surface or grid, which aids in viewing its interior. This is especially useful if the region to be meshed lies on the inside of the surfaces. The cutplane dialog box can be opened by clicking on the **Cutplane**  button in the **Workplane Toolbar**. Clicking on the  button next to the **Add Cut Planes** button adds the cutplane in the GL. The check box next in the added cutplane is used to toggle the view of the cutplane in the GL. The **lock** button  is used to lock the current position and orientation of the cutplane. When unlocked, the cutplane can be clicked on and dragged along its normal in the GL. The **flip** button can be clicked to flip the direction in which the surfaces are clipped. The **remove cutplane**  button is used to delete the cutplane.



The **Follow View** check box can be checked to make the workplane always remain parallel to the screen irrespective of the camera rotation and orientation.

There are some **Additional Parameters** such as **Opacity**, **Angle** and **Size** which are defined for each cutplane. The Opacity specifies the transparency of the workplane. The Angle specifies the angle the cutplane is rotated along its axis. This does not change the direction in which the surfaces are clipped, but is instead done for visualisation purposes. **Size** can be used to make the cutplane larger or smaller. All these parameters have sliders next to it which can be moved to change their values.

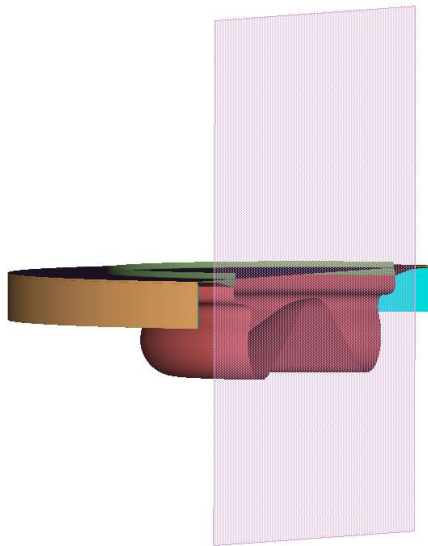



Figure 2.99: Cutplane on a piston bowl

### 2.4.7 Handles

Workplane axes and vertices that are used for translating and resizing the workplane can be toggled using **Handle**  button in the workplane toolbar.

## 2.5 Toggle Surface Section

Most of the time, the user may want to hide multiple surfaces of the topology. Though surface groups can be used for this purpose, sometimes it is easy to toggle the unnecessary surface by scrolling through them. Each surface of the topology has a visibility flag which is used to hide/show the surface. By default, all the surfaces are visible. These flags are used to filter the visible surfaces, and some tools help you to modify these flags internally and help you with a better visualization. Note that the surface groups act as the final layer of filters, i.e. the surfaces which



are not appended to the current surface group or reference group are invisible irrespective of their visibility flags.

Tools under the **toggle surface** section, as shown in Figure 2.100, provide a simple way to scroll through all the surfaces of the topology. The indices or IDs of the surfaces are used to scroll through them. The field shows the *current* surface ID. The *current* surface is defined as the one that is sea blue. *Current* surface is central to corner assignment, and a corner assigned to the *current* surface is shown with a white dot at its centre.




Figure 2.100: Toggle surface section

Next to the surface ID is the Label of the surface. If the label is set to the current surface, the same will be displayed or 'NONE' will be displayed as shown in Figure 2.100. Labels can be set under the settings dialog box (Section 3.4.7) for individual surfaces and a group of surfaces.

The **backward scroll** button  is used to scroll to the previous surface, i.e. the surface whose index is one less than the *current* surface and select it as the *current* surface of the topology. Similarly, the **forward scroll** button  is used to scroll to the next surface (a surface whose index is one more than the *current* surface). Note that when the *current* surface is changed, the index of the new *current* surface is shown at the bottom, i.e. in the **surface ID** field between the scrolls. By providing a surface index or surface ID in this field you can directly make that surface as a *current* surface without scrolling.

Note that when the *current* surface is scrolled, the corners assigned to the new *current* surface are automatically shown with the white dots. When the *current* surface is scrolled, it is always visible irrespective of the visibility flag of the surface. But the visibility flag of the surface is not modified. Hence, the surface remains visible/invisible when the *current* surface is scrolled further.

The *current* surface can be toggled using **toggle**  button next to the forward scroll. When this button is clicked, the visibility flag of the *current* surface is flipped, and the surface is toggled. Note that since the visibility flag of the *current* surface flag is modified, the surface remains hidden/visible when the *current* surface is scrolled further.

Note that the surfaces which are hidden are referred to as *Hidden surfaces*.

## 2.6 Toggle Display Section

As discussed in *section 2.1*, the right-side panel is divided into many sections. Of those sections, the **toggle display** section is one of the common sections present in all the tabs at the top on the right side. You can hide/show the surface, topology, grid, and workplane related objects in the graphics window from the toggle display section. Figure 2.101 shows the toggle display section with various options. By default, it is collapsed; click on the title bar of the section to expand the section.

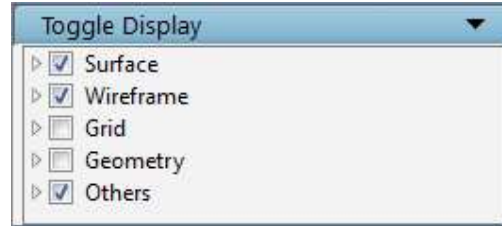


Figure 2.101: Toggle display section

- **Surface**

In the toggle display section, the first branch is **Surface**. By default, surfaces are visible only in the **Surface** tab and **Topology** tab. Sometimes, you may want to hide the surfaces and the surface-related objects while building the wireframe for better visualization, or you would desire to view surfaces to compare them with

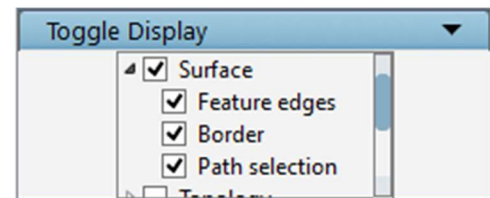


Figure 2.102: Surface sub-menu

the grid. GUI provides two ways of toggling the surfaces, locally or globally. The **toggle display** section helps in toggling globally, while the **toggle surface** section will help in toggling locally as discussed in *section 2.5*. In this toggle display section, there are three sub-branches under the surface branch: **Feature edges**, **Border**, and **Path selection**. The *feature edges* are the lines highlighted on the surface triangulation where the given feature angle exceeds, which is highlighted in baby pink colour. The *border* is the path around the border of the surface that is highlighted in olive colour, and the *path selection* is the path selected on the surface by the user using the path toolbar to segment the surface. These tools are explained in detail under *section 3.4*. Each one is provided with a checkbox that will hide/show the respective option. The surfaces can be toggled by unchecking the surface checkbox, as shown in Figure 2.102. To toggle a particular surface option, expand the surface main branch and uncheck the checkbox of the corresponding option.



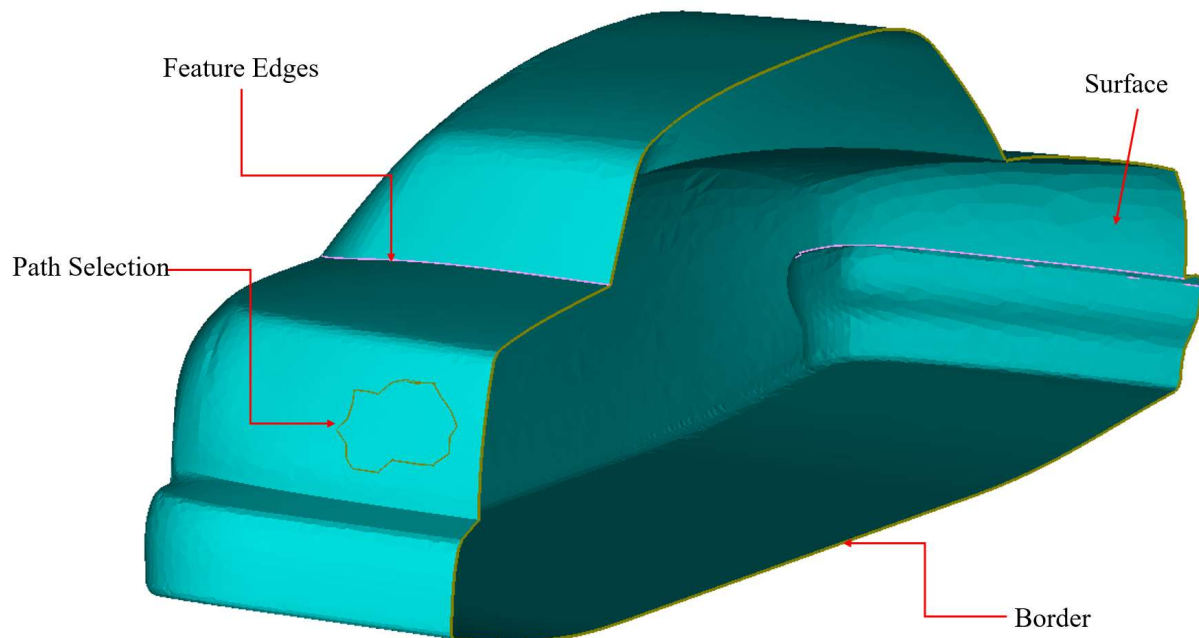


Figure 2.103: Surface entities displayed

- **Topology**

There are various options to hide/show the topology-related objects such as **Wireframe**, **Corner ID's** and **Error** as shown in Figure 2.104. By default, topology is only visible in the **Topology** tab; it is hidden in the rest of all tabs. Sometimes, it might be desired to hide the topology to obtain a better view of its bounding surfaces. The topology can be hidden by unchecking the topology check box. The wireframe can be toggled using the checkbox of the wireframe. When the corners are hidden, their indices (ids) are also hidden. The indices of the corners can also be toggled separately using the checkbox of corner ids. The highlighted edges and face diagonals can be shown/hidden using the option toggle error. **Exclusion** is the line created in pink colour when a face is excluded from the wrap. The face exclusion lines can be toggled by unchecking the Exclusion. **Shrunk blocks** that are shown in smaller shaded boxes can be switched off using the checkbox of Shrunk Blocks. This can also be toggled using the Blocks button in the global dock. Refer to Section 2.3.22 for more information.

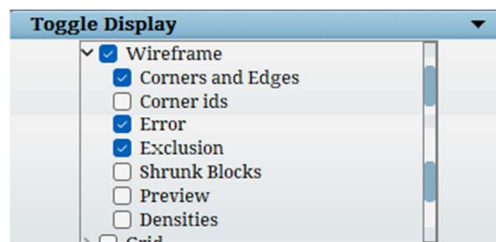


Figure 2.104: Topology sub-menu by default is shown



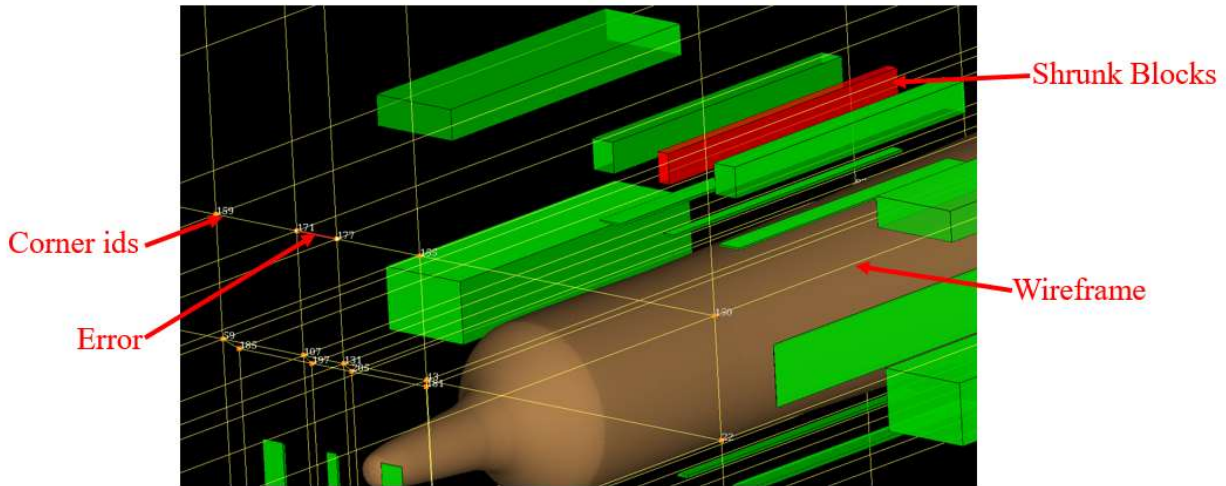


Figure 2.105: Topology entities

Topology errors are automatically detected and highlighted in red when you try to launch an incomplete *.fra* file. The error and highlighted display point precisely to the problem area that prevents Ggrid from running. To remove this highlight, uncheck **Error** under the **Topology** section in the Toggle display section. Errors are displayed only when you attempt to start the gridding process of an incorrect topology. **Preview** is the mesh preview which can also be disabled globally using the Preview checkbox. **Densities** is a colour legend that will plot different densities set in the topology as a gradient of colours. You can click on any particular colour gradient to view the corresponding topology edges. By default, the density legend is switched OFF. Refer to section 4.9.1 for more information on Density legend.

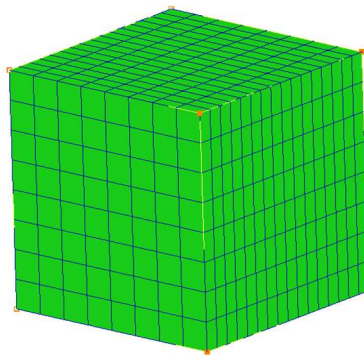


Figure 2.106: Mesh Preview

- **Grid**

By default, the grid is only displayed in the **Grid** and **Property** tab, where it is viewed and analyzed. Sometimes you may want to hide/show the grid and its entities. Unchecking the Grid check box can hide the grid; checking it back can show the same. Similarly, the blocks of the grid can be toggled by checking the box next to **Blocks** under the **Grid** branch.

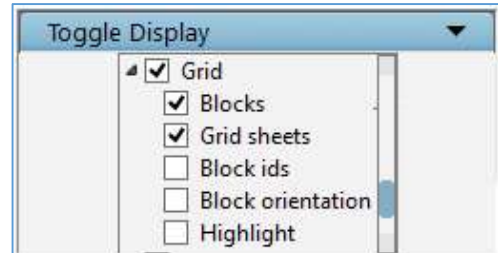


Figure 2.107: Grid sub-menu

Similar to surfaces, each block is given a unique number called block ID, and each block has unique IJK positioning. To see the IJK axes for each activated block, check the **block orientation** under the grid branch in the toggle display panel. The IJK axes will appear near the centre of each block. When the grid blocks are hidden, their IDs and orientation are also hidden. To show/hide the block indices, use the checkbox of **block IDs**. Checking the **block orientation** box can toggle the block orientations. Grid sheets are useful in viewing grid cross-sections, which can be toggled by checking the sheets box. In the grid tab, the spacing tool which gives the distance between two selected nodes will be highlighted with a red line connecting the two selected nodes. Checking the **Highlight** check box under the grid branch can toggle this line.

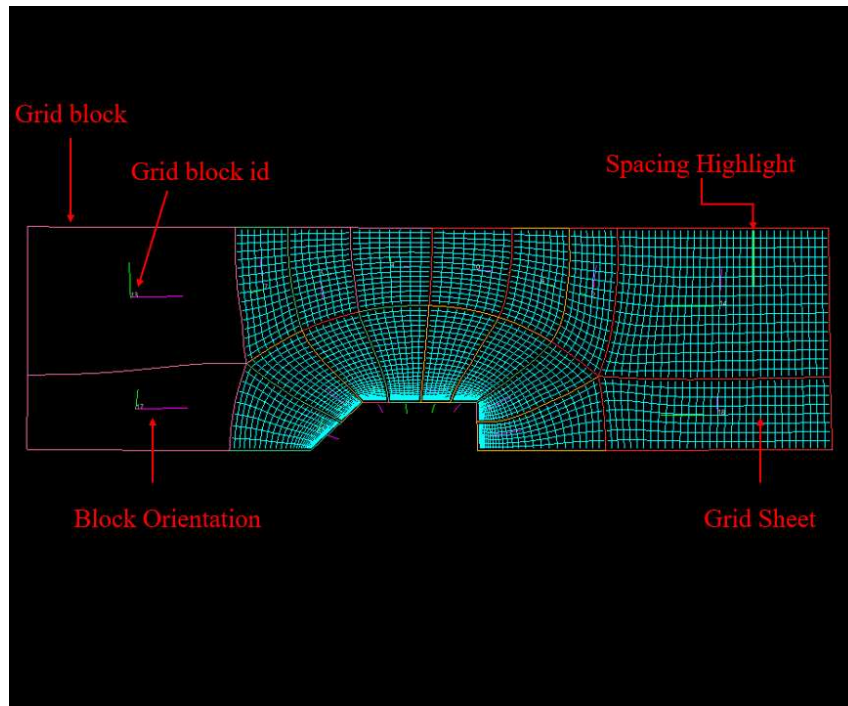


Figure 2.108: Grid entities

- **Geometry**

All the geometry related entities are grouped under the '**Geometry**' branch, as shown in Figure 2.109. **Faces** represent all the face definitions of the geometry; unchecking it will display

only the connectivity lines of the geometry. **Mesh** represents the triangulation or surface mesh of the geometry which can be shown/hidden by checking/unchecking the Mesh checkbox. The connectivity of the faces is represented in green, yellow and red colour depending on the number of faces it is connected to. Checking/unchecking the **Border** will show/hide the connectivity lines. Hiding the connectivity lines displays the geometry in shaded grey colour.



Figure 2.109: Geometry sub-menu

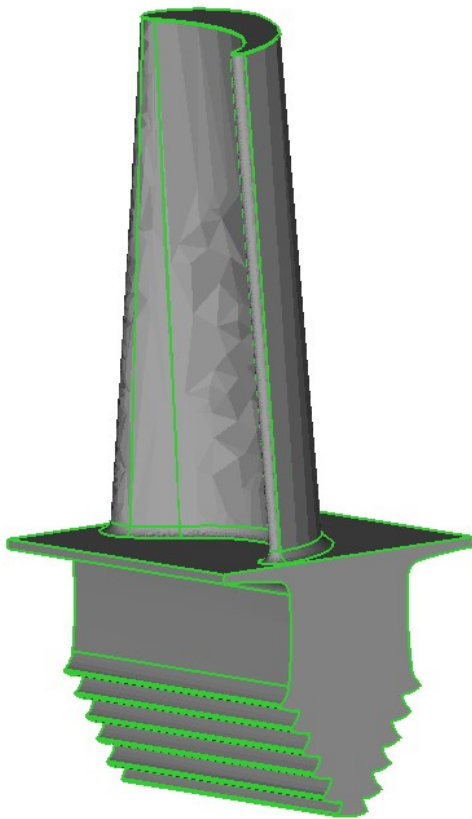


Figure 2.110a: Geometry with Face



Figure 2.110b: Geometry with mesh

- **Others**

Other than the surface, topology and grid entities visible in the graphics window are Global axes, Centre of Rotation axes, Workplane and Workplane axes classified under the 'Others' branch in the Toggle display section. By default, these are visible in both **Surface** and **Topology** tabs

except for the **global axes** and **centre of rotation axes** that are visible in all tabs. The entire workplane can be toggled to avoid screen clutter by unchecking the box of **Workplane** and the **Workplane axes**. Note that workplane axes can be shown only when the workplane itself is visible. The global axes and centre of rotation axes can be toggled by checking the **Axes** check box.

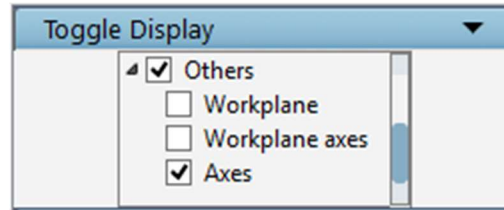


Figure 2.111: Others sub-menu

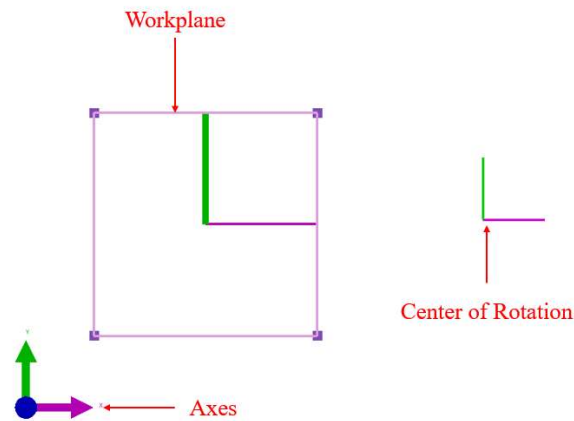


Figure 2.112: Other entities

**NOTE:**

1. If the workplane is used to clip the surfaces or grid sheets and if it is hidden, the clipping would be disabled.
2. Toggling off or on any entity is independent of tabs; if you toggle off the workplane in the surface tab, it would still be visible in Topology or any other tab.

# CHAPTER 3

# SURFACE

The foremost desired objects to generate a grid are surfaces. Before gridding a region, the user must often read on several surfaces. *GridPro* can read surfaces from roughly eight different formats. Surfaces loaded into GUI must be analyzed and repaired if required so that they can be used for grid generation. All operations related to surface restructure and repair are done in the **Surface** tab. Starting with the first tab, i.e. **Surface** tab, where you edit the surfaces or prepare the already loaded surfaces that can be readily used for grid generation. The surface tab provides surface repairing tools, surface probing tools and surface reconstruction tools. Each one is explained in different sections of this chapter.

Before you start working with surfaces, you must know all the surfaces *GridPro* works with. The surfaces in *GridPro* have two kinds of classification: One is based on the format and the other based on the orientation. It is classified into implicit (analytic) surfaces and explicit surfaces based on the formats. It is divided into internal and external surfaces based on the orientation. The surface classification based on formats is explained in detail under *section 3.1*; similarly, the other classification based on orientation is explained under *section 3.2*.

In the **Surface** tab, tools are divided into two sections such as Edit and Check/Repair. Edit section includes those tools which are used for modifying the already loaded surfaces. Check/Repair section includes tools that help to check whether the surface is smooth enough to be used for the gridding process and also performs the modification based on the requirement. In this chapter, all these tools will be discussed.

### 3.1 Surface Types

As mentioned above, Implicit and Explicit surfaces are classified based upon their surface definition.

- **Implicit Surfaces**

*Implicit surfaces* are defined as equal potential surfaces of a scalar-valued analytic function of the (x, y, z) position vector. These surfaces are defined by the equation,  $u(x, y, z) = 0$ . Hardwired into GridPro are four special cases: the *plane*, *cylinder*, *ellipsoid* and *periodic* surfaces. *Plane* refers to as a plane surface. *Cylinder* refers to a cylindrical surface. *Ellipse* and *sphere* refer to as ellipsoid surfaces. *Periodic* surfaces are rotationally symmetric and translationally symmetric surfaces.

- **Explicit Surfaces**

*Explicit surfaces* are defined by parameters. GridPro contains four explicit surface types: the *-linear*, *-quad*, *-tria*, and *-tube* surface types. Explicit *-linear* surfaces are single patch bilinear parametric surfaces or surfaces composed of multiple patches of bilinear parametric surfaces. A *-quad* surface is composed of unstructured quadrilateral elements, while *-tria* surfaces are composed of unstructured triangular elements. A *-tube* surface is a surface of revolution about a cartesian curve.

Except for *-tria* surfaces, all of these implicit and explicit surfaces can be created in GridPro. Implicit surfaces are grouped under the **Surface** button in the **Create** section of the **Topology** tab, as shown in Figure 3.1. The surfaces can be either 2D or 3D. Depending on the dimension of the workspace, the default dialog box opens for surface **Create** options. i.e. plane dialog box for 3D and Line for 2D. The list of surfaces that can be created for 2D is

- Line
- Circle
- Ellipse

and for 3D surfaces are :

- Plane
- Cylinder
- Ellipsoid

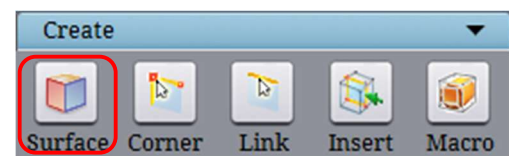


Figure 3.1: Create Section

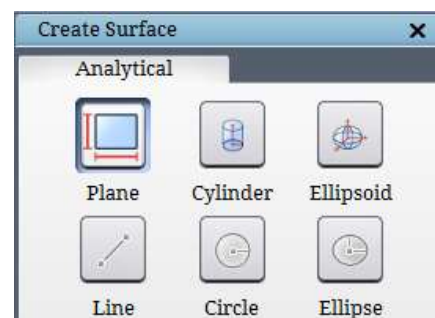


Figure 3.2: Surface creation dialog box

Even though these buttons are self-explanatory, they are explained below, corresponding to their classification. Starting with Implicit 3D surfaces:

### 3.1.1 3D Implicit Surfaces

- **Planes**

Planes are categorized under 3D implicit surfaces. Planes represent one of the simplest surfaces. Planes can be defined either by an equation or by a centre point and a normal direction. The general equation for a plane is  $ax + by + cz - d = 0$ . Planes are inputted in the TIL code according to this equation. However, the plane's centre and normal direction are required in the UI. Like the workplane, surface planes are represented by rectangles. The centre of this rectangle is the point that helps define the plane. The normal of a plane helps to define the plane's orientation. The infinite plane is displayed as a square of finite size for visualization purposes. Apart from the centre and normal, the plane size is also required to create the plane surface. It is just a visualization parameter. The larger the size, the larger the rectangle will appear. To input a



plane, click on the **Plane** button in the *Create Surface* dialog box, as shown in Figure 3.2. By default, the plane related parameters are shown if a 3D workspace is opened, as shown in Figure 3.3.

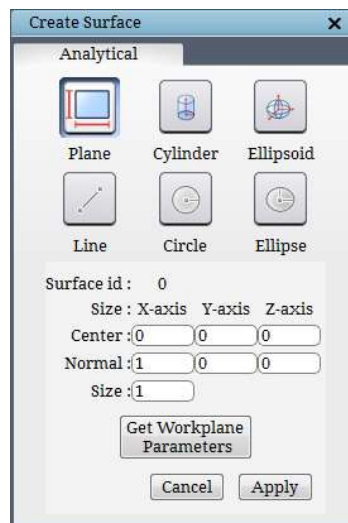


Figure 3.3: Plane dialog box

The surface id is an automatically generated number that starts from 0. It cannot be modified manually. The fields corresponding to the centre, normal, and plane size can be manually edited. The plane normal and the centre field require three real numbers separated by a space representing the value on all three coordinates in the order x, y, and z. The size field requires a positive real number taken as the diagonal length of the plane. After inputting the coordinates for the plane, click on **Apply** to create a plane in GUI. The newly created surface would be the current surface shown in sea-blue colour. And the surface id is displayed at the top of the dialog box as 'Index'.



Figure 3.4 shows two planes created with centre 0,0,0 and 10,0,0. The normal for both the plane is the same, i.e. 1,0,0. The sizes of the planes are 1 and 10. The size of the plane is increased to 10. Plane is an infinite surface, and the size parameter is purely used for visualization purposes.

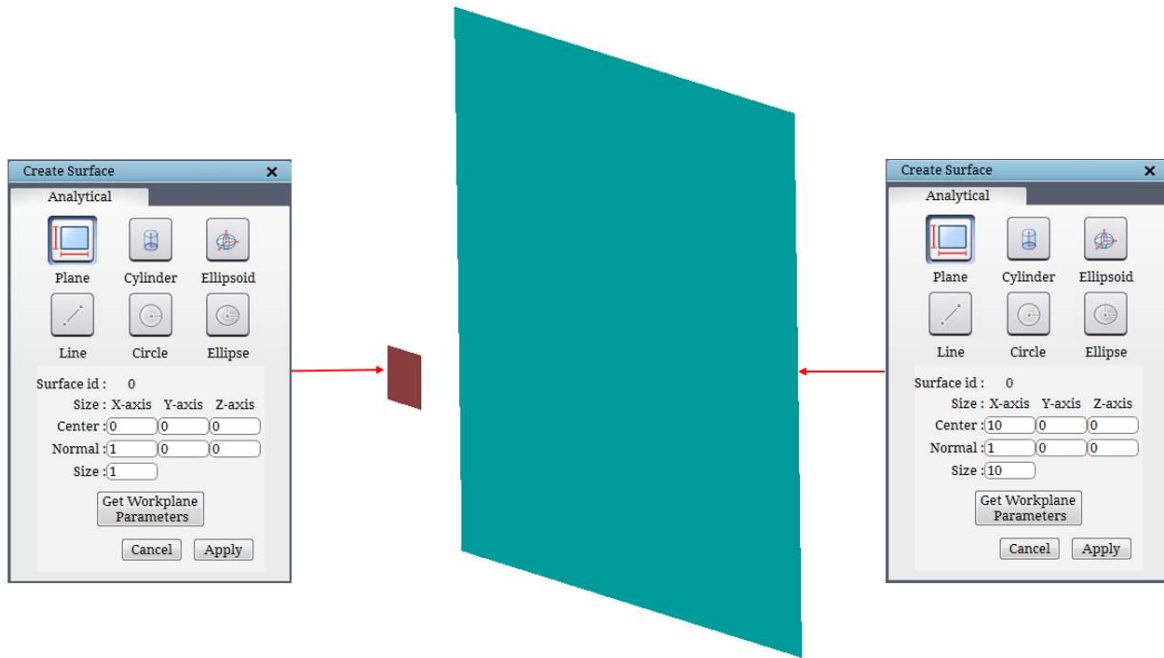


Figure 3.4: Plane with different size

The workplane (of current graphics window) data can also be used to create the plane surface by clicking on the **Get workplane parameters**. A workplane is defined by its centre, normal direction, and size (length \* width). This data is directly copied to the fields corresponding to centre, normal and plane size. Note that the graphics window is active even when this dialog box is open, and hence the workplane can be resized, reoriented if necessary, and then the workplane data can be used.

- **Cylinder**

**Cylinder** is also classified under 3D implicit surfaces. It is a special case of an



ellipsoid whose radius is  $\infty$  along with its axes. When clicking on **Cylinder** button, Create Cylinder parameters appears on the right-side panel, as shown in Figure 3.5. The centre, major and minor axis and radius along those axes, the axis of the cylinder, and power of the cylinder define a cylinder. A cylinder is infinite in its length, but its length (or view scale) is also required for visualization purposes. The radius, power and length field require a single real positive number representing the radius on two axes; power determines the shape or cross-section of the cylinder (Circular, elliptic, etc.) and the length defines the length of the cylinder displayed in the

graphics window. The centre, axis, and the length direction field require three real numbers separated by a space representing the coordinates of x, y, and z-axis for the corresponding input.

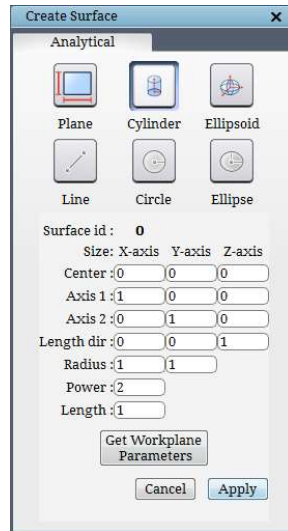


Figure 3.5: Cylinder dialog box

Figure 3.6 shows the effect of length. Centre of the cylinder along the z-direction is the same for both surfaces. Length parameter is changed from 10 to 20.

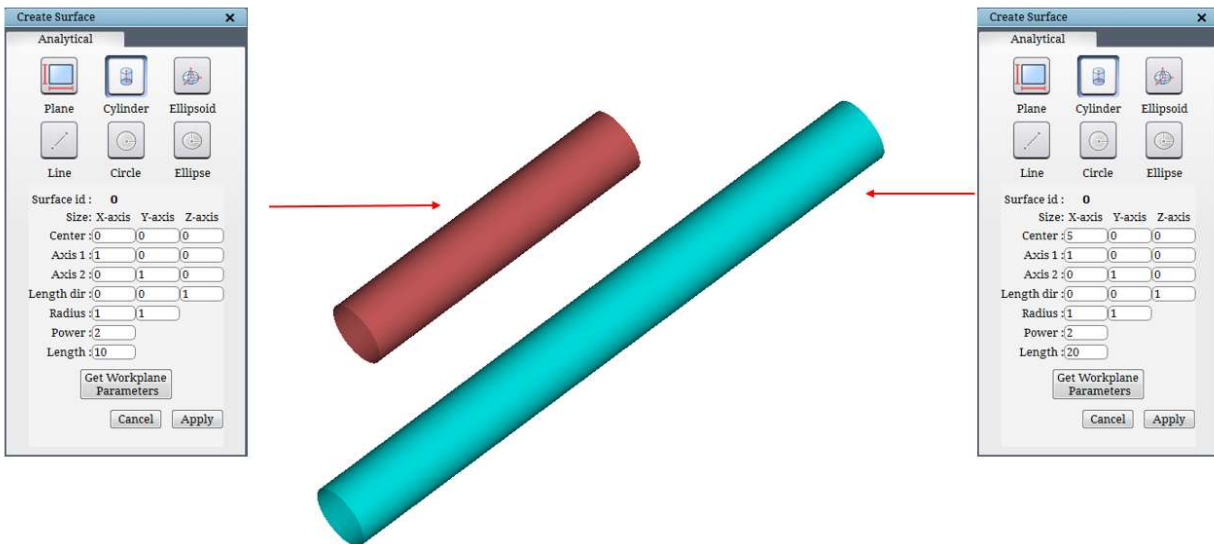


Figure 3.6: Cylinder with different lengths

The workplane data can also be used to create the cylinder surface by clicking on the **Get Workplane Parameters**. The centre of the workplane is copied to the 'centre' field of the cylinder. The workplane X-axis and Y-axis orientations are used as the *major* and *minor* axis of the cylinder and the workplane length & width are copied to the radius fields. The Z-axis of the workplane is used as the direction along the length of the cylinder. The power of the ellipse and length of the cylinder remains unaltered when the workplane data is used.

**NOTE:** The "power" option defines the relationship that provides the shape of the built-in surface. An ellipsoid (includes cylinder, circle, ellipse) is defined by a second-order function that renders an oval shape. A super-ellipsoid is defined by a higher-order function. For a power equal to 2, according to the ellipse equation, a circle would be formed; if power is more than 2, a super-ellipsoid is formed, which looks more like a rectangular prism, as shown in Figure 3.7.

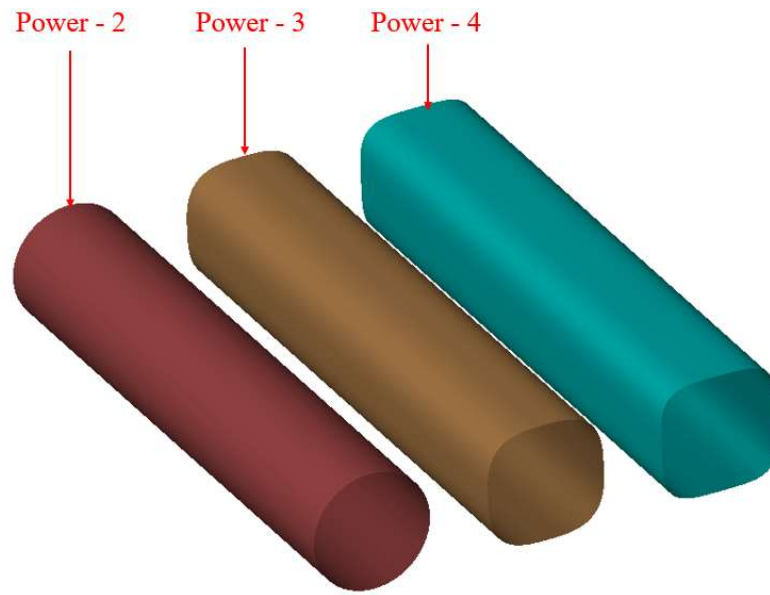


Figure 3.7: Cylinder with different power

- **Ellipsoid**

An ellipsoid is defined by the centre, three axes, their radii and the power of the



ellipsoid. To input an ellipsoid, click on the ***Ellipsoid*** button Ellipsoid in the **Create Surface dialog box**. Figure 3.8 shows the ellipsoid dialog box of the ellipsoid button on the right-side panel. The radius and power field require a single real positive number representing the radius on all three axes, and power determines the shape of the ellipsoid. The centre and axis fields require three real numbers separated by a space representing the coordinates of x, y, and z-axis for the corresponding input.

Surface id : **0**

Size :	X-axis	Y-axis	Z-axis
Center :	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
Axis 1 :	<input type="text" value="1"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
Axis 2 :	<input type="text" value="0"/>	<input type="text" value="1"/>	<input type="text" value="0"/>
Axis 3 :	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="1"/>
Radius :	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="text" value="1"/>
Power :	<input type="text" value="2"/>		

Get Workplane Parameters

Ok Reset Cancel Apply

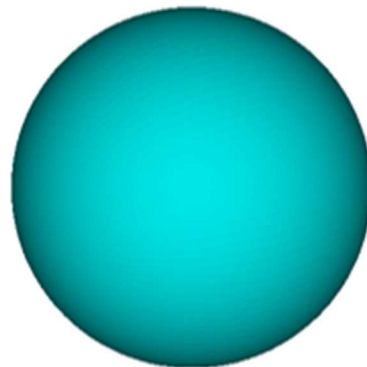


Figure 3.8: Ellipsoid with equal radius in all directions

Surface id : **0**

Size :	X-axis	Y-axis	Z-axis
Center :	<input type="text" value="10"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
Axis 1 :	<input type="text" value="1"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
Axis 2 :	<input type="text" value="0"/>	<input type="text" value="1"/>	<input type="text" value="0"/>
Axis 3 :	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="1"/>
Radius :	<input type="text" value="5"/>	<input type="text" value="3"/>	<input type="text" value="5"/>
Power :	<input type="text" value="2"/>		

Get Workplane Parameters

Ok Reset Cancel Apply

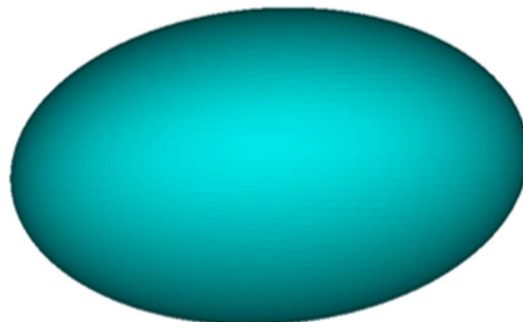


Figure 3.9: Ellipsoid with equal radius in two directions

Surface id : **0**

Size :	X-axis	Y-axis	Z-axis
Center :	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
Axis 1 :	<input type="text" value="1"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
Axis 2 :	<input type="text" value="0"/>	<input type="text" value="1"/>	<input type="text" value="0"/>
Axis 3 :	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="1"/>
Radius :	<input type="text" value="5"/>	<input type="text" value="3"/>	<input type="text" value="1"/>
Power :	<input type="text" value="2"/>		

Get Workplane Parameters

Ok Reset Cancel Apply

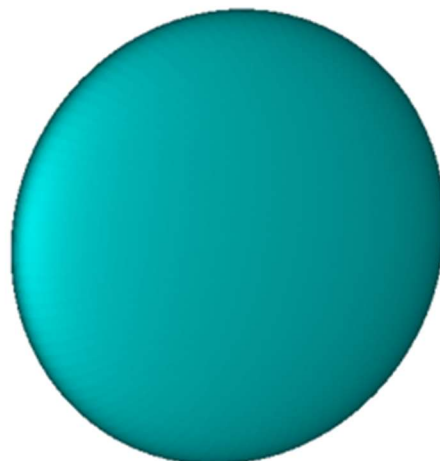


Figure 3.10: Ellipsoid with radius different in all directions

Figure 3.8 shows the ellipsoid created for equal radius in all directions. Figure 3.9 shows an ellipsoid created for equal radius in two directions. Figure 3.10 shows an ellipsoid created with different radius in all directions.

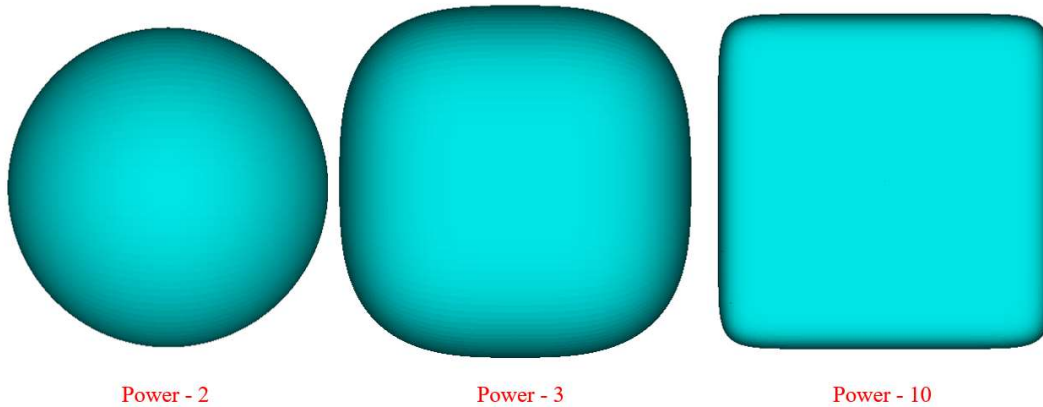


Figure 3.11: Effect of power for ellipsoid on the same parameter

The dialog box of the axis shows the three axes, i.e. axis 1, axis 2 and axis 3. The first column in the Radius field belongs to axis 1, the second column belongs to axis 2, and the third column belongs to axis 3. So, it does not matter if you specify the x-axis direction on axis 1 or axis 2 or axis 3. It will take the values based upon the axis vector you input. For example, Figure 3.12 shows 3 dialog boxes, for which output will be the same, i.e. an ellipsoid with radius 10 in the X direction (1, 0, 0) will be created and radius 5 in the other two, i.e. Y and Z directions. The same ideology applies to cylinder surfaces as well.

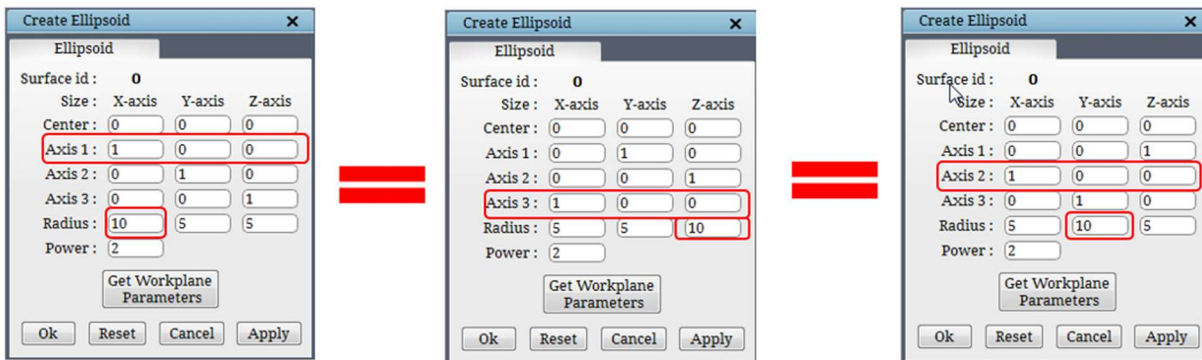


Figure 3.12: Same output for different dialog box


Similar to the cylinder, the workplane data can also be used to create the ellipsoid surface by clicking on the **Get Workplane Parameters**. The centre of the workplane is copied to the centre field, 'centre' of the ellipsoid. The workplane X, Y, and Z-axis orientations are used as the axes of the ellipse, and the workplane length, width, and diagonal length are copied to the radius

fields of X, Y and Z, respectively. The power of the ellipse remains unaltered when the workplane data is used.

- **Symmetrical surfaces**

Some geometries are translational symmetric or rotational symmetric. Rather than computing the whole grid about such an object at once, GridPro can grid a "section" of the object and then duplicate that section to construct the grid about the whole object easily. This process saves time and effort and allows to construct a topology for only a fraction of the region that needs to be gridded.

The required "section" of symmetric objects is created with periodic surface types, called *Rotational* and *Translational*. The *Rotational* surface is used for rotationally symmetric geometries, and the *Translational* surface is used for translational symmetric geometries. These surfaces comprise nothing more than the periodic borders of the object. The grid generated within the given period can be used to generate a complete grid across the periodic boundaries. Similar to any other surface, appropriate corners must be assigned to these surfaces. Unlike other surfaces, the user has to use the

**Periodic Assignment**  button in the surface assignment section (refer to section 4.10.4) from the right-side panel of the topology tab to create and

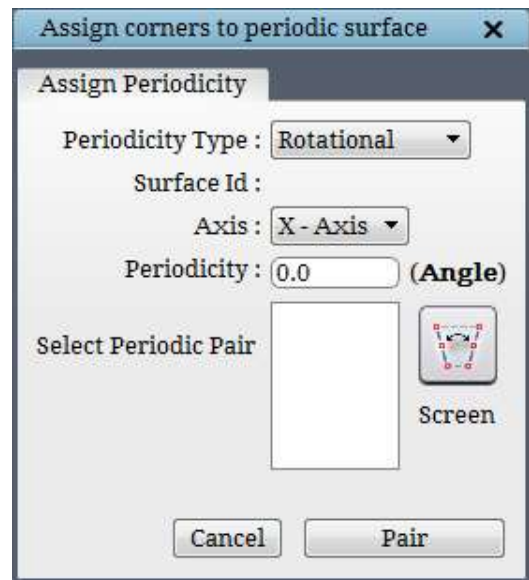



Figure 3.13: Periodic surface dialog box

assign corners to the periodic surfaces. Once the **Periodic**  button is clicked in the **Surface Assignment** section of the Topology tab, it shows the dialog box as shown in Figure 3.13.

- **Rotational**

A rotational periodic surface is defined by its periodicity, orientation, and centre. When the **Periodic** button is clicked, the dialog box opens as shown in Figure 3.13. Origin is taken as the default centre of the surface. The orientation of the axis can be changed from the Axis drop-down list. The centre and orientation can be modified by the transformation of the surface. In general, the periodic surface requires a transformation for its appropriate orientation (Transformation of surfaces is discussed in section 3.4.9). Hence, two parameters are required for generating a default Rotational Periodic surface (which passes through the origin of the global coordinate system): the periodicity of the surface and axis alignment. The periodicity of the surface should be a factor of 360. The periodic surface can be used only once in a topology



file. The rotational periodic surface doesn't have any representation, but in GUI, it has been displayed as a Torus shape for convenience. The axis extending through the torus centre hole is the periodic axis. The torus serves solely as a visual aid and in no way represents the physical properties of a surface. The torus can be scaled to any size using transformation.

Figure 3.14 shows the periodic 2d topology with assignments highlighted for the periodic surface. The periodicity for the shown topology is taken as 60.

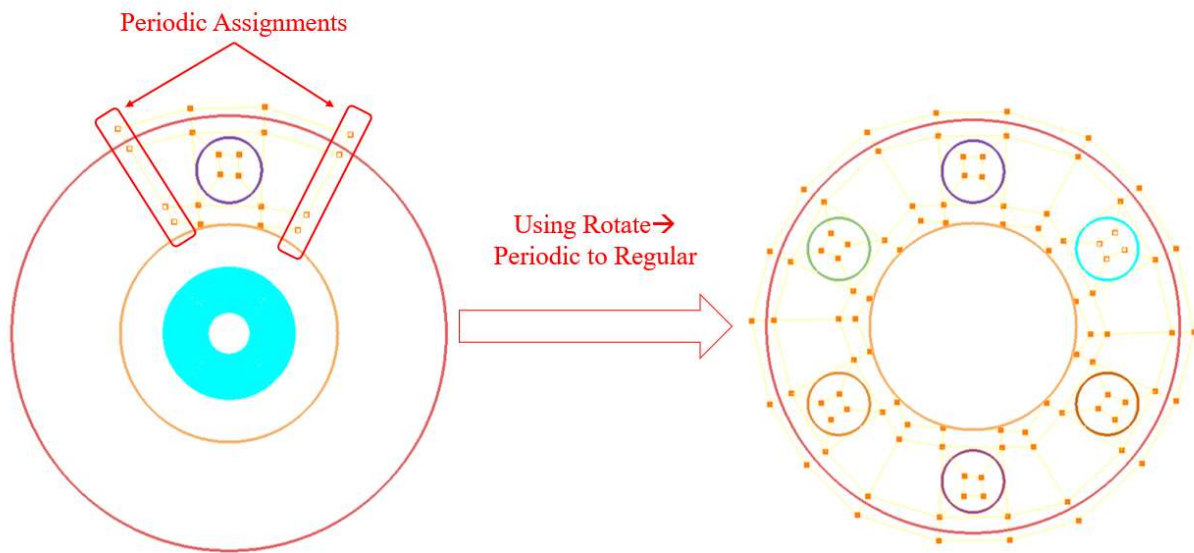


Figure 3.14: Using Rotate option to rotate the Periodic Topology

A complete periodic topology can be used for generating a full topology based on the periodicity specified as shown in Figure 3.15 by using the **Transform → Rotate** option in the Topology tab.

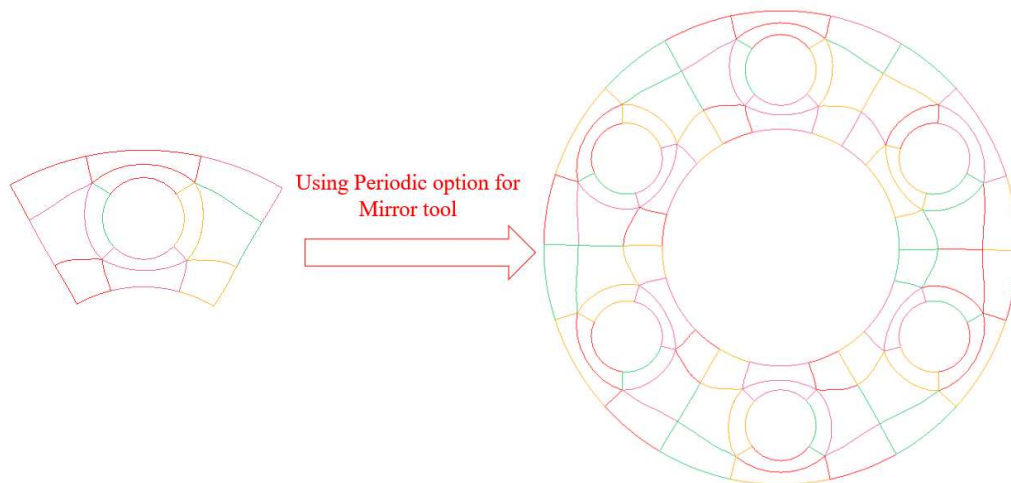


Figure 3.15: Using Mirror option to generate a full grid



A full grid can be generated using the periodic grid as shown in Figure 3.15 by using the Periodic option specified under the **Mirror** tool in the Grid tab.

- **Translational**

A translational periodic surface is used to create a grid with linear periodicity. The grid generated within the given period (distance), can be used to generate an infinite grid across the periodic boundaries. It is defined by its periodicity (which corresponds to the length of the section) and orientation. Its centre is the origin of global system coordinates, which can be transformed using the transformation option (discussed later). Figure 3.16 shows the dialog box of the translational periodic surface. Similar to Rotational, the Translational object also doesn't have a representation. Still, GridPro displays it as a plane with an arrow representing the periodic side in case of 3D and as a straight line in case of 2D for convenience. The periodicity is on the axis perpendicular to the line/plane.

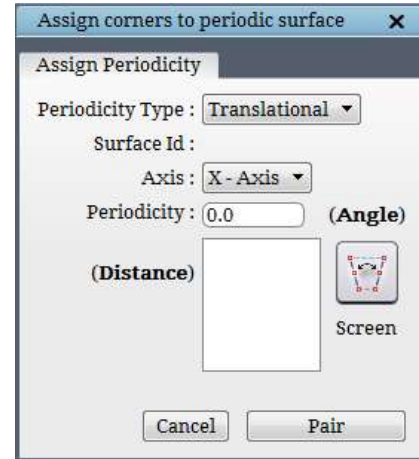


Figure 3.16: Periodic surface XYZ tab

**NOTE:** Both the periodic surfaces are defined in the same way irrespective of their dimension (2D or 3D). The surface assignments for a periodic surface is discussed in section 4.10.4

It is not advisable to use a V-shaped or wedge-shaped structure for a 180-degree periodicity. In such a scenario, two faces of the same block are assigned to the periodic surface, which forms 180 degrees. Hence it becomes a highly skewed grid.

### 3.1.2 Implicit 2D surfaces

Similar to implicit 3D surfaces, various 2D implicit surfaces are available in UI. All 2D built-in surfaces can be seen in Figure 3.2.

- **Line**

The very first surface is **Line**. It is similar to a **Plane** in 3D. It is infinite and is defined by a centre and normal. For visualization purposes, the infinite line is displayed as a line segment of




finite length depending upon the size defined by the user. When the **Line** button is clicked, the Create Line dialog box appears as shown in Figure 3.17. Note that 2D topology lies on the XY plane, and hence the centre and normal fields require only two coordinates (x and y) to represent the line. The size field requires a single real positive number that defines the length of the line.



- **Circle**

**Circle** is defined by its centre, radius, and power. Click on



the **Circle** button  in the **Create Surface** dialog box. Figure 3.19 shows the **Create Circle** dialog box which appears on the right-side panel. The centre field requires two real numbers separated by a white space representing the centre of the circle and the radius. The power field requires a single positive real number representing the radius of the circle and the shape of the circle. As discussed earlier (refer cylinder), if the power is more than 2, the shape would look more like a rectangle.

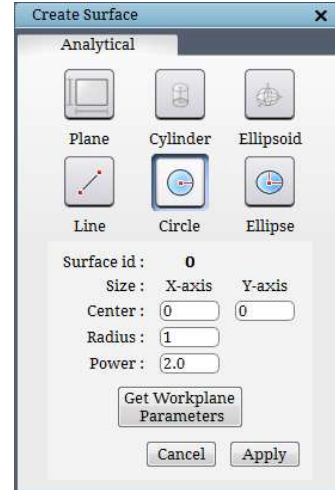


Figure 3.19: Circle dialog box

Power value affects similar to cylinder and ellipsoid as shown in Figure 3.20.

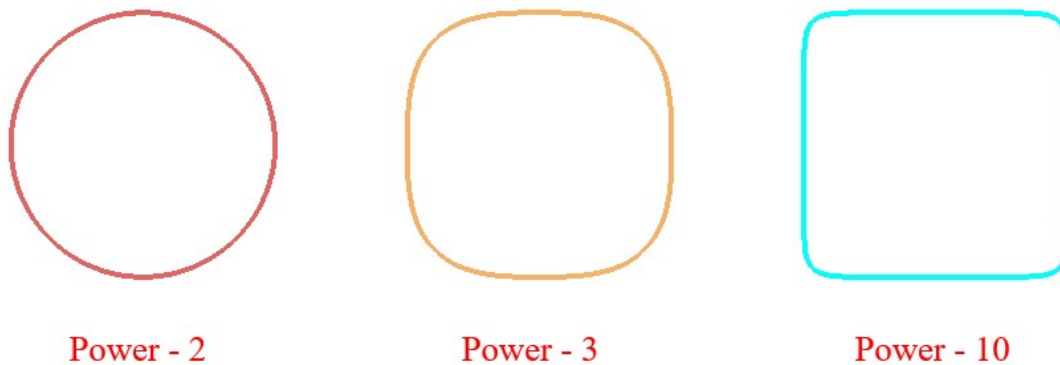



Figure 3.20: Effect of power value in cylinder

The workplane data can also be used to create the circle surface by clicking on the **Get Workplane Parameters**. The centre of the workplane is copied to the 'centre' field of the circle. The workplane diagonal length is copied to the radius of the circle.

- **Ellipse**

**Ellipse** is similar to ellipsoid in 3D. It is defined by its centre, major and minor axis and radius along their respective axes and the power of the elliptic surface. Clicking on **Ellipse**



button  in **Create Surface** dialog box opens ellipse parameters, as shown in Figure 3.21. As discussed earlier, the centre, the major axis and the minor axis fields require two real numbers each (representing the x and y coordinates/components) separated by white-space and the radius

& the power requires a single positive real number representing the radius along the X and Y axis and the shape of the ellipse.

Similar to other surfaces, the workplane data can also be used to create the ellipse surface by clicking on the ***Get Workplane Parameters***. The centre of the workplane is copied to the centre field of the ellipse. The workplane length (along the workplane x-axis) is copied to the major axis radius field. The workplane width (along the workplane y-axis) is copied to minor axis radius field. If both the axes are perpendicular to each other and the radii are equal, the ellipse is equivalent to a circle.

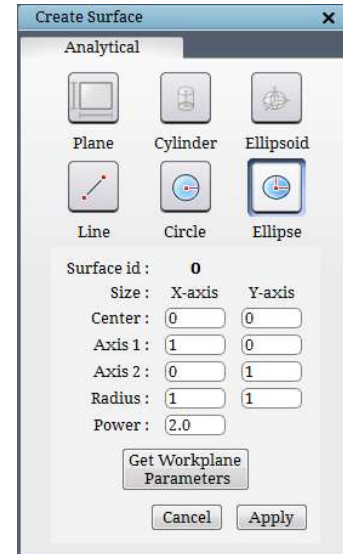


Figure 3.21: Ellipse dialog box

### 3.1.3 Explicit surfaces

GridPro can create a few explicit surfaces as well. Explicit surfaces which can be readily used in GridPro are in ***quad, tria, tube*** or ***lin*** format. Except for ***tria*** (triangular), the other three can be created using the options available under **Surface Tools** in the top menu bar.

- **Tube**

A ***tube*** can be defined by the revolution around a centre curve. The revolution for each point on the centre curve is performed in the normal plane of that point with respect to the tangent of the curve. Hence a ***tube*** is represented by a set of points on the curved centre line or axis and a variable/constant radii at all the cross-sections of the tube. Surface creation tools are discussed in *section 4.11*.

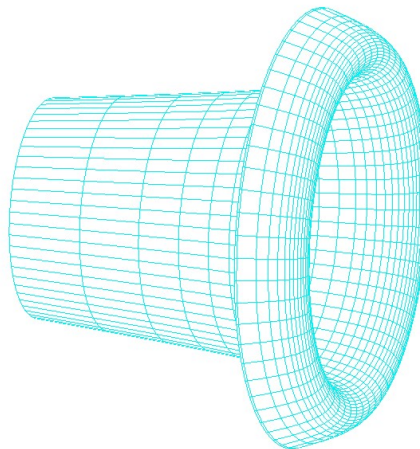


Figure 3.22: Tube Surface

- **Quad**

Another 3D explicit surface that can be created in GUI is *quad* surface. A *quad* surface is composed of unstructured quadrilateral elements. 4 points define each *quad* element. The *quad* files in GridPro is represented by *.quad* as an extension. A *quad* file consists of the total number of points, coordinates of each point, number of quads in the surface, the node numbers of the quads and the property id of the surface if it has. The positive side of the quad surface is determined by the first quad in the quad list. It is the side where when one faces the first quad, the listing order of the points that define the quad rotates anti-clockwise. Figure 3.24 shows an example of a quad surface with two quads.

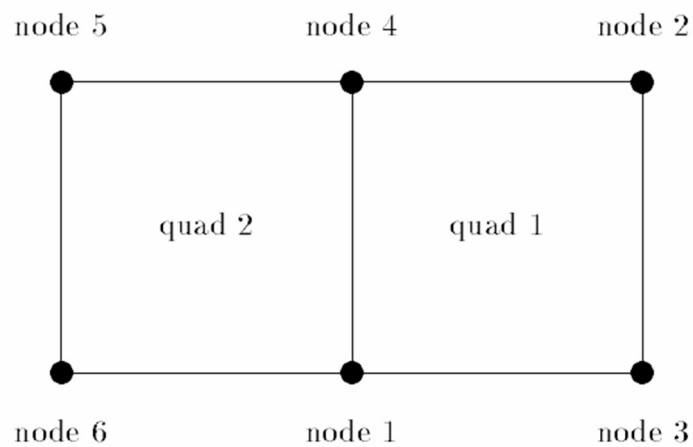


Figure 3.23: Surface with two quads

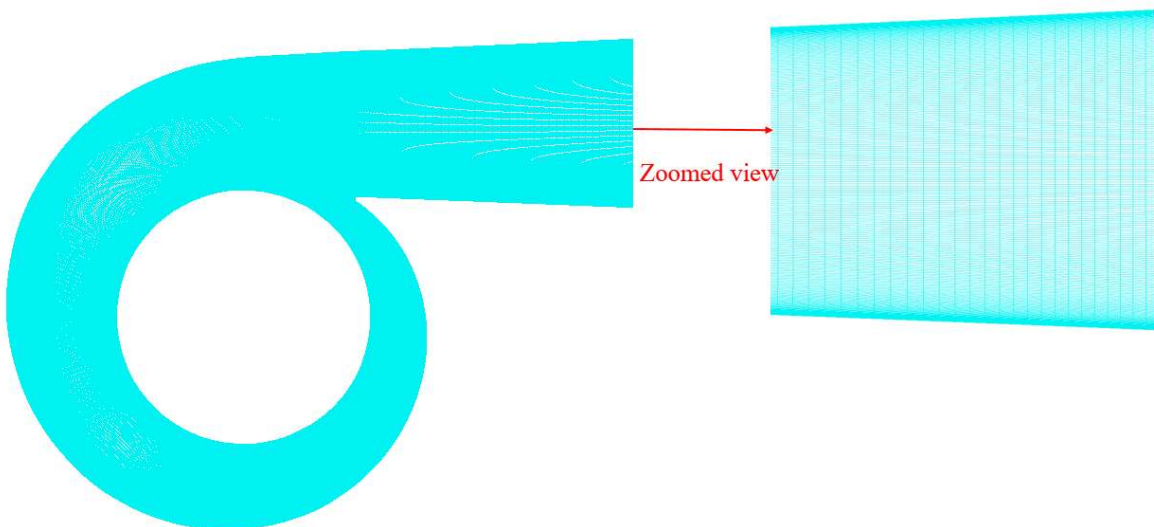


Figure 3.24: Quad surface

- **Lin**

The only explicit 2D surface that can be created in UI is in *.lin* format. *.lin* are linear surfaces defined by an  $I \times J$  array of points for some  $I$  and  $J$  in the file. The bilinear interpolation determines the surface points within the array cells. No collapsed or merged data points are allowed. It is represented either by *.dat* or *.lin* as an extension. In *GridPro*, the curves generated by the controlnet curves option in 2D built-in surfaces panel are saved in *.lin* format. Apart from that, *.dat* and *.lin* files which are generated from the CAD packages, can also be loaded into *GridPro* with some modifications in the file, i.e. the total number of points should be mentioned in the first line of the file. Figure 3.25 shows an example of a linear file with 6 surface pieces.

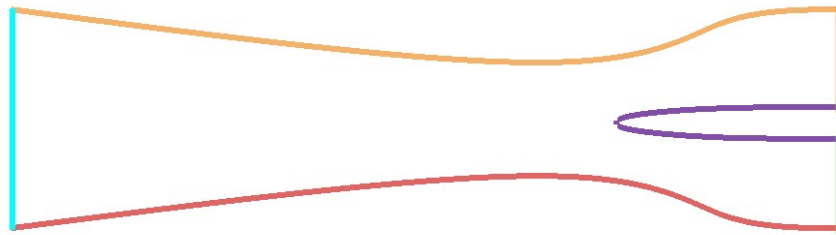


Figure 3.25: Linear file with 6 surface pieces

- **Tria**

*Tria* is the only explicit 3D surface that cannot be generated in GUI, but it can be loaded into GUI. A *tria* surface is composed of unstructured triangular elements. Each triangular element is defined by 3 points. The *tria* files are represented by *.tria* as an extension. A *tria* file is similar to a quad file that consists of the total number of points, coordinates of each point, number of triangular elements in the

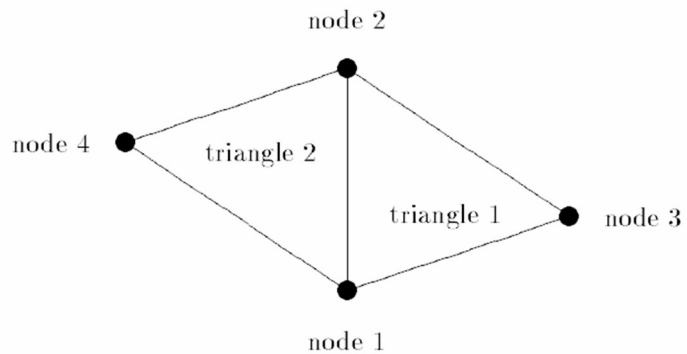


Figure 3.26: Tria surface with two triangular elements

surface, the node numbers of the elements and the property id of the surface. No degenerated sides are allowed for any of the triangular elements. Figure 3.27 shows an example of a tria surface with two triangular elements. *Quad* files generated in GUI can be converted into *.tria* files using some conversion tools. In addition to that, the geometry formats such as *step*, *stl*, *iges*, etc., are converted to *.tria* files using the conversion tools before it is loaded into the GUI.



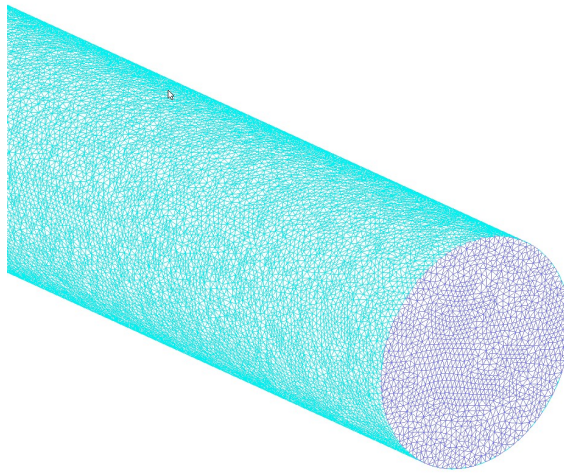


Figure 3.27: Tria surface

## 3.2 Surface Orientation

As mentioned earlier, surfaces can be classified based upon orientation. The **orientation** defines the direction of the normal vector to the surface. The orientation of a surface determines the region to be gridded, either in the region outside the surface, inside the surface, or both. Thus, the surfaces are classified as *external* and *internal* surfaces.

The *external* surfaces have a grid only on one side, which are either inside the surface (internal flow geometry) or outside the surface (external flow geometry). The external surfaces act as the boundary surfaces for a grid. It does not have a grid on the other side.

The *internal* surfaces are surfaces that have grid blocks on both sides. i.e. surfaces that are defined inside the grid to represent the fluid/solid regions or to enhance the grid lines which have a grid on both sides. Thus, at that point, there would be two normal vectors: one pointing to each side. So, surfaces that have a grid on both sides are internal to the grid and, as such, are quite naturally called internal surfaces.

All the implicit and explicit surfaces except the periodic surfaces can be used as internal or external surfaces, depending on the need.

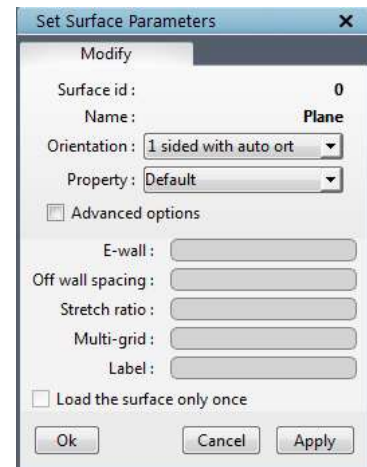


Figure 3.28: Surface Settings dialog box



Orientation can be changed using the **Settings** button. The dialog box of the settings button is shown in Figure 3.28. The drop-down next to the **Orientation** heading appears with 8 options when clicked is shown in Figure 3.29. The items are a combination of "+ side", "- side," 'auto' and '1side' & "2 sides".



- 1 sided with auto ort
- 1 sided with +ve ort
- 1 sided with -ve ort
- 2 sided with auto match
- 2 sided with +ve match
- 2 sided with -ve match
- 2 sided without match
- 2 sided with +- match

Figure 3.29: Surface orientation

Each item dictates the direction of all vectors normal to the surface. The + direction is outward, and the – direction is inward. If a surface is oriented in the "1 sided +ve" direction, the normal vectors at each surface node point to the "+side" of the surface. If a surface is oriented in the "1 sided -ve" direction, the normal vectors at each surface node point to the "-side" of the surface.

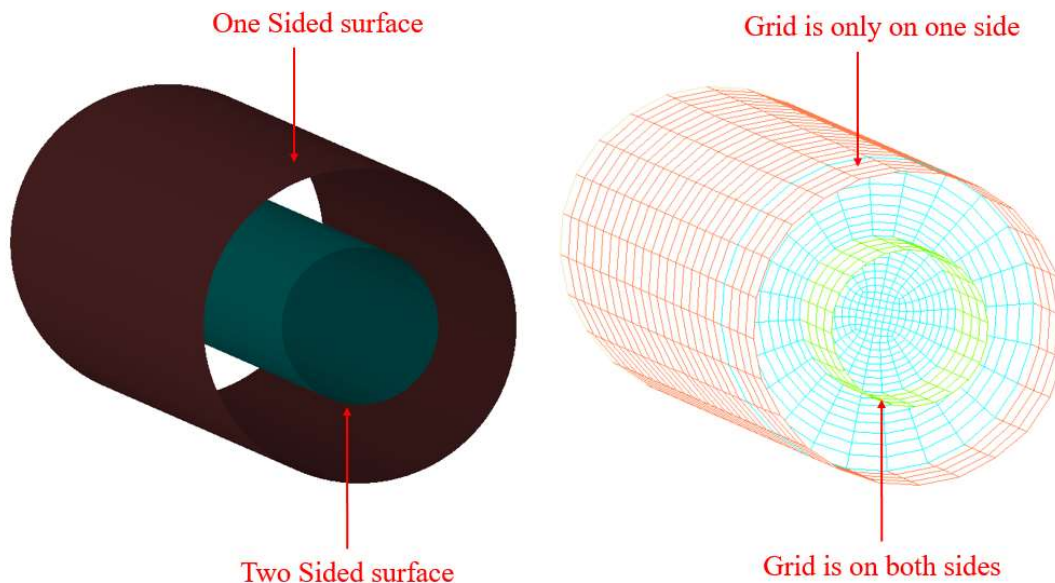


Figure 3.30: Surface orientation with the grid

Similarly, if a surface is given '1sided auto-match' orientation, GridPro will automatically calculate which side has to be meshed using the topology positions of its assignments. To impart a double orientation to the surface, select '2 sided with auto-match'. In the grid generation process, GridPro will optimize the grid quality and produce a grid sheet that is smoothly embedded within the overall grid, and it would be hard-pressed to find any discrepancy.

## 3.3 Surface Probing Utilities

Geometries created using CAD software are sometimes not perfectly suitable for CFD analysis as they may be having some leaks or patches. Sometimes geometry is perfectly fine but requires some modifications to be readily used by grid engines. The GUI contains operations allowing you to manipulate the GridPro-read surfaces.

In the surface tab, two tools are classified under surface probing tools, namely **Border** and **Angle**. Both the tools are located under the **Check/Repair** section in the Surface tab.

### 3.3.1 Border



**Border** button is used to visualize the boundary of the current surface. It will place an *olive* colour path around the entire border of the current surface. By switching off, the button will remove the same path. The border for the current surface is shown in Figure 3.31.

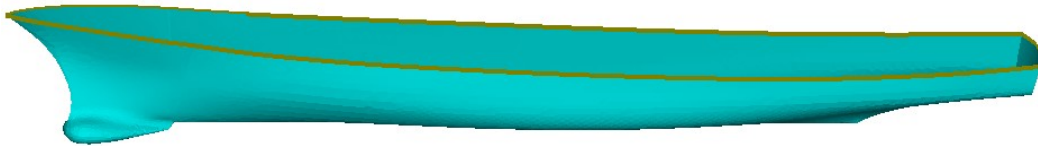


Figure 3.31: Surface boundary (border)

### 3.3.2 Angle

The **Angle** operation illustrates the surface cell's levels of significance. The angle between the normal of the two planes slanting off from a surface cell determines the cell's level of significance. The smaller the angle, the larger the significance level. The **Angle** tool is available with two options: incremental and above the specified. By the default, **Angle** tab is displayed, as shown in Figure 3.32,



when the **Angle** button is clicked.

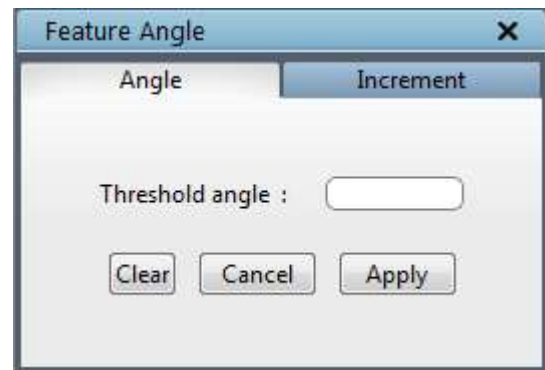


Figure 3.32: Angle tab dialog box

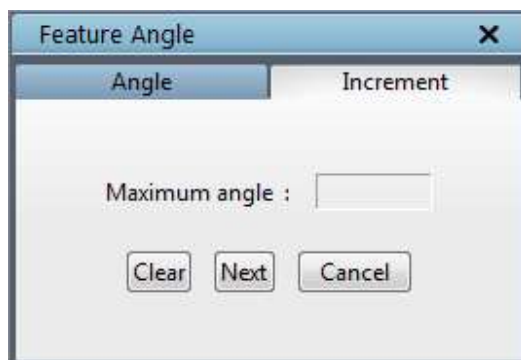


Figure 3.33: Increment tab dialog box

Once the threshold angle is given, it calculates the angle in all the surface nodes and creates a path in baby pink colour on the current surface, wherever the cells have an angle more than the specified value. To

generate a smooth grid, a feature angle of more than 45 degrees is not desirable. Another option is the **Incremental**, available in the Increment tab as shown in Figure 3.33. Press **Next** to display the path and maximum feature angle of the current surface. Every time you press **Next**, all cells with the next (lower) level of significance will be highlighted with baby pink colour paths and the angle value is shown in the maximum angle field. Additionally, a message will appear at the status bar specifying the angle, as shown in Figure 3.34. To delete the paths created by the **Angle** operation, press the **Clear** button.

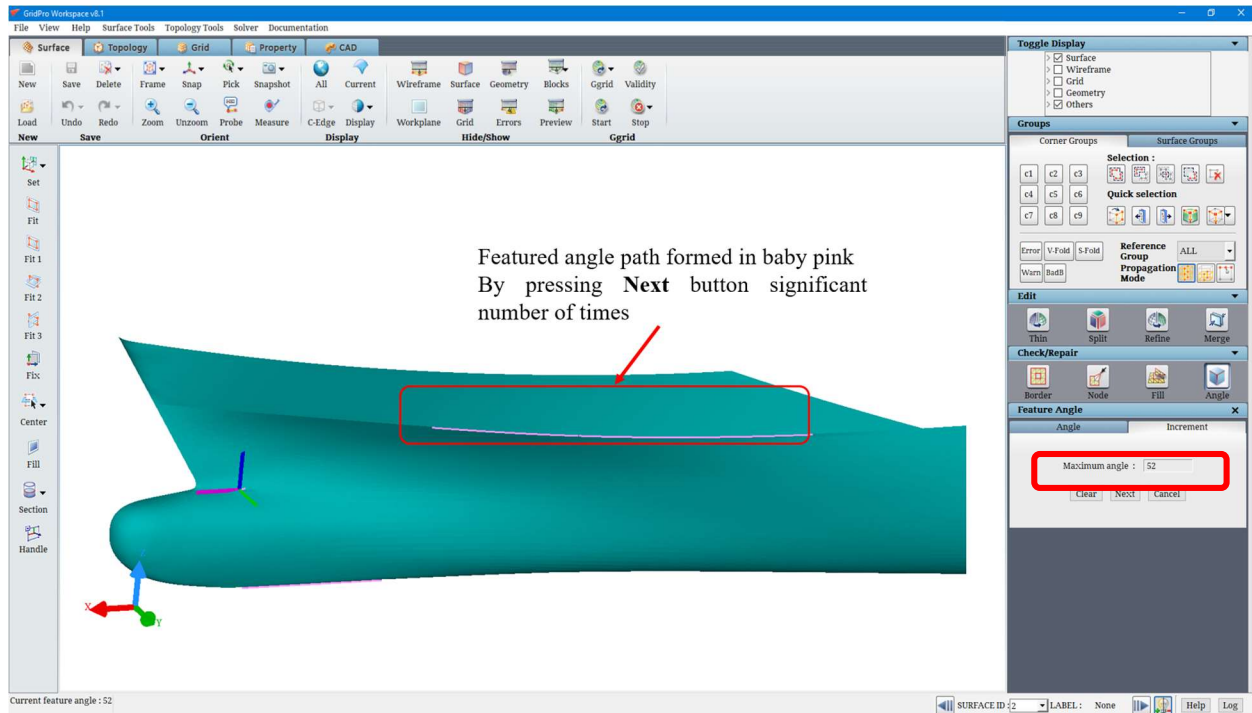


Figure 3.34: Feature angle path formed on the Current surface

### 3.3.3 Surface Mesh Quality

The foremost desired objects to generate a grid are surfaces. The quality of a surface mesh heavily influences the ability of the grid smoother to generate good quality meshes. This will enable the user to understand the quality of the surface and can check if the surface needs to be re-triangulated. The **Surface Mesh quality** dialog box can be opened from the **Surface Tools** drop down in the menubar.

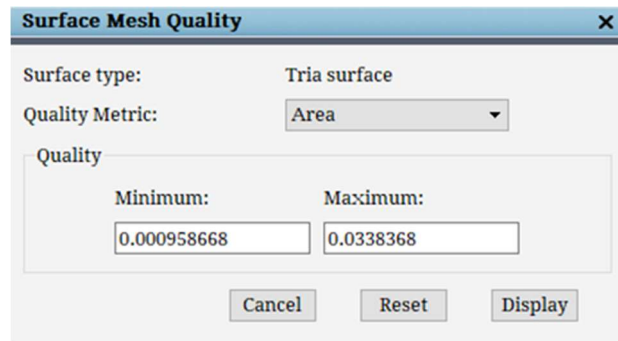


Figure 3.35: Surface mesh quality dialog box

Let's take a look at the various Quality Metrics available:

**Area** - It calculates the surface area of each of the triangulations, and shows from the least area to the maximum area.

**Aspect Ratio** - Aspect ratio is the measure of stretching of a cell. It is calculated by dividing the long edge of the triangle by the short edge.

**Aspect Frobenius** - The Aspect Frobenius is the sum of the edge lengths squared divided by the area and normalised. A unit equilateral triangle has a value of 1.

**Minimum Angle** - The minimum angle of edges of all the triangles.

**Maximum Angle** - The maximum angle of edges of all the triangles.

**Edge Ratio** - The edge ratio is calculated as the maximum edge length by minimum edge length.

**Scaled Jacobian** - The Jacobian ratio measures the deviation of an element's shape from an ideally shaped element.

**Gaussian Curvature** - Gaussian curvature is a curvature intrinsic to a two- dimensional surface. Gaussian curvature is a measure of how curved or warped a surface is at a specific point. It quantifies the amount by which a surface deviates from being flat at that point.

**Mean Curvature** - The Mean curvature is a measure of the average curvature of a surface at a given point.

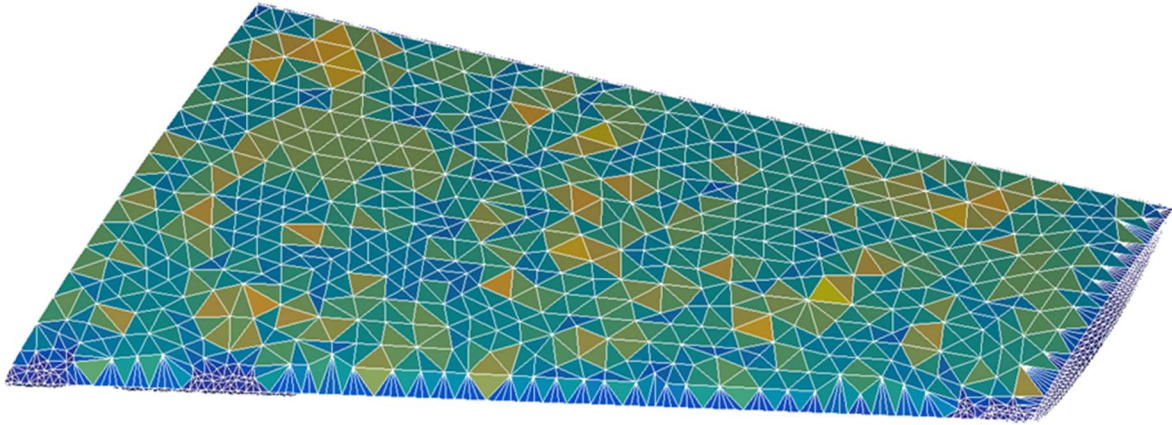


Figure 3.36: The surface mesh quality colour display.

Once we select the required quality metric, we can select the **Display** option. The colour of the current surface changes. The values inside the **Minimum** and **Maximum** text boxes can be manually changed if the user needs to have a closer look only at the cells which have a quality metric within a certain range. Clicking on **Display** after entering the new values to be checked regenerates the colour view on the surface according to the new range of values. A legend is also displayed in the GL showing the colour against the quality metric.

## 3.4 Surface Reconstruction Utilities

GridPro has a specific definition of a surface. So the user has to modify a read-in surface to adapt it for use on GridPro. GridPro contains features in the form of CAD functionalities that allows performing this modification. All the tools used for surface reconstruction or surface modifications are grouped under the **Edit** section of the Surface tab in the GUI. A set of 3 buttons are positioned in the Topology tab under the **Edit (surface)** section. Since these 3 buttons are often used, they are positioned in the Topology tab to provide a better workflow. Some tools are also present in the **Surface Tools** menu.

### 3.4.1 Thin

Sometimes, geometries loaded into the GUI have fine triangulations, which will slow down the GUI performance. In such cases, the number of triangulation has to be reduced without compromising the surface shape in order to increase the GUI performance. **Thin** button helps to reduce the number

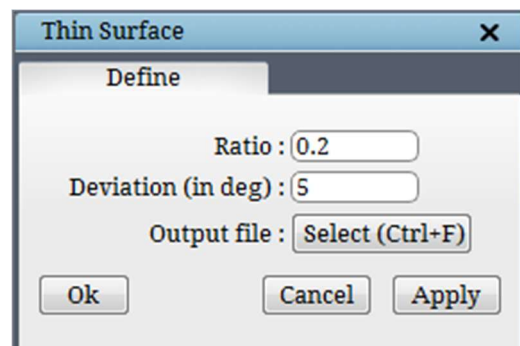



Figure 3.37: Thin dialog box



of triangulations based on angle or ratio. **Thin**  button is located in the right-side toolbar under **Edit** section in the **Surface** tab. If clicked, a dialog box opens, as shown in Figure 3.37. In the *Deviation* field, specify the maximum allowable surface cell angle. In the *Ratio* field, specify the ratio by which the number of cells must be reduced. Any one of the options can be specified; if both are specified, it will try to reduce the number of cells iteratively to satisfy both conditions. Figure 3.38 shows a surface with and without **Thin**. While thinning the surface, ensure that the surface shape is not lost; reduce the angle/ratio to avoid losing the surface shape.

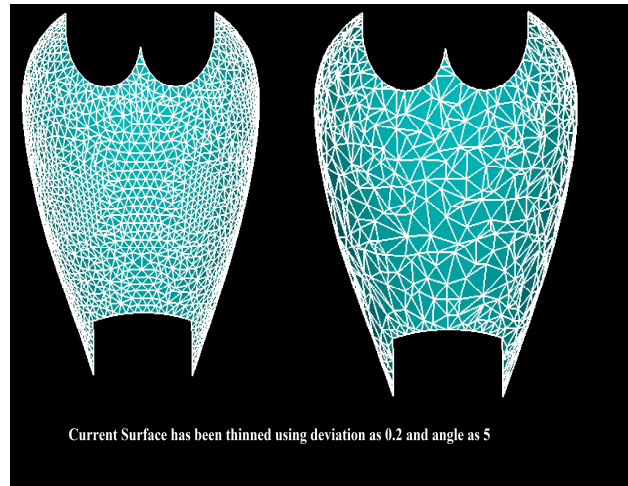


Figure 3.38: Before and after applying thinning tool

### 3.4.2 Split

Any geometry loaded into the UI would be a single surface by default. The surface has to be split into different pieces to capture all its features, such as sharp edges. GridPro has different ways of splitting, of which the first option is 'Split disjoint surfaces'.

- **Split disjoint surfaces**

If the loaded surface has disjoint surfaces and yet is represented as a single surface, the surface has to be split using '**Split disjoint surfaces**' first. This would split the disjoint parts of the geometry into different surfaces. **Split disjoint surfaces** is located under the **Surface Tools** menu. Mostly while loading the surface, it prompts the user with a dialog box as shown in Figure 3.39 to split the surface. Clicking on 'Continue' will load the disjoint pieces as a single surface. You can also use this tool on the active surface group to split it after loading the surface; hence, the surface that has to be

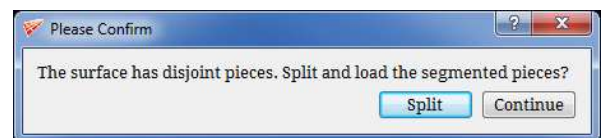


Figure 3.39: Dialog box for split disjoint surface

split should be grouped under a surface group and switched on while executing the command. Refer to *section 4.4* for more information on groups.

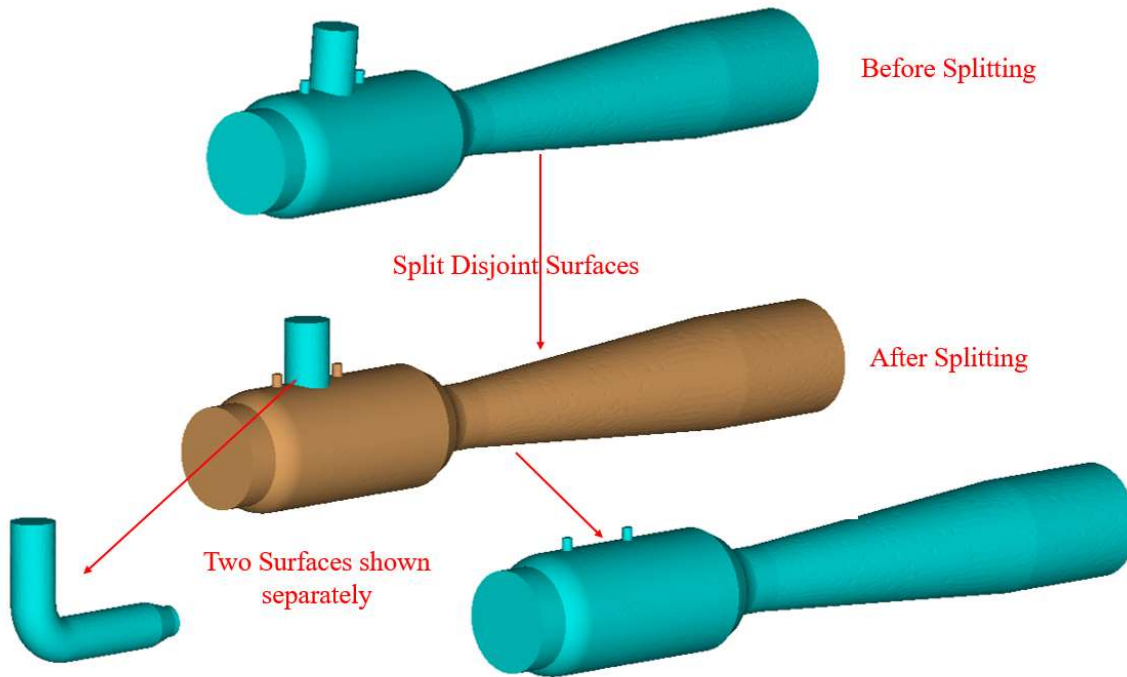



Figure 3.40: Before and after splitting disjoint surfaces

- **Feature Angle**

**Angle** button in the **Check/Repair** section helps to find out the surface cells angles. If the feature angle is more than 45 degrees, it has to be repaired to avoid a skewed grid. If the path formed by the feature angle is closed, it can be split using the **Split** button under the **Edit** section. **The split** operation offers three options; one is based on the path formed by **feature angle**, the other is a path created manually called **Sketching**, and the last one is by trimming the surface cells called **Selection**.

In **Feature Angle**, under **Split**  button as shown in Figure 3.41, a surface is split based upon feature angle. If **Based on Geometry Feature** is selected, it will split the current surface if it forms a closed path with a feature angle of more than 45 degrees. Another option in the **Feature Angle** tab is to specify **Threshold angle** which would split the current surface above the specified angle if it forms a closed path. Press **Split and Load** to split the current surface. From v8.1, we can split the 2D curves based on the angle/slope discontinuity.



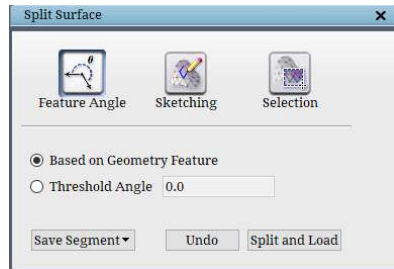


Figure 3.41: Feature Angle tab for the Split button

- **Sketching**

To split up the surface, form a closed curve around it using options available in the **Sketching** tab under the **Split** button, as shown in Figure 3.42. Creating curves is an essential tool in the surface manipulation process. A curve is either a straight line between any two adjacent surface nodes or a straight or bent line composed of paths between any two nodes. The curve appears as maroon coloured lines. The endpoint of a path appears as a white square. Figure 3.43 illustrates examples of paths.

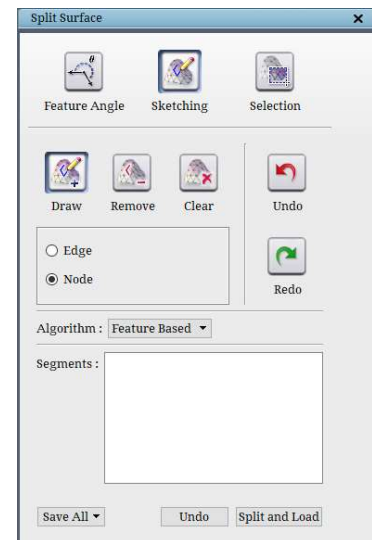


Figure 3.42: Sketching button

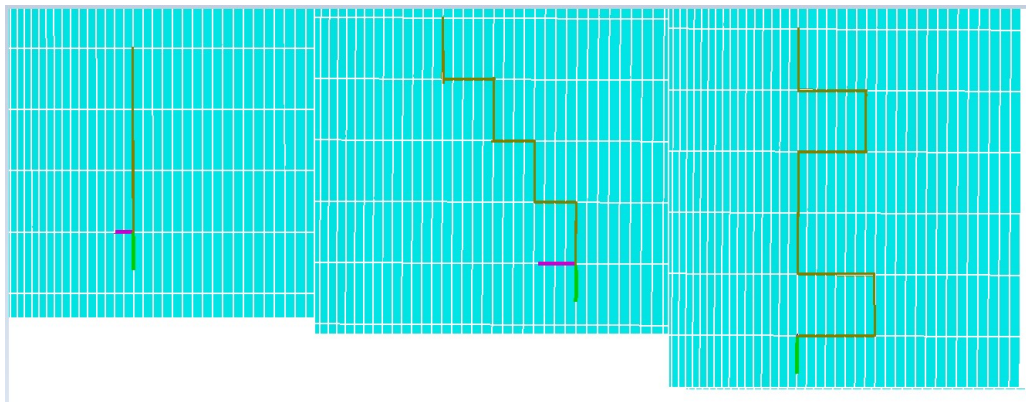






Figure 3.43: Various ways a path can be created

To create a curve, the user can either use the **Draw→ Edge** or **Draw→Node** path button. Edge path button helps to pick a path between two nodes one at a time. After selecting the edge button under the draw section, click (with the left mouse button) between the two nodes to add to the curve. Keep clicking between the nodes to add it to the curve. Similar to the Edge path, Node path allows creating a curve. Instead of picking a path between nodes, click on nodes, and it will form a curve from the starting node to the second node that has been selected. It will try to select the path based on the feature-based algorithm. It can also be changed to choose based on the shortest path. If the line connecting the two points is not along a feature-based surface cell, GridPro will automatically trace the shortest path among the intervening cells. Notice that once used, the Edge path and Node path button will remain depressed (and activated) until it is re-clicked. Thus, after clicking two nodes, clicking a third will extend the path that was just created.

By repeatedly clicking nodes while adding the path button is depressed, a long and precise curve can be created as per the requirement. The two operations **Remove** and **Clear** button manage paths. The remove path  operation deletes an arbitrary part of a whole path. To perform the deletion, depress the remove path button, and drag a box on the part of the path to be deleted using the left mouse click. If you want to continue again, click on the Draw path button, which will automatically trace back and pick the last point of the existing path so that you can click on the next desired node for the path to trace. Notice that the remove path  remains depressed until it is re-clicked. The Clear path  button, when clicked, deletes all the paths. The undo path  redo operation does the undo, redoing of path creation.

Users can choose the splitting process based on two **Algorithms**, namely **Feature Based** and **Shortest Path**. Feature-based traces the path based on the slope discontinuity, and the shortest path traces the shortest path between the selected nodes. Once a closed curve is formed, it is automatically segmented and displayed as different surfaces. And if two disconnected curves are created and if it forms a closed loop that can be split into multiple pieces, the tool automatically segments that as well. Similarly, you can split as many surfaces as desired by forming a closed-loop every time. The split segments are logged in the **Segments** section, as shown in Figure 3.44.

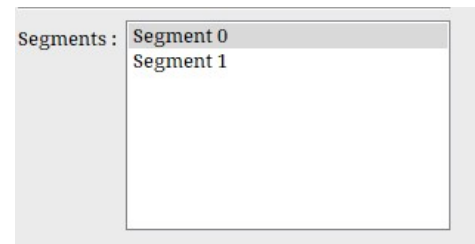


Figure 3.44: Segment section

Figure 3.45 illustrates examples of segmented surfaces. If *Split* succeeds, the two segments are displayed in a different colour (one piece being the current surface will have a sea blue colour). You can continue segmenting pieces as much as desired. Like surfaces and sheets, pieces are either current or not current. The current piece is in sea blue. The split pieces can be scrolled through, and any piece can be selected as current surface using in the **Segments** section. To save, click on the **Save** button. There are multiple modes of saving surface files. Based on the need, save can be Save All, Save Segment, Save Path, etc. Often, segmenting surfaces necessitates using many paths; saving these paths will guide the user when the surface has to be revisited. The *Save Path* button saves a path to the file name.

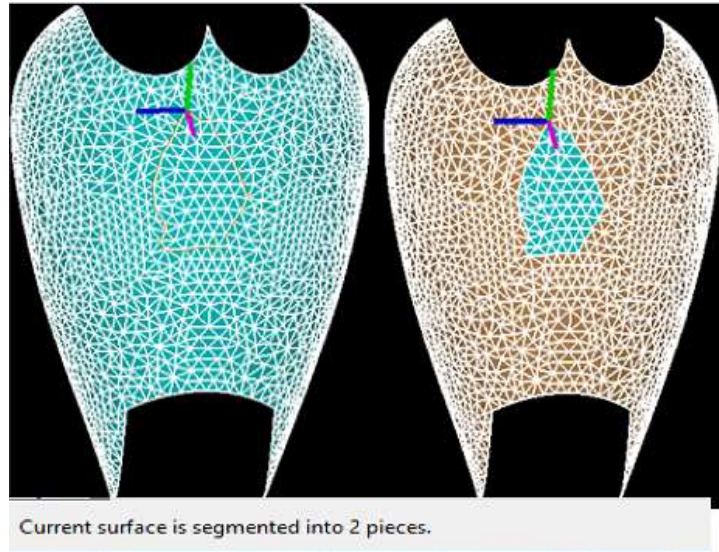





Figure 3.45: Surface split into two pieces


- **Selection**

The third tab under the **Split** button is **Selection**, as shown in Figure 3.46. It is similar to **Sketching**; instead of selecting a path, surface elements are removed to split the current surface. Surface elements can be removed one by one

from the current surface by using  **Remove→By Cell**. After selecting **By Cell** option, clicking on visible surface elements will hide the selected elements. This button remains depressed. It needs to be clicked to switch it off. **Remove→By Box** is used for removing elements from the current surface. By selecting **By box** option and dragging a box using right-click will hide all the elements inside the box. Once the elements are removed, the segments section displays them as Visible and Invisible. All the trimmed cells will be displayed in grey colour and are kept under Invisible. The button will switch off once the operation is done. To add

elements,  button can be used. Similar to Remove, Add also has two options; By Cell and By

Box. A box must be dragged after clicking on the *By Box* option to add elements. Using  button, the selected elements of the current surface can be retained.  button will flip the visibility of all surface elements of the current surface. All surface elements can be restored for

the current surface using  button. All these modifications made on the *Current* surface can be saved using the option under the **Save** drop-down. As explained in Sketching, Selection also

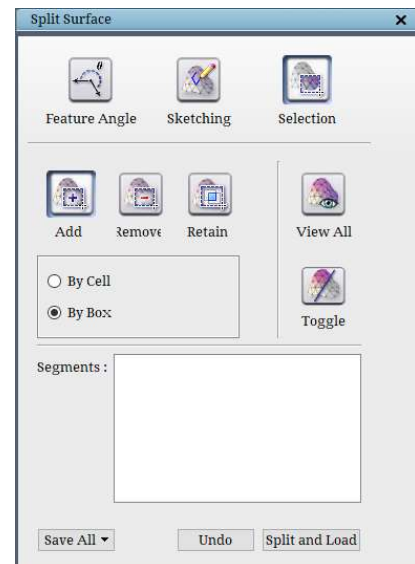


Figure 3.46: Selection dialog box

has 3 kinds of save options: Save All, Save Segment & Save and Load all Segments. The **Split and Load** option splits the segments and reloads the segmented surfaces.

### 3.4.3 Refine

Triangulation on the surface also plays a role in the quality of the grid, because the grid block nodes will try to position themselves on the surface nodes. So, when you need a fine



triangulation on the surface, **Refine** button can be used instead of going back to the CAD software. It is located under the **Edit** section of the surface tab, and Figure 3.47 shows the Refine Surface dialog box. Only a current surface can be refined in the GUI, so select that surface as the Current surface first whenever you want to refine a surface.

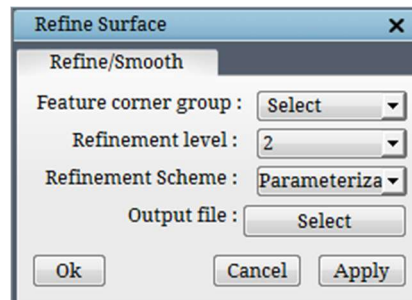


Figure 3.47: Refine dialog box



The three options under **Refine** button give the user better control to modify the surface better. The first option, *feature corner group* is a corner group on which the refinement will not be applied. This helps in retaining the sharp features of the surface where refinement is not required. The second option is the *refinement level*. The third option is the *refinement scheme*; by default, 'Parameterization' is chosen. You can choose one of the three whichever suits your need. Each scheme employs a different algorithm. Finally, you need to enter the output file name to execute the command. The refined surface will automatically load into the UI after executing the command.

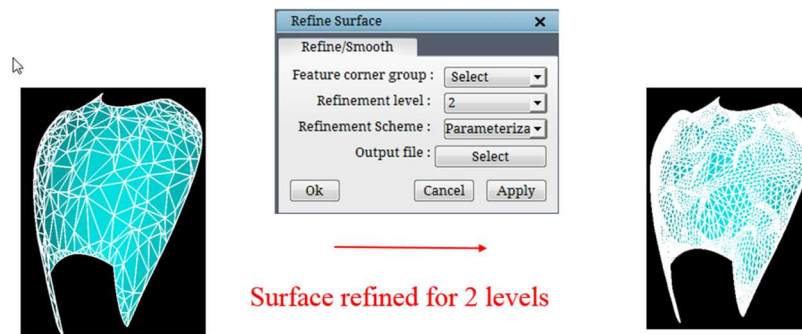


Figure 3.48: Surface before and after refining

### 3.4.4 Merge

If one surface can be split into two surfaces using the *Split* tool, two surfaces having the same boundary faces can be merged into one surface using the *Merge* tool located under the **Edit**

section. If *Merge* button is clicked, a dialog box opens, as shown in Figure 3.49. Select the two surface numbers that must be merged, provide the output file name, and click on **OK**. It will merge the two given surfaces, and it will be replaced by a single merged surface.

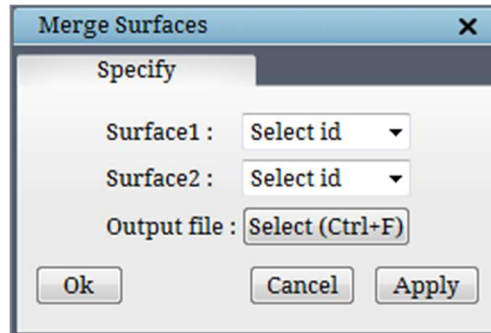


Figure 3.49: Merge Surfaces dialog box

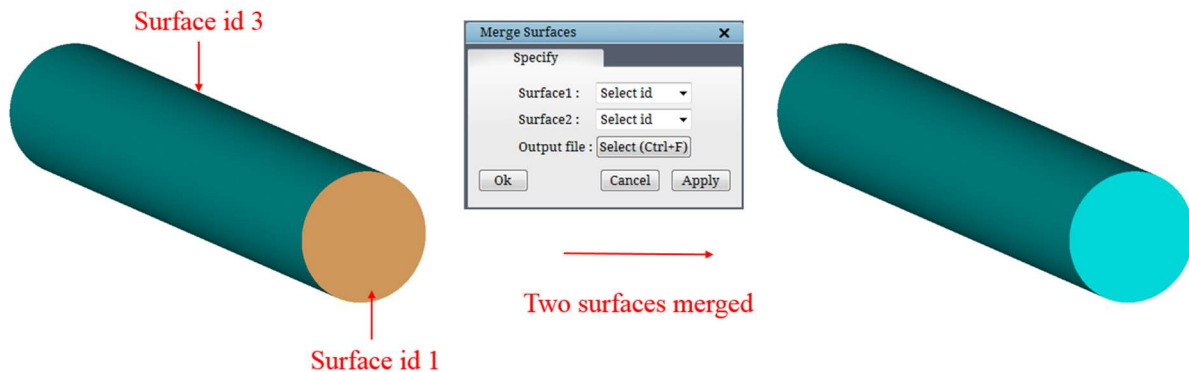



Figure 3.50: Surface before and after the merge

### 3.4.5 Duplicate

Duplicate tool  is located under the *Edit (surface)* section of the Topology tab, and it is used to duplicate the *current* surface to create a new surface and append it to the current topology. This is very useful if a new surface with properties similar to the Current surface is needed. When this option is executed on a built-in surface, a dialog box corresponding to the current built-in surface opens in the right-side panel to modify the surface position and append the surface. For instance, creating a plane parallel to the current planar surface can be duplicated, and the centre can be modified appropriately. In case of explicit surfaces, it opens a confirmation dialog box to copy and append the surface. Once confirmed, it opens the surface transformation

dialog box, as shown in Figure 3.52. The duplicate operation can be undone, and it is also stored in the undo stack.

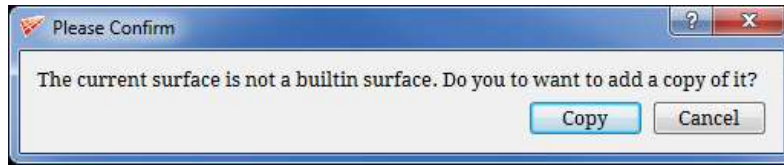


Figure 3.51: Copy confirmation dialog box

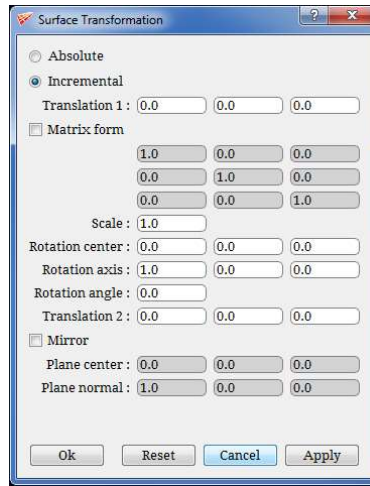


Figure 3.52: Surface Transformation dialog box

### 3.4.6 Settings

**Settings** tool is applicable for all the surfaces in GUI. It is used for *defining orientation, property* and *clustering parameter*. It can also be used for labelling surfaces. If

the **Settings** button under the *Edit (Surface)* section of the Topology tab is clicked, the Set Surface Parameters dialog box opens, as shown in Figure 3.53. The surface id and name of the surface is shown first in the dialog box, which helps as a reference while changing the parameters. For built-in surfaces, it displays the same name for all similar surfaces. i.e. for all cylinders, the name of the surface will be shown as 'Cylinder'.

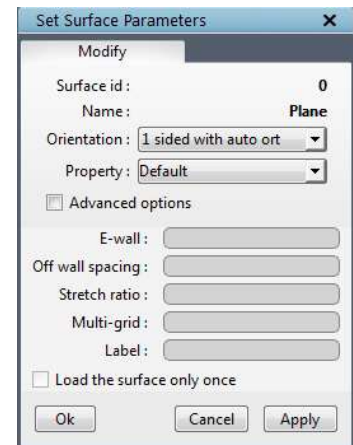


Figure 3.53: Surfaces Settings dialog box



The orientation can be changed using the drop-down list next to the 'orientation'. It has a list of options, as shown in Figure 3.29. Orientation parameter guides the Ggrid engine to grid which side of the surface. If the surface has a one-sided orientation, it can have grid blocks only on one side of the surface which is automatically determined by the grid generation engine. If the surface has a two-sided orientation, it is an internal surface with grid blocks on both sides of the surface. By default, all the surfaces manually appended to the topology have a one-sided orientation. But the surfaces read from an existing topology/macro will retain their orientation from the loaded topology. Note that when the surface orientation is changed, it is saved to the undo stack since it changes the topology state. Boundary condition or property can be set to a surface by selecting one of the options from the drop-down list under the property. It inherits the properties based on the solver selected in the **Solver** menu. The properties set on the surface will be reflected in the grid, which can be modified later in the property tab, based on the desired export solver format if required.

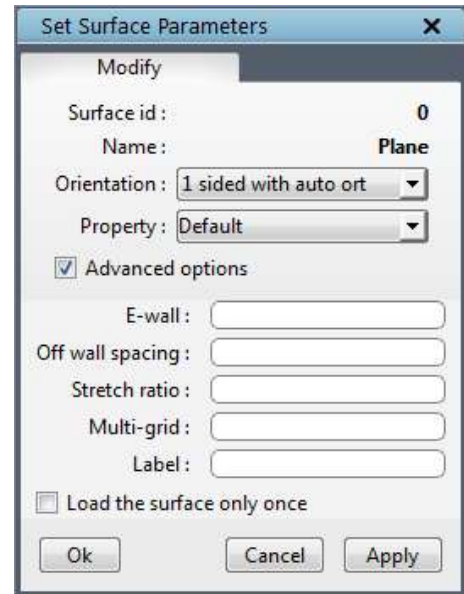


Figure 3.54: Surface Settings parameters with Advanced options enabled

Clustering parameters for a *current* surface can be set by checking advanced options in the properties dialog box. Checking advanced options enables the clustering parameters, as shown in Figure 3.54. Clustering parameters refine the region near surface boundaries. **E-wall** parameter is the off-wall spacing used for clustering within the grid generation engine. E-wall attracts the grid nodes towards the surface based upon the number of grid points and e-wall value provided.

The **Off wall spacing** deals with the number of grid points on a local scale (specifically, near boundary surfaces). The number you insert in the **Off wall spacing** field is the length of the cell edge measured between a boundary grid point and the next closest "off-wall" grid point. Off-wall spacing will add the grid points based upon the stretch ratio and off-wall spacing provided to smooth out the grid lines.

The ratio between this length and the cell edge length measured between the off-wall grid point and the next off-wall grid point is the number you input in the **stretch** ratio field. This number becomes the ratio for a geometric progression of cell edges, which propagates outward from the boundary surface along off-wall grid points. Typically, the value of off wall spacing is less than 0.001 of the length of the boundary. The **stretching** value usually lies between 1.1 and 1.5. See Figure 3.55.



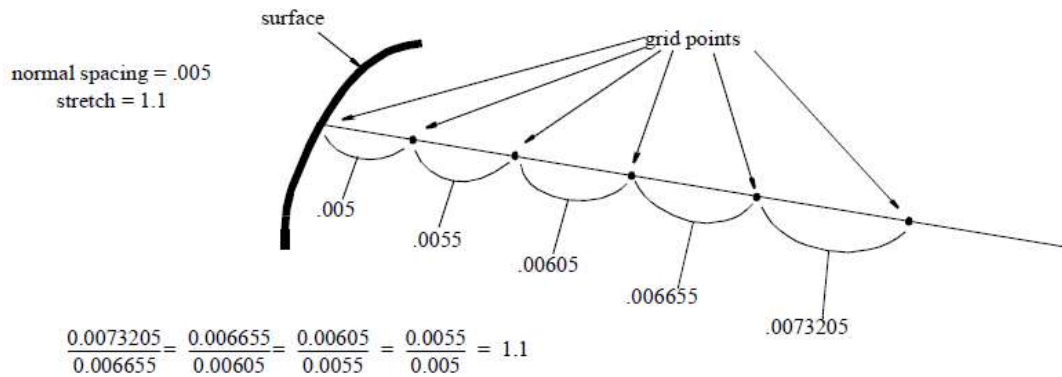


Figure 3.55: Stretching value for grid points

The parameter **Multi-grid** ensures that the number of cells in the normal direction is a multiple of this parameter. It is a unique value for the entire topology, and hence to change the m-grid parameter, use a positive integer value.

Also, a label to the surface can be specified by defining the label in the field next to the **label**. The parameter **label** is a tag attached to the surface. A given surface can have only one **label**, but multiple surfaces can have the same label name. These labels attached to the surface can be loaded in association with property in the Property tab using the **Label** option explained in *section 6.1*. The flag **load only once** is used by the grid generation engine if two (or more) built-in/non-built-in surfaces are identical in type, location, and transformation. Once it is enabled, when you load a fra file as a template and load multiple files together, the duplicate surfaces are deleted automatically, and all its surface assignments are transferred to the original surface.

### 3.4.7 Replace



**Replace** current is used to reposition or modify the surface parameter inputs for the CURRENT surface. It is one of the 3 buttons positioned under the **Edit (surface)** section of the Topology tab. Once clicked, a dialog box opens, as shown in Figure 3.56, where you need to select the surface type you want to reload; once you select a surface type, it opens the respective dialog box where you can modify the inputs. If the surface type 'External' is chosen, it opens the file menu from which you can select the desired surface. This helps in loading a new surface in place of the existing one without losing the surface assignments. Note that the changed position (specific to built-in surfaces) using the replace button can be undone from the **Topology** tab.

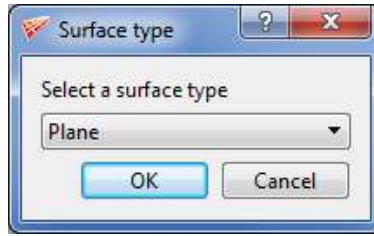


Figure 3.56: Surface type dialog box

### 3.4.8 Transform


**Transform** is one of the necessary operations to execute the desired surface transformations. It works on any surface in GUI, whether implicit surfaces (such as plane, line, circle, cylinder, ellipse, ellipsoid, xpolar or xyz) or explicit surfaces (such as *lin*, *quad*, *tria*, *tube*). This operation can carry out surface scaling, translation, rotation, and mirroring. From

v6.7, this operation is integrated into the **Transform**  button under the Topology tab. Please refer to section 4.6 for more information.

### 3.4.9 Node

GUI allows the user to move a node of a *Current* surface. To move a surface node, click



on **Node**  button under **Check/Repair** section in the Surface tab. Move Surface Nodes dialog box opens as shown in Figure 3.57. Nodes can be moved w.r.t to workplane orientation, either along the workplane or normal to the workplane. By default, it can be moved parallel to the workplane direction. If **Along** button is highlighted, the nodes move parallel to the workplane. Click on the **Normal** button in the dialog box to move nodes perpendicular to the workplane.

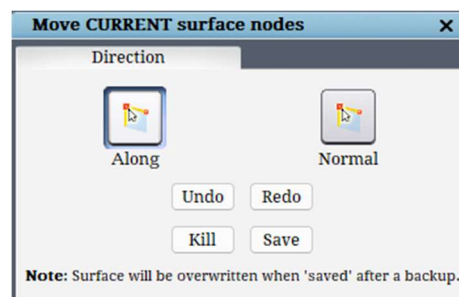



Figure 3.57: Move surface dialog box

Similar to topology undo and redo operations, node displacement operations can also be reverted using **Undo** and **Redo** buttons in the Move surface dialog box. Clicking on **Kill** button will delete all the node displacements. All the node displacement operations done on a surface can be saved using the **Save** button.

### 3.4.10 Fill

GridPro allows the user to merge segmented surfaces (pieces) and fill the holes created by segmentation. GridPro fills holes with triangles; hence it works only for tria surfaces. To fill a

hole in a surface or piece, select the surface as *current*. Click on **Fill**  button located under the **Check/Repair** section in the Surface tab. Figure 3.58 shows the dialog box of the *Fill* button. Create a path around the region that needs to be filled using the available path creation options, then click on the **Fill** button in the dialog box. *Tria* elements will automatically fill the hole. Filled holes are not part of the surface until the modifications are saved using **Update Surface** button.

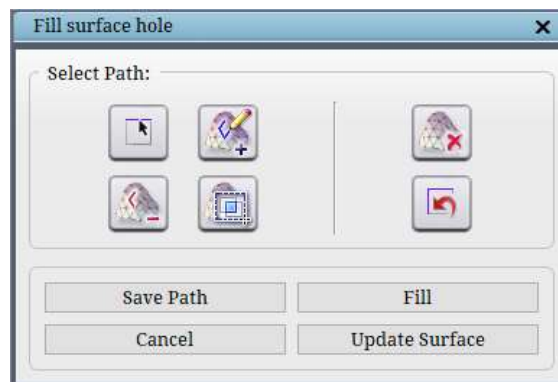


Figure 3.58: Fill Surface hole dialog box

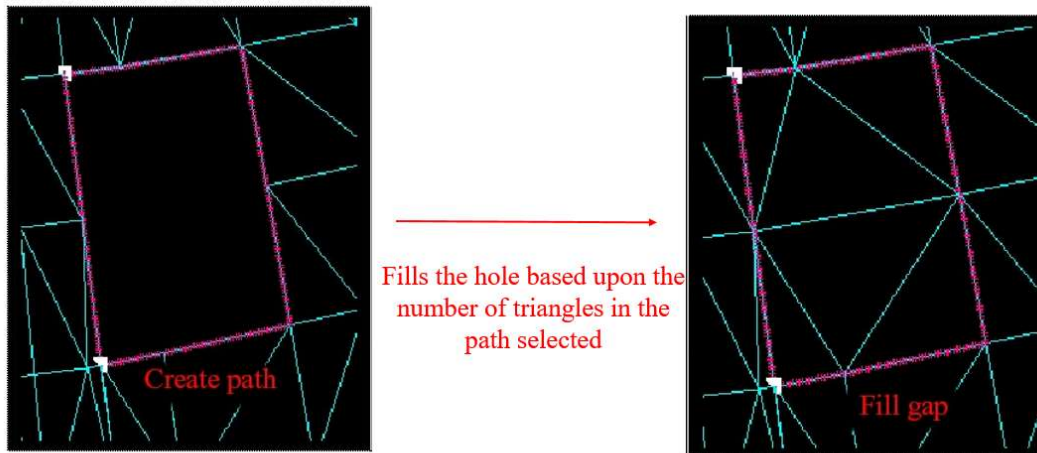


Figure 3.59: Fill Surface hole dialog box

### 3.4.11 Repair 2D Curve

Sometimes, when you load a \*.dat or \*.lin file, notice that two nodes (coordinates) will be located at the same position, either at the origin or endpoint. GridPro cannot generate grids where two surface nodes are overlapping. It will report an error message in the terminal, as shown in Figure 3.60, while running Ggrid. You need to find and remove the overlapping nodes from the surface file to resolve such cases. Using the **Repair 2D curve** tool, you can resolve this issue. Go to the **Surface Tools** menu and select **Repair 2D curve** option, which opens a dialog box as shown in Figure 3.61. It works only on the *Current* surface. You have to input the tolerance value. The newly modified surface will have the original surface name. The previous surface file would be backed up to 'filename.~1~'. You can save up to four backup files. Note: This button is enabled only in the 2D workspace.

```
found collapsed x'prod <0> on quad3198<4265 4269 4270 4266>,exit
```

Figure 3.60: Error message

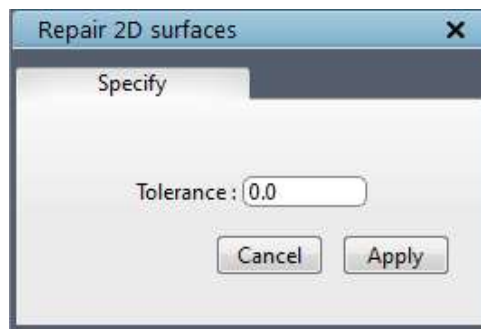


Figure 3.61: Curve button dialog box

### 3.4.12 Revolve Curve

There might be cases where a surface with a circular cross section such as the hub or shroud of a turbine blade might be incomplete. The CAD software from which the surface was extracted, might provide only one surface with the periodic section, and not the entire hub/shroud. However, the user might require the entire surface as GridPro generates its own periodic boundary which might be different. It would also be required when the periodicity of the case is changed. To facilitate this, the **Revolve Curve** tool has been added under **Surface Tools** in GridPro.

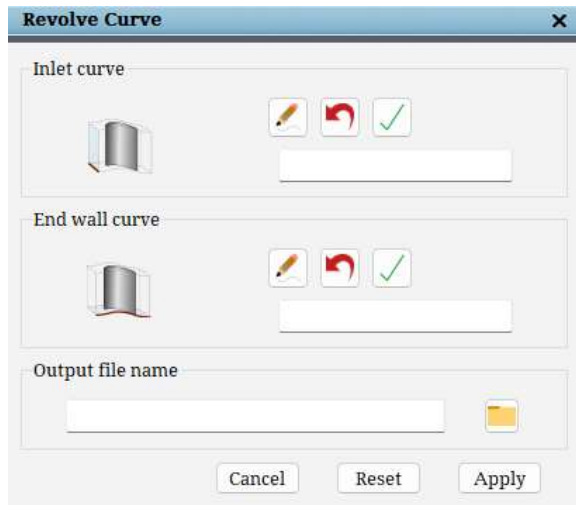





Figure 3.62: The Revolve Curve Dialog box

To generate the complete surface using the tool, the **Inlet Curve** and **End Wall Curve** must be selected on the surface.

This can be done by clicking on the  button and selecting the path for each of the required curves. The  button is used to undo the path selection, and the  button is used to confirm that the selected path is the correct one. The **Output file name** section is used to specify the output file name. The surface created after revolving is a .tube surface.

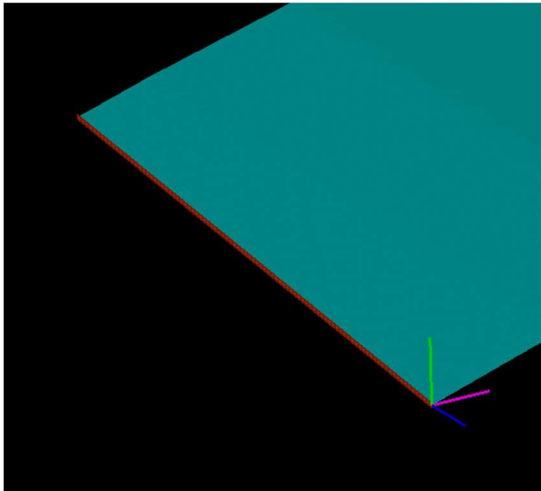


Figure 3.63: The selected path highlighted in brown.

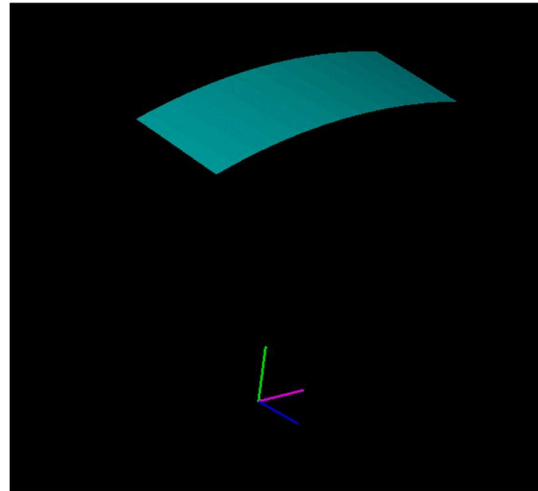


Figure 3.64: Before revolving the curve.

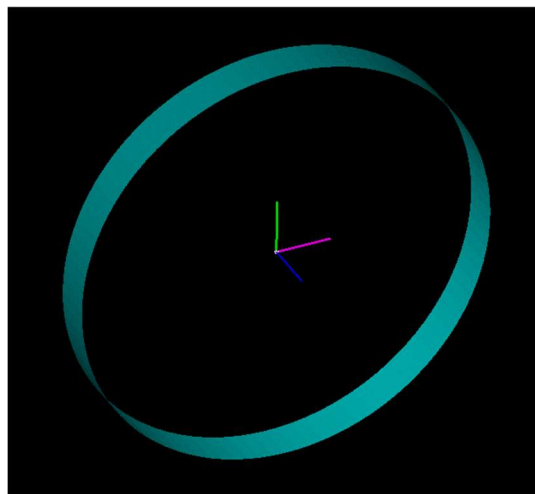


Figure 3.65: After revolving the curve.

### 3.5 Right Click Menu

Apart from the properties options, the often-used options are kept under the surface right-click menu, as shown in Figure 3.66.

- **Centre of rotation** picks the right-clicked position on the surface as the centre of rotation of the screen.
- **Make current** option is used to select any surface as *current*. Selecting this option on a *current* surface will ignore the selection.
- **Reload** option is used to modify the position of the built-in surfaces or to modify the surface definition of explicit surfaces.

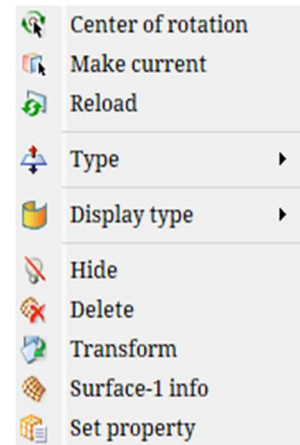


Figure 3.66: Surface Right click menu

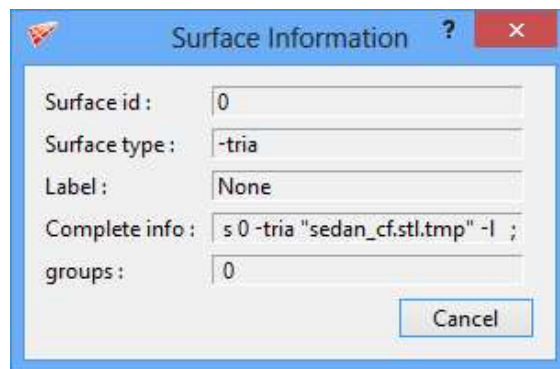


Figure 3.67: Surface information

- **Type** option is used for changing the surface orientation.
- **Display type** shows the list of display types available in GridPro, using which the surface display of selected surface can be changed. Refer to *section 2.3.15* for more information.
- **Hide** option hides the selected surface. To know how to unhide/show the surface, refer to *section 2.5*.
- **Delete** option deletes the selected surface that can be restored by clicking on the UNDO button in the topology panel.
- **Transform** button can be used to translate, rotate, scale and mirror the surfaces explained in *section 3.4.8*.
- **Surface info** shows the information on surfaces such as id, type, label, position and group number to which it is grouped, as shown in Figure 3.67.
- **Set property** helps in setting the property to a surface without switching to the Surface tab.

# CHAPTER 4

# TOPOLOGY

In order to grid regions, it must be covered with wireframe topologies that crudely follow their boundaries. The majority of *GridPro* operations deal with creating and modifying topology. Remember, a topology consists of wireframe and surface assignments.

*GridPro* is a unique multiblock grid generation program because it revolutionizes how volumes are gridded. Usually, multi-block grid generation programs require users to assign topological attributes (separate parts of a topological framework) laboriously to individual blocks, whereas *GridPro* automatically assigns separate parts of a topological framework to individual blocks.

Remarkably, *GridPro* can also grid loosely positioned topology. In other words, *GridPro* will grid topology whose position only roughly conforms to the corresponding geometry. A wireframe topology does not have to fit the boundaries for the region to be gridded exactly. However, the topology itself should be precise. If a topology is positioned too far from a surface, the gridding process might take longer than it normally should. Thus, as long as your topology is positioned with some care, the surfaces will be gridded with relative ease. When creating topology, you would do well to remember this golden rule: *the topology for a region has to be precise but does not have to be positioned precisely, just positioned to follow the geometry of its bounding surfaces roughly.*

*GridPro* grids volumes in both 2 and 3 dimensions. Although all real-life objects are three dimensional, sometimes two-dimensional grids can serve as useful models.

This chapter deals with the topology building and various tools which help in creating the topology. These tools are classified based on their functions; they are

1. Topology Creation Utilities,
2. Automatic Topology Creation,



3. Topology Grouping Utilities,
4. Topology Modifying Utilities,
5. Topology Transformation Utilities,
6. Topology Enhancing Utilities,
7. Corner Reposition Utilities and
8. Topology Optimization Utilities.

All the utilities are located on the right-side panel of the Topology Tab. Under **Create** section, buttons related to Topology Creation Utilities are located. Under the **Edit (Surface)** section, 3 buttons that are often used are located. They are Duplicate, Settings and Replace. Under the **Edit (Topology)** section, buttons related to Automatic topology creation utilities, Modifying utilities, Transformation utilities, Optimization utilities, Enhancing utilities and Projection utilities are located. Grouping utilities have dedicated sections such as **Groups**. The topology consists of wireframe and surface assignments. A separate section called ‘**Assignment**’ consists of the tools that help in assigning the corners to their respective surfaces. The tools that help in creating the surfaces using the corners and edges are classified separately under surface creation utilities which are located under the **Surface Tools** menu. This chapter also explains the gridding process.

## 4.1 Topology Building

Like all other multi-block grid generation programs, *GridPro* requires you to first sketch out the topology for a surface before inputting that topology. You must know how a volume will be decomposed into blocks; *GridPro* can only work with your results, not create those results. In other words, you should decide upon the pattern of grid points before you can instruct *GridPro* to generate the corresponding grid.

The first step in creating any topology is always defining the region that must be gridded. Then, decomposing the boundary geometry into surfaces and the surfaces describe a block pattern; the resulting wireframe topology must represent this pattern. Although the real-life objects are 3 dimensional, creating topology in two dimensions first better prepares you for creating topology in three dimensions(in case of symmetry in 3<sup>rd</sup> axis).

In two dimensions, gridded blocks are represented by convex quadrilaterals or quads. In three dimensions, gridded blocks are represented by convex hexahedrons or hexes. When you create a quad or hex, *GridPro* will automatically recognize it as such. Unlike other grid generation programs, *GridPro* does not require you to identify a geometrical object after you create it. *GridPro* will read only topologies containing “full face matching.” These topologies include intersecting quads with intersecting faces having the same dimensions or intersecting hexes with intersecting faces having the same dimensions. In other words, one side of one quad

cannot intersect one side of two other quads, and one face of one hex cannot intersect one side of two other hexes.

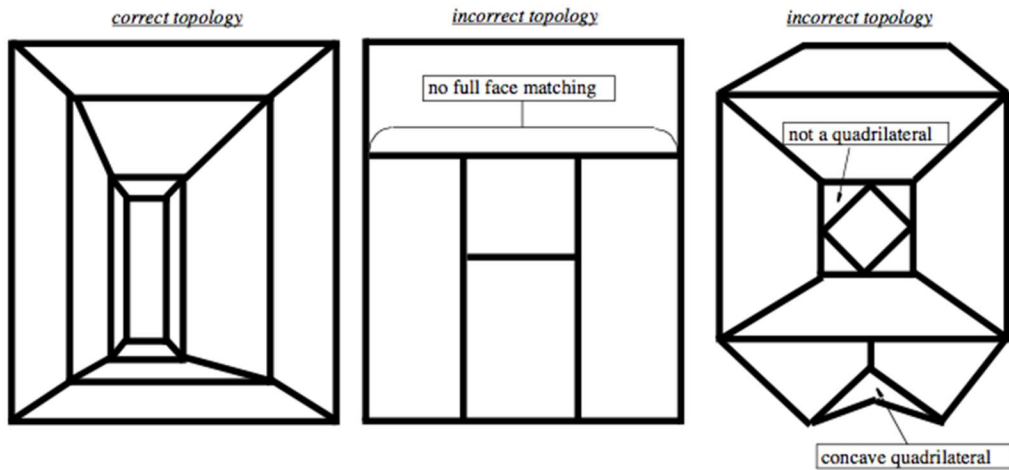


Figure 4.1: Topology examples

Now that you know the topological form expected from *GridPro*, you should make sure that you understand how to decompose a region and create a topology. Because surfaces can be complex, they may require complex topologies. Unfortunately, no set of rules can define how all topologies could be constructed. However, there are some rules that define how most topologies should be constructed when there is no other special feature or concern. When you create topologies about angles in surfaces, you should consider the concavity or convexity of the angle. This is seen from the viewpoint of the region to be gridded rather than from outside that region looking inward. In the following examples, the dashed structures represent surfaces, and the solid structures represent a possible, correct topology for those surfaces which is on the side that is to be gridded. Differences between correct topologies and a discussion of what topology to use will be covered later. In general, preferred topologies about angles between 0 and 90 degrees have the structures shown in Figure 4.2.

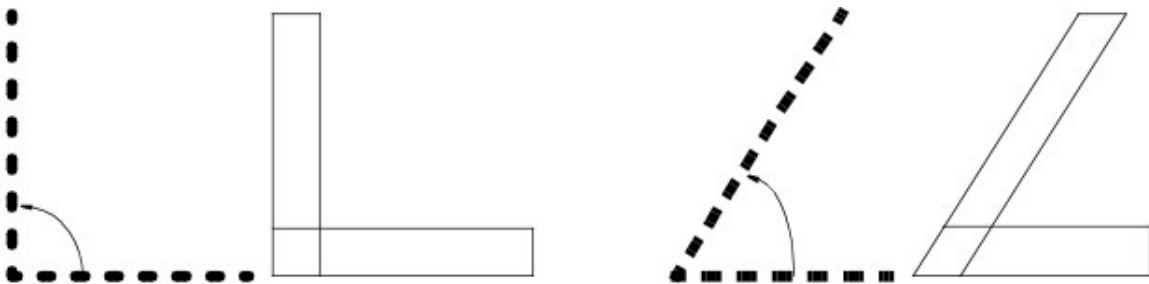


Figure 4.2: Topology for angles between 0 and 90 degrees

Angles between 90 and 180 degrees typically have two different structures. These structures are shown in Figure 4.3; the topology on the right side of the figure is used more often for these angles:

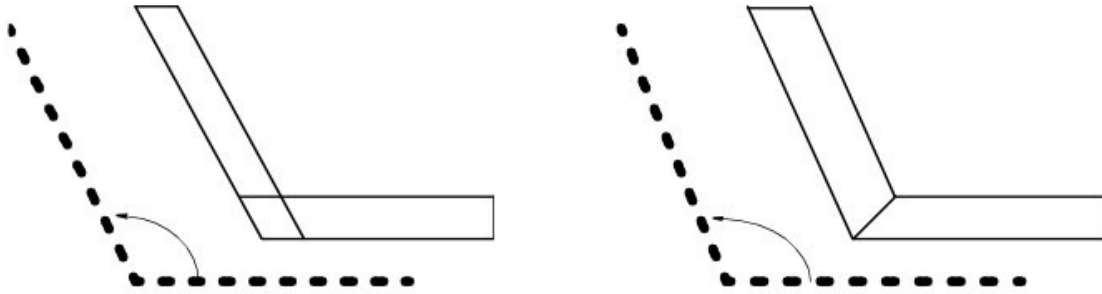


Figure 4.3: Topology for angles between 90 and 180 degrees

Preferred structures for angles between 180 and 270 degrees are shown in Figure 4.4.

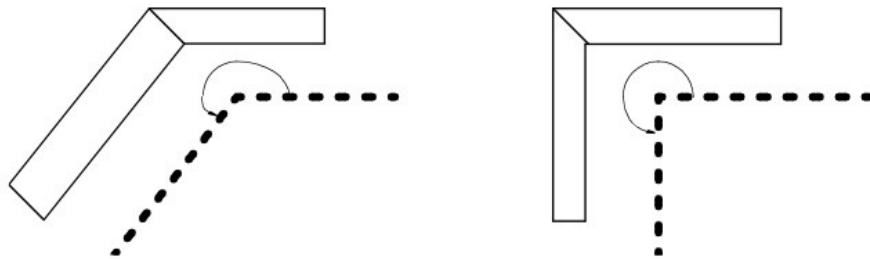


Figure 4.4: Topology for angles between 180 and 270 degrees

Angles between 270 and 360 degrees can have several structures, the most common of which are illustrated in Figure 4.5.

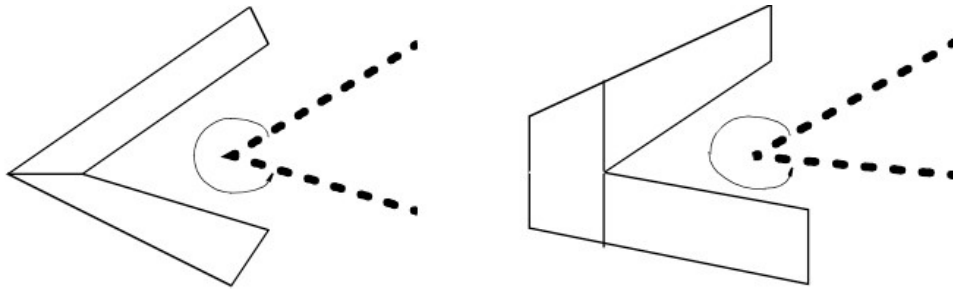


Figure 4.5: Topology for angles between 270 and 360 degrees

Notice that all of the topologies are created using *convex* quads, and all quads have full face matching. Figure 4.6 incorporates ideas you have seen:

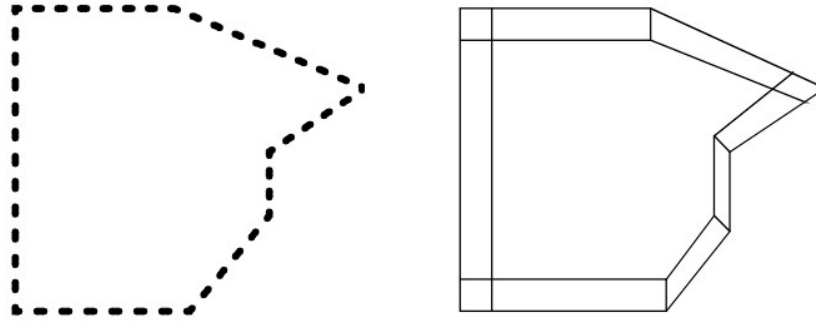


Figure 4.6: Example showing the above ideas incorporated

If a surface is smooth, decompose the surface, as you normally would, then create an “inner layer” between the smooth surface and the decomposed block. The “inner layer” is illustrated in a dark shade. Figure 4.7 illustrates the correct topology for a smooth surface:

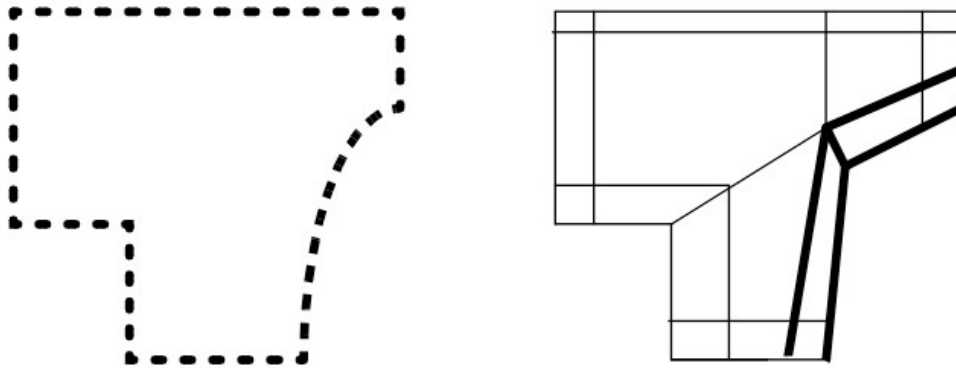


Figure 4.7: Topology for a smooth surface

Notice how the darkened lines follow the path of the circular region. In Figure 4.7, the topology conforms well to the corresponding geometry. Also, this example shows how topology is wrapped around the continuously curved portion of the surface. The figure formed by the dark lines is considered a partial “wrap”. Looking at the angle variations above, you will notice that the angles close to 180 degrees are wraps. In fact, at every point of a smoothly varying surface, the angle is computed from the corresponding surface tangent line. Look back at how topologies are created for angles between 90 and 360 degrees. Notice that some of those topological configurations are “wraps” as well. In Figure 4.8, the partial wraps formed for the continuously curved surfaces are illustrated in dark shade.

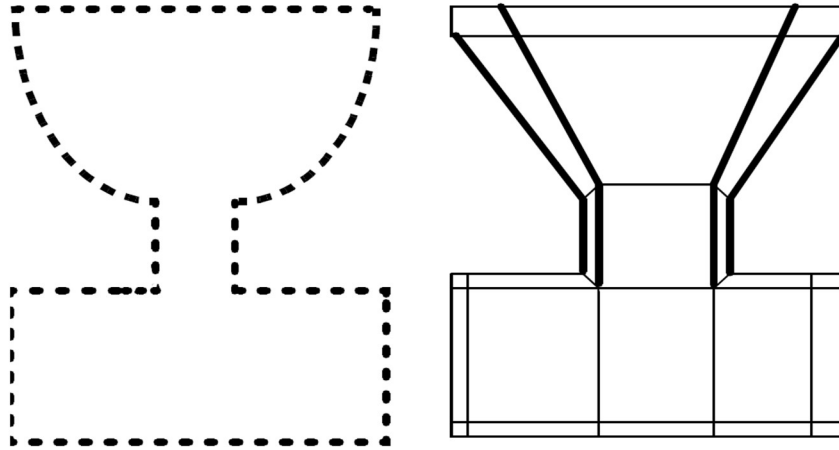


Figure 4.8: Topology for continuous curves

Notice that in the above example, wraps are formed around the 270-degree angles as well. Figure 4.9 displays the full wraps in a dark shade.

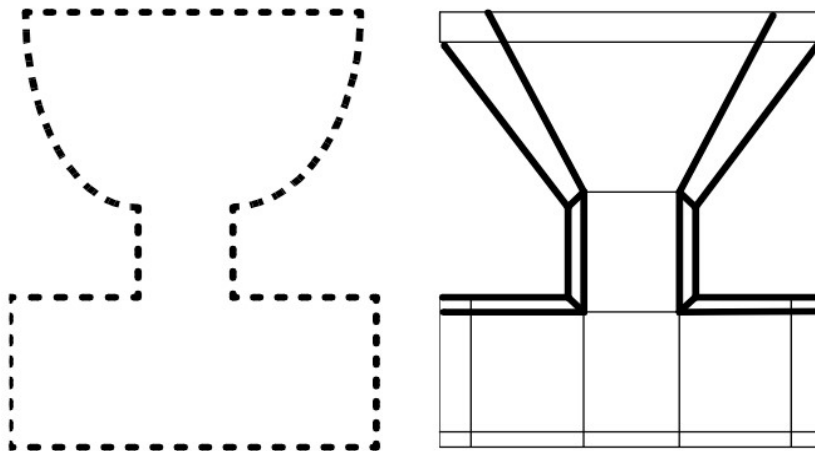


Figure 4.9: Topology with wrap layer highlighted

Figure 4.10 displays the full wraps in dark shade from the first example in this chapter.

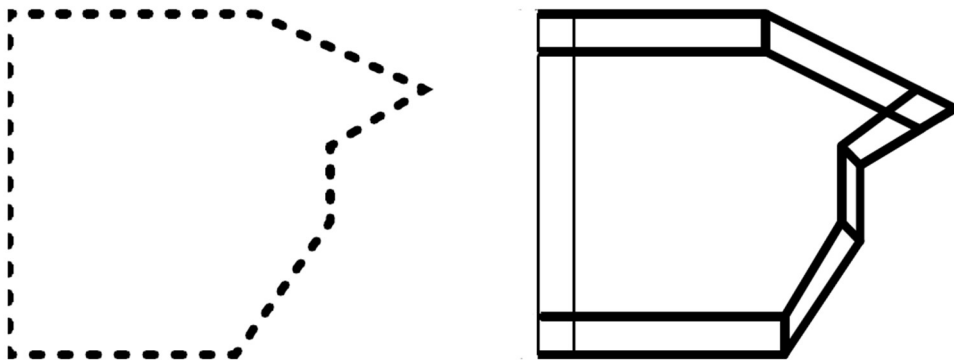


Figure 4.10: Topology with wrap layer highlighted

In order to develop an intuitive understanding of what topology to create, you should learn through examples. You will successfully prepare regions for gridding by first learning the rudimentary rules described above. Of course, to obtain the phenomenal grid accuracy *GridPro* can provide, you have to learn the nuances of different surfaces gradually and be able to produce topological frameworks using intuition rather than rules.

All real-life configurations have three dimensions. Therefore, to use *GridPro* effectively, you must understand the structure of three-dimensional topology and become proficient in manipulating three-dimensional topology.

The approach to wireframe building is of two types, one is top-down and the other is bottom-up. Anything can be followed for any case, but it is advisable to use a bottom-up approach for complex cases.

## 4.2 Topology Creation Utilities

As discussed in Chapter 4, the tools related to the topology are grouped into different categories. Of which the first category is the *topology creation utilities* grouped under the **Create** section of the **Topology** tab located in the right-side panel, as shown in Figure 4.11. Topology creation utilities include Surface, Corner, Link, Insert and Macro.

### 4.2.1 Surface

The foremost desired object to generate a grid is surfaces. Before gridding a region, users must often read several surfaces. *GridPro* can read surfaces from roughly eight different formats. Of these eight analytical implicit surfaces, one of the formats includes plane, cylinder & ellipsoid in 3D and line, circle and ellipse in 2D. All these surfaces can be created using the ***Surface*** button under the **Create** section. Please refer to Chapter 3 for more information on each option under the ***Surface*** button.

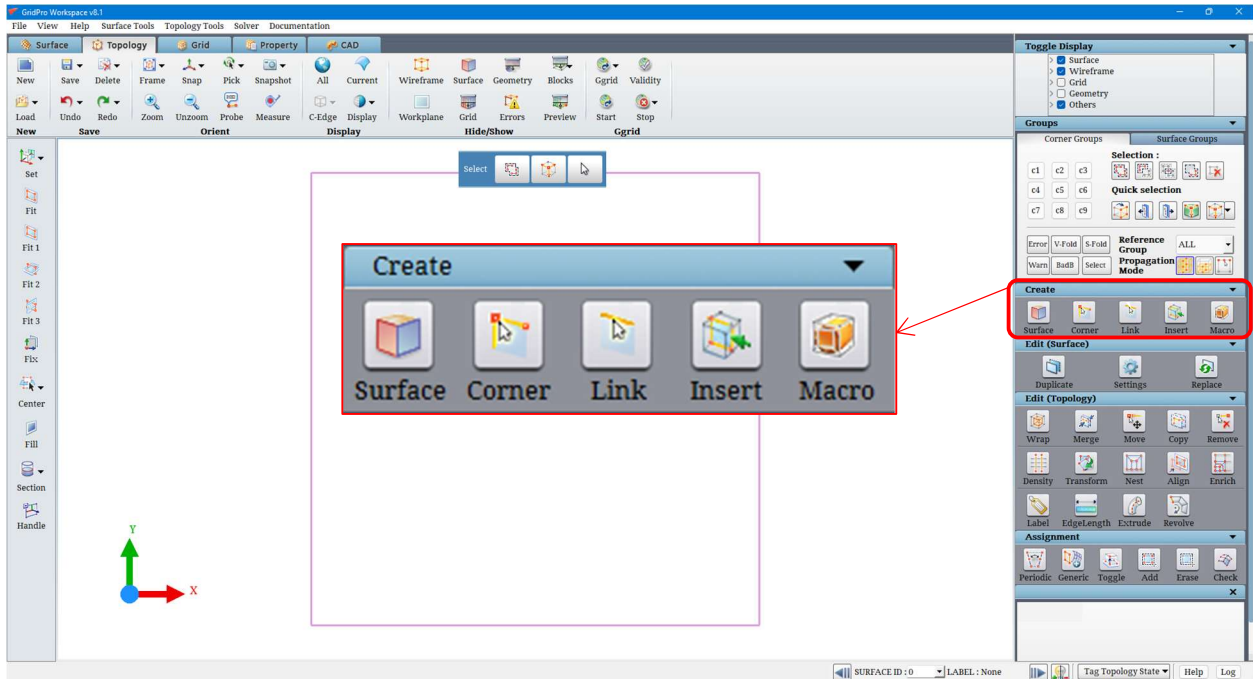


Figure 4.11: Create section

## 4.2.2 Corners

The wireframe is created entirely by corners and edges. Any two or three-dimensional structure can be constructed with just corners and edges. Corners are displayed as small orange squares, as shown in Figure 4.12. Edges are displayed simply as yellow lines with two corners as endpoints.

To input a corner, switch on the **Corner** button in the **Create** section of the topology tab. Figure 4.13 shows the *Create Corners* dialog box. Select the mode of creation, i.e. either using a mouse click or by specifying the coordinates. By default, the mode of selection is set to the mouse that is specified as **Create On** and using the mouse click, the corner can be either created on the **workplane** or **current surface**. By default, the flag **workplane** is selected, and so the corners will be created on the workplane. Selecting the **Current surface** flag under the **Create on** tab allows you to input the corners on the CURRENT surface.

Once the mode of creation is set, click the left mouse button at the desired location on the workspace. An orange square should appear at the location of the cursor. You can also create by depressing the <C> key and clicking the left mouse button on the workspace. Edges can then be created after corners are created, or you can choose to create both corners and edges together. TO do that, you can switch on **With Edges** button and start clicking on the screen. The edges will

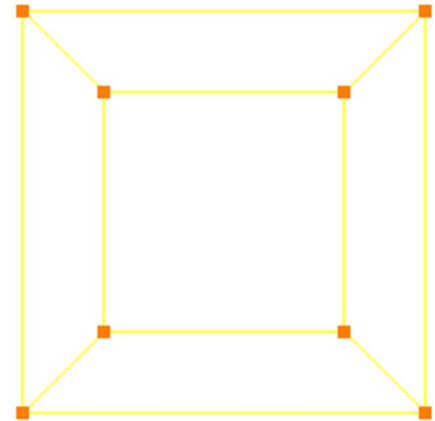


Figure 4.12: Wireframe



automatically be created once you create corners 1 and 2. Based on the mode selected, either Workplane or Surface, corners and edges will be created correspondingly. After creating a few corners, if you click on the first corner which is created, a closed-loop edge will be generated between the first and lastly created corner instead of creating a new corner.

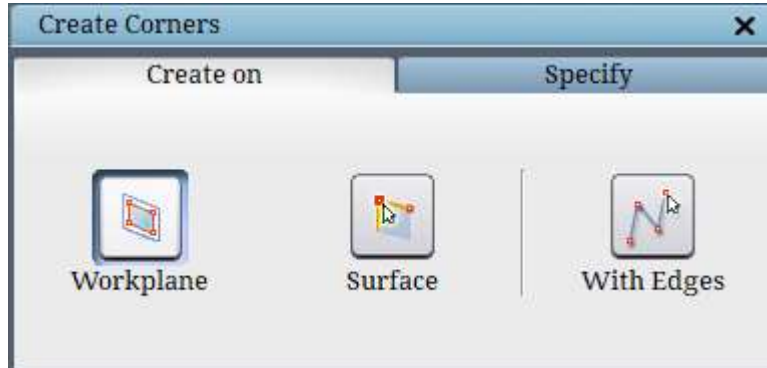


Figure 4.13: Corner dialog box

If the workplane is perpendicular to the screen when creating corners on the workplane, an error message pops out, as shown in Figure 4.14, because corner cannot be created when the workplane is perpendicular to the screen.



Figure 4.14: Error message when workplane is normal to screen

Similarly, an error message as shown in Figure 4.15 appears when the corner creation mode is on the surface but the clicked position is outside the surface.

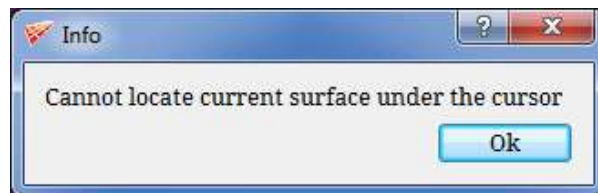


Figure 4.15: Error message when cursor is not on surface

Many times you would like to create the corner exactly at a particular location instead of arbitrarily. In such circumstances, you can switch the tab to **Specify** as shown in Figure 4.16, enter the coordinates in the provided fields, and click **apply** to create the corner.

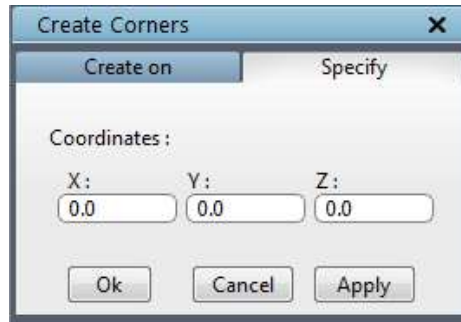


Figure 4.16: Specify tab dialog box

There are three shortcut options located under the right-click menu on corners. When you move the cursor to the corner, the cursor changes to corner selection mode (circular ring); right-click on the mouse, which will show the options as in Figure 4.17. The **centre of rotation** option fixes the centre of the rotation of the screen to the corner position. An axis similar to the global axis whose centre appears on the corner, which represents the centre of rotation, is fixed in that place.

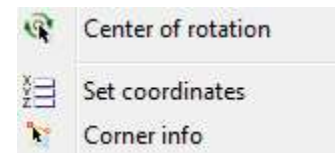


Figure 4.17: Right click menu of corner

The *set coordinates* option allows you to redefine the corner position with specific coordinates that open a dialog box to enter the coordinates for the corner to position. Figure 4.18 shows the dialog box of the set coordinate option.

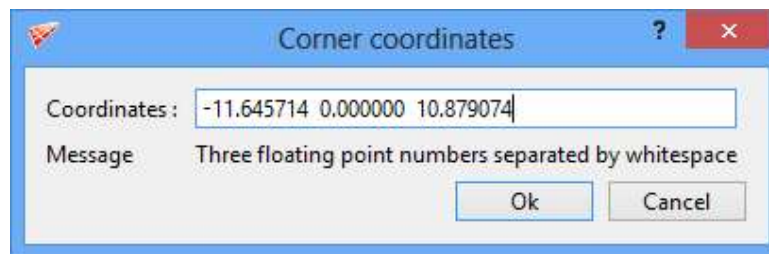


Figure 4.18: Set coordinate dialog box

The *corner info* option displays all the information about the corner, such as corner id, corner location, corners which are linked with it, surfaces to which it is assigned and the corners paired with it in case of a periodic surface, as shown in Figure 4.19.

NOTE: Corners and edges are also used in surface creation which will be discussed in *section 4.11*.

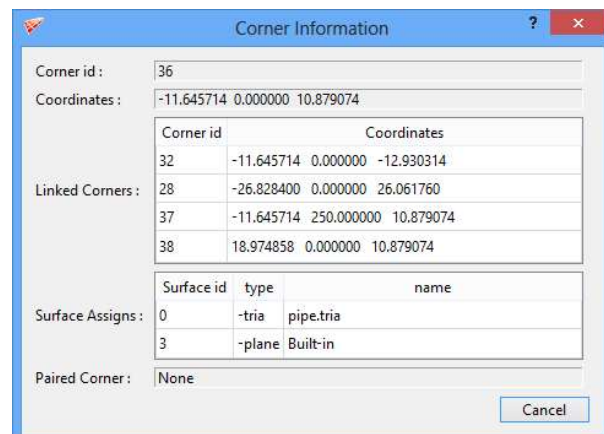


Figure 4.19: Corner info dialog box

### 4.2.3 Link

When you want to create an edge between two corners, **Link** button can be used. Depress the **Link** button located on the left side toolbar under **Create** section in the **Topology** tab. Move the cursor to one corner, and click on it using the left mouse button (in order to select a corner, the cursor must become a ring), then move the cursor to the other corner and click the left mouse button. An edge should appear between the two corners, as shown in Figure 4.20.



Figure 4.20: Corners with and without edges

In order to create a string of edges, all you must do is depress the **Link** button for the duration of the process and click the mouse strategically. By clicking the endpoint corner of the string, clicking on every successive corner you wish to link, and finally clicking the other endpoint corner of the string, you can form many edges quickly and efficiently as shown in Figure 4.21.

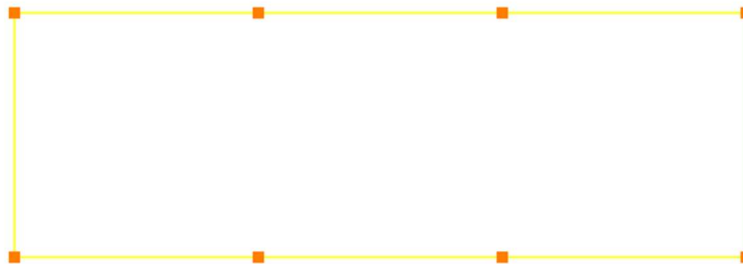


Figure 4.21: Strings of corners linked

Simply dragging a box after depressing the Link button around the group of corners that has to be linked can also create edges, as shown in Figure 4.22. Note that it will not create a closed-loop in order to avoid creating a triangular block.

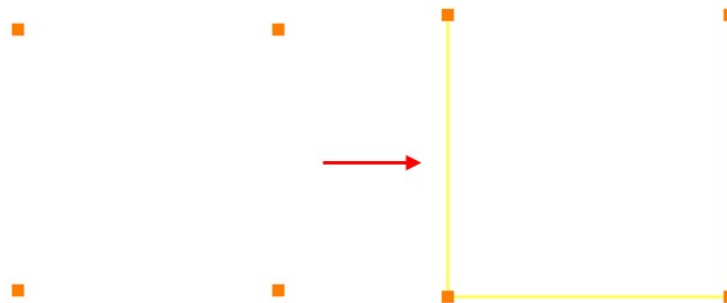


Figure 4.22: Corners linked after dragging a box around them

The other way to create edges is by checking **Groups** options in the link dialog box shown in Figure 4.23. Keep the corners to be linked in two different groups and select those two corner group numbers (Refer *section 4.4* for more information on corner groups), and click on *Apply* to link them. By default, the **Select** option is selected, which enables you to simply select two corners or a group of corners to be linked.

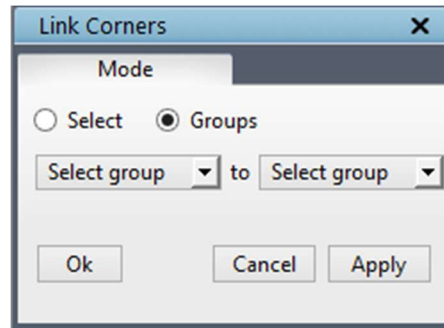


Figure 4.23: Link dialog box with Groups mode selected

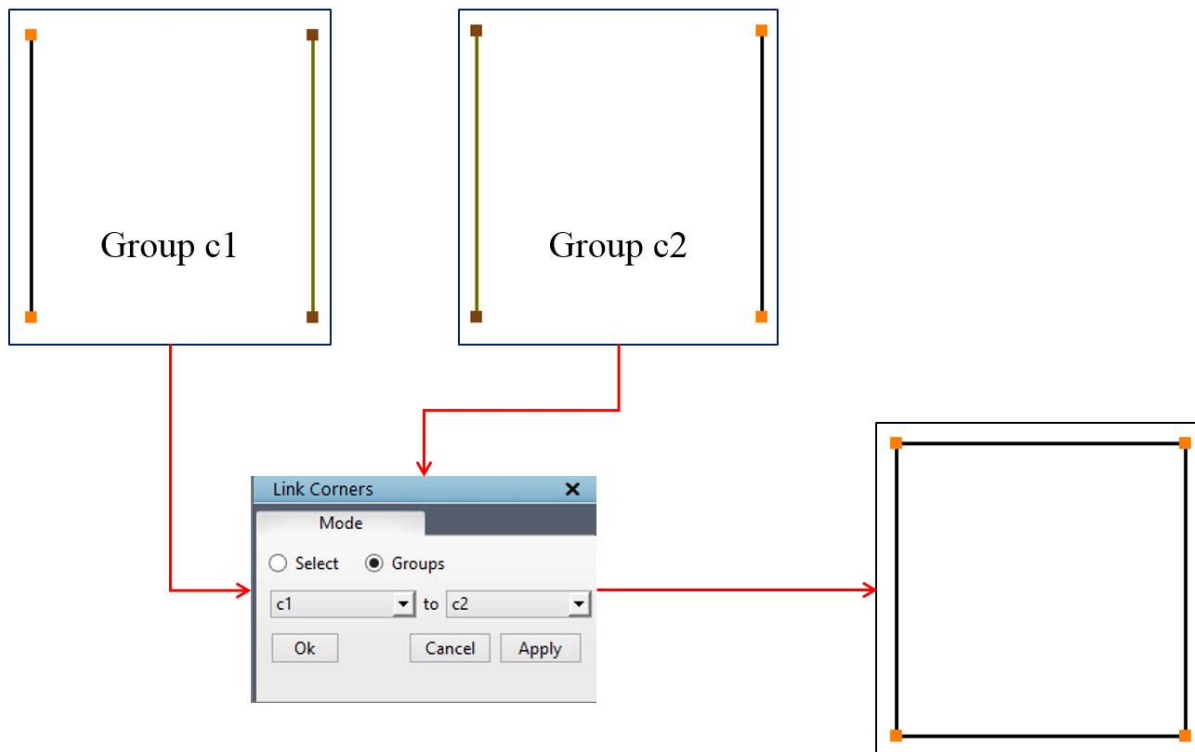


Figure 4.24: Linking corners using Groups

NOTE: Linking does not work on inactive corners.

## 4.2.4 Insert

When you create topologies for multiple surfaces, you will often construct grid-like wireframes. Since adding corner layers individually for these wireframes consumes time and sacrifices precision, GridPro has a feature that enables you to create these (often multiple) corners/sheets quickly within the boundary of an already existing wireframe.

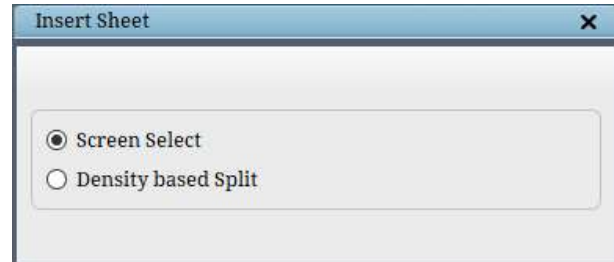


Figure 4.25: Insert dialog box

In order to insert a sheet on an existing wireframe, click on the **Insert** button in the **Create** section, and a dialog box opens in the pop-up section as shown in Figure 4.25, which asks whether screen selection or density-based inserts have to be created. For screen select, you can simply click on edge at the desired location to split the blocks. The inserts need a propagation mode to split; the propagation mode can be selected from the Groups section



Corner/Sheet inserts are created using a feature of the propagation mode. The three propagation modes are All, Group and Edge. By default, the **ALL** mode is on. After depressing the **Insert** button, click on an edge where you would like to insert. The insert sheet will propagate till the two boundaries of topology, i.e. a corner is created on that edge at the point you clicked and proportionally on every edge topologically parallel to the edge you clicked, as shown in Figure 4.26. It can also be done using the shortcut key by holding the **'I'** key on the keyboard and clicking on the edge where the insert has to be done.

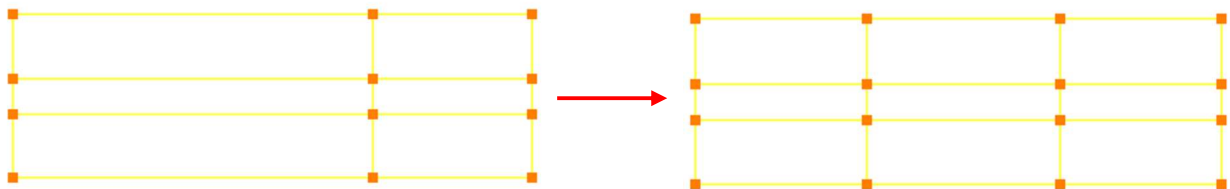


Figure 4.26: Insert propagation in All mode

If **Group** propagation mode is selected and a corner will be created on that edge at the point, you click and proportionally on every edge in the active group that is parallel to the edge you clicked. The inserted corner won't be propagated beyond the active corner group, as shown in Figure 4.27.

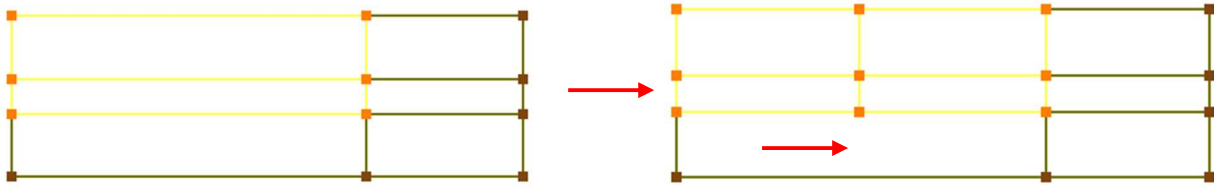


Figure 4.27: Insert propagation in group mode

If **Edge** propagation mode is selected, which means ‘1 edge insert’, i.e. it allows you to input a single corner on an edge in order to split that edge.

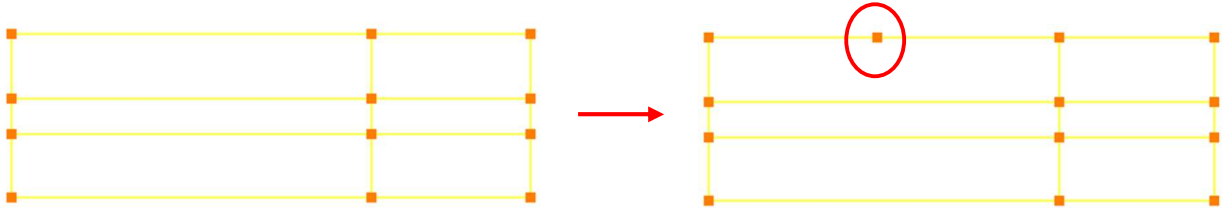


Figure 4.28: Insert propagation in edge mode

If mode **All** or **Group** is ON, and you click on an edge, not only are corners formed proportionally along every edge parallel to the one you clicked; but edges are also created between the newly formed corners. If the **Group** mode is used and a group is not activated, the mode functions similar to **All** mode.

The other way to do an insert on an edge is based on the density. Select the **Density based Split** option which will open the dialog box as shown in Figure 4.29. With the help of this density based tab, you can either do the multiple inserts on all the edges or only to the edges which are grouped in the current group. Every edge has a default density of 8, which is called as *base density*. Based on the base density you provide, it will determine the number of inserts. So the given base density should always be less than the default density in order to calculate the number of inserts. You can also use this option on selected corners by keeping them in a corner group and inputting them in the fields provided. If no corner group is selected, it will consider the entire topology.

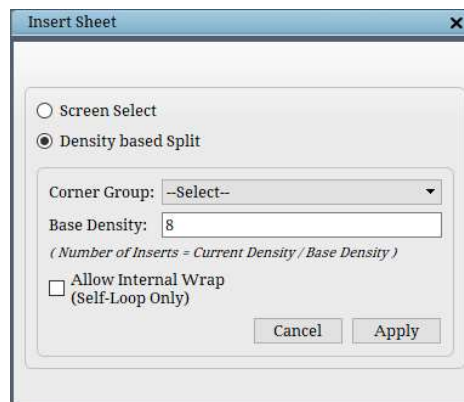


Figure 4.29: Density-based dialog box

Note: Insert can be executed only on the active corner group, but the inserted sheet can propagate through the inactive corners. The insert propagation is based on the selected propagation method.

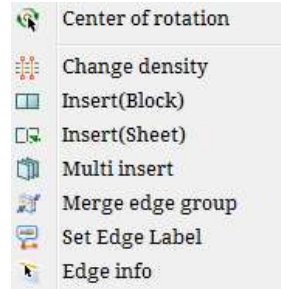


Figure 4.30: Right-click menu of edge

In addition to the **Insert** button in the **Create** section, you can also insert using the right-click mouse button. Figure 4.30 shows the list of options in the right-click menu of an edge. Right-click on an edge where you want to insert and select the **Insert (Block)** option from the list. This insert option's propagation depends on the mode selected in the insert button. The other option in right-click menu is **Insert (Sheet)** which inserts the sheets in such a way it exactly replicates the adjacent sheets, whereas the **Insert (Block)** option insert the sheets in such a way it avoids the extra link. For E.g. when you do Insert (Sheet) on a wrapped topology as shown in Figure 4.31, an extra link is created because the outer four corners are also considered as a separate block; hence it adds an edge connecting those two edges, but the Insert(Block) option evaluates such cases and avoids creating such links.

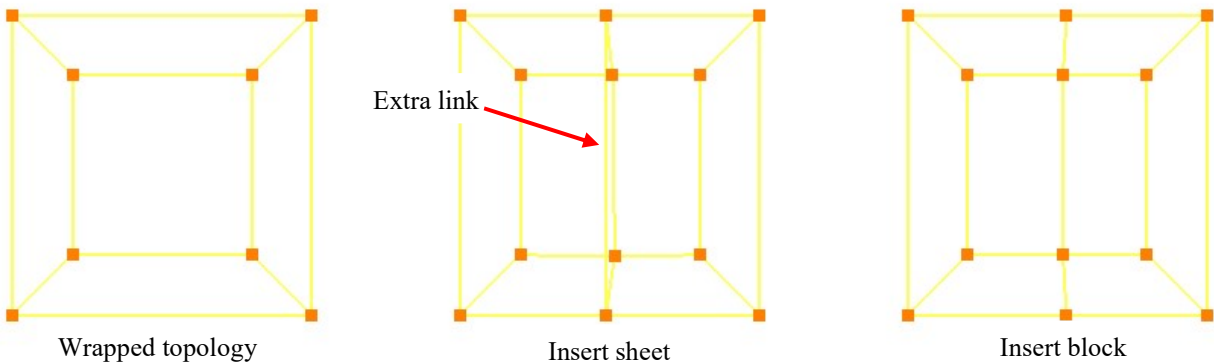


Figure 4.31: Difference between Insert sheet and Insert block

**Edge info** that displays the information about edge id, the corner id and its coordinates of the corner to which the edge is linked, length of the edge and the density of the edge.



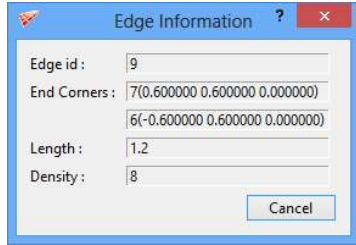


Figure 4.32: Edge info

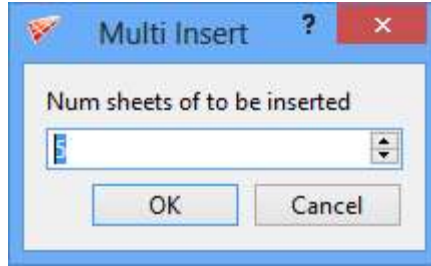


Figure 4.33: Multi insert dialog box

Similar to COR in the corner right-click menu, the COR in the edge right-click menu fixes the COR of the screen to the selected position on the selected edge. **Multi Insert** option inserts the multiple sheets based on the number inputted in the *Multi Insert* dialog box as shown in Figure 4.34.

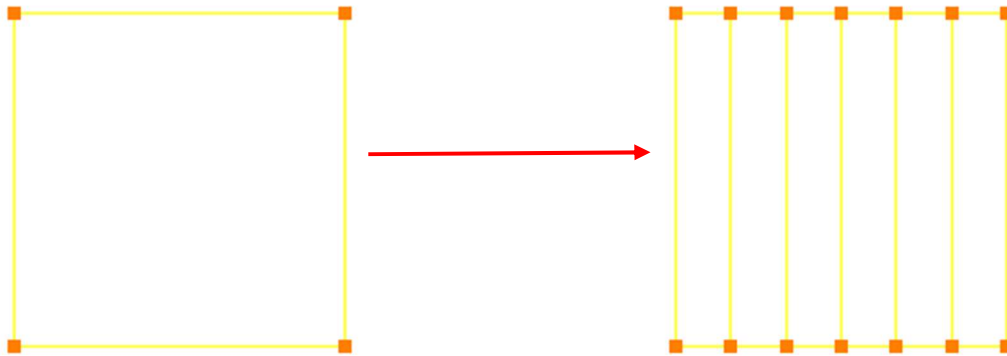


Figure 4.34: After Multi insert

Other two options under the right-click menu are Merge edge groups and Set Edge label. **Merge edge group** merges the adjacent topology sheets of the selected edge and its parallel edges. It is similar to **Merge** button functionality, as explained in *section 4.5*, and it is very easy & interactive to handle. If the corners of the given edge and all of its parallel edges can be merged together, it executes the merge operation, which otherwise will pop up an error message. Similar to block and surface labels, the **Set Edge label** sets a label for any given edge and all of its parallel edges. Only the fra files which are saved after setting the labels will have the edge label information, and the same should be used for subsequent operations related to edge labels. It primarily helps in modifying the density of desired edges from the schedule file while running Ggrid.

## 4.2.5 Macro

Macros are pre-assigned topologies that can be often used for simple geometry creations and to create the outer domain. Therefore, the software offers some topological

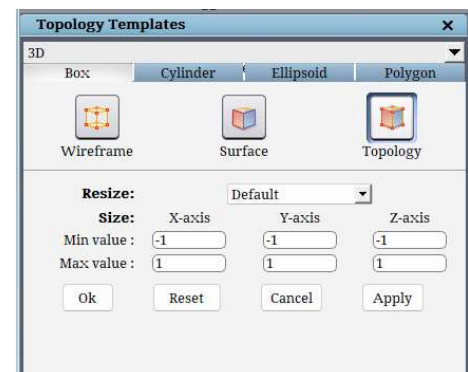


Figure 4.35: 3D Macro options

templates from which you can input objects such as surfaces and/or wireframe. The **Macro** button is located in the **Create** section of the right-side panel in the **Topology** tab. Similar to built-in surfaces, macro has different options for 2D and 3D. In the case of 3D - **Box**, **Ellipsoid**, **Cylinder** and **Polygon** macro are available as shown in Figure 4.35 and in the case of 2D - rectangle, circle and ellipse macro are available as shown in Figure 4.36. Both for 2D and 3D, you can load a user-defined topology. Depending on the dimension of the workspace chosen, the corresponding macro is enabled, i.e. if the workspace is opened as 3D, 3D macro buttons are enabled; if a 2D workspace is opened, 2D macro buttons are enabled. The dialog box of each macro allows you to choose the surface and/or wireframe you would like to create. Each option has its macro dialog box which is located in the right-hand side panel.

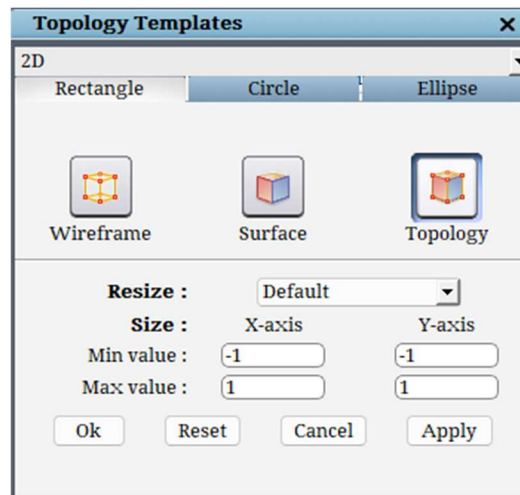


Figure 4.36: 2D Macro options

- **Box**

When **Macro** button is clicked, by default **Box** macro dialog box opens, as shown in Figure 4.35, if it is a 3D workspace. The box macro allows you to create automatically a three-dimensional region bounded by 6 planes. You can create only surfaces or only wireframe or both, i.e. topology, by selecting one of the options in the pop-up. By default, **Topology** is selected, which provides a wireframe box structure and six planes along with respective corners assigned to its respective surfaces. The wireframe option creates only the wireframe and the surface option creates only the surfaces.

The **Resize** dropdown list allows you to resize and/or fit your surface and/or wireframe accordingly. Figure 4.37 shows the list of options in the resize dropdown. This resize option automatically fills up the dimensions fields in the dialog box of the respective macros. The default option allows you to manually input the values in the below fields for the given macro option. The **fit active surfaces** option calculates the minimum and maximum boundary of the active surfaces in the GUI along all three

- Fit active surfaces
- Fit current surface
- Fit curr surf group
- Fit workplane
- Default
- Last setting
- Enlarge 100%
- Enlarge 10%
- Enlarge 1%
- Reduce 1%
- Reduce 10%
- Reduce 50%

Figure 4.37: Resize dropdown list

axes and updates it in the below value fields, whereas the **fit current surface** option uses only the current surface boundaries. Similarly, the **fit current surface group** option uses the surfaces in the active surface group to calculate the boundaries. The **fit workplane** uses the workplane position and its dimensions. The **Last setting** option allows you to use the last used values for a macro in the current session. The **default** option takes the predefined values, .i.e. -1 to 1 for min and max respectively. The next six options increase or decrease the existing values in the fields below the **resize**, depending on the ratio you select. You can also set the dimensions of these regions manually by changing the numbers in the X, Y, Z min and max fields. Once the values are entered, click on **Ok** to create the box macro and close the dialog box. Clicking on **Apply** will create the box macro and retain the dialog box to proceed further. Click on **Cancel** to close the dialog box without creating the box macro. These three options work the same way for all the macros.

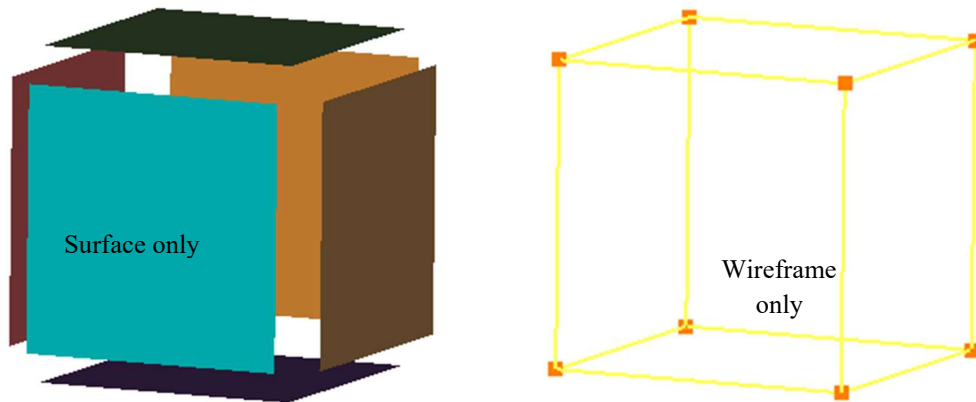


Figure 4.38: Box macro

If either both **Apply** and **Ok** is clicked consequently or **Apply** alone clicked twice, a warning message pops up as shown in Figure 4.39 to avoid duplication of the macro again. You can create a duplicate of the macro by clicking on 'Yes' in the pop-up window.

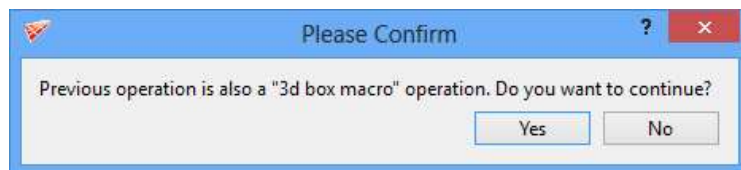


Figure 4.39: Warning message

- **Cylinder**

Next option in 3D macro is Cylinder. The **Cylinder** macro allows you to automatically create a cylinder with a centre, radius, length, and rotation axis that you provide. Two planes will automatically be positioned to bind a truly cylindrical region. The position of these planes depends upon the length of the cylinder. Figure 4.40 shows the cylinder macro dialog box. First two fields (topology type and resize) in the dialog box are the same for all the macro options.

The remaining fields in each macro box require you to input dimension values belonging to each macro.

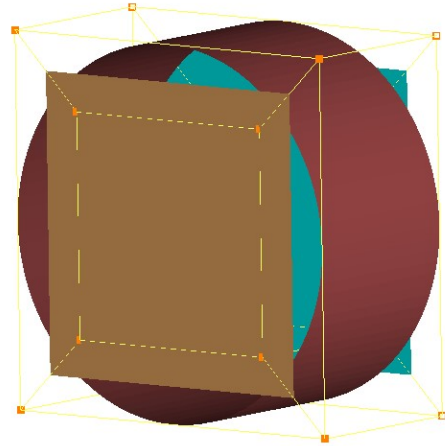
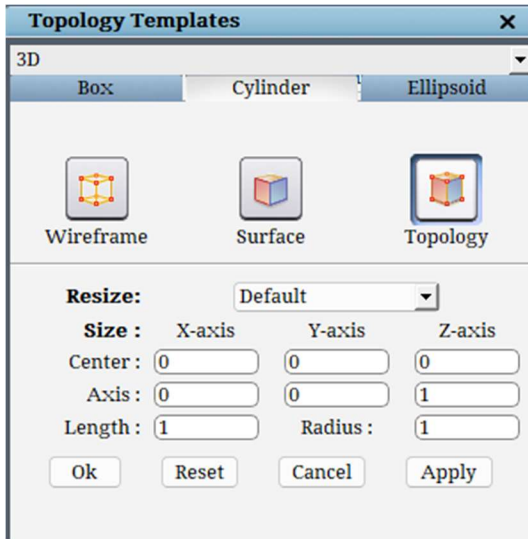


Figure 4.40: Cylinder topology macro

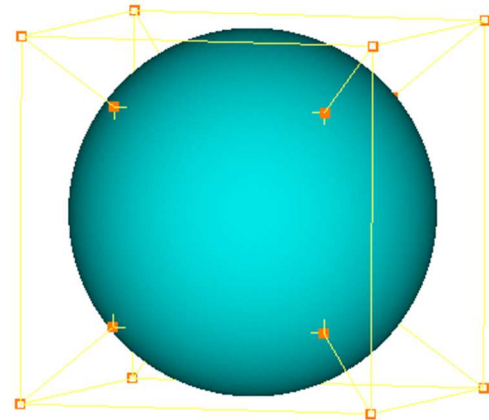
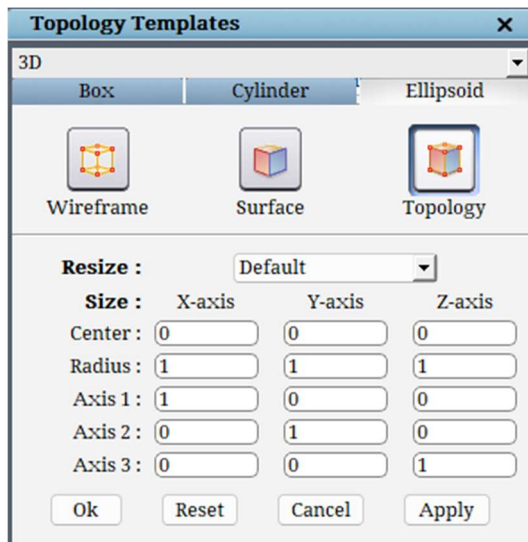


Figure 4.41: Ellipsoid topology macro

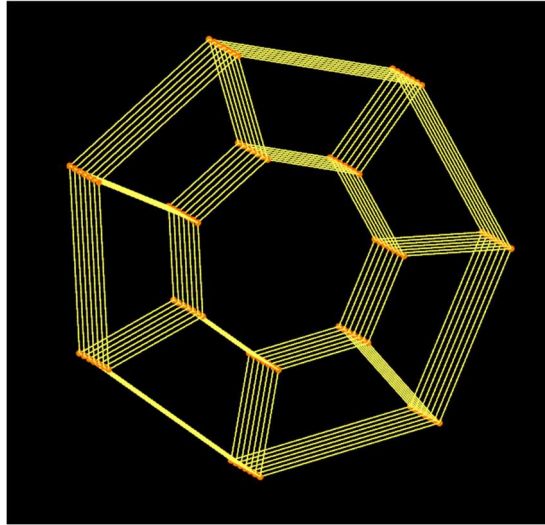
- **Ellipsoid**

The ***Ellipsoid*** macro enables you to create an ellipsoidal shape region with a centre, axis and radius. Sphere is a special case of ellipsoid; when all the three axes have the same radius, a sphere is created. By changing the radius value in any one of the axes, an ellipsoid of different shapes can be created. Figure 4.41 shows the Ellipsoid dialog box. As mentioned above, the dimension field requires a centre, axis and radius value in all three directions. These fields can be

filled manually or automatically using the *resize* dropdown list by choosing one among the list that best suits your need.

- **Polygon**

The ***Polygon*** macro can be used to create a polygonal wireframe. Unlike the other 3D macros, only a wireframe can be created using the polygon macro. Certain user inputs need to be provided to create this.



*Figure 4.42: Polygonal wireframe created using polygon macro*

The very first option for a polygonal macro is the **Axis** on which the macro has to be created. Either the default x,y or z axis could be chosen or the tool can evaluate the axis with respect to the workplane, by selecting the **Workplane** option under **Evaluate Axis**. The user also has the option to manually enter the axis, by clicking on the **User defined** radio button and entering the axis direction vectors.

After the Axis has been defined, the **Centre** of the polygonal macro around which the macro has to be created needs to be specified. The centre coordinates can be entered manually or, the tool can pick the centre, by either calculating the **Workplane** centre, **Current Surface** centre or **Surface Group** centre. This is done by selecting the respective option under **Evaluate centre**.

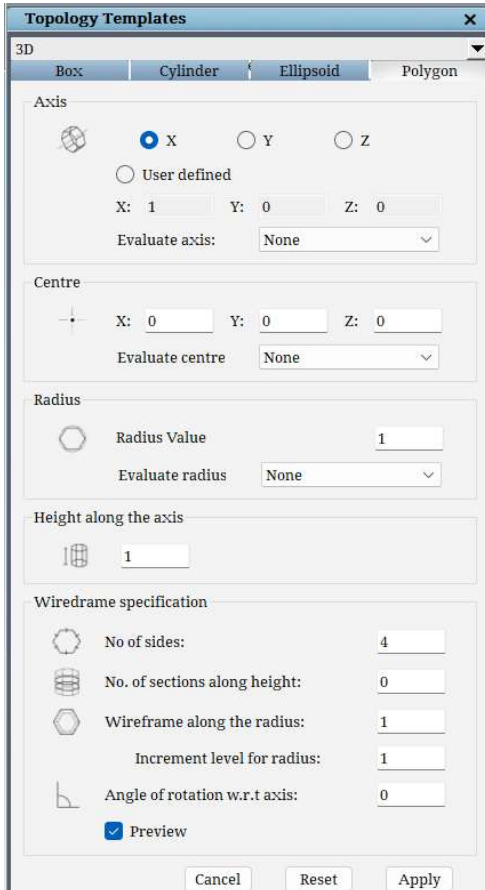


Figure 4.43: Polygonal Macro options

Then the **Radius** of the macro needs to be defined. It can either be input manually by entering the value in the **Radius Value** text box, or it can be automatically evaluated. Much like the **Centre**, the **Radius** has the same options to evaluate the radius.

Once the radius is done, the **Height** of the macro to be created can be entered. Here the height of the macro along the axis is defined. This option has to be manually entered in the text box.

Finally, the user needs to enter the wireframe specification. Here the user has the option to define the **Number of sides** required, the **Number of Sections along height**. The **Wireframe along the radius** can be used to define the number of sections required along the radial direction. The next option, **Increment level for radius** lets the user choose if they need multiple concentric macros or just one layer. If multiple layers are required, the distance between each layer is defined by this option. And finally, the wireframe can be rotated along the axis if required by entering an **Angle of rotation w.r.t axis**.

The **Preview** check box can be used to see a preview of the wireframe. Changing the parameters dynamically changes the preview shown.

- **Rectangle**

Only if a 2D workspace is opened, 2D macro options are enabled. You cannot create a 2D macro in a 3D workspace. Figure 4.44 shows the dropdown list of the same.

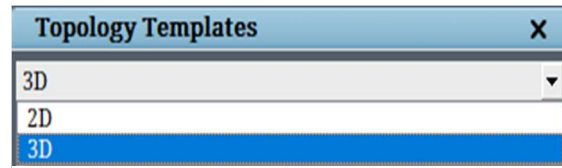


Figure 4.44: Dimension dropdown

The first option in 2D macro is **Rectangle**. The rectangle macro allows you to create automatically a two-dimensional region bounded by 4 planes (you must be working on the plane  $z = 0$ ). Since it is 2D, you only require 4 values x min-max and y min-max. Figure 4.36 shows the dialog box of the rectangle macro.

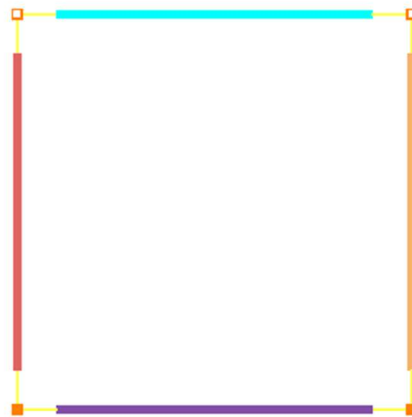


Figure 4.45: Rectangle topology macro

- **Circle**

The Circle macro enables you to create a circular surface and its corresponding topology. It can be created with a centre and a radius. Circle centre in x and y coordinates and radius needs to be defined. Figure 4.46 shows the dialog box of the circle macro.

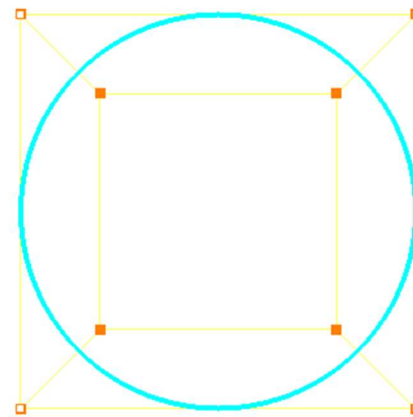
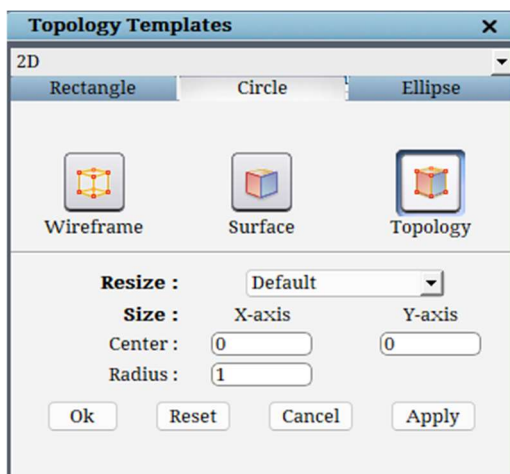


Figure 4.46: Circle topology macro



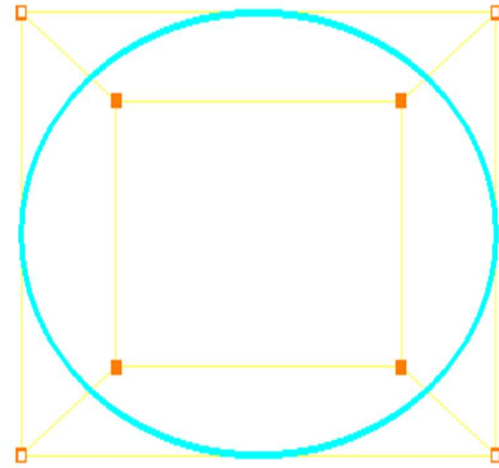
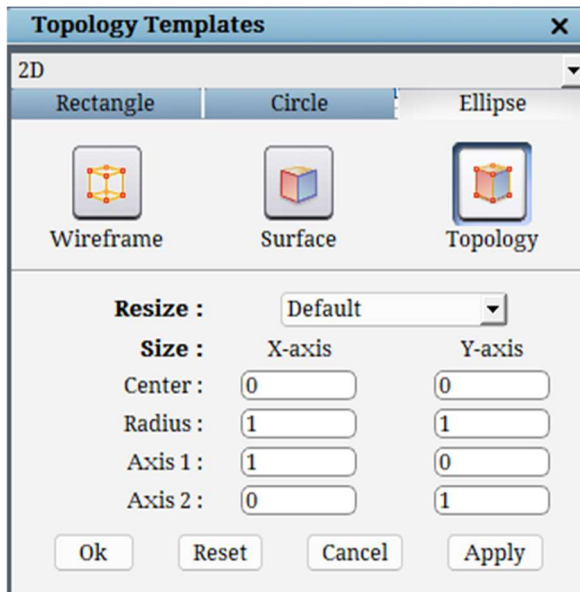


Figure 4.47: Ellipse topology macro

- **Ellipse**

The *ellipse* macro enables you to create an elliptical region of any shape with a centre, axis and radius. To change the shape of the ellipsoid, change the radius value of axis 1 and axis 2. You will create a circle by giving the same radius value on both fields. Instead, you can use the circle macro. Since it is a 2D surface, only two coordinate axes are required. Figure 4.47 shows the Ellipse dialog box.

### 4.3 Automatic Topology Creation

Automatic topology creation tools help in creating topology automatically using the surface definition. There are two automated tools in the GUI, namely *Holes* and *Extrude*. Holes helps in the automatic creation of topology and internal surfaces for surface holes. Extrude helps in the automatic creation of topology for any geometry which is tubular in nature.

### 4.3.1 Holes

**Holes** or **Hole topology** helps in creating topology and internal surfaces automatically. For. E.g., consider an injector that has many pipes connecting two surfaces, as shown in Figure 4.55. In such cases, topology has to be built individually for each pipe using insert and wrap. To ease this process, you can automatically create an individual topology for each pipe using the Holes command located under the *Topology Tools* menu. Clicking on **Holes** will open the dialog box, as shown in Figure 4.48.

It has many sections, of which the first one is to select the type of holes. There are four scenarios for which the hole topology can be used.

1. **2 sided** – when you want a grid on either side of the hole.
2. **1 sided** – when you want a grid only on one side of the hole. This will also create a flat internal surface such that it closes the hole on the other side.
3. **Mushroom** – when you want a grid only on one side of the hole but not the entire length of the hole. This will not create any internal surface on the other side of the hole. It will be like a hanging topology that has to be taken care of by the user.
4. **Punched** – when you want a grid only on one side of the hole but not through the hole.

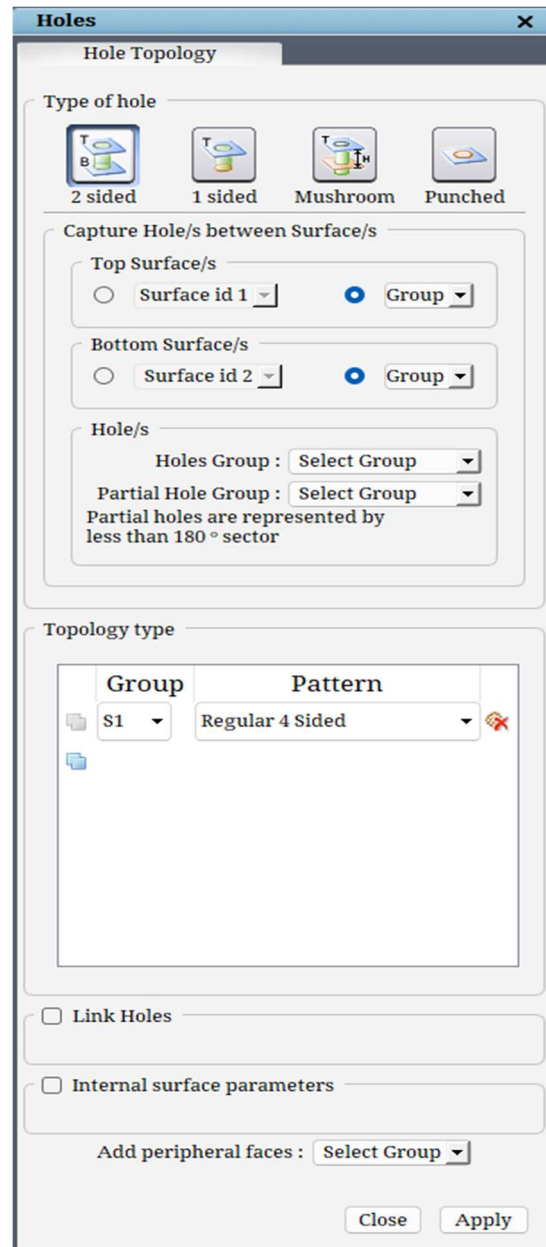


Figure 4.48: Hole topology dialog box

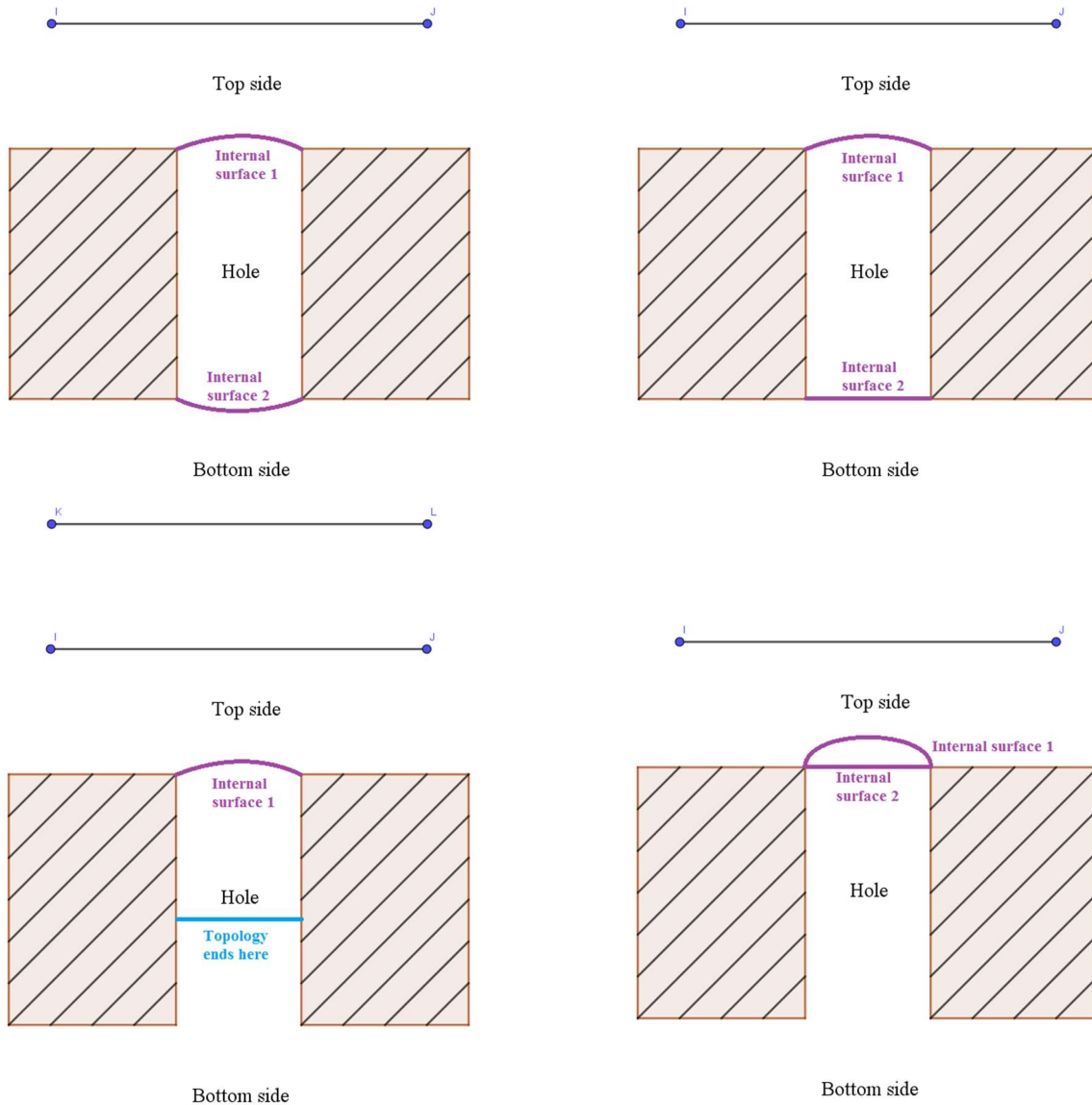


Figure 4.49: Type of Holes

For each type, the input parameters vary, which is addressed in the sub-section, ‘**Capture Hole/s between Surface/s**’. Three parameters such as top surface id/group, bottom surface id/group and hole/s surface id/group have to be input for both 2 and 1 sided hole types. For the mushroom and punched hole type, you only need top surface id/group and hole/s surface id/group. From v8.1, we can create hole topology for partial holes as well. i.e. half or quarter holes. The holes represented by an angle lesser than 180 degrees should be given as input for the **Partial hole group** and the rest under the **Holes group**. Note that the partial hole topology works only for circular holes.

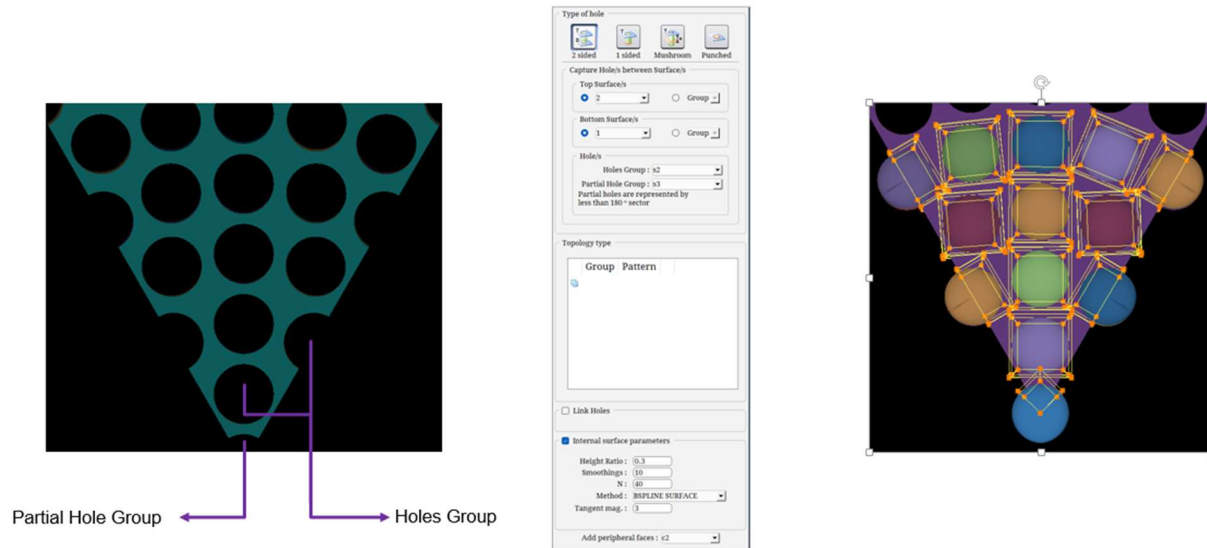


Figure 4.50: Hole topology for partial/half holes

Next section is the **Topology Type**, which can either be 4 sided, 4sided with longitudinal insert or 4sided with Bi-directional insert. 4 sided means the output topology has 4 faces like in cube(excluding front and rear), 4sided with Longitudinal Insert will have a block split along the longitudinal direction and 4sided with Bi-directional Insert will have block split on either direction.



Figure 4.51: Type of Topologies

These are the mandatory input parameters necessary for generating the hole topology. More options are grouped inside Link Holes and Internal Surfaces checkbox for advanced controls. Enabling the **Link Holes** checkbox opens the required parameters, as shown in Figure 4.53. Link Holes help in linking all the individual hole topology based on the pattern provided. The pattern can either be calculated automatically using the pre-defined patterns or can be edited manually after getting the respective hole centres. There are 5 different pre-defined patterns which are as follows:

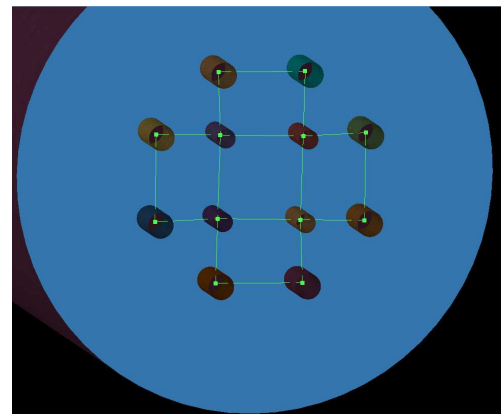


Figure 4.52: Pattern calculated using Rectangular option

1. **Hole centres** –Creates a corner at every hole centre for linking them manually if the user does not get the desired pattern from the pre-defined options.
2. **Linear** – Captures the linearly arranged holes.

3. **Circular** – Captures the holes which are arranged in a circular manner, for example, Holes in Injectors.
4. **Rectangular** – Captures the holes which are arranged in a rectangular pattern.
5. **General Tria** – Captures the holes using the triangulation (surface mesh) of the surface.

All these options capture the hole centres perfectly; only the linking pattern of the holes vary as per the selected option. Once the option is selected, a green dot appears at each hole centre and is linked together as per the selected option. If none of the options has given the desired linking pattern, users can select hole centres which will create the green dots at each hole without linking them. Enable the **Edit pattern** checkbox to convert the green dots to corners and group them in the desired corner group using the **Edit in CG** dropdown. Now you can link them as per the pattern you desire and generate the hole topology. Make sure the pattern that you provide are either linked one to one or like a quad structure. Please avoid creating a triangular loop or any other structure. Once the green dots are converted to corners, it has to be deleted before proceeding with the topology building. It can be done by enabling the Delete pattern in CG checkbox.

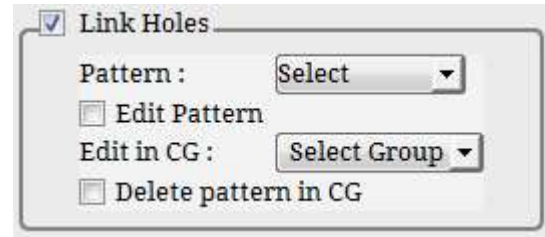


Figure 4.53: Link Holes dialog box

The next section, **Internal surface** parameters, provide the options to control the internal surface shape. Enabling the respective checkbox opens a dialog box, as shown in Figure 4.54. It is same as the parameters given in the Internal Surface creation. Please refer section 4.11.8 for more information on those parameters.

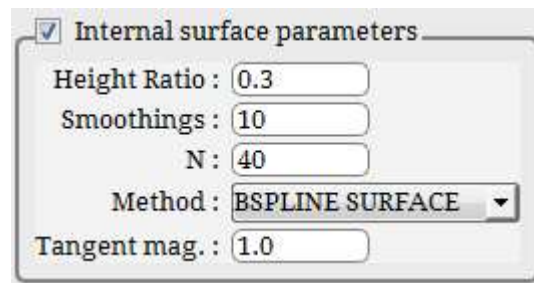


Figure 4.54: Internal Surface dialog box

The last option is to group all the outer corners of the hole topology to proceed with the topology building for the rest of the regions. If a corner group is selected from the dropdown, all the outer corners of the output topology will be grouped in the corner group.

Please refer *Hole\_Topology.pptx* for detailed information on the tool to execute it from the command line.

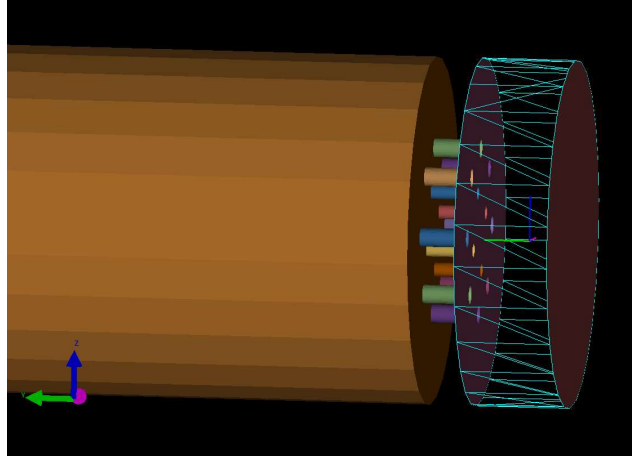


Figure 4.55: Injector Surfaces

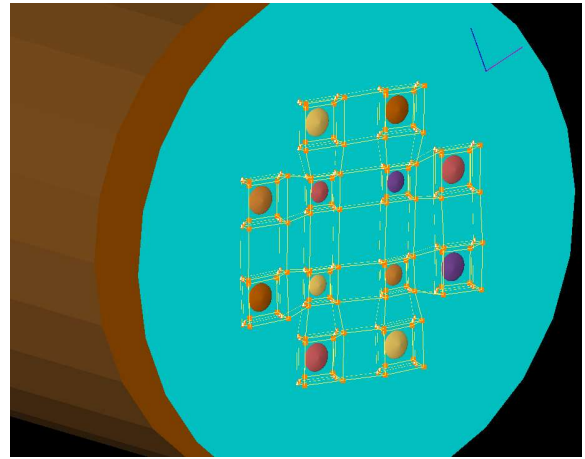
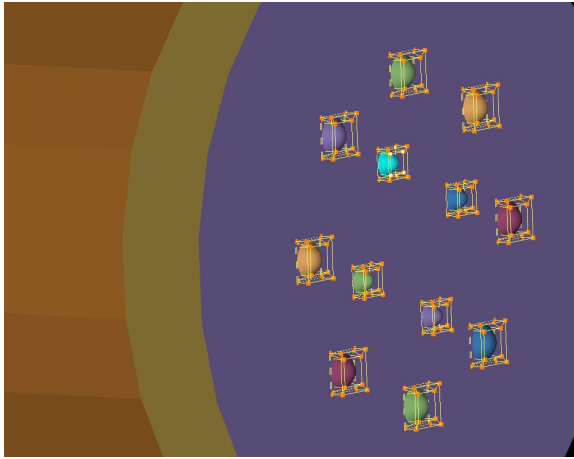


Figure 4.56: After executing hole topology command with and without links

### 4.3.2 Extrude



**Extrude** button is located under *Edit(Topology)* section, which helps in automatic topology creation for any given geometry which is tubular in nature. For e.g. Consider a duct that has circular/elliptical cross-section throughout. Given a cross-sectional 2D topology faces, the extrude tool automatically traces the surface curvatures and copies the given 2D faces smartly and links them together. Clicking on **Extrude** will open a dialog box, as shown in Figure 4.57. It has three sections, of which the first section is called as the **source** where the input 2D faces should be selected under **Input face group**. The second section is called **Guideline**, which has two input options, namely Reference Type and Surface/Wireframe group. The **Reference Type** has three options such as Surfaces, Curve and wireframe. If **Surface** is

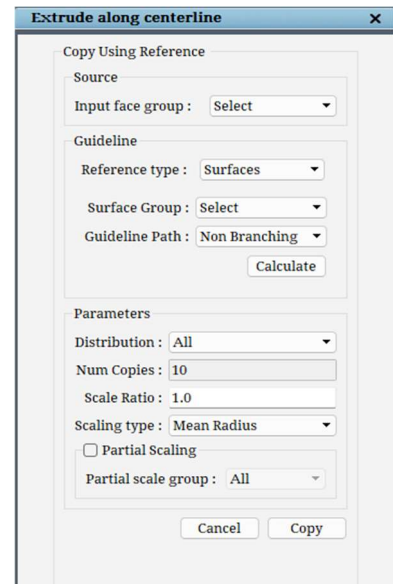


Figure 4.57: Extrude dialog box



invoked, then the user needs to provide a surface group under **Surface Group** option whose centerline has to be calculated by clicking on Calculate button manually. This creates a chain of corners representing the centerline of the geometry which will be used as a reference path for copying. If the centerline is not branching, you can proceed with the subsequent options and generate the topology by clicking on Copy button. In case if the geometry is branching, select '**Branching**' option under '**Guideline Path**' and click on calculate. The resultant centerline would also be branching, which you may have to edit and delete the branching edges and use them as a Reference Type by selecting **Wireframe** option from the dropdown list. There is one more option under Guideline Path, which is called '**Branching at the end**'. As the name suggest, if the branching is at the end of the geometry. For example, Y exhaust or carotid bifurcation. However, for both the branching cases, you need to delete the branching edges and use them as a Wireframe Reference type for extruding the faces.

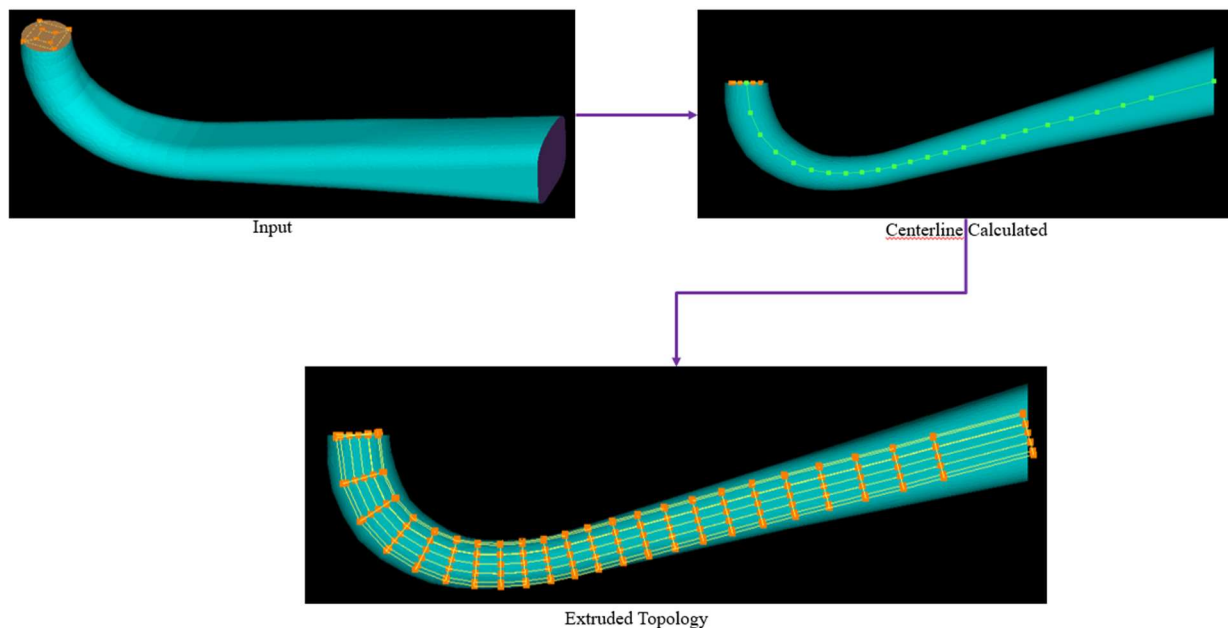


Figure 4.58: Extrude using reference type - Surface with uniform scaling

If **Wireframe** is invoked, a corner group which represents the centerline of the surface should be selected under **Wireframe Group** option. If **Curve** is invoked, user needs to provide the curve's surface id under **Surface Group**, whose path will be used for copying.



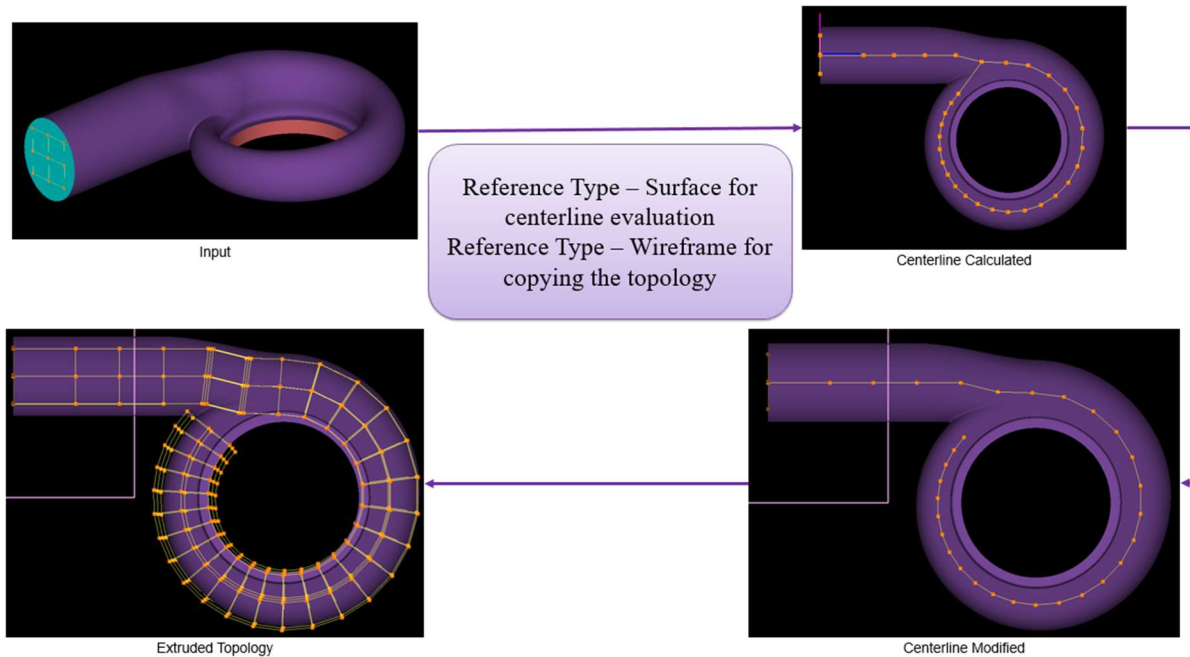


Figure 4.59: Extrude using reference (surface) - Branching corners deleted - new reference type (wireframe)

The last section is called **Parameters** which has many options to control the size and shape of the topology creation along the length of the geometry. The **Num Copies** option decides the no. of copies to be created and copied if the **Distribution** is set to *Uniform*, whereas if the distribution is set to *All*, the block faces are copied at all the nodes of the centerline. The curvature sensitive option is yet to be implemented. Next, the scaling ratio and scale type help in scaling the 2D faces based on the cross-sectional radius of the geometry to where it is copying. The **Scaling Type** has three options under the dropdown such as min, max and mean radius. At every cross-section, the radius of curvature will not be the same all-around 360 degrees. Consider an elliptical cross-section; the radius at every angle is different; for such cases, the min, max and mean radius helps in determining the scaling radius. However, the *Mean Radius* is used as default scaling type. Finally, you can choose to scale the entire 2d input block faces or a particular face group. Enabling the **Partial Scaling** checkbox enables the **Partial Scale Group** dropdown where you can choose the corner group to be scaled. If the reference type is set to *Surfaces*, then the extruded topology automatically scale the input faces based on the geometry's cross sectional radius.

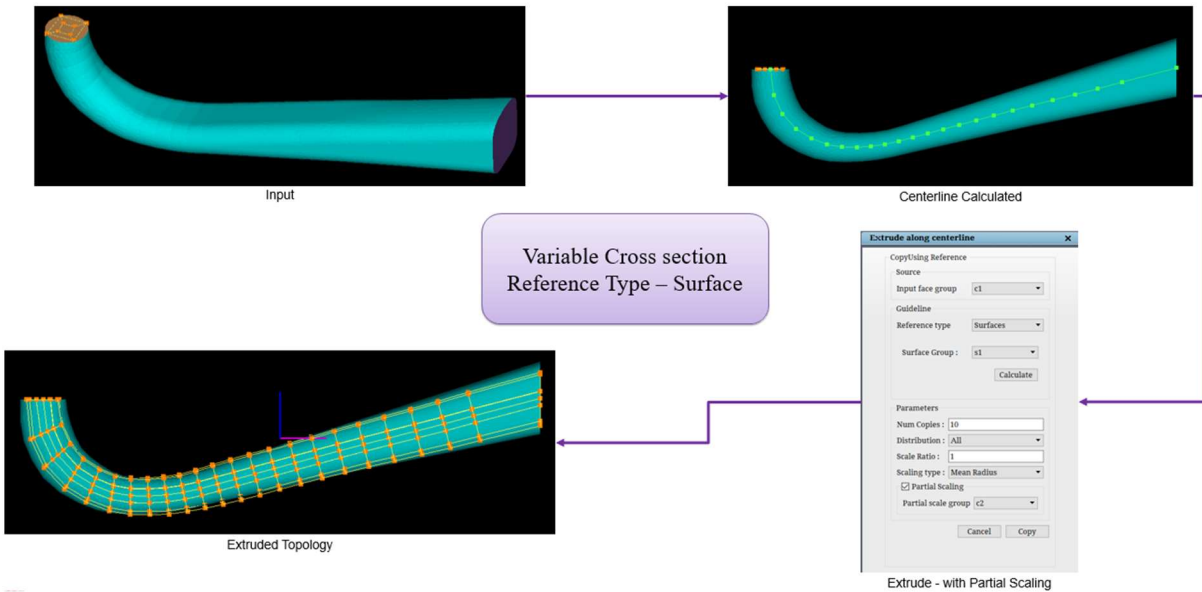


Figure 4.60: Extrude with Partial Scaling

\*It should be noted that this tool fails to work for branching pipe/tubular geometries. Hence you need to manually delete the branching edges. Press Esc to go to edit mode, which will convert green highlight to corners and edges.

### 4.3.3 AutoMesh Volute

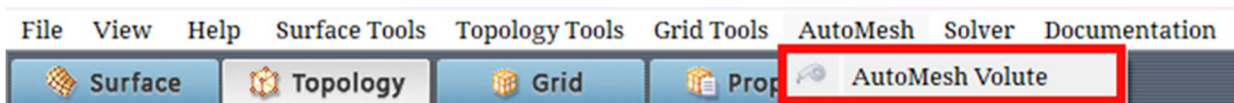
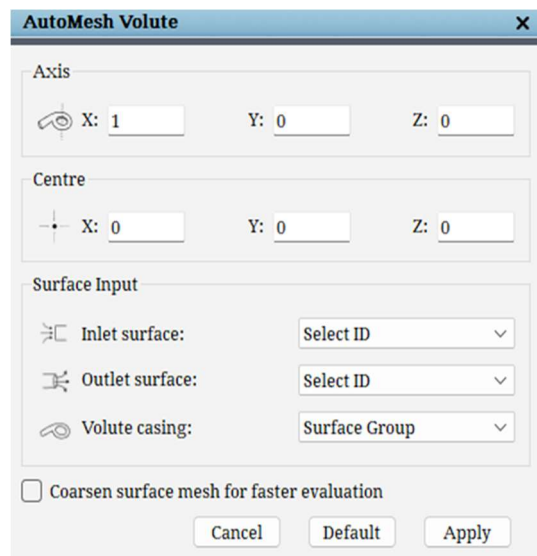


Figure 4.61: Accessing AutoMesh Volute from the menu bar

A Volute is a curved funnel that increases in area as it approaches the discharge port. GridPro has developed a tool which is automatically able to generate a topology and then mesh on a volute geometry. This greatly reduces the time required in generating a structured mesh on volutes. The tool can be accessed from the **AutoMesh** section of the menu bar. Clicking on **AutoMesh Volute** opens up a dialog box on the right side.



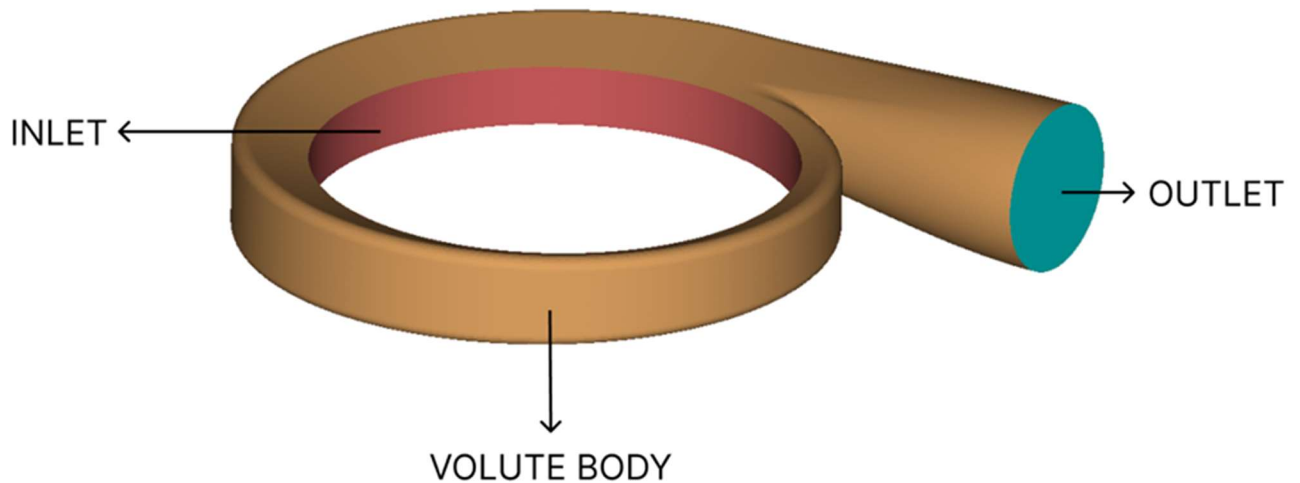
Here, the respective parameters can be entered. The **Axis** represents the axis passing through the eye of

Figure 4.62: The AutoMesh Volute dialog box

the volute. Similarly, the **Centre** of the volute is the centre of the eye through which the axis passes.

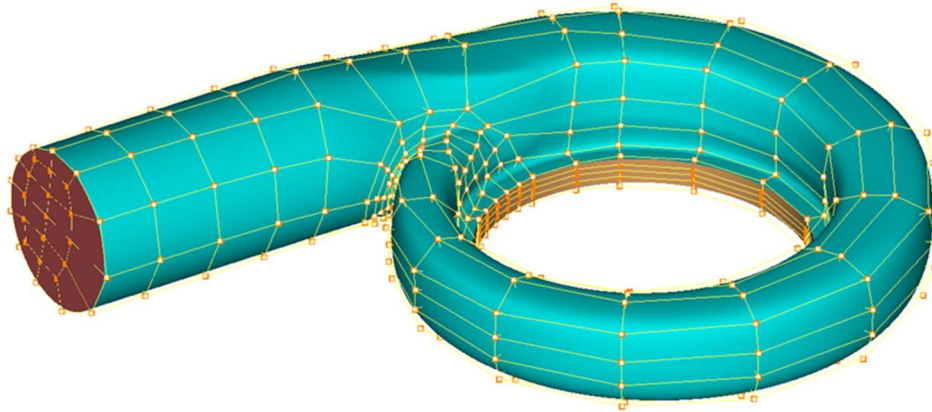
Coming to the **Surface Input** section, there are 3 parameters which have to be entered by the user here. They are the **Inlet surface ID**, the **Outlet surface ID** and the **Volute casing surface group ID**.

There is also a checkbox to **Coarsen surface mesh for faster evaluation**, which can create the topology faster on geometries with a large number of triangles.

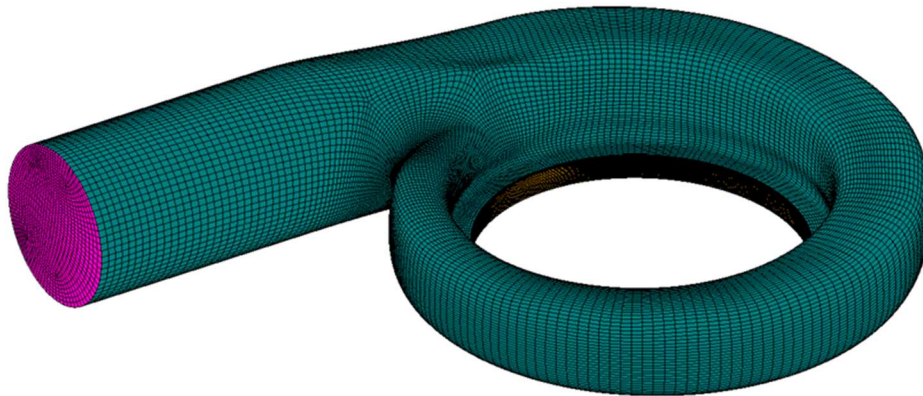


*Figure 4.63: The surfaces to be input*

Clicking on **Apply** runs the automation code which outputs a valid topology. GGrid can be run on this topology to create a mesh which can then be used.



*Figure 4.64: The topology which is automatically created using the automation tool*



*Figure 4.65: The grid created using the generated topology*

## 4.4 Topology Grouping Utilities

One tool that helps you create three-dimensional topologies is the workplane; another tool is a group. Often, you will repeat procedures for creating topologies because grid topology often requires the user to repeat operations. *GridPro* has developed the group feature in order to help you circumvent the procedural repetition so prevalent in the field of grid generation.

More than any other feature, groups help you manipulate three-dimensional topology and use graphics effectively. Creating a typical topology composed of hundreds of corners and edges without utilizing groups is ill-conceived, time-consuming, and unnecessary. Similar to

topology groups, you can group surfaces as well. The surface group is as much useful as corner groups as it helps you to differentiate groups of surfaces from all the surfaces loaded into GUI.

### 4.4.1 Corner Groups

Before learning what a group can do, you should first learn how to access a group. A maximum of nine groups can exist for any topological components in a single topology file. Each group is numbered, one through nine, with 'c' as a prefix for corner groups and 's' as a prefix for surface groups. Groups are located below the toggle display section in the right-side panel. Groups are divided into two tabs, such as **Corner**

**Groups** tab and **Surface Groups** tab. The corner groups tab can be seen in Figure 4.66. When you depress one of these buttons, the number on that button activates that group number. Therefore, when the [c6] button is depressed, group #c6 is active. Only one group can be active at a time. When you switch ON a corner group, a message appears at the status bar with the no. of corners and no. of faces grouped in that corner group.

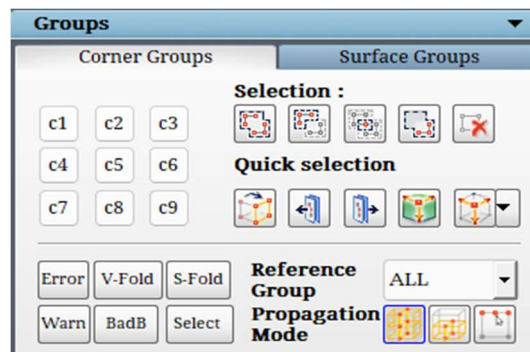





Figure 4.66: Corner Groups tab

Corners can only be added to or removed from the active groups. Initially, no corners exist in any group. Groups can be thought of as empty containers. Therefore, at first, you can only add corners to groups. After depressing a (one through nine) group number, you may add corners to that corresponding group by depressing the  button, i.e. second icon in the selection section in corner groups tab. This  operation allows you to drag a red box with the right mouse button about corners. The corners within this box will be added to the group number depressed. Just as the  button allows you to add corners to

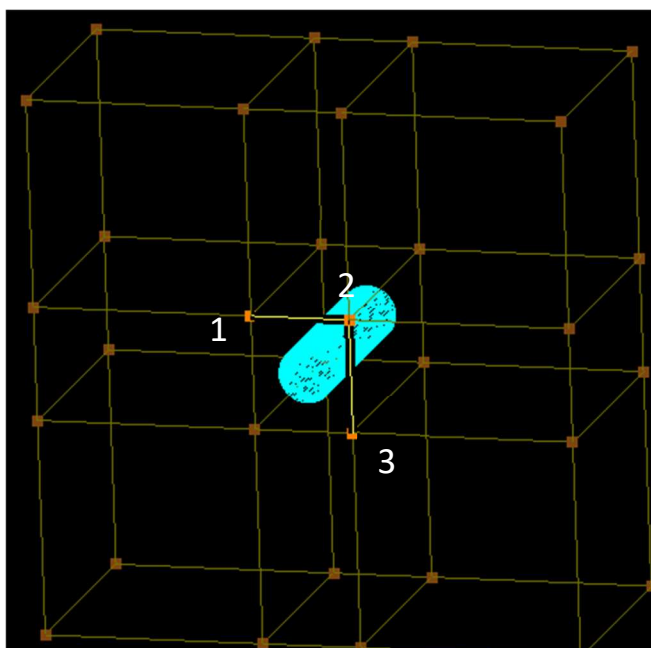










Figure 4.67: Grouping corners by picking individually

groups, the  button lets you subtract corners from groups. This  button requires you to drag a red box around corners. Of course, you can only take corners out of groups to which you have already added corners. All the buttons under the Selection section are persistent, i.e. it acts



as a switch ON and OFF button. The functionality continues until you switch it off manually by clicking on the button again. In addition, to add or subtract a group of corners using  and , you can also add or subtract corners by picking a corner. Switch ON the  button and click on a corner which is not in the group that will add to the active corner group. Similarly, switch ON the  button and clicking on the corner which is already there in the group will subtract that corner from the group. This operation is also persistent; it will be effective until it is switched off manually again by clicking on it.

The next two, intersection and equal operation, help you efficiently add corners too or remove corners from groups. When a group already containing corners is active, depressing the  button allows you to drag a red box with the right mouse button. Any corner/s inside this box that is part of the active group will be selected, and all the other corners outside the box get unselected, no matter what their position or state, will be cleared from the group (if they are not already added in the group). The  button (similar to equal sign) acts like a solution of addition, subtraction and intersection of corners of different groups. It can do multiple additions of corners from specific groups or topology or subtraction in a single operation.

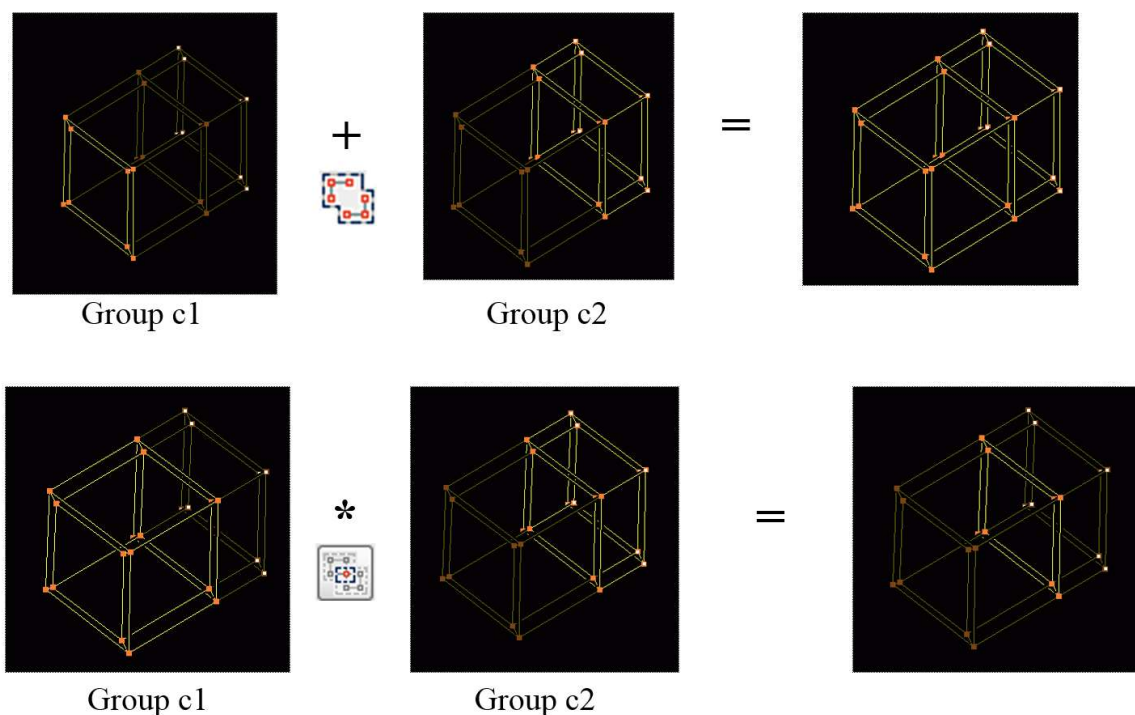





Figure 4.68: Grouping using two different corner groups

Groups often confuse beginners because groups, at first, may appear to be abstract. In effect, the activation of a group is like the creation of an empty container. This empty container

discards its abstract label when you fill the container with corners and edges. As already described, you can add corners to groups with the  button and the red box created by dragging the cursor over the corners that you wish to add.








In addition to adding/subtracting corners to/from a group, you can add/subtract two groups, i.e., add the corners in a group number to another group number by using the  button. Similarly, subtraction and intersection between groups can also be done using the respective buttons. This avoids the duplication of your effort in grouping the corners again and again.

When a group already containing corners is active, pressing the  button will automatically subtract all corners in that group. It clears the selection of the respective group. This helps in reusing the group again and again.

From v6.7, a new widget is implemented with specific buttons, as shown in Figure 4.69. These buttons are more like separate groups, which help in a quick selection of the desired corners.



Figure 4.69: Quick selection widget

Depending on the button switched ON in the selection section, the buttons either add, subtract or intersect with the selection.  represents all the corners in the UI. If  button is ON and you click on All, it will add all the corners to the active group.  represents the surface assignment group as explained in *section 0*,  denotes the function span out external as explained in *section 4.4.5*,  represents the A key functionality or sheet selection function as explained in *section 4.4.6*,  represents the mode of grouping is set to screen and  helps in closing the widget. All these options are explained in the following sections.

## 4.4.2 Grouping Modes

Grouping modes allow you to select the corners or edges that you would like to display while hiding the others (not removing). You can either view all the corners in the UI irrespective of the groups, view only the corners grouped in the active group, view the active group with reference to another group. To achieve this, GUI provides special grouping modes. You can change the grouping display mode as per your requirement. These grouping modes are located under [All] dropdown list in the **Groups** section. There are 13 group display modes available, as shown in Figure 4.70. Each one represents a topology mode, one that controls how active groups are distinguished from inactive groups. You can select one of the modes from the dropdown list. By default, the grouping display mode is All. When **ALL** mode is active, the

ALL
GROUP
REF-1
REF-2
REF-3
REF-4
REF-5
REF-6
REF-7
REF-8
REF-9
ERROR
Volume Folds
Surface Folds
Warnings
Bad Blocks

Figure 4.70: Grouping modes



corners in the active group and the edges connected between those corners are highlighted while all of the surrounding topologies are dimmed. **GROUP** mode refers to Active Group. When the group display mode is active, the corners in the active group and the edges connected between those corners are highlighted while all of the surrounding topologies are hidden. **REF** groups mode refers to Reference to other selected group. When Ref-1 mode is active, the corners in the active group and the edges between those corners are highlighted while all of the topologies in group #1 is dimmed and rest all topology entities are hidden.

Similarly, if Ref-2 mode is active, the corners in the active group and the edges between those corners are highlighted while all of the topologies in group #2 is dimmed. Same applies to Ref-3, Ref-4 ... Ref-9. In particularly complex topologies, you may have difficulty focusing on one set of corners. Setting the group mode to reference group (Ref-1 to Ref-9) localises your work,

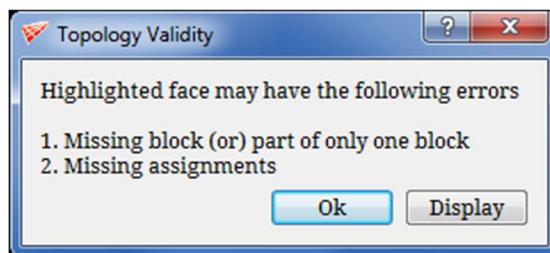


Figure 4.71: Validity check

eliminates screen clutter, and effectively manipulates specific corners. Error group mode refers to the corners which are highlighted by the Validity button as invalid topology. The respective corners are automatically grouped into **ERROR** group, and it can be displayed by clicking on the 'Display' button in the 'Validity check' pop-up box, as shown in Figure 4.71. **Warn** group refers to the corners which will output a bad quality. These are highlighted by the Validity button as singular topology. For example, the topology may have reported mildly, mediumly, or very severe singularity and the corresponding edges would be highlighted by the validity button. Similar to **the Error group**, **the Warn group** automatically groups those singular corners into it. It can be displayed by clicking on the Display button in the Validity Check pop-up box. In addition to this, bad topology faces can also be added while checking the quality of the grid, as explained in *section 5.5*. The **V-Fold** group displays corners/ blocks which have grid volume folds, while **S-Fold** group displays corner sheets which have surface folds. V-Fold and S-Fold groups have content only if there are volume or surface folds generated while checking the quality of the output grid; else, they are empty. Finally, **BadB** is a group which contains all the faces/blocks of the topology whose positions are not good.

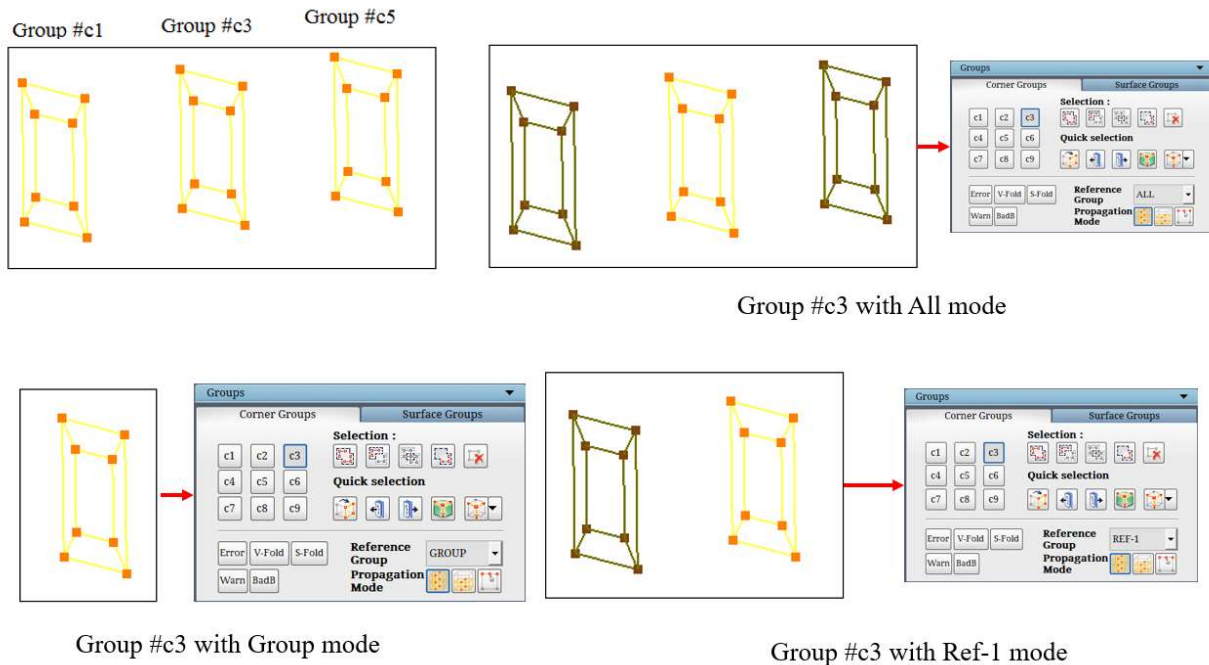




Figure 4.72: Different grouping modes

### 4.4.3 Surface Assignment Group

The  button is a special group. However, this group always contains the corners assigned to the current surface and is thus named the surface assignment group. Therefore, you can change the contents of the group just by changing the current surface. Often, when surfaces are close together, the corners assigned to each surface appear to be right on top of one another. Using the  button will allow you to pick out the corners assigned to one of those surfaces and allow you to manipulate just the wireframe about that surface. Like any other grouping operation, you can use the corners in the group to add/subtract to/from any group, but corners cannot be added/subtracted to/from this group like any other group.

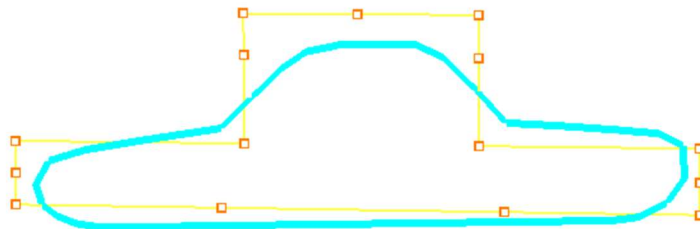






Figure 4.73: Surface assignment group

#### 4.4.4 Group Retrieve Option

Similar to undo or redo options for topology building, the corner grouping actions within a group can be retraced. If you have added corners to a group using more than one red box (adding corners to a group at different times), you can retrace your previous steps by pressing the  and  buttons found in the **Quick selection** section. Practically speaking, these buttons are very useful. By enabling you to visit previous selection in a group, these buttons allow direct access to the parts of a group that was the result of your actions (e.g. wrap, copy, add, subtract, etc.).  shows previous selections while  shows forward selection.

#### 4.4.5 Displaying mistakes in topology and grid

The **Error**, **V-Fold**, **S-Fold**, **Warn** and **BadB** options help to view the blocks and faces having issues. Some of these options are also available under **Reference Group**. **Error** option displays mistakes resulting in invalid topology such as wrong assignment, missing corners, etc. **Warn** displays mistakes in a valid topology like singularities assigned to surfaces, absence of internal surfaces, etc. **V-Fold** and **S-Fold** options display the blocks and faces containing grid volume folds and surface folds, respectively, formed in the grid generated using the topology. **BadB** groups all the faces of the topology whose block positions are not good.

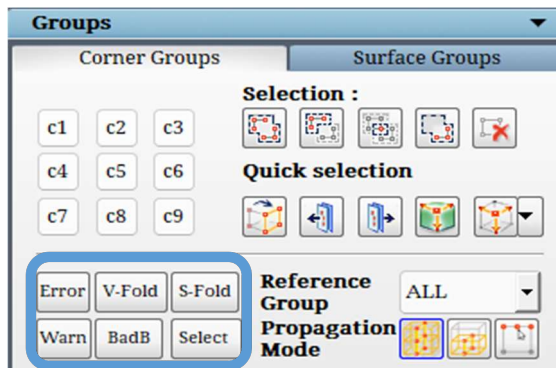




Figure 4.74: Other groups

#### 4.4.5 Span out

You can directly add or remove sheets of corners from an active group by using span out or peel off button. The drop-down list under this button is shown in Figure 4.75. It has nine options, five options belonging to span out, which means adding corners to an active group and four options belonging to peel off, which means subtracting corners from an active group. Using **span out** layer 1, you can add one layer of corners in all directions connected to active group from the reference group. Similarly, you can add 2 layers or 3 layers or 4 layers of connected corners to active group from the reference group. [Reference group refers to the corners

displayed as dimmed in addition to active group, it depends on your selection of group mode display (A or R)]. **Span out all**, adds all the corners in reference topology to the active group. This option is very useful for checking the connectivity between the topology. If you have two distinct pieces of topology, you can separate them by using this feature. **Span out blocks** , groups all the remaining block faces of the already grouped faces but are not completely defining its blocks. It helps in grouping the blocks accurately without adding or missing any unnecessary faces. There is one other option, located under the quick selection widget as shown in Figure 4.69, which is **Span out external** . It adds all the external faces to a corner group with a click of a button. All these span out options work only if the active corner group has pre-selected corners in it.

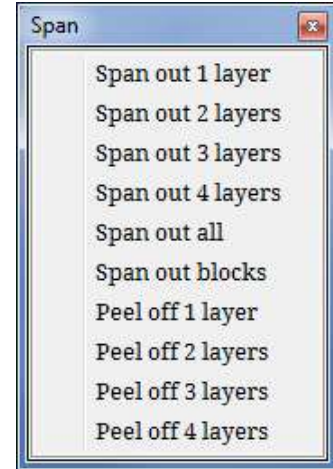


Figure 4.75: Span dropdown list

**Peel off** works opposite to Span out. Peel off subtracts corners from an active group in a similar fashion as span out adds corners to an active group. Similar to span out, to carry out peel off, you require some inactive corners in addition to the active corner for reference.

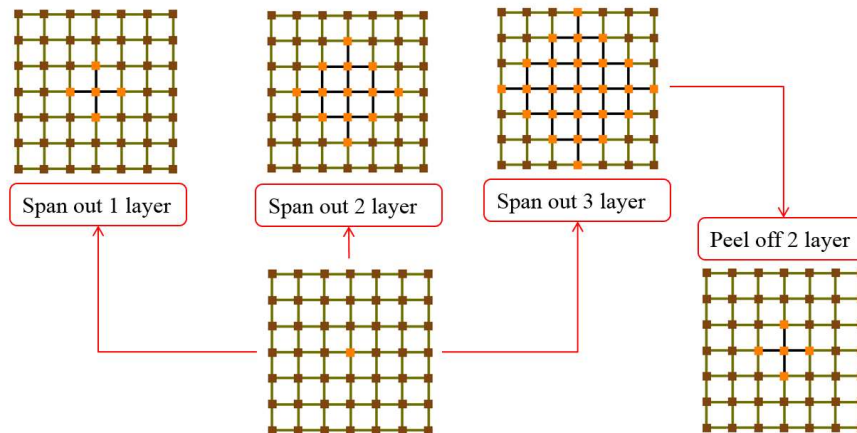







Figure 4.76: Examples for Span out and Peel off options

#### 4.4.6 A KEY

In addition to all the buttons for grouping operations, keyboard shortcut such as **A** key is available to select a sheet from an active group of corners. It can also be called as **sheet selection** button . It helps you to visualize how corners are propagating throughout the topology. You can pick a cross-sectional sheet from an active group of corners by holding **A** key and clicking on an active edge near the sheet which you want to select. You can use , , and  options along with **A** key to add, subtract or intersect sheets from an active corner group, respectively. Figure 4.77 shows the sheet selected after using **A** key .

Further, the corners picked by **A** key or **Insert** depends on the **Propagation Mode** available in **Groups**. By default, it is **All**, where all the corners in the topology are available for picking by **A** key sheet picking option. With **Group** Propagation mode, only corners in a particular group are available for picking, while in **Edge** propagation mode, only the edges in connectivity are available for picking.

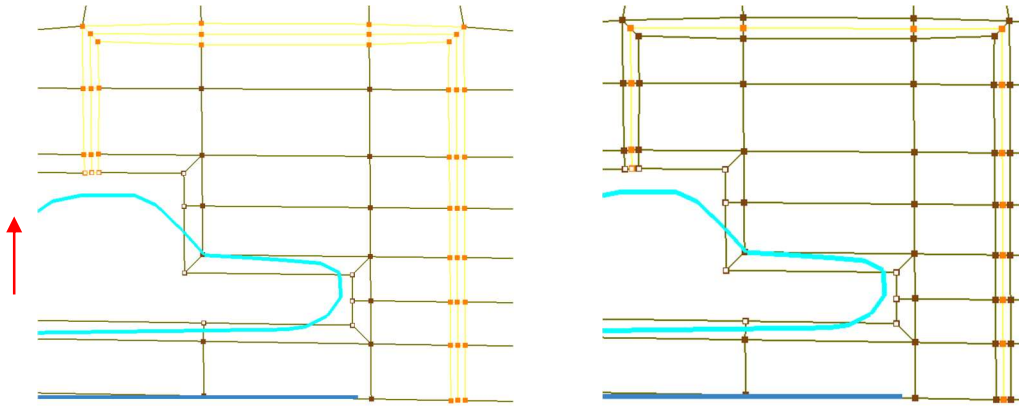



Figure 4.77: Sheet selection

#### 4.4.7 Selection Pane

Selection pane is a GL widget which has three buttons to aid easier grouping. Grouping the desired faces for any operation, such as a wrap, copy etc., is always a critical task that too on complex cases where there are lots of blocks, it becomes a more time-consuming process. Sometimes you may end up grouping some extra faces accidentally, which you may not realize until it's too late. With the help of selection pane, the grouping operation is made more accessible. The pane has three buttons as shown in Figure 4.78.



Figure 4.78: Selection Pane

The first one is selection group which is a temporary selection group where you click and group the desired faces, which will be grouped under a group called 'Select' . The faces grouped under select can be used for subsequent operations. However, the group will be reset whenever the selection group button is clicked.

The second option is the face group, which lists all the available face groups of the existing topology. The face groups are same as the one listed under the generic button, and these are evaluated based on various topological rules. In some cases, it may not be able to evaluate the face groups due to those rules, during which a pop up will be shown as in the

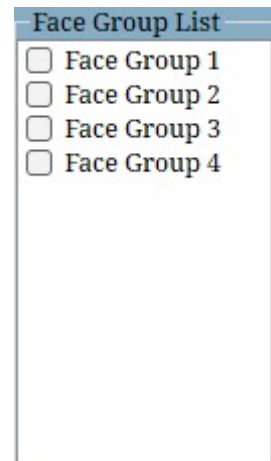


Figure 4.79: Face Group List

Figure 4.79.

The last option in the selection pane is the surface selection which helps in selecting the desired surface to be the current surface. All the surfaces are displayed in different colours, but the selected surface (current surface) is always displayed in sea blue colour. Clicking on the surface selection turns all the surfaces to grey except the current surface; once the desired surface is selected, the current surface colour will be switched to the selected surface. All the surfaces are displayed with their respective colours. For easier identification of individual surfaces, the border of the surfaces is highlighted in olive green colour as shown in the Figure 4.80.

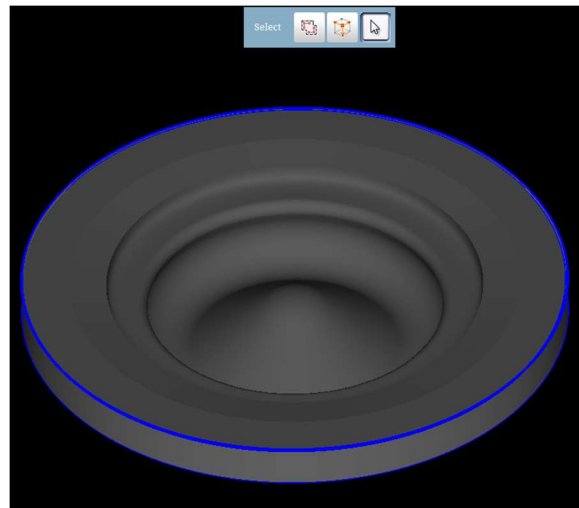


Figure 4.80: Surface Selection Mode

#### 4.4.8 Surface Grouping Tab

You can access surface grouping options by depressing **Surface Groups** tab in the **Groups** section. In **surface group** tab, similar to **corner groups**, there are nine groups, 6 grouping operation buttons and 11 grouping display modes which functions similar to corner group options. Surfaces belonging to active groups are displayed in different colours, whereas inactive surfaces are displayed as white grey colour to distinguish between them. In addition to all these operations, ALL group is also available located below selection operations in surface groups tab. It displays all surfaces in GUI as active surfaces and also enables you to use this ALL group to add all the surfaces to an active group.



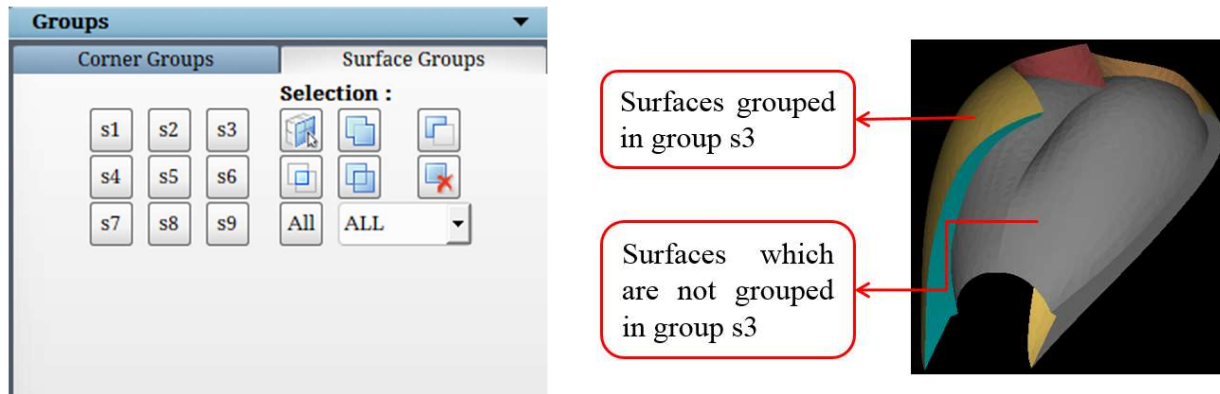



Figure 4.81: Surface Groups

Surface group operations	Corner group operations	Description
	-	Add/subtract individually
		Add single or multiple surfaces and corners
		Subtract surfaces/corners from the active group
		Retain surfaces/corners which are selected from the active group
		Add the selected surfaces/corners or selected group
		Clear the active group.
All	-	Group which is having all the surfaces/corners already grouped in it.
ALL	ALL	Display mode for the surface/corner group.
		A key - Groups the adjacent topology sheet

Table 4.1: Surface and corner group operations

#### 4.4.9 Label

**Label** refers to labelling a given group of corners or surfaces for easier selection and visualization. **Label** button is accessible from the **Edit(Topology)** section in the right-side

panel. Clicking on the **Label**  button will pop up a dialog box, as shown in Figure 4.82. You can choose to label either blocks or surfaces or both. By default, Blocks label is checked;

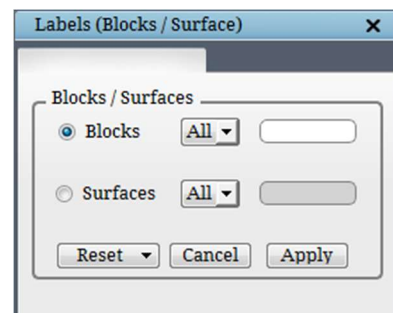


Figure 4.82: Labels dialog box



select the corner group, enter a name to label the blocks, and click **Apply**. Once it is labelled, the same can be seen in **Group** mode under the **Reference Group** dropdown list once it is labelled. Once the topology is saved, it can also be seen in the \*.fra file.

Similarly, the surface groups can also be labelled. The surface labels will be helpful to assign properties to multiple surfaces easily in a single step using the ‘labels\_to\_properties’ command-line tool. Refer section 20 in *GridPro Utilities Manual v1.7*. Further, provision is made to label individual surfaces under the **Settings** option in **Edit(Surface)** mode. For more details, refer to section 3.2 Surface Orientation. Surface labels for surfaces collectively as a group can be seen in **Surface** modes under Reference group. The label gets displayed in the bottom right corner of the UI for individual surfaces, just besides surface id, as shown in Figure 4.83.



Figure 4.83: Surface label name displayed at bottom-right corner in UI.

## 4.5 Topology Modifying Utilities

Merge, Move, Copy, and Remove are the four buttons located under **Edit (Topology)** section that can be classified under Topology Modifying Utilities. In addition to these four buttons, Split topology, and Adapt blocks to surface, which helps in modifying the topology, are also explained. They are located under the **Surface Tools** menu in the Menu bar.

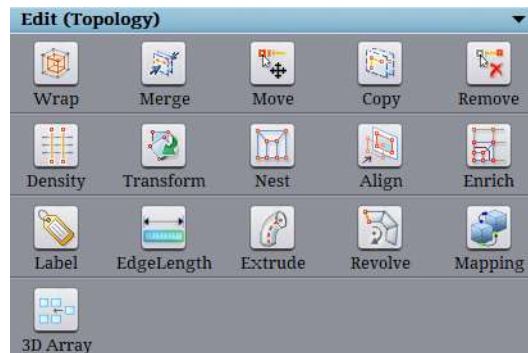


Figure 4.84: Edit (Topology) section

### 4.5.1 Merge



Merge can be done for two corners or two sheets of corners. Sometimes you would like to merge two different wireframes of a similar pattern with the same number of corners into one. Merge can be done in two ways by dragging a box around the corners using mouse right-

click, i.e. **Screen Select** or placing corners into different corner groups and selecting the respective groups, i.e. **Group**.

**Screen select** tab provides you three options to position the selected corners after merge, as shown in Figure 4.85. First is **To Centre** option that will position the merged corners to the centre of the input corners. Second is **To Selection 1** option that will position the merged corners to the first selected corners' position. Similarly **To Selection 2** will position the merged corners to the secondly selected corners'.

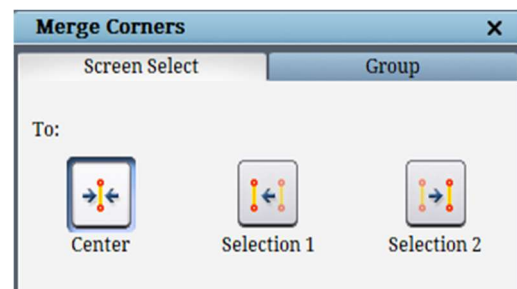


Figure 4.85: Screen select tab dialog box

After selecting the position where corners need to be placed after merging, select the corners using mouse right-click. A corner or sheet of corners can be selected. A single drag box will work if two separate sheets of corners are used, but you need to drag two boxes separately when the sheets of corners are linked to each other.

Figure 4.86 shows the corners merged at different positions.

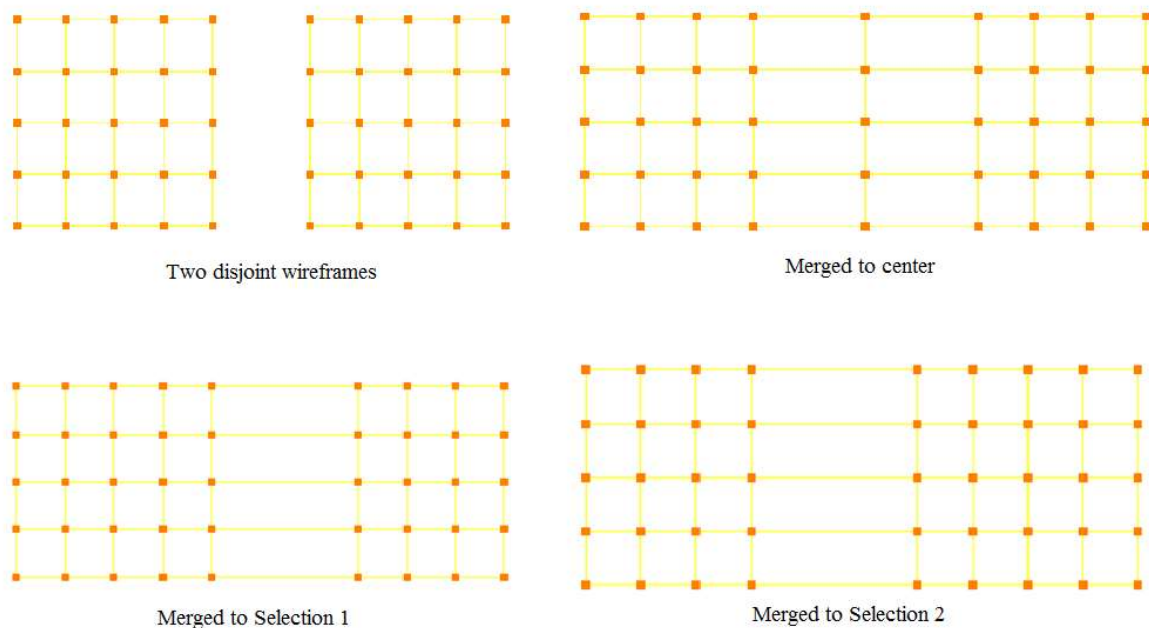


Figure 4.86: Corners merged into different positions

The other way to merge is by using **Groups**. Switch to **Group** tab under Merge dialog box as shown in Figure 4.87. First, place the two sheets of corners into two different corner groups. Then select the two respective corners groups. Then decide at which position you want to place the merged corners. Enable **To centre** if you want to merge it to the centre of the two selected corner sheets. If **To centre** is not enabled, it will position the merged corners on the second selected group.

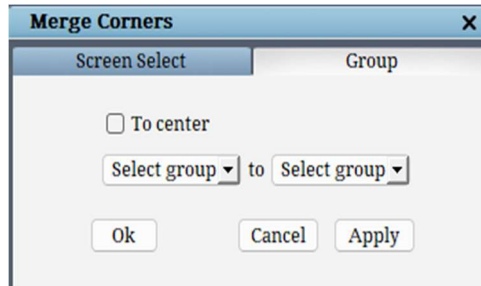


Figure 4.87: Group tab dialog box

The third way to merge is by using the **Merge edge group** option under the right-click menu on edge, as explained in *section 4.2.4*. If two adjacent topology sheets can be merged, simply right-click on the edge which connects these two sheets and select 'Merge edge group' to merge the adjacent sheets together. If it can't be merged, an error message pops up.

## 4.5.2 Move

The corners need not be positioned precisely in *GridPro* to obtain a grid. But positioning it reasonably reduces the grid generation time. So more often, you would like to move the corners in order to position it reasonably, especially when you try to converge the grid generation process. You can translate a corner by moving the cursor to the corner and dragging the corner by using the left mouse button. Figure 4.88 shows the

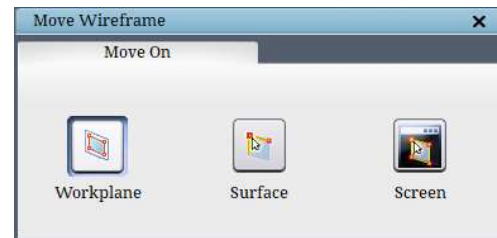




Figure 4.88: Move on dialog box

dialog box of the **Move** button. **Move on** tab, by default, allows you to move corner parallel to workplane. A corner can be moved along and normal to workplane, surface and screen (only parallel). The flags, *parallel*, *perpendicular*, *workplane*, *current surface* and *screen*, allow you to move the corner either parallel or perpendicular to the workplane, current surface or screen.

The flags such as **parallel** and **perpendicular**   are shown as float icons on the workspace once the move button is switched ON. The Move on **Screen** option works similar to workplane option, with the workplane being aligned to screen Z always. Even if the screen is rotated, it aligns back the workplane to screen Z again. Suppose the surface and parallel flags are selected, and you move the corners outside the surface. In that case, it displays a status bar message "*Cannot move the corners. Clicked position is not on the current surface*" and retains the original position of the corner.

You can also move a corner group along with the workplane. In order to move a corner group, first, fit the workplane centre to the centre of the active corner group using the **Fit** buttons in the left side toolbar and then switch on the handles of the workplane by clicking on the **Handles** button on the left side toolbar. Then switch on the **move** button and move the

workplane to the desired location where the grouped corners have to be moved. You can also shrink the corner group size (i.e. the edges connecting the corners will shrink) by shrinking the workplane size using the workplane nodes. While moving the corner groups using workplane, the flags selected in the dialog box is ignored. Note that the move button is not automatically undepressed after being used; it should be switched OFF manually by clicking on it again.

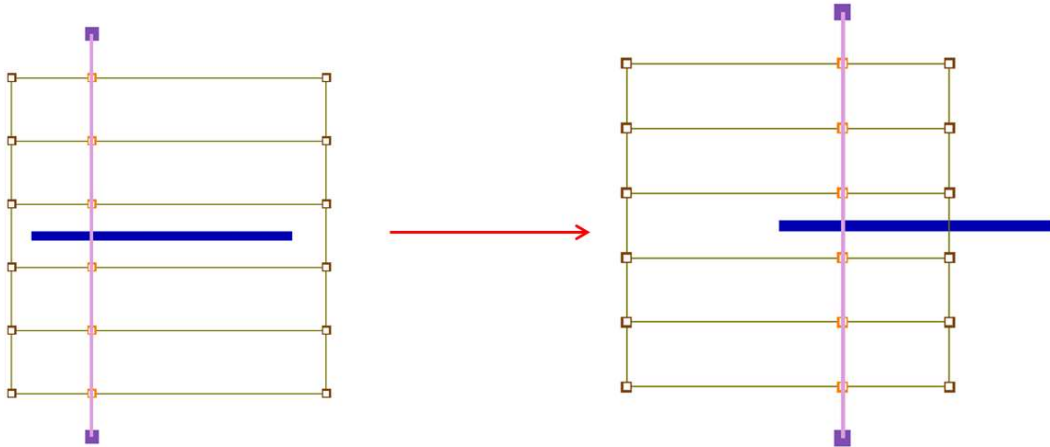


Figure 4.89: Corner group moved using workplane

### 4.5.3 Copy

It is easy for geometries with the same cross-section or symmetric sections to build the wireframe in 2D and extrude it to 3D. In order to extrude/copy the existing wireframe, GridPro

provides the **Copy** tool, which is located in the **Edit (Topology)** section of the right-side panel in **topology** tab. Figure 4.90 shows the Copy Wireframe dialog box.

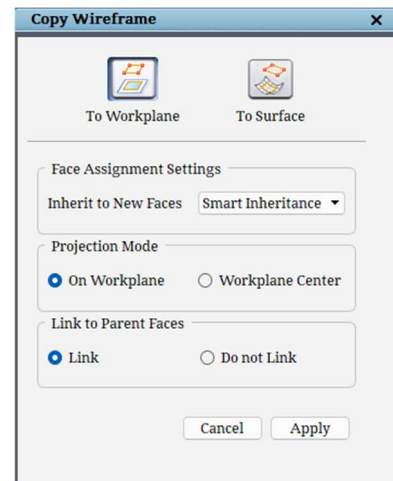


Figure 4.90: Copy dialog box

- **To Workplane**

The “**To Workplane**” option allows you to produce transformed copies of the current corner group by using the workplane. A group must be activated to use this button. The operation **Link to Parent Faces**→**Link** represents perhaps the most often used tool. In addition to the translation of the grouped corners, this option links the duplicated set of corners to original set of corners. Essentially, the **Link** operation saves you time because the operation creates a duplicate of a wireframe in a different plane, then creates edges between the wireframe and its duplicate automatically.

The other operation, **Do not Link**, simply duplicates the group structure on the workplane such that the centre of mass of the copy of the group and the centre of the workplane rectangle coincide. NOTE: *After this action, a new group is formed, one comprised of the original group and the other, its duplicate. This new group replaces the old group. In order to retrieve the old group, you must retrace an action in the new group by pressing the backup group button in the corner grouping wizard.*

Another often-used feature is **Projection Mode** along with **Link to Parent Faces** → **Link** option. The checkbox next to **On Workplane** should be enabled to use this feature in addition to **Link to Parent Faces**. This operation duplicates the active group by first moving its centre of mass to the workplane centre and then projecting each corner onto the workplane in the direction of the original movement. Then, links the original corners to the corners which compose the flattened copy of the group. This operation has tremendous value; inputting corners along a curved trajectory seldom can be accomplished with the same efficiency using other operations. Like the other operations in the **Copy** dialog box, this operation adds duplicated corners to the original group.

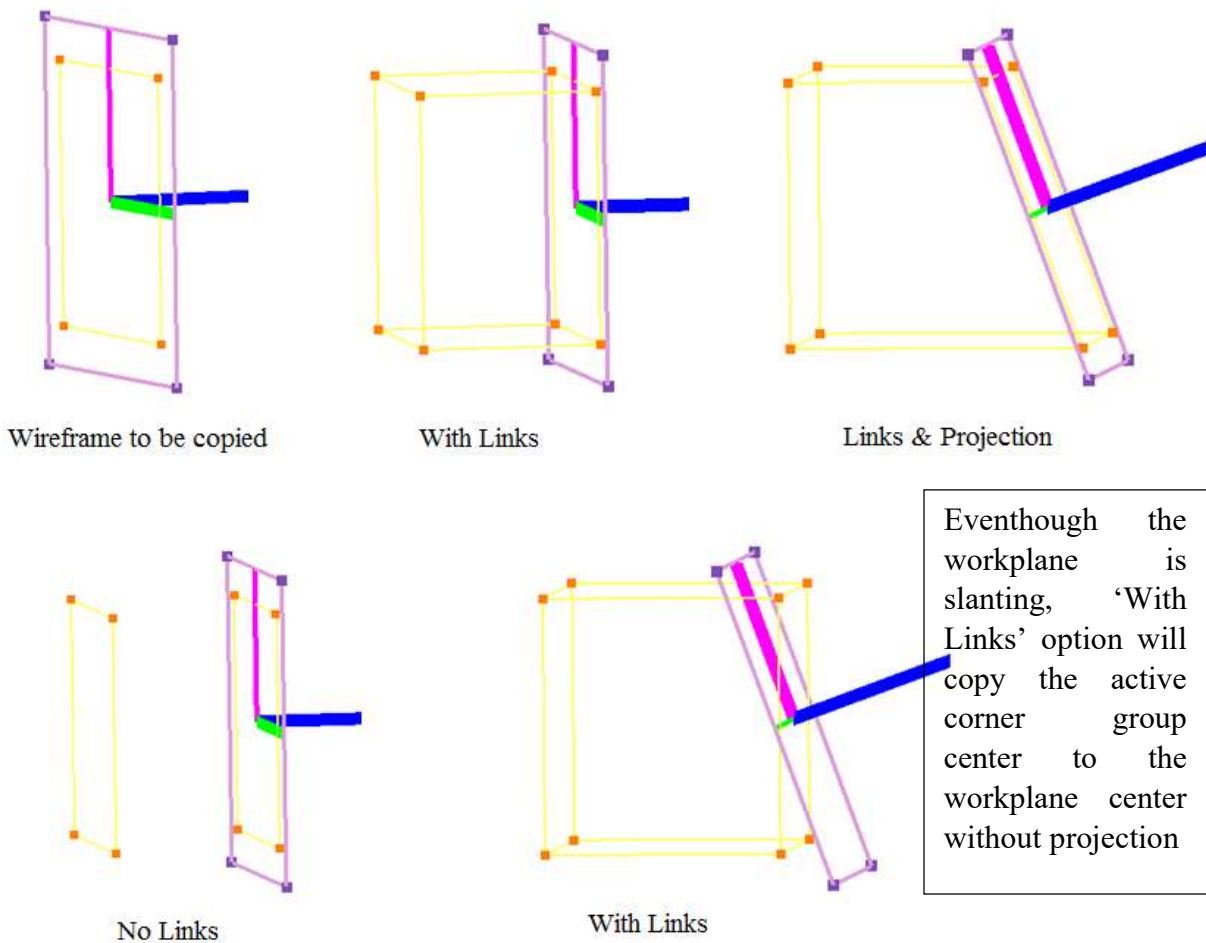


Figure 4.91: Various copy options

When **Workplane to Centre** is invoked, the corners in a group gets copied and translated to the workplane centre, but the projection is not done. Figure 4.74 shows the different modes of copying corners to workplane.

All the above 4 operations can choose various ways to carry forward surface assignments if there are any in the given corner group. It is same as the **Inheritance** option in **Wrap** button. **Smart Inheritance** automatically decides whether to copy or transfer the assignments to the copied group based on the neighbouring surface assignments. **Inherit All** copies all the surface assignments to the newly created corners. **Do not inherit** retains the surface assignments with the input group and does not copy/transfer to the newly copied group.

- **To Surface**

This option works similar to **To Workplane** except for the **Projection Mode**. Usage of options like the **Face Assignment Settings** and **Link to Parent Faces** are exactly same as into **Workplane**.

When the **Projection Mode** is **On Surface**, the copied corners are projected on to the surface, while when the sub-option **With Offset** is invoked, the projection on to surface happens, but the corners are placed not exactly on the surface but slightly at an off-setted distance. This is a useful tool as the offsetting helps to avoid wireframe edges/faces crisscrossing the surface.

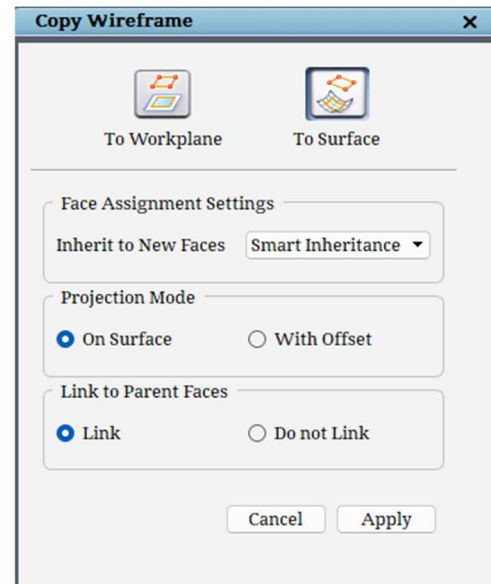


Figure 4.92: Copy To Surface dialog box

## 4.5.4 Revolve

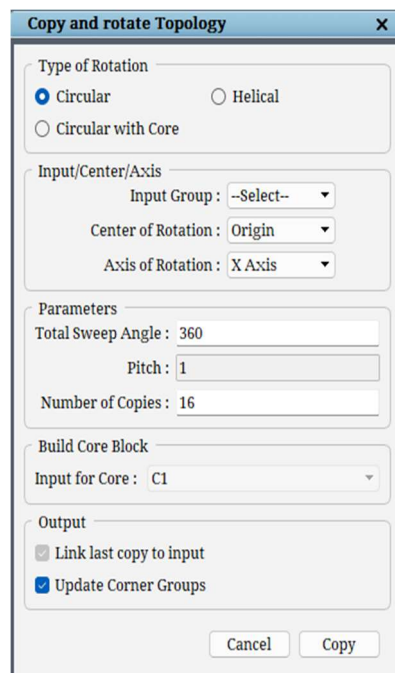


Figure 4.93: Copy+Rotate dialog box

**Revolve** option is used for copying and rotating the given wireframe using the angle, number of copies and pitch. This tool can be useful where geometries have the same cross-section but in a cylindrical or helical path. When you have the same cross-section, you can create a 2D wireframe on a plane and then copy it to the end of the cross-section using the copy drop back edges button if it is on the same axis. But when the axis is changing in a cylindrical or helical path, it becomes difficult to copy and move it to position the wireframe approximately to the geometry. So in such cases, the corner sheet rotation can be used



to copy the wireframe. From v8.1, the button is located under Edit section of the Topology tab.

The **Type of Rotation** defines whether it is a **Circular**, **Circular with core** or **Helical** path. If helical is chosen, the **Pitch** distance option gets enabled otherwise, it is disabled. The **Centre of Rotation** and **Axis of Rotation** should be provided in their corresponding fields. Then the maximum **Angle** till which the wireframe has to be revolved can be provided in the next field. The **Number of Copies** is the number of times the wireframe has to be revolved within the mentioned angle, and the angle is divided equally for each instance. For e.g., if the **Number of Copies** is 2, then you will have the wireframe at the starting point and then at the ending point; if it is 3, then you will have one more layer at the middle of the two endpoints. The **Pitch** can be provided in the corresponding field, which helps in revolving the wireframe in a helical structure. If you want to link the first and the last copy of the wireframe (NOTE: other adjacent copies are linked by default), you can enable the checkbox in front of **Merge Final copy to Initial**.

Similarly, enabling **Update Corner Groups** adds the subsequently created corners to its corresponding corner groups based on their 2D section. This helps in automation purposes. Finally, invoking **Circular with Core** option helps in completing the topology at the centre core by creating and linking the topology automatically. For e.g. consider a nacelle where the geometry cross-section is rotationally symmetric, and once you rotate a 2D cross-sectional wireframe, you manually link the faces and create blocks at the centre. From v6.8, this can be automatically built with a click of a button provided you input the 2D layer of the sheet, which has to be linked after rotation.

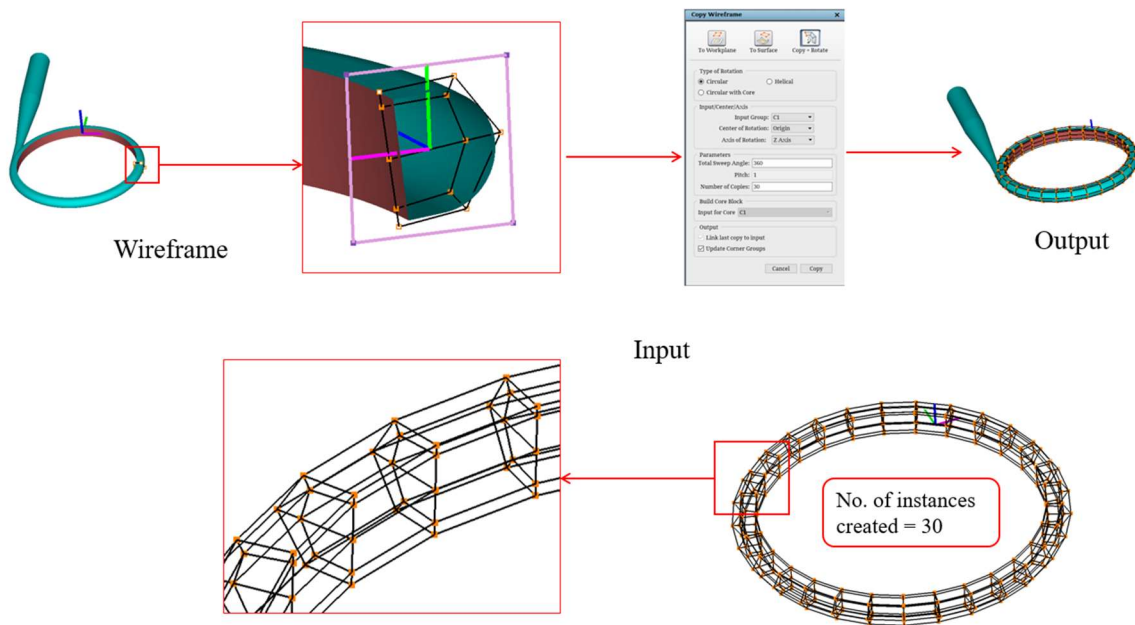


Figure 4.94: Rotate → Faces



### 4.5.5 Remove

Remove button helps in deleting the unwanted corners from the wireframe. In order to remove/delete

a corner, switch on the **Remove** button in the Edit section of the right-side panel in the topology tab and choose the mode of removal from the dialog box. There are two flags available; one, **corners** which will remove both corners and the edges associated with the selected corners; second, **edges**, which will remove only the edges and retain the corners. Once the mode is selected, select the corners or edges to be deleted

either by picking one by one or dragging a box around it. If the **corners** mode is selected and you drag a box around a bunch of corners, all the corners within the selected box will be deleted along with the corresponding edges. If the **edges** mode is selected, only the edges within the selected box will be deleted, and the corners will be retained in the same place. GridPro offers shortcut keys for most frequently used operations. Hold the '**R**' key on the keyboard and select the corners to be deleted. If you want to remove an edge, Hold the '<R>' key or switch on the remove button and click on the edge to be deleted. The cursor will become a two-sided vertical arrow when you move over an edge.

Note: Remove operation does not work on inactive corner groups.

Another operation is also embedded with the remove button, i.e., **Faces**. **Faces** removes a sheet of corners if its pattern replicates the adjacent sheets. It is the reverse of the insert, where it inserts a sheet similar to adjacent sheets. But **Faces** remove the sheets if it is similar to the adjacent sheets. It helps remove the extra blocks to reduce the total number of blocks in the wireframe. In order to remove a sheet, group the corners of the sheet in a group number, then click on the **Remove** button and click on the **Faces** option in the remove dialog box. The corner group should be active in order to execute Faces. It outputs an error message as shown in Figure 4.96 if removing sheets are not possible because of different adjacent layers. To avoid such mistakes in grouping, use 'A' key to group the corners, which gives the sheet of corners without much human effort.

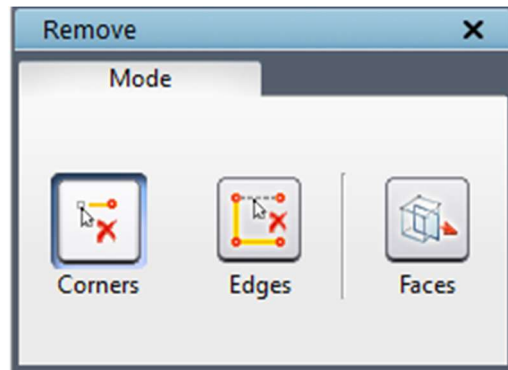


Figure 4.95: Remove dialog box



Figure 4.96: Error message

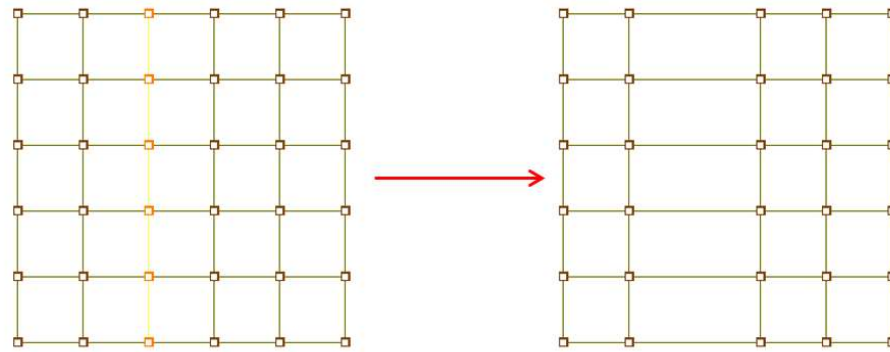


Figure 4.97: Remove → Faces

### 4.5.6 Split Topology

Split topology is a utility which splits a valid topology into disjoint pieces based on the given surface group with each piece as a valid topology individually. It is located under the **surface tools** menu. This is helpful when you want to run the gridding process faster on a bigger case. You can split the bigger case into smaller topologies and run Ggrid individually. The individual grids can be merged together using ‘weld’ utility. Clicking on the ‘Split topology’ option in the dropdown opens a dialog box, as shown in Figure 4.98.

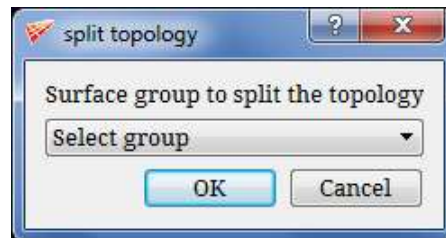
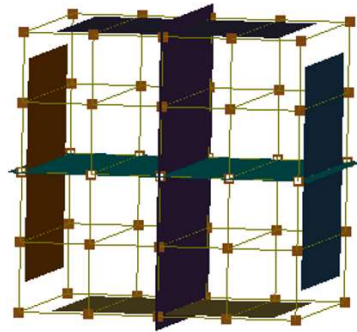
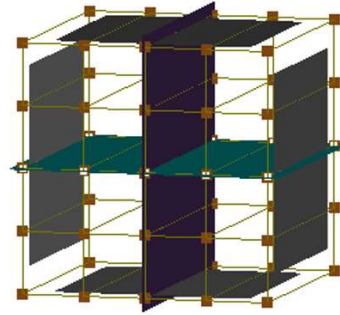


Figure 4.98: Split topology dialog box

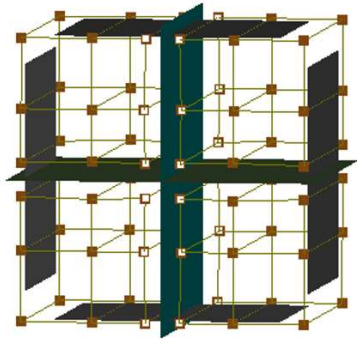
For an easier understanding, a simple box macro is chosen for an explanation. Consider a box topology with two internal surfaces in the middle, which splits the topology into four pieces, as shown in Figure 4.99.



Topology



Internal surfaces are grouped in s1



After split topology

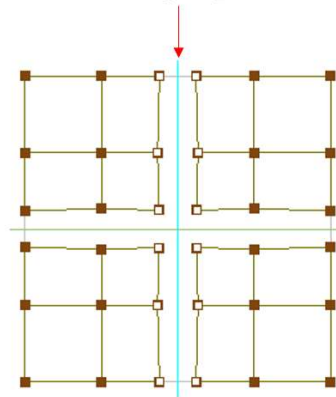


Figure 4.99: Split topology

### 4.5.7 Adapt Blocks to Surface

The **Adapt Blocks to Surface** tool is used to automatically trim blocks around a given geometrical surface, which creates a geometry fitted topology from an originally cartesian blocking pattern. The Adapt Blocks to Surface tool can be accessed through the **Topology Tools** option in the menu bar.

The images below explain the trimming process in more detail.

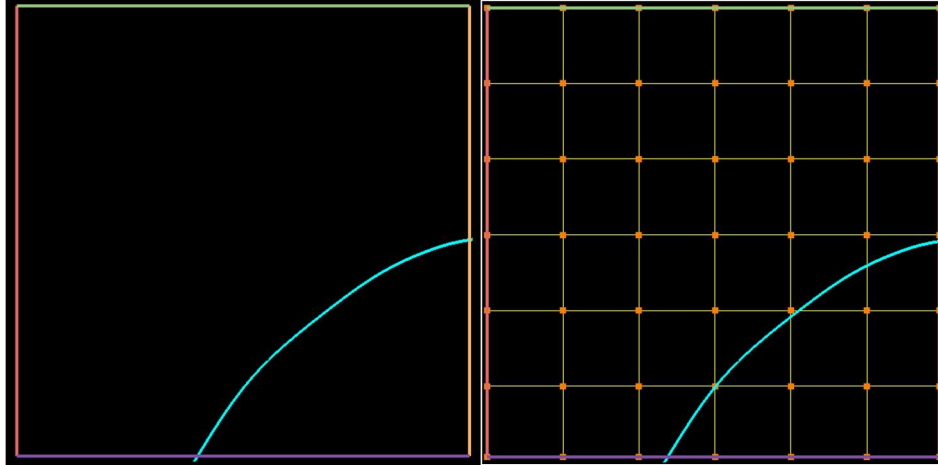


Figure 4.101: The square boundary and the curve to be trimmed

Figure 4.100: Cartesian block structure

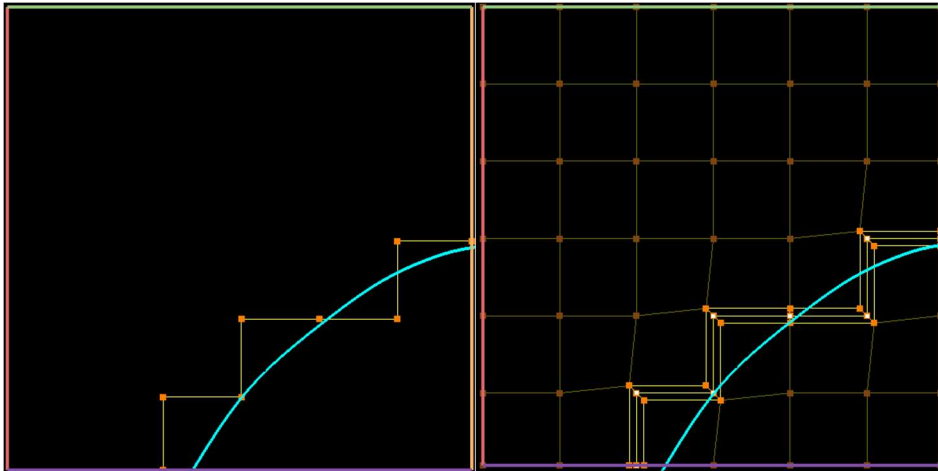


Figure 4.103: Block edges identified for trimming

Figure 4.102: After trimming

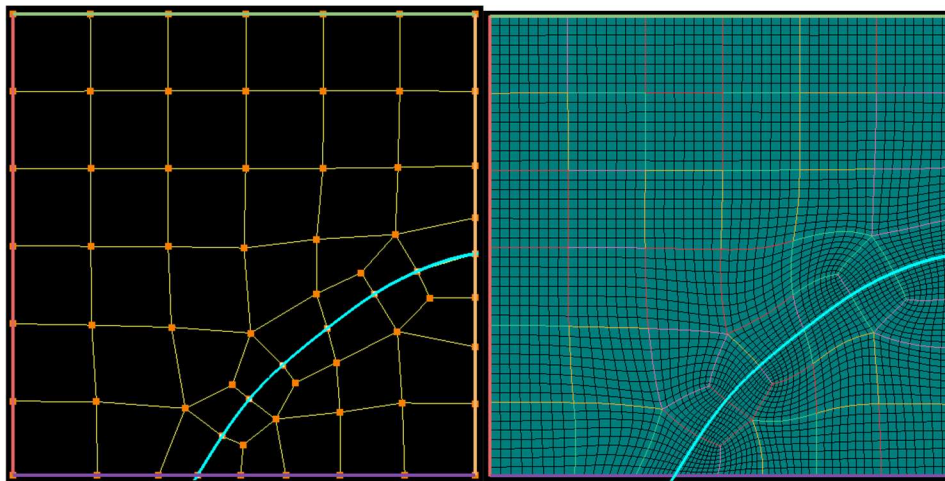


Figure 4.105: After smoothing

Figure 4.104: The generated grid

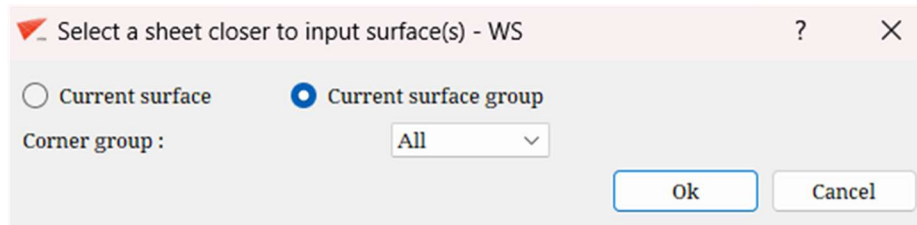


Figure 4.106: Adapt Blocks to Surface dialog box

The **Current Surface** and **Current Surface Group** radio buttons refer to the surface(s) to be used for trimming. The **Corner group** drop down can be used to specify which corners can be used while trimming.

The trimming process works for both 2D and 3D topologies.

## 4.6 Topology Transformation Utilities

Operations such as Translate, Rotate, Mirror, Scale and General transformation under **Transform** button and **Edge Length** button are grouped under topology transformation utilities that will be discussed here.

Clicking on the transform button opens a dialog box, as shown in Figure 4.107. The first section contains all the common input parameters such as corner and surface group options which to enable transform current surfaces and create a copy. The next section contains all the buttons related to transform. The third section opens the dialog box related to the buttons which are selected in the second section.

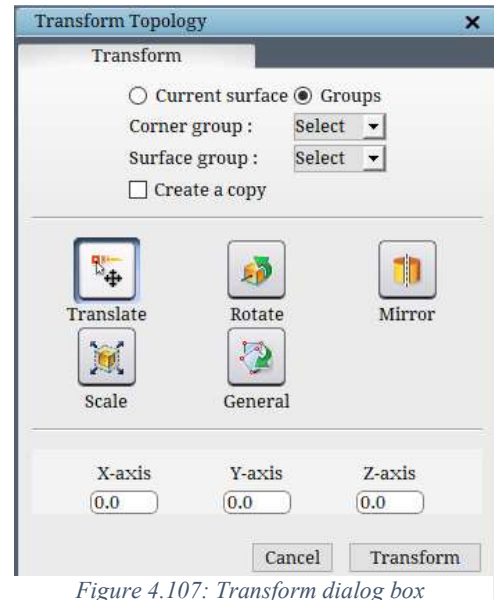


Figure 4.107: Transform dialog box

### 4.6.1 Translate



When **Translate** button is clicked, it opens a dialog box, as shown in Figure 4.107 with input parameters for X, Y, and Z direction. It gives control to translate the active corner and/or surface group to a given location. Enter the coordinates in the given fields to translate the corner and surface group. You can also copy the given groups to the new location by enabling the 'Create a copy' checkbox.

If the Groups option is enabled, only the groups which are selected under corner and surface group dropdown list will be translated. So make sure the right group is selected before executing the operation.

## 4.6.2 Rotate

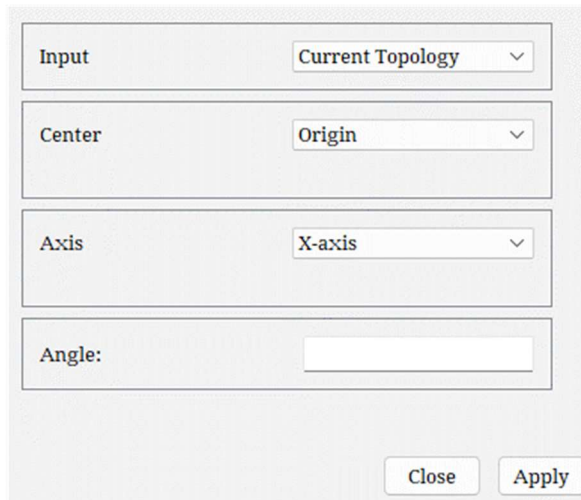


Figure 4.108: Rotate dialog box

necessary for the operation. By default, the Current topology option is selected. Next, you will have to decide whether to rotate the active corner group, active surface group or both axis. It can be chosen by enabling the corresponding check box next to each option. Then enter the centre, axis and angle of rotation in the respective field and click on **apply** to rotate. The centre of rotation is taken as origin for these axes. By default, the **origin** option is selected, which allows you to rotate a corner and/or surface group about the user-defined axis. The user has to define the centre of rotation, the axis of rotation and angle in the respective fields. Selecting the **workplane** option will rotate the corner and/or surface group along the workplane axis. You need to mention the angle of rotation. The centre of rotation is evaluated from the workplane centre, and the axis of rotation is always considered as workplane z-axis. Checking the ‘**Create a copy**’ checkbox will rotate and copy the topology; otherwise, the original topology will be rotated.

Sometimes a part of the geometry alone looks periodic, but the other parts would be linked with it. In such cases, you cannot do a periodic topology because you need topology on all periodic surfaces in order to continue the topology for the rest of the geometry. For e.g. blower where the blades are placed in a periodic manner, but you would like to have a mesh for the outlet as well. For such cases, GridPro offers a tool called **periodic to regular**, which can copy/duplicate the topology created for the part into a full topology, just like how a part of



When **Rotate** button is clicked, Rotate Topology dialog box opens, as shown in Figure 4.108. It deals with rotating the given corner group and surface group to the desired angle with the given rotation centre and the axis of rotation. It can rotate along the default global axes such as **x**, **y** and **z**-axis and about workplane. Once the dialog box is opened, select the input topology type, whether it is Current topology or Periodic topology. If Current topology is selected, all the below options would still be enabled, and if the periodic topology is selected, all of them would be disabled as it is not

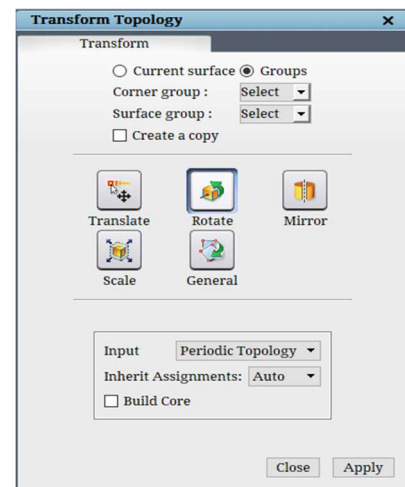


Figure 4.109: Rotate → Periodic



periodic geometry is gridded and revolved to get a full grid. This option is embedded in the **Input topology** dropdown itself; selecting the **periodic topology** option will disable all other options in the dialog box, as shown in Figure 4.109. To use this option, you need to create a wireframe for the part of the periodic surface and then rotate it to the full geometry, which will use the periodicity given for the periodic surface. Once it is rotated fully, the periodic surface will be removed from the UI. Once the option is selected, you will be prompted with the message for confirmation of the execution of the command. From V8.0 onwards, you can build the core blocking automatically without creating a singularity while converting the periodic topology to full topology. Enable the **Build Core** checkbox before execution to ensure that the core blocks build automatically

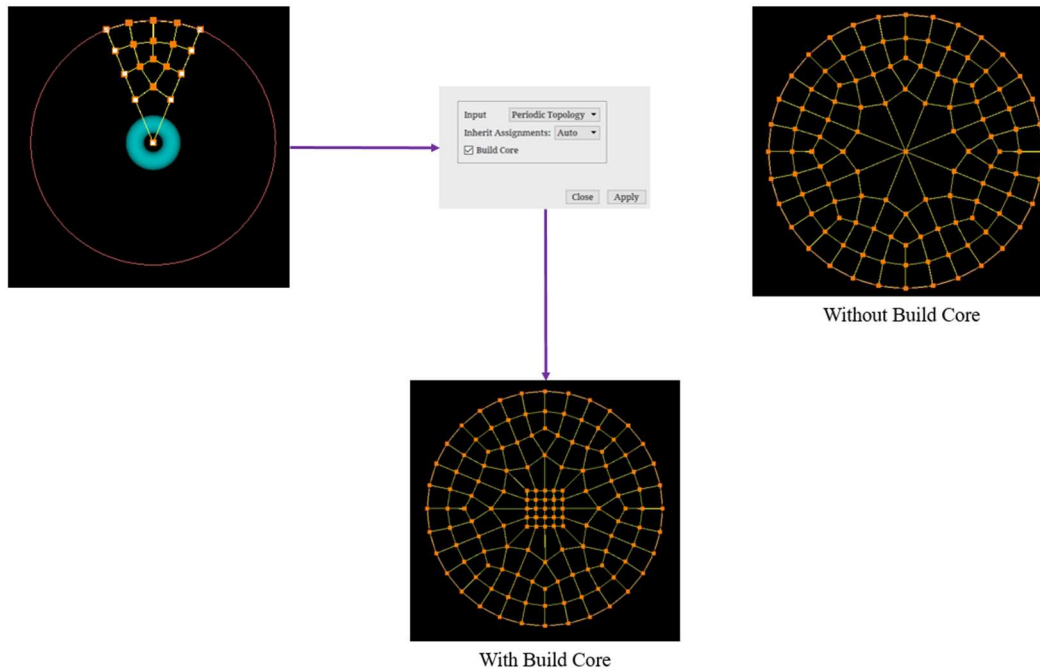


Figure 4.110: Periodic to Full Topology

### 4.6.3 Mirror



Mirror tool **Mirror** is helpful when you work on symmetrical geometry. It can mirror and/or copy and mirror the given active corner and/or surface group. The corresponding group must be active while executing this operation. It can also be used when part of the geometry is symmetrical. To mirror an object, a plane of symmetry and the normal symmetry is required. Figure 4.111 shows the *Mirror Topology* dialog box. Similar to rotate, the mirror also has some predefined options apart from the user-defined options. Selecting the **user-defined** option gives the control to define the plane of symmetry and the normal of symmetry. A point on the symmetry plane needs to be entered in the centre field, which will calculate the plane of symmetry while the normal of symmetry needs to be given in the normal field. Selecting the



**workplane** option mirrors the corner and surface group with respect to the workplane. It assumes the workplane position as the plane of symmetry and the z-axis of the work plane as normal of the symmetry. The pre-defined three options mirror the corner and surface group with respect to the global plane, i.e. **XY**, **YZ** and **XZ** plane. The normal for these planes is assumed implicitly using the options itself, i.e. XY plane defines that the normal is on XY plane, and the plane centre for these planes would be the origin. Checking the ‘**Create a copy**’ checkbox will mirror and copy both the wireframe and surfaces; otherwise, the original topology and surfaces will be mirrored. **Add to groups** option adds the corresponding mirrored corners/surfaces to its respective corner/surface group. For e.g. the corresponding mirrored corners of corner group c2 will get added to c2 if the checkbox is enabled.

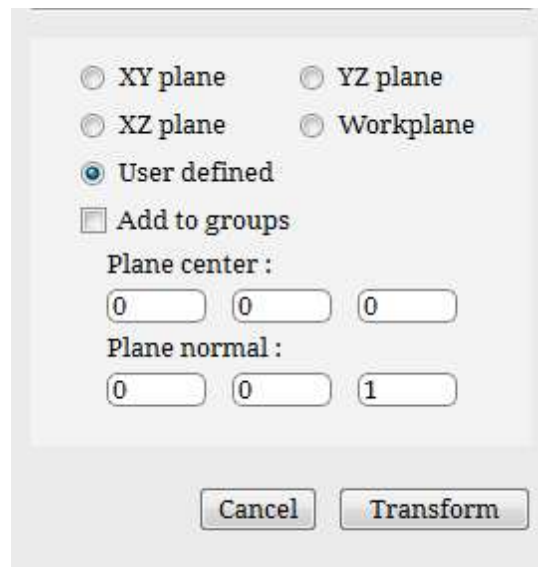


Figure 4.111: Mirror dialog box

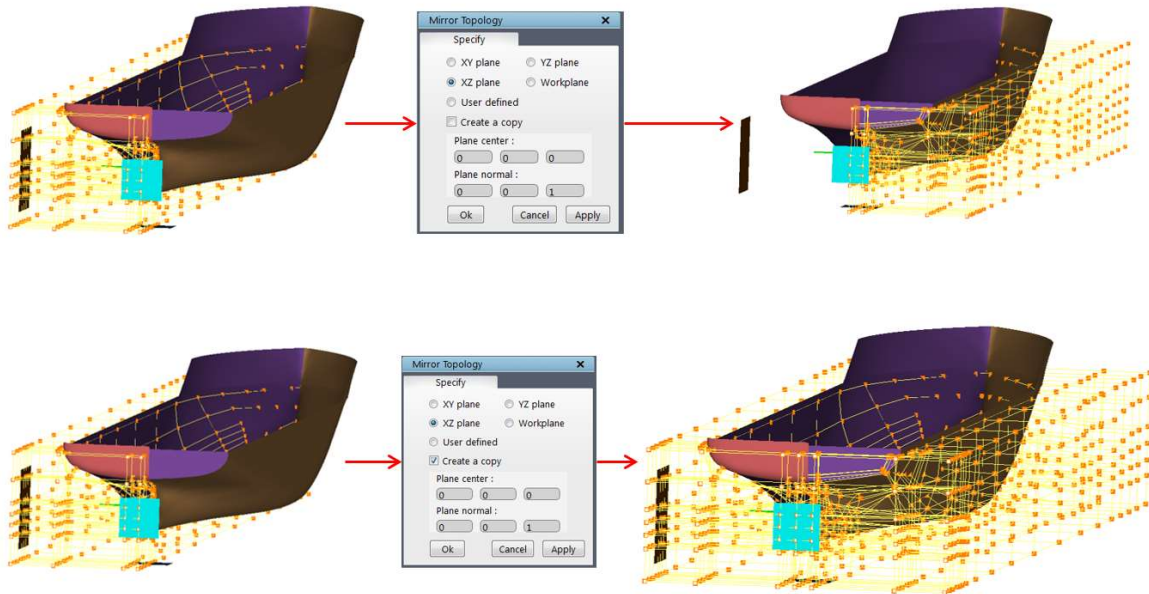



Figure 4.112: Mirror and Copy & Mirror corner and surface groups

#### 4.6.4 Scale



Scaling  can be done on any active corner and/or surface groups by giving the scaling centre and the ratio of scaling in all three directions. Figure 4.113 shows the dialog box of the **Scale** button. You need to provide the **centre** of scaling in the centre field and the **ratio** of scaling in the ratio field. Each axis is provided with a separate field. There are no pre-defined options similar to Rotate and Mirror except the workplane. Clicking on '**Get workplane Centre**' will automatically find the centre of workplane position and input it in the Centre field.

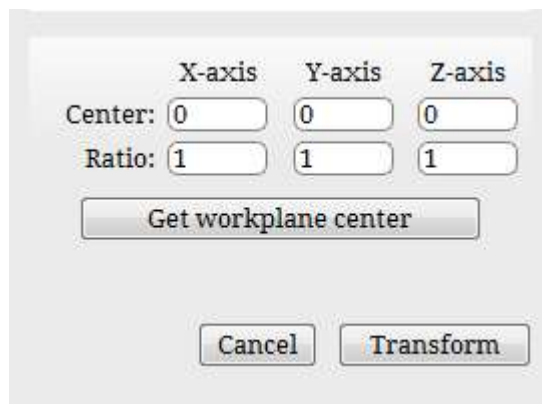


Figure 4.113: Scale dialog box

#### 4.6.5 General Transform

All the above-discussed transformations are put together in this transformation tool. Figure 4.114 shows

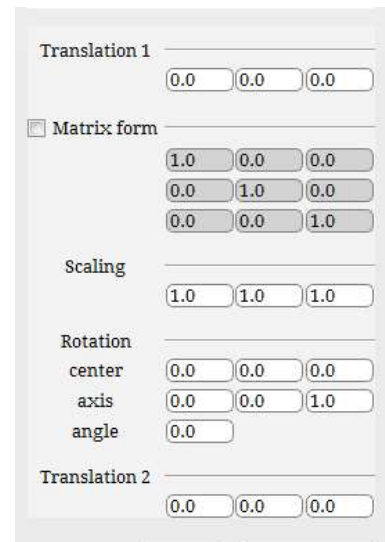


Figure 4.114: General Transform dialog box

Topology Transformation dialog box. As you can see, you can translate, rotate and scale both the surface and/or corner group. Unlike the above transformations, you don't need to keep a group active, but the corners and surfaces should have been grouped in some group. The dialog box provides fields to input the groups/group numbers to be transformed for both **corner** and **surface group**. Next is the **translation 1**, where the translation coordinates can be provided to translate the surfaces and corners. Then the **scaling** where the **scaling ratio** is given. By default, the origin is used as scaling centre, and the corresponding ratio is used for the 3 directions. Then **rotation**, which can be defined similar to the rotation button by using rotation centre, rotation axis and the angle of rotation. **Translation 2** fields can be used to apply any surface translation after the rotation discussed above is executed. This option is helpful when you want to scale the group with centre other than the origin. You can translate the group to the origin using translation 1 and apply to scale and then revert the topology back to the same position using the translation 2. All these transformations will be executed one after the other in the order as listed in the dialog box.

Instead of executing the transformations step by step, you can use the transformation matrix to specify the translation and rotation together. As you can see, there is a separate field in the dialog box to input a matrix below the translation 1 field. By default, the **matrix form** fields are disabled; check the matrix check box to enable the input fields, which will obviously disable all the other fields except the corner and surface groups.

#### 4.6.6 Edge Length

**Edge Length** is a topology repositioning tool which is located at the last row of the *Edit(Topology)* section. It helps in adjusting the length of the selected edges by moving either side of the topology sheet based on the options selected in the dialog box. Clicking on the **Edge Length** button opens a dialog box, as shown in Figure 4.115. The first option is a workspace input where the edge ID and its corresponding linked corners are displayed when you select the required edge whose length (also the length of its parallel edges) has to be increased/decreased. The second option in the dialog box is **Reference** section, which has two inputs **Adjust Group** and **Reference Group**. These two inputs help in deciding the edges whose length has to be changed. Selecting *All* under Adjust group will change all the length of the parallel edges of the selected edge. Selecting *Group* will enable the **Reference Group** input where you will have to select a corner group; only the parallel edges length which falls under the given reference group will be changed. Selecting *Edge* will only change the length of the selected edge and leave all its parallel edges unaltered. The third option in the dialog box is the **Direction**. Selecting the direction will highlight the topology sheet with a green coloured arrow, which will be

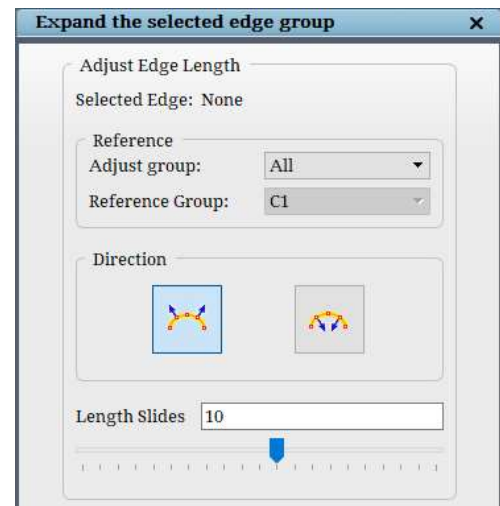


Figure 4.115: Edge Length dialog box

highlighted. The direction will highlight the topology sheet with a green coloured arrow, which will be

repositioned to change the edge length. The final option is the **Length Slides**; it is represented in percentage, which can be adjusted using the slider and fix the length of the selected edge based on visual perception.

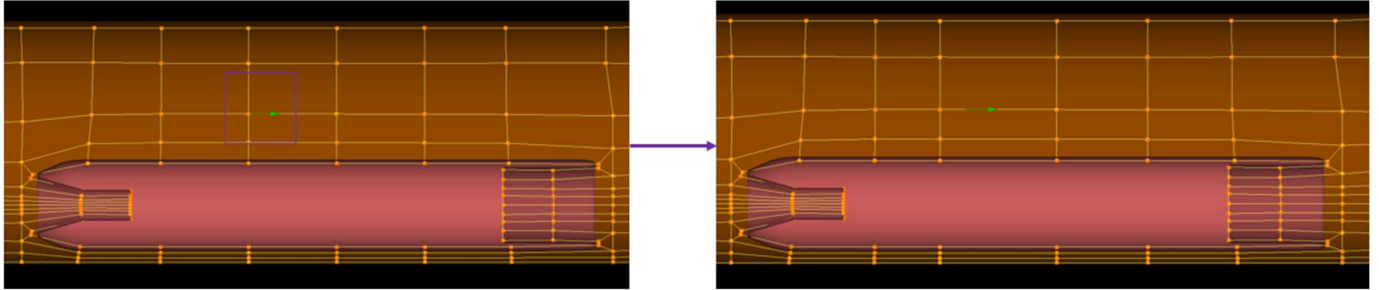


Figure 4.116: Example for Edge Length

## 4.6.7 Topology Mapping

Topology mapping is a very valuable, time saving feature. Previously, for topologically similar geometries, one has to move the topology using the transform option. This can be a time consuming and tedious process. To counter this issue, **Topology Mapping** has been implemented in GridPro.

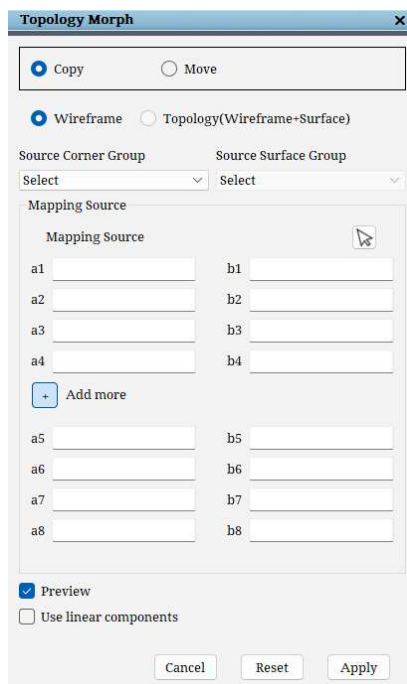




Figure 4.117: Topology Mapping dialog box

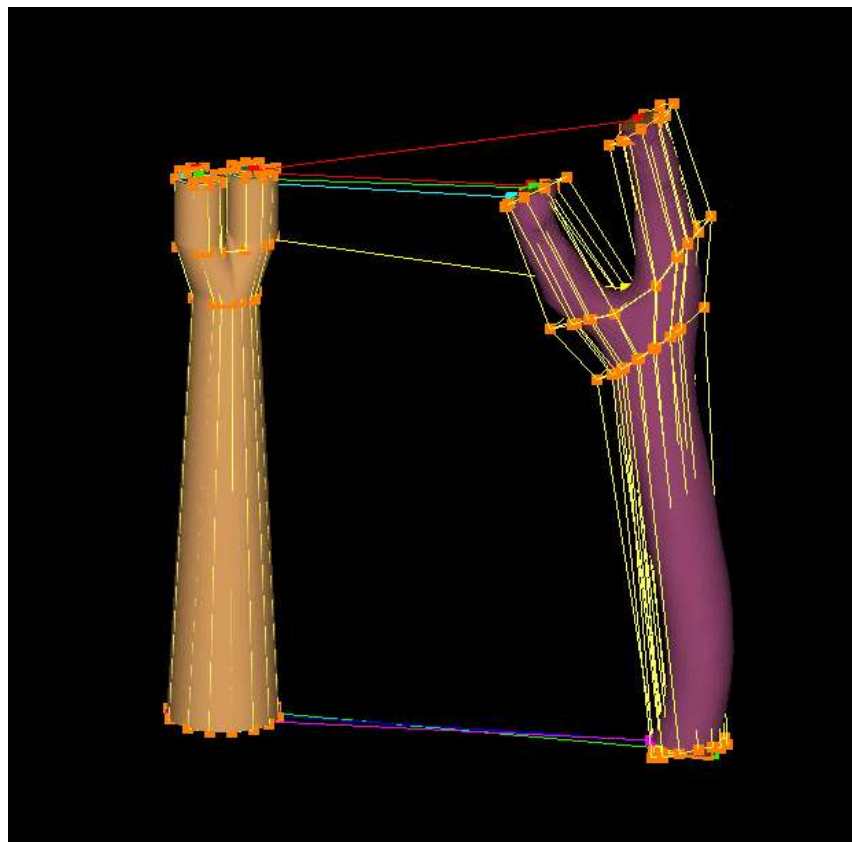
Topology mapping can be accessed by clicking on the Mapping  button in the **Edit (Topology)** section in the **Topology Panel**. The user has a choice to either **Copy** or **Move** the **Wireframe** (Corners and edges only) or the 'Topology' (Wireframe + Surfaces). Corners to be mapped are added into a corner group and the respective corner group must be selected from the **Source Corner Group** dropdown. Similarly, if the Topology is being mapped, the surfaces to be moved or copied must also be added into a surface group which must be selected from the **Source Surface Group** dropdown.

To start the mapping process, click on the **Mapping Source Button** . It automatically moves to the a1 option.

**a** here represents the source of the mapping and **b** represents the destination. To select the first mapping source, select a point on the surface of the topology to be copied. The cursor in the dialog box then moves to **b1**, then we have to select a point at the corresponding location on the surface where the topology is to be mapped. Both selected points will be displayed along with a line joining the corresponding points.

A minimum of 4 points is required to map the topology onto another set of surfaces, every time we select a point the preview updates and gives us a clearer picture of where the topology would be seated after mapping. In case a point has been incorrectly selected, go back to the point which needs to be reselected, select and erase the coordinates, make that point active and then select the new location in the UI.

If the preview of the mapping looks correct, click on **Apply**. If the four points do not define your topology perfectly, then we have the option to add four more points to map the topology more precisely. Also, in some cases, the mapping fails without the use of linear components. In such a case, check the **Use Linear Components** check box.



*Figure 4.118: Mapping the topology from one set of surfaces to another*

### 4.6.8 3D Array

When we are working with cases of similar geometrical patterns or shapes, it would be much easier if we just create the topology for one pattern, which could then be automatically copied over to the other similar geometries. The **3D Array** tool enables us to copy a topology in multiple directions. Let's take a look at how to work with the 3D array tool.

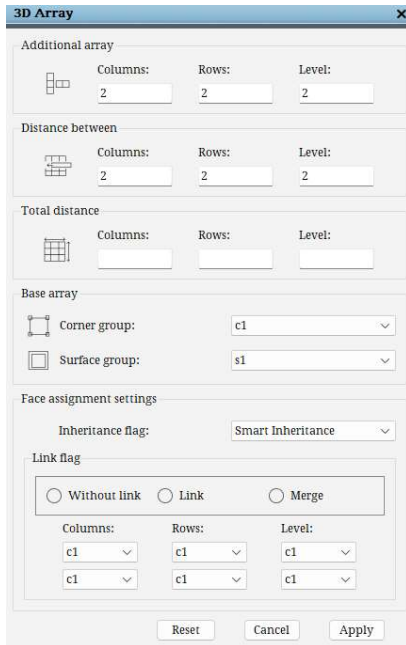


Figure 4.119: 3D Array dialog box

The first half of the dialogue box is to specify the direction and number of copies required of the base topology. The first three boxes under the **Additional array** section specify the number of **Columns**, **Rows** and **Levels** required for the base topology to be copied to. This specifies the number of copies to be added in each direction.

The next section, **Distance between**, specifies the distance between the base topology and the copied topology. By default they are set to a value of 2.

Similarly the **Total Distance** section is to determine the maximum limit upto which the copied topology can extend or be shrunk to.

Next we move to the next subsection for the 3D array. Under **Base Array** select the **Corner group** where the corners required to be copied are saved in. Similarly the same to be done with the

**Surface group**. Enter them in the base array option.

In the **Face Assignment settings** specify the surface assignments. The **Inheritance Flag** works the same as how it used to work for options like copy and wrap. Inheritance option determines how the surface assignments have to be carried forward. It consists of 3 options provided in a dropdown. The **Smart Inheritance** option removes the surface assignments from the original corner group and assigns them to the copied corners. The **Inherit All** option retains the assignments with the corner group used for copying and also carries forward to the copied corners. **Do not Inherit** retains the assignment with the corner group

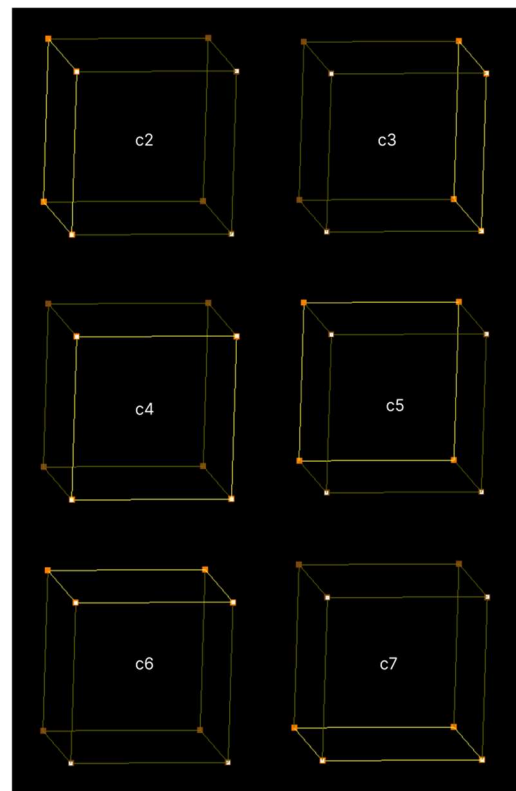


Figure 4.120: The corner groups selected for row, column and level



used for copying and will not be carried forward to the wrapped corners.

The **Link Flag** is used to specify whether the copied topologies are to be linked or merged to one another. When the **Without Link** option is selected, all the topologies are copied without any links between themselves, ie, they are independent disjoint topologies. When **Link** is specified, the corner groups need to be specified in the **Corner, Row** and **Level** section below. These corner groups should have the set of corner faces which need to be linked or merged when copied. Using **Link** links the copied topologies together, whereas using **Merge** merges the faces which are specified together.

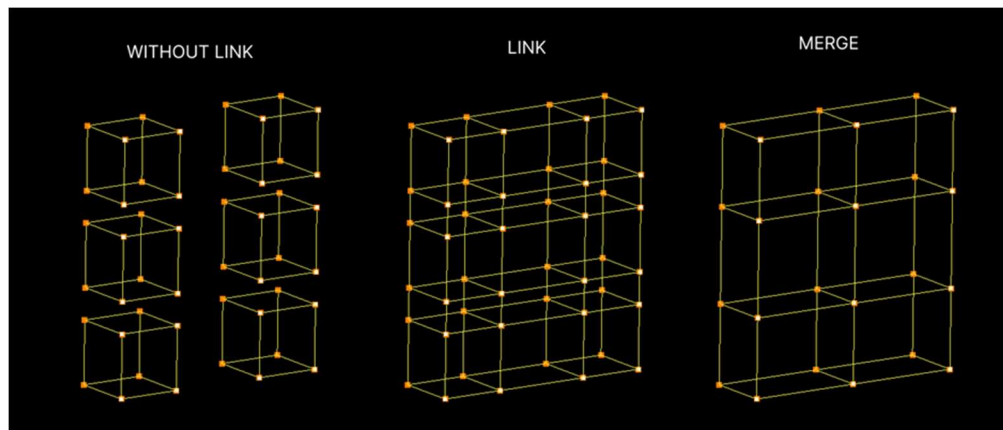


Figure 4.121: Outputs of the different link flag options

The **Reset** button at the bottom of the dialogue box only resets the dialogue box, not the function. To undo a 3D array copy, you will have to use the undo option in the global deck.

## 4.7 Topology Enhancing Utilities

Topology enhancing utilities are used to modify the topology in order to get a better quality grid. There are two utilities which enhance the topology quality; they are **Wrap** and **Heal**. **Wrap** helps in achieving an O grid topology, whereas **Heal** helps in fixing the singularities in the topology automatically.

### 4.7.1 Wrap

The utility **Wrap** is a dilated or shrunk duplication of a string of segments or quadrilaterals translated from and linked to the original string such that no links intersect one another. Because wrapping strings or faces of topology is common and can be a laborious process, *GridPro* has developed a feature that allows you to create a wrap automatically using groups. Since you must place a wrap around every curved boundary, as well as most boundary angles, you will more likely be wrapping wireframe than not.



Before you learn how to wrap, you must first learn what to wrap. On a plane, a closed or open string of segments is wrap-able. Each corner in the string must be the endpoint of one or two edges, not three or more edges. All edges must be part of the same string, and no more than two corners can define an edge. There is no restriction on the direction of the string or of the angles formed from connected edges. In space, a closed or open string of quadrilaterals is wrap-able. Each connected edge in the string must be common to one or two quadrilaterals, not common to three or more quadrilaterals. All quads must be part of the same string.

Sometimes, you may want to exclude a face of a wireframe in order to achieve a certain wrap. Excluded faces contain a diagonal represented by a dashed red line; these faces are ignored by *GridPro* during the wrapping process. Thus, a string of one-dimensional segments, which form a rectangle, could be considered as a two-dimensional quad if the face is not excluded. If you depress the “F” key and click two opposite corners of a face, a dashed red line will appear, signifying that the face is excluded. From v8.0 onwards, you can also group the faces to be excluded and use it wherever required. Face exclusion is covered extensively in a later section.

A wrap is better understood by sight than by explanation. However, the definition provides a good backdrop to the pictures. Figure 4.122 illustrates wrap-able one-dimensional objects, and Figure 4.123 illustrates unwrap-able one-dimensional objects.

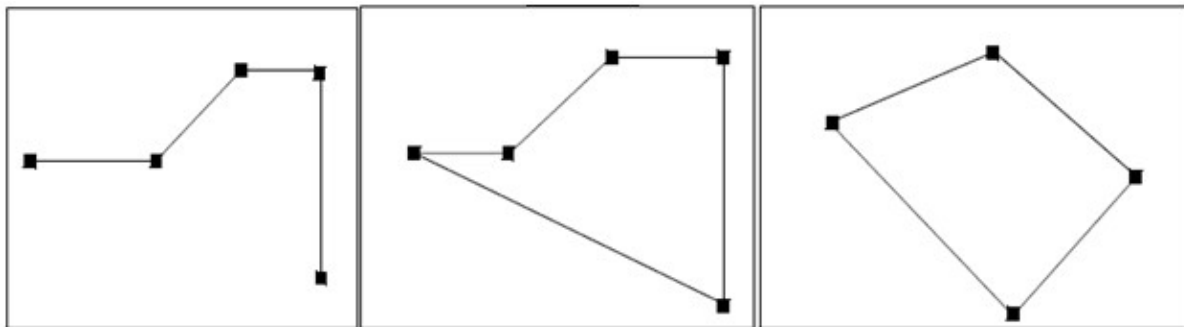


Figure 4.122: Wrap-able 1D wireframes

*GridPro* can automatically wrap pieces of a wireframe in one swift topological action; you do not need to input the corners and edges of a wrap one by one. However, these pieces of wireframe must be wrap-able and inside a group. If these conditions are met, you have a wrap-able group.

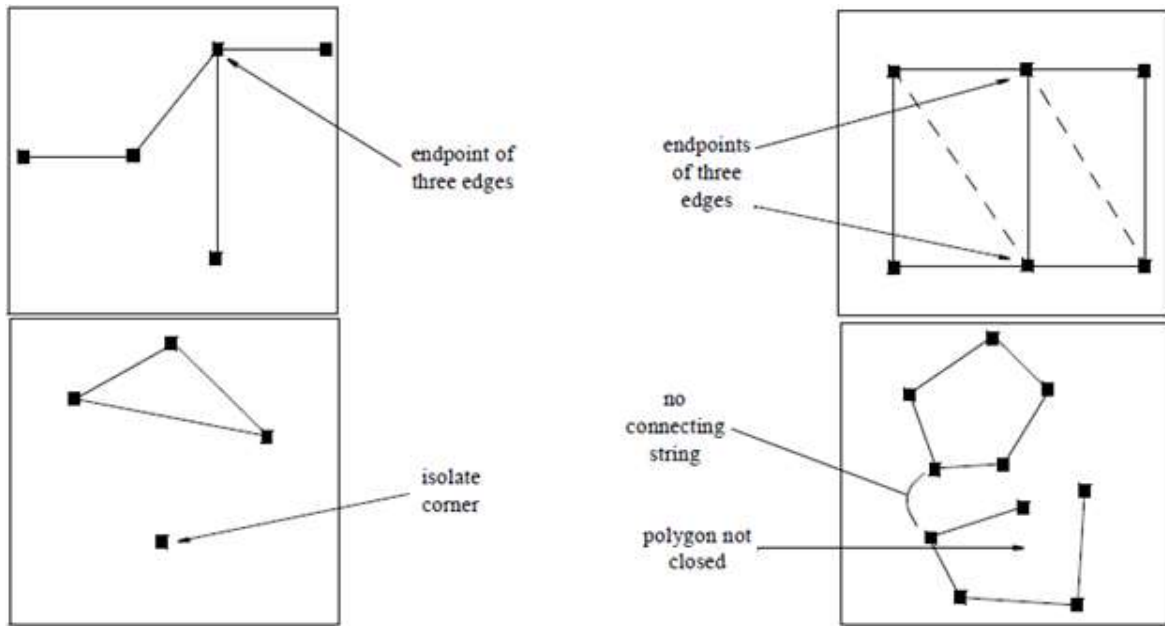


Figure 4.123: Unwrap-able 1D wireframes

In order to wrap a group, switch on a corner group (Refer *section 4.4*), and then press the **Wrap** button located in the **Edit (Topology)** section. The button opens a dialog box shown in Figure 4.124. By selecting one of the values in the dialog box, you can dilate or shrink your wrap by the percentage shown. In the status bar, you will be informed if a group is wrap-able or unwrap-able.

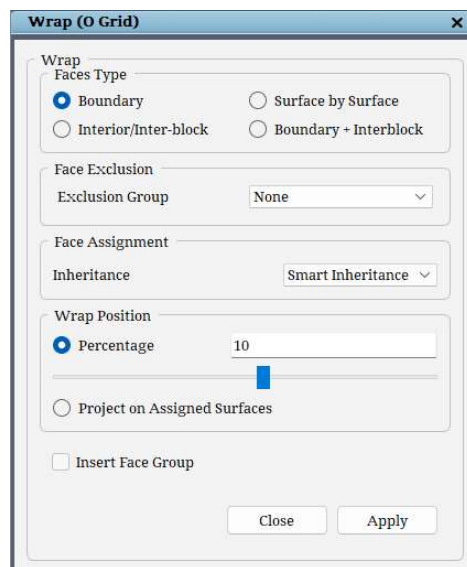


Figure 4.124: Wrap dialog box

There are four kinds of wrap: Boundary, Interior/Inter-block, Boundary + Inter-block and Surface by Surface.

1. **Boundary Wrap:** This is a wrap which extrudes from the existing topology either on the positive normal side or the negative normal side.

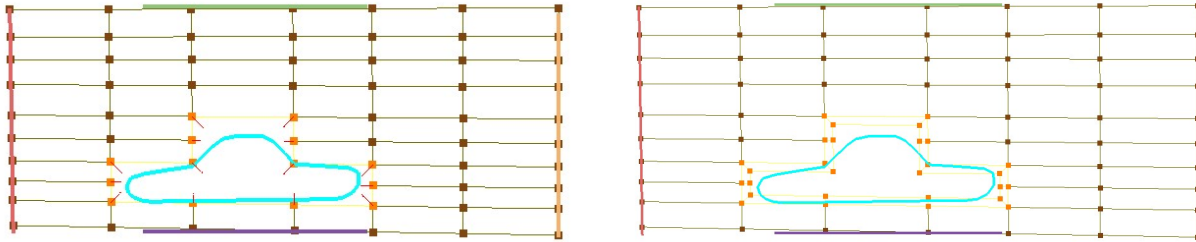
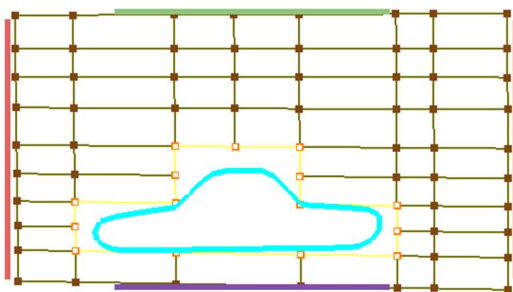
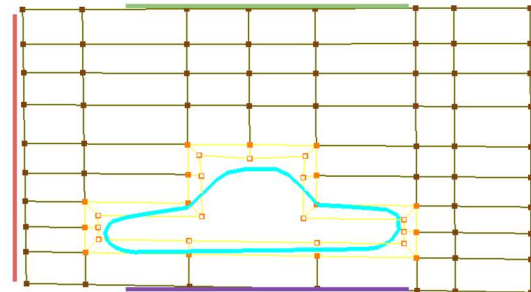


Figure 4.125: Before and after External wrap

For all the four types of wrap, the percentage of the wrap layer should be mentioned in the percentage field. By default, it takes 10% as a standard value, this value is the measure of the average length of all the edges in the corner group used for the wrap. It can also be adjusted using the **slider** provided below the input field. The **slider** primarily helps in picking the direction of the wrap, either inwards or outwards. The **Face Assignment Inheritance** option determines how the surface assignments have to be carried forward. It consists of 3 options provided in a dropdown. The **Smart Inheritance** option removes the surface assignments from the corner group used for wrap and assigns them to the wrapped corners. The **Inherit All** option retains the assignments with the corner group used for wrap and also carries forward to the wrapped corners. And **Do not Inherit** option retains the assignment with the corner group used for wrap and will not be carried forward to the wrapped corners. The option **Project on Assigned Surfaces** helps to project the wrapped corners on to the respective surfaces. This option is active only for External wrap and is in a deactivated mode for the other wrap types.



Before wrap



Smart Inheritance

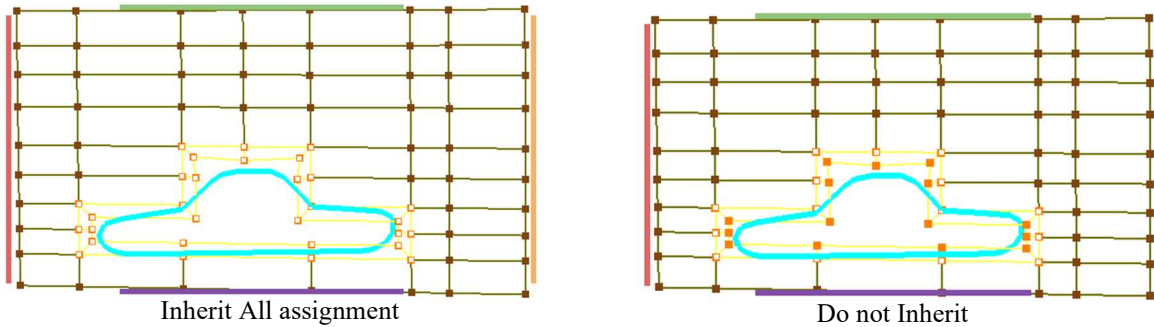


Figure 4.126: Different surface assignment propagation type

2. **Interior/Inter-block:** This wrap is inserted between the existing topology, and topology connections are adjusted so that the topology remains valid. This is helpful when there is a need for local refinement without affecting the far-field boundaries.

Enabling the option, **Insert Face Group** will add an insert in the middle of corners used for wrap and the wrapped corners. This option is active only for **Internal wrap**.

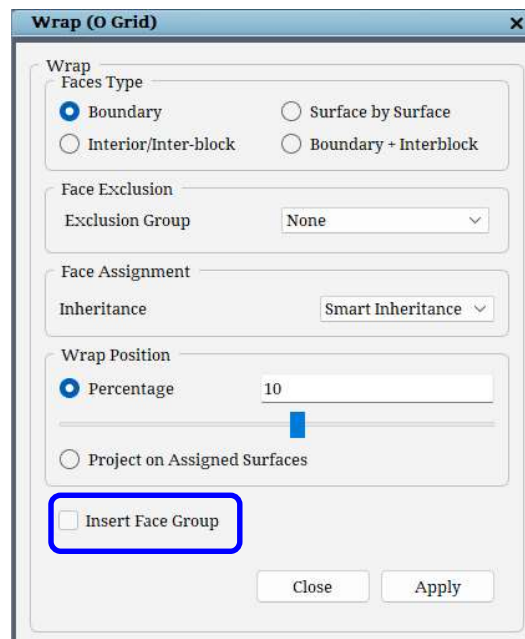


Figure 4.127: Wrap with Insert Face Group option.

Internal wrap with Smart Inheritance option automatically moves the surface assignments of the internal surface to the wrapped corners and reassigns the surface assignments of the external surface to output a valid topology. If you get a singularity topology, then do the internal wrap in the opposite direction, which reassigns the assignments in the opposite direction, which in turn outputs a valid topology.

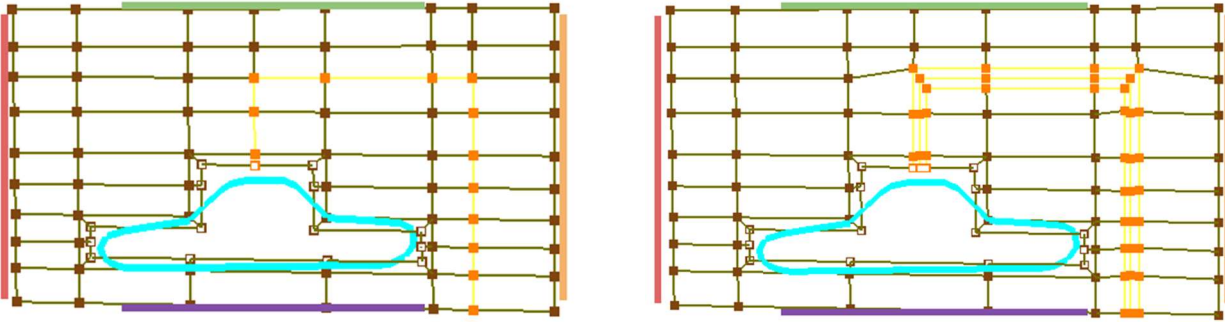


Figure 4.128: Before and after Internal wrap

3. **Boundary + Interblock:** This wrap type takes a corner sheet representing both internal and external faces as input, duplicates it and continues to maintain one-to-one connectivity with rest of the wireframe. Usually, the corner sheet chosen represents a T-type of structure, as shown in Figure 4. 129

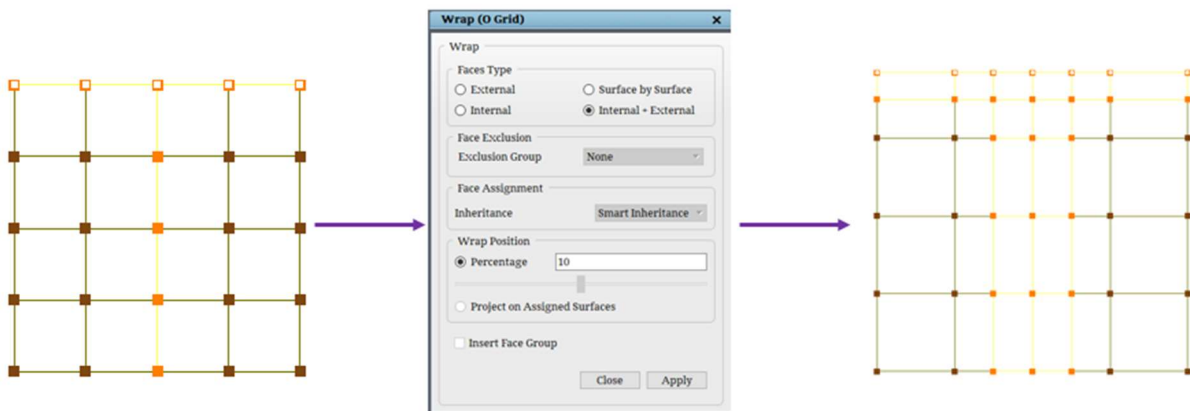


Figure 4.129: Internal + External Wrap

4. **Surface by Surface:** In this wrap type, the corners sheets assigned to all external surfaces are picked and wrapped at one shot, generating an H-type topology, as shown in Figure 4.130

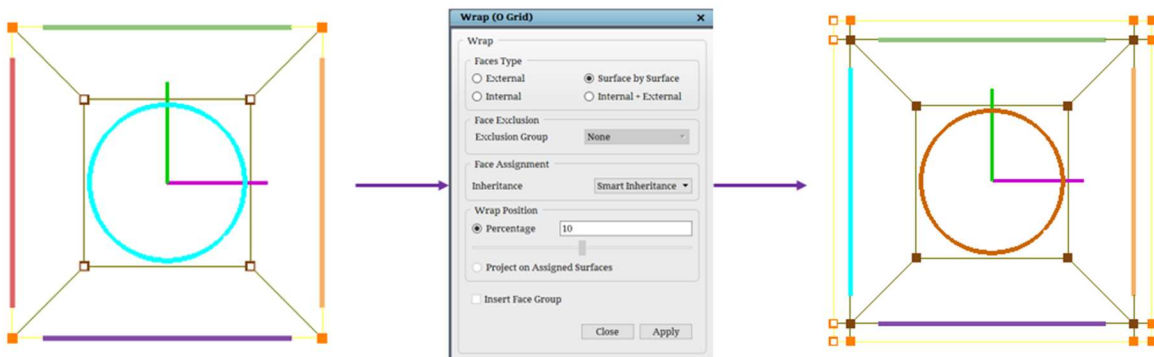


Figure 4.130: Surface by Surface Wrap

- **Face Exclusion**

Now that you have learned what wrap is and how it is used in the topology, this subsection describes exceptions arising from the rules stated previously regarding the wrap of a surface. In order to account for these exceptions, you must *exclude* faces of a wireframe. A face here is defined as the area enclosed by four corners and the edges among them. *Face exclusion* is necessary in two cases: first, you must exclude faces so that *GridPro* can correctly read the dimension of a surface being prepared for wrapping. For example, a wireframe rectangle can be thought of as a string of four segments (a one-dimensional wrapping) or as a quadrilateral (a two-dimensional wrapping). Therefore, you must specify the dimension of the wrap. In order to convey to *GridPro* what you are wrapping, you must exclude faces of a wireframe when the edges of those faces are to be wrapped as one-dimensional segments.

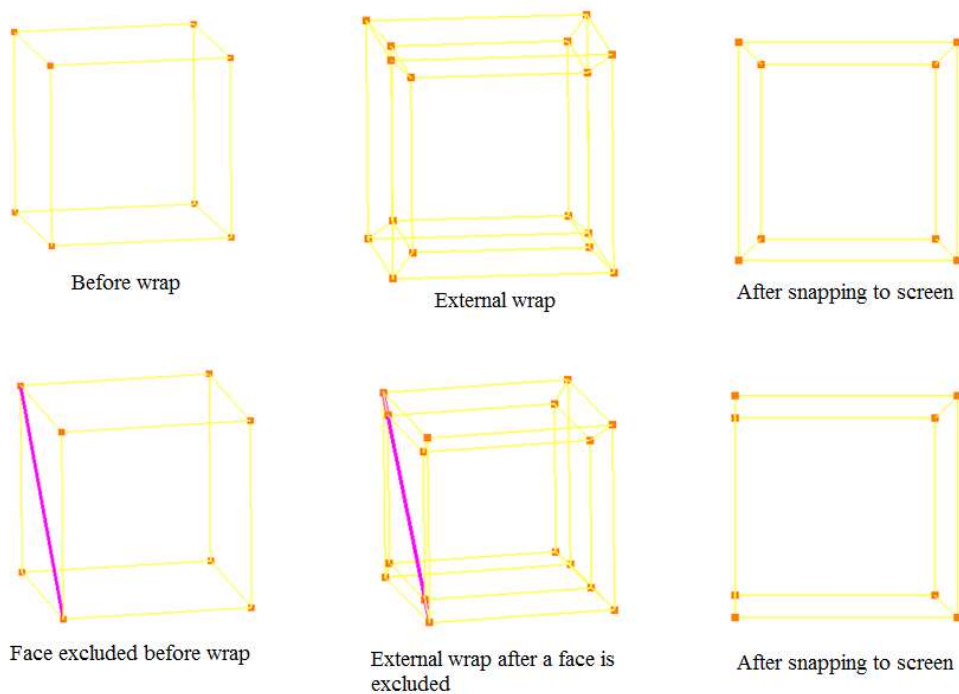


Figure 4.131: Wrap on face excluded wireframe

Second, you must exclude faces to exclude a portion of the wrap. Figure 4.131 illustrates how wrap is affected by excluding faces. You might believe that in order to wrap the outside faces of a three-dimensional object, you have to exclude the object's internal faces. However, *GridPro* automatically excludes these faces for you. See Figure 4.132.

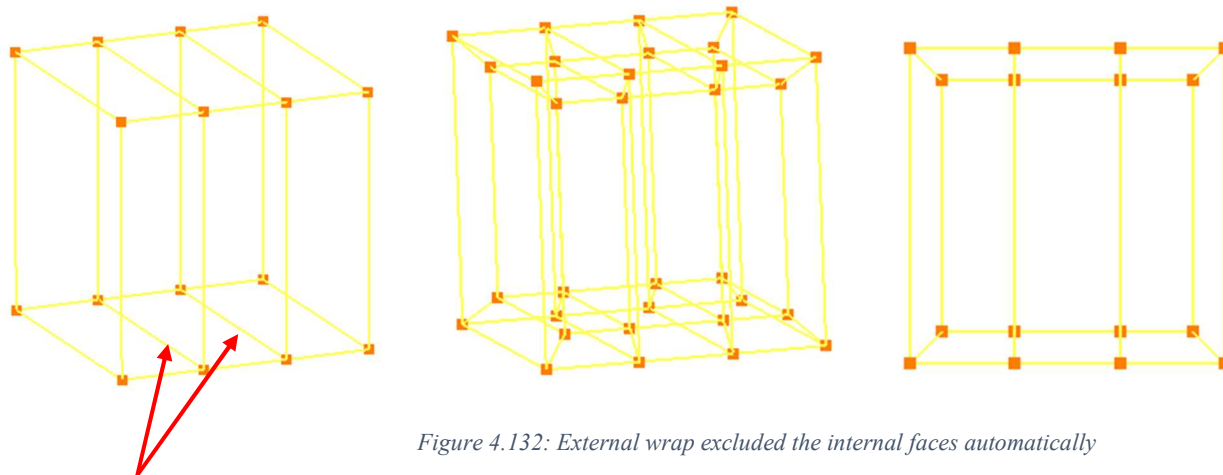


Figure 4.132: External wrap excluded the internal faces automatically

**Faces are not excluded** Face exclusion doesn't have an exclusive button in the GUI. It is accessible only using the shortcut keys. In order to exclude a face, you must hold down the **<F>** key and click two corners of a diagonal of the desired face. A pink line appears between these corners, signifying that the quad, or face, is excluded. It is a feature in topology building, where before the grid generation process is started, all the face exclusion (pink line) in the topology is removed. In order to remove the pink line, hold down the **<F>** key, select both the corners again. Often, exclusion markers will clutter your screen. To remedy this, you can hide the exclusion markers by unchecking exclusion in the topology tree of the toggle display section, which is explained in *section 2.6*.

From V8.0 onwards, you can choose to group the faces to be excluded and select it as an input under Face Exclusion Group.

## 4.7.2 Heal

- **Singularity**

In 3D, a regular grid point in a structured grid has 6 neighbouring grid points, which form a well-defined index coordinate system. For multi-block structured grids, most of the grid points are regular. However, singular grid points are introduced for steering the grid lines to better fit a complex geometry.

Singularities can be classified on the topological level instead of for grid points. There are two types of topological singularities: *singular block corners* and *singular block edges*. A singular grid point is either on a *singular corner* or a *singular edge*.

For 3D cases, the term singularity usually means singular edges. On the other hand, for 2-d, singular corners are the only type of singularities. A singularity is ranked with the number of neighbouring blocks. For internal edges (i.e. edges not on surfaces) in 3D, a regular one has 4 neighbouring blocks. An internal edge having three neighbouring blocks is called a singularity of

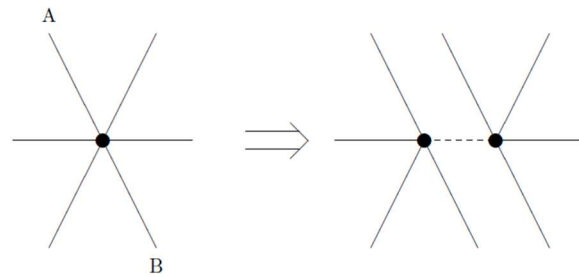


degree 3. Similarly, an internal edge having five neighbouring blocks is called a singularity of degree 5. It is more difficult to generate good grids around a singularity as the degree moves farther from the regular number 4.

*GridPro* restricts the degree of a singularity to be either 3 or 5 to 7. However, one should almost exclusively limit the singularities to degree 3 and degree 5.

A singularity of degree higher than 5 can always be split into several singularities of degree 5 using a general procedure without completely redesigning the topology. The tradeoff is increasing the number of blocks.

A simpler procedure will work most of the time. For a 2D case, Figure 4.133 is a local view showing how a singularity of degree 6 can be split into two singularities of degree 5.



*Figure 4.133: 6 degree singularity split into two 5 degree*

Singularities of degrees higher than 5 are used for the cases where their use will significantly reduce the number of blocks. When a sufficiently high degree of symmetry is present in the surface geometry, and the block topology near these singularities, the blocks around a singularity will be more or less uniformly distributed. So you should avoid putting singularities on surfaces.

In other words, you should try your best to design a topology that wraps around the surfaces. The following situations are considered to be regular on surfaces (3D): an edge on a single surface having two neighbouring blocks; and an edge on the intersection of two surfaces having only one neighbouring block. Any other surface edge is a singular surface edge and should be avoided in the design. For 2D cases, the explanation given w.r.t edges can be considered as corners. Thus, the topology design in Figure 4.134 has 4 singularities on the surfaces. They are the 4 corners of the outer box. A better design should avoid them, as shown in Figure 4.135. In some situations, the need to use regular corners or edges on surfaces may result in a bad topology. However, one does not have to use singularities on surfaces to solve the problem. An alternative is to redo the surface division.

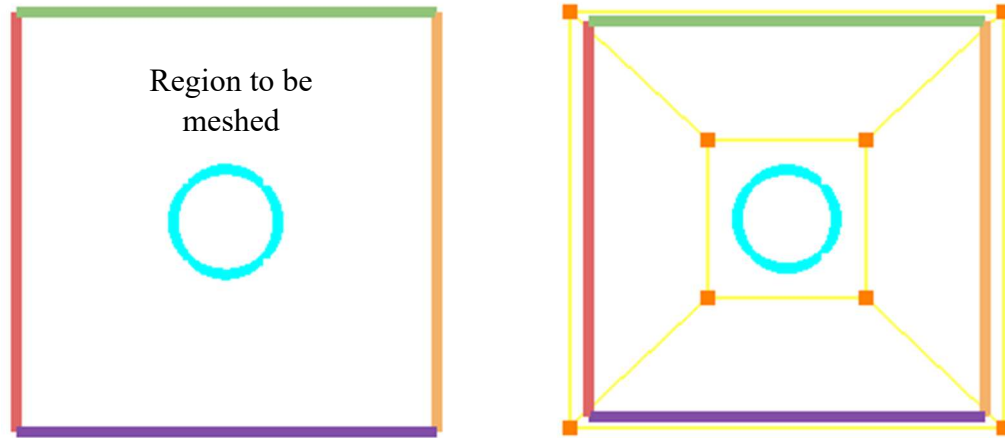


Figure 4.134: Topology with singularities on surface

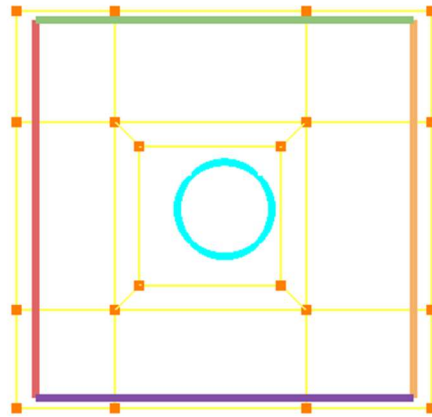


Figure 4.135: Modified topology without singularity

Based on the effects of these, the singularities are split into 3 types. They are mildly, mediumly and very severe singularity.

- **Mildly severe singularity:**

A singularity is ranked with the neighbouring blocks and its surface assignments. A corner is fixed if it is assigned to two or more surfaces. So a corner can be fixed only on the surface intersection. For an ideal block structure, there should be only two edges that can emerge out of a fixed corner in case of external surface intersections. If 3 or more edges emerge out of a fixed corner, then it is called mildly severe corner, and the singularity is a mildly severe singularity. In 3D, if there are 3 or more faces emerge out of an edge, then it is called *mildly severe edge*.

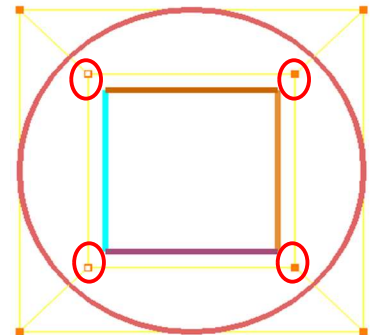


Figure 4.136: Mildly severe topology

NOTE: The above explanation is applicable only for the corners at the surface intersection. In other words, it is applicable only for the fixed corners.

- **Mediumly severe singularity:**

The corners which are being assigned to the external surfaces should have two adjacent blocks on either one of its normal sides, whereas in case of internal surfaces, it should have two blocks on both of its normal sides. In other words, there should be three edges emerging out of the assigned corner in case of external surfaces and 4 edges emerging out in case of internal surfaces in 2D. For 3D, instead of an edge emerging out of a corner, it would be a face emerging out of an edge.

From the above explanation, if 4 edges are emerging out of an assigned corner for an external surface or 5 edges emerging out for an internal surface in 2D is called a *mediumly severe singularity*.

NOTE: The above explanation does not apply to the corners which are at the surface intersection. In other words, it does not apply to the corners which are having more than one surface assignments.

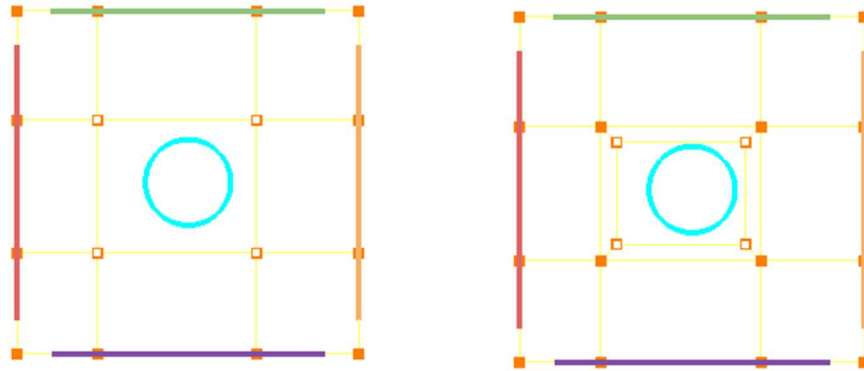


Figure 4.137: Mediumly severe topology and its solution

- **Very severe singularity:**

From the above explanation given in the mediumly severe, if there are only 2 edges emerging out of an assigned corner for an external surface or 3 edges emerging out for an internal surface in 2D, it is called a *very severe singularity*.

NOTE: The above explanation does not apply to the corners which are at the surface intersection. In other words, it does not apply to the corners which have more than one surface assignments.

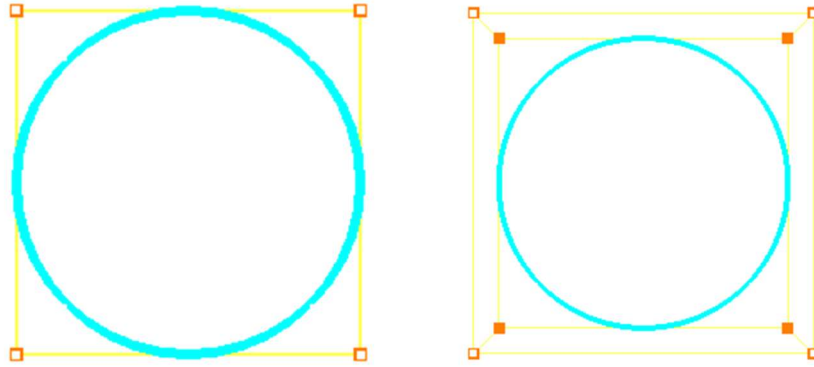


Figure 4.138: Very severe topology and its solution

All these singularities can be solved manually or automatically using the Heal button located under the **Edit (Topology)** section. Figure 4.139 shows the Autofix singularities dialog box. All three can be solved using a single dialog box in a single operation. The mediumly and very severe singularities are solved by doing a wrap either internally or externally, while the mildly severe singularity is solved using an internal surface. These internal surfaces have to be replaced with the user-created internal surfaces as these surfaces cannot be used for generating the grid. The mildly severe can also be solved in two other ways by giving overlapping assignments and using surface by surface wrap.

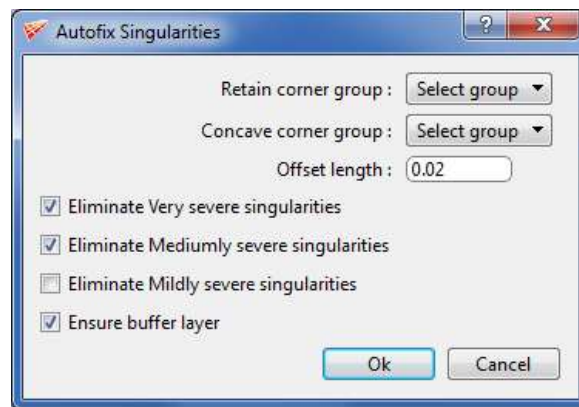



Figure 4.139: Heal dialog box

The first field is **retain corner group**, which contains the corners whose singularity has to be retained. Next is the **concave corner group**; if the singularity is in the concave corner group, it can be solved using surface by surface wrap, without internal surface, as mentioned above. **Heal** button also solves the surface by a surface wrap in the concave region. So corners with singularities in the concave region should be grouped and given in this field. You can either solve all the three singularities at a single time or step by step. The next three options are the checkbox indicating which singularity has to be solved. By default, it solves both **mediumly** and **very severe**. If you want to solve mildly severe in addition to that, then enable the checkbox of **mildly severe**. The next option is again a checkbox to input whether to create **buffer layer** or not.

By default, it creates a buffer layer; you can avoid creating it by disabling the checkbox. The last option is the ratio which determines the newly created corner position.

## 4.8 Corner Reposition Utilities

As you encounter increasingly complex topologies, you may need to project/align corner groups onto the workplane or surfaces or grid. Projection often helps in visualization and positioning purpose. There are different aligning tools available in GridPro that are grouped under

the **Align**  button. When Align button is clicked, Align Wireframe dialog box opens, as shown in Figure 4.140. The entire dialog box is split into three sections. i.e. Based on **Workplane**, **Surface** and **Grid**. There is also an additional tool called **Smooth Topology** located in the Topology Tools dropdown in the menu bar, which automatically smooths a valid topology. This helps in saving time in repositioning corners manually.

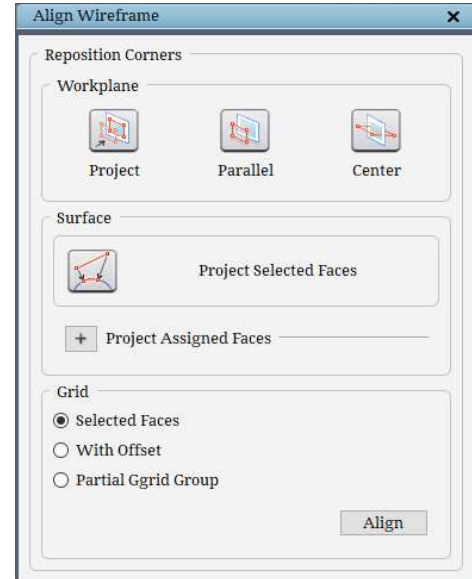


Figure 4.140: Align To dialog box

### 4.8.1 Workplane

**Workplane** provides an option to project the group of corners on/to the Workplane and its centre.

- **Project**



The **Project**  positions the corner group to the workplane position, i.e. it projects the given corner group to the workplane position.

Note the difference between the group fit operations and the options under the project button. The group fit button deals with operations that align the workplane to the active corner group; the project button deals with operations that align the active corner group to the workplane.

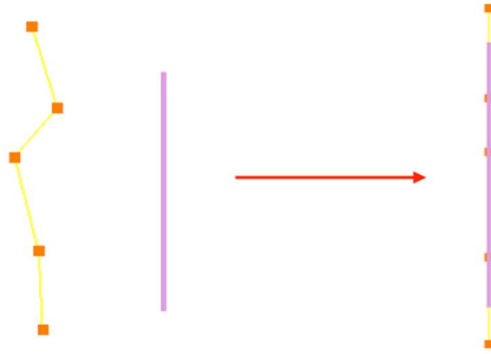


Figure 4.141: Before and after aligning on workplane.

- **Parallel**



The **Parallel** operation projects the active corner group to the workplane such that the centre of mass of the active corner group and the workplane are relative. Figure 4.142 shows the projection of corners to workplane. Note that projection on workplane does not place the corners at workplane position.

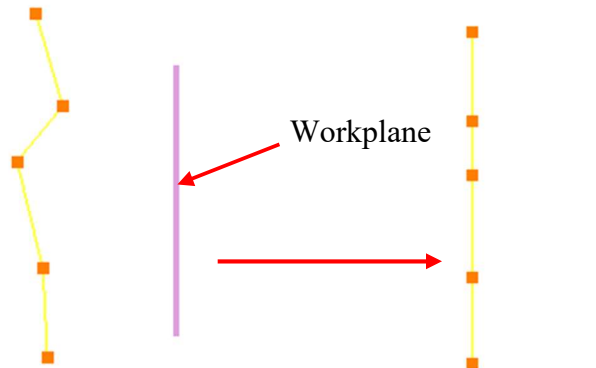


Figure 4.142: Before and after aligning to workplane

- **Centre**



The **Centre** operation translates the centre of mass of active corner group to the centre of the workplane.

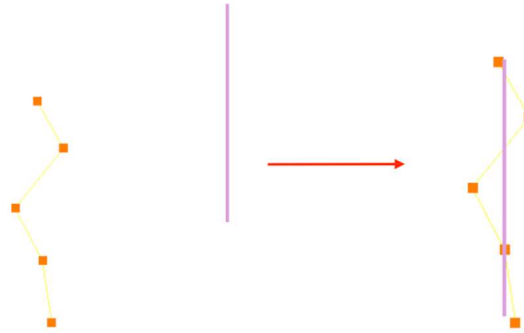


Figure 4.143: Before and After aligning to Workplane Centre

## 4.8.2 Surface

- **Project Selected Faces**

When you align a corner group on a surface, all corners and edges of that group are projected onto the current surface. So, a corner group should be enabled/active, and it should contain the corners to be projected. An empty corner group cannot be used for projection, which will pop up an error message. Note that the corners can only be projected to the *current* surface.

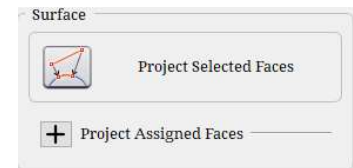


Figure 4.144: Project Selected Faces option in Align dialog box

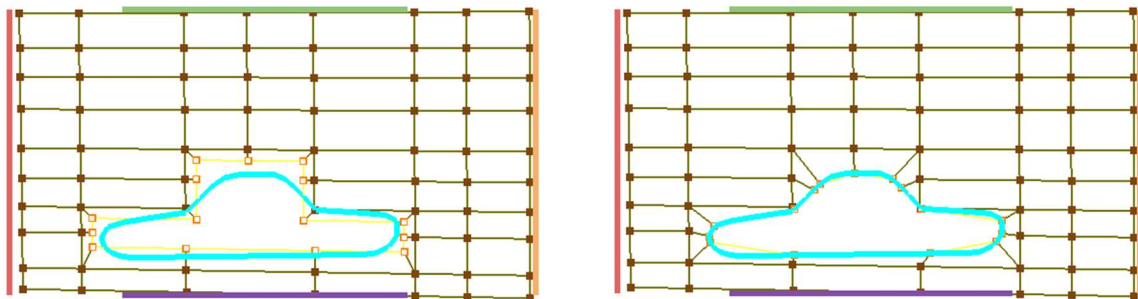


Figure 4.145: Before and after aligning on current surface

When project on the surface is clicked, a dialog box pops up asking for number of smoothing as shown in **Error! Reference source not found.** Figure 4.146. A smoothing algorithm will run on the projected corners based on the number of smoothings given.

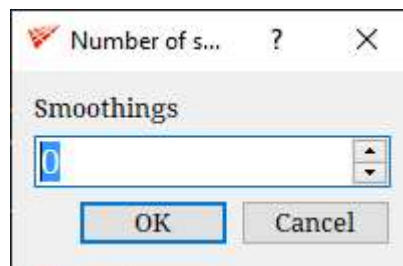


Figure 4.146: Number of smoothing dialog box



- **Project Assigned Faces**

This option is an extension of the above Project Selected Faces, where corners assigned to surfaces are automatically picked and projected to respective surfaces. Either assigned corners of current surface or assigned corners of all surfaces can be picked, or corners corresponding to surfaces in a surface group can be picked for projection. The input here is the surface/s or surface group. The corresponding assigned corners are automatically grouped for projection. Further, the projected corners can be placed on the surface or offset by a small distance. Also, the offset projection mode can be Face Offset or Edge Offset.

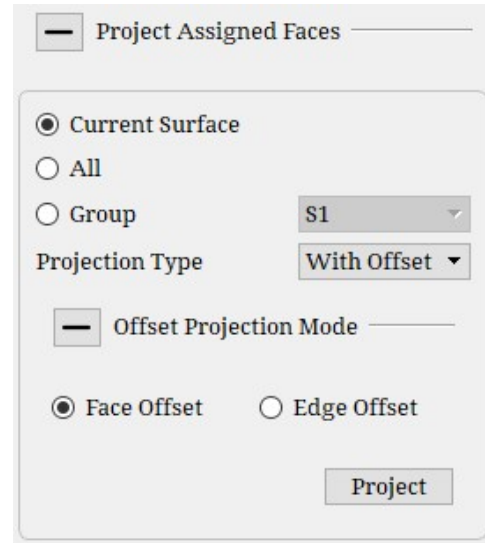


Figure 4.147: Options to do assigned corner projection to surface

### 4.8.3 Grid

- **Selected Faces**

Under **Grid** is a kind of projection that synchronizes the position of the topology blocks with the grid blocks existing in the grid tab. A multi-block *GridPro* grid should exist in the grid tab that should match the blocking pattern and the block count of the topology that has to be projected. Both the topology and the grid should exist in the current session of the *GridPro*. Figure 4.149 shows the topology before and after sync.



Figure 4.148: Various options to do align to grid.

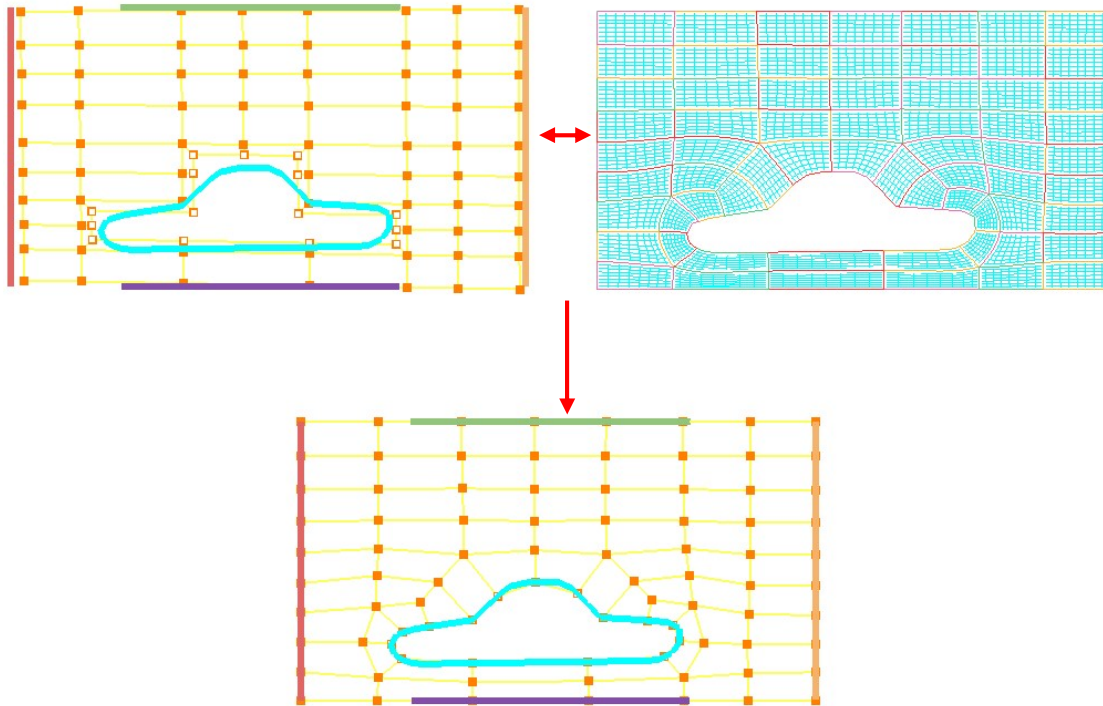


Figure 4.149: Before and after aligning to grid

If the grid blocking pattern is not matching with the topology, an error message pops up, as shown in Figure 4.150.

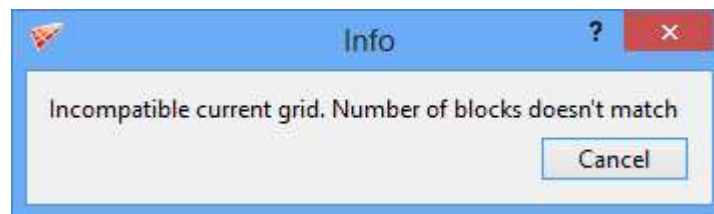


Figure 4.150: Error message

- **With Offset**

This option, along with aligning the wireframe in accordance with the grid, also offsets the corners assigned to the surfaces by a small distance. This is a very useful tool when we want to achieve better repositioning of corners using the grid and later want to re-run Ggrid. If the corners were sitting on the surface, there is a high possibility of facing grid convergence issues, as some of the edges/faces might be cutting the surfaces, especially in high curvature regions, resulting in surface or volume folds. Offsetting pushes the assigned corners by a small amount to ensure that the edges do not cut across the surfaces.

- **Partial Ggrid Group**

This is similar to **Selected Faces** operation. This operation is used along with Ggrid groups. You may have run Ggrid on part of topology to position them better. Once you get the grid for the same, use **Align→Grid→Partial Ggrid Group** to achieve that. Upon clicking on the button, it shows a confirmation pop-up, as shown in Figure 4.151. Click Ok to align the given group to the grid loaded in the UI. It must be executed on the entire group used for Ggrid. This helps in faster convergence for the bigger topology.

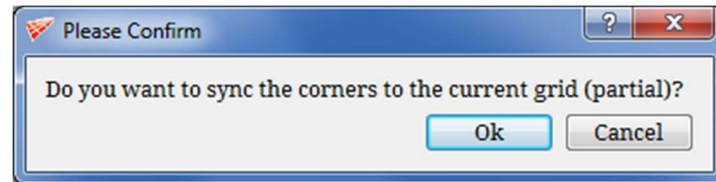


Figure 4.151: Confirmation message for Group2Grid

#### 4.8.4 Smooth Topology

The grid smoother Ggrid is able to converge much faster if the topology corners are placed close to their final locations. In order to facilitate this, the **Smooth Topology** tool has been added into GridPro. The tool smooths and repositions corners in a valid topology. The new topology serves as a better input to the grid smoother and aids in faster grid generation. The Smooth Topology tool can be accessed via the Topology Tools dropdown in the toolbar.

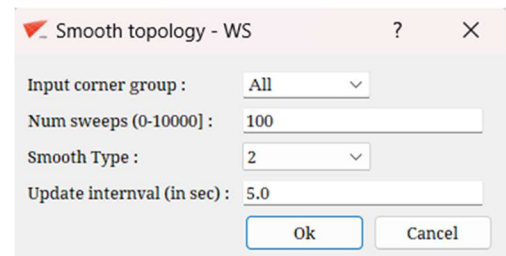


Figure 4.152: Smooth topology dialog box

There are a few options in the dialog box which are explained as follows. The **Input corner group** dropdown presents the corners on which the topology smoother is to be run. It can be run either on a partial corner group or on all corners of the topology. However, in both cases, it is important to note that the topology must be valid. The **Num sweeps** option contains the number of sweeps the smoother is run on the topology. In the **Smooth Type** dropdown, we are able to select the type of smoothing algorithm applied to smooth the topology.

- 0 - Only the interior corners are smoothed using a PDE scheme.
- 1 - Smoothing using layered harmonics.
- 2 - Smoothing using PDE's.

The last option, **Update Interval (in sec)** specifies the amount of time after which the topology is updated in the GL, while the smoothing process is running.

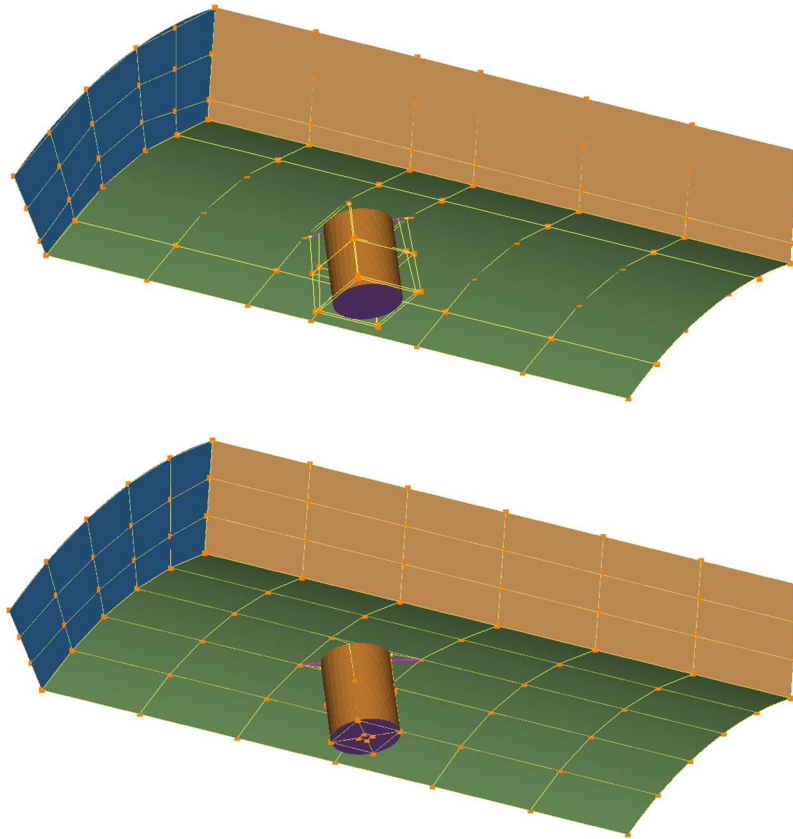


Figure 4.153: The topology before and after smoothing

## 4.9 Topology Optimization Utilities

Topology optimization utilities are used to enhance the topology in order to optimize the grid according to the user's need. Three tools are classified under Optimization utilities such as Density, Nest and Enrich. Density allows you to refine the grid by modifying the number of grid points. Nest allows you to refine specific regions in the grid with cells of 1-1 connectivity. Enrich utility, or compact enrichment utility, refines the grid in a particular region without affecting the far-field.

### 4.9.1 Density

The density of a grid is of great importance as a grid can be used for analysis only if the gridded region is sufficiently

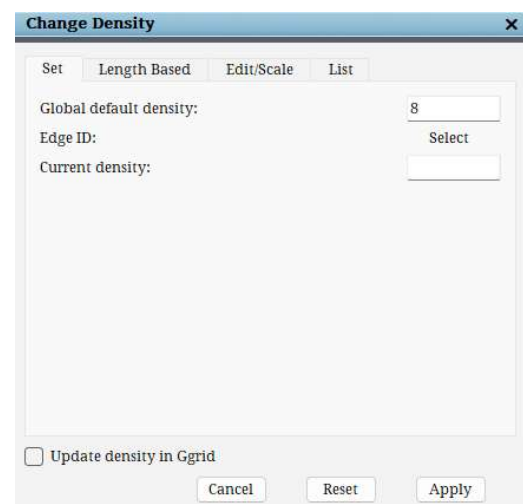



Figure 4.154: Density dialog box

resolved. Before you begin the gridding process, you can specify the density of the grid you wish to create. Density is a number that describes the number of grid cells along an edge (and thus all compatible edges).

The grid cell count is directly proportional to the grid's overall density. You must set the density of the grid in the **Topology** tab; you can either see the results of the density modification by switching on the **preview** button, which is explained in section 5.1.6 or by running the grid

generation process. By accessing the **density**  button under **Edit** section in the **Topology** tab, you can alter the grid density of sets of topologically parallel edges or all edges. You cannot alter the density of only one edge or a pair of topologically perpendicular edges. There are four tabs in the Density dialog box - **Set**, **Length Based**, **Edit/Scale** and **List**.

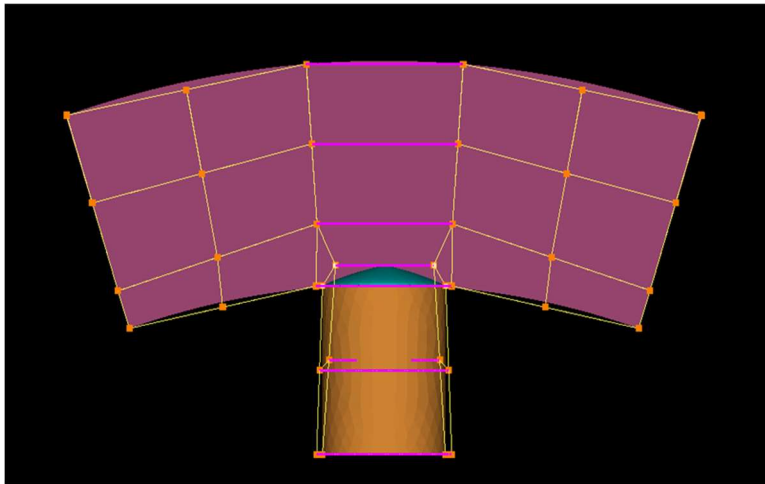


Figure 4.155: Parallel edges highlighted

edge and all parallel edges will be highlighted in pink. The density of all these edges will be changed to the new value. The new density can be specified by changing the value in the **Current Density** text box.

The **Length Based** panel can be used to change the density of the entire topology based on the length of the edges in the topology, by specifying a reference edge and reference density. The software automatically calculates the density of all edges in the topology based on the given inputs. This is especially useful in cases where the wireframe closely matches the actual location in the grid. The **Reference edge** can be selected by clicking the reference edge. The selected reference edge is also highlighted in pink colour, however, you will notice that only the selected edge, and not all parallel edges are highlighted in this way. The density of the edge can be specified by changing the value in the **Current Density** text box, and the maximum density that is to be present in the topology can be specified in the **Max density** text box.

In the **Set** tab, we have the option to change the **Global Default Density**, which is the default density of all edges. By default, it is set to be 8. To change it, just change the value in the text box and click on **Apply**.

The density of a particular set of perpendicular edges, can be changed by clicking on the edge whose density is to be changed. Upon clicking on the edge, that



To understand the usage of this, let's take an example. Let us take 4 edges in the topology with lengths  $L$ ,  $2L$ ,  $2.5L$  and  $3L$ . Let's take the first edge with length  $L$  as the reference edge, and assign it with a density of 4. We shall also set the **Maximum density** to be 11. On clicking **Apply**, we see that the edge with length  $2L$  has its density set to 8 ( $=2 \times 4$ ), the edge with length  $2.5L$  has its density set to 10 ( $=2.5 \times 4$ ), and the edge with length  $3L$ , should have had its density set to 12 ( $=3 \times 4$ ), but due to the maximum density constraint of 11, its density will be set to 11.

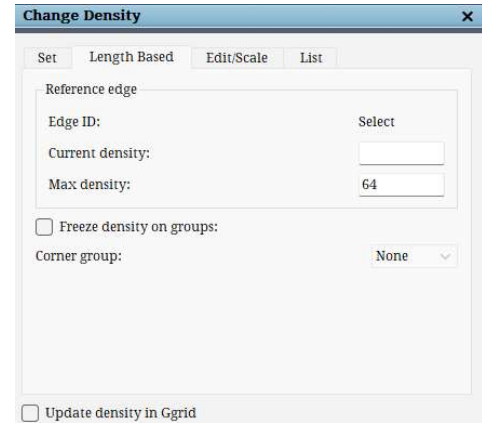


Figure 4.156: Length Based Density Dialog Box

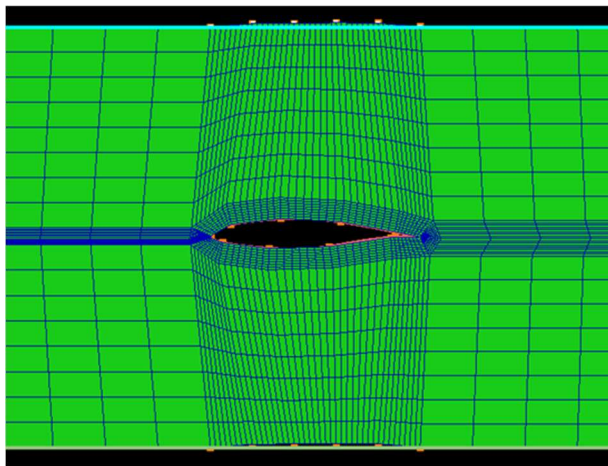


Figure 4.157: Before Length based density setting

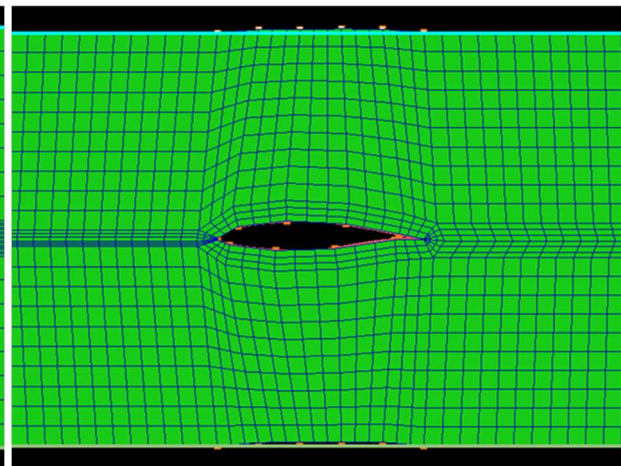


Figure 4.158: After Length based density setting

You can also change the density of multiple edges at the same time in the **Edit/Scale** panel. Under the **Change** dropdown list, the edges are grouped based on their density value and listed.

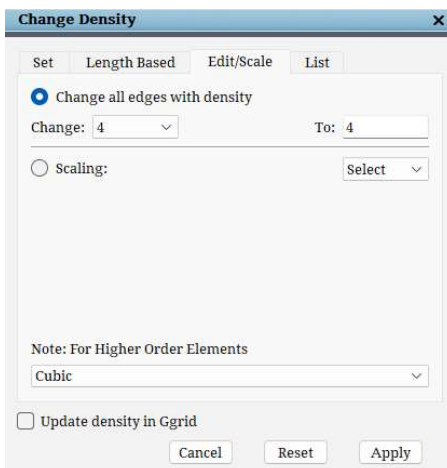




Figure 4.159: Edit/Scale panel

Clicking on **Change all edges with density**, changes the density of all the edges from one value to another. For e.g. If you want to change all the edges whose density is set to 12 to 16, then select 12 under the dropdown and enter 16 in the input box and click **Apply**. Now all the edges whose density was 12 are modified to 16. Similarly, you can also choose to scale the density of all the edges by a factor. Enabling the **Scaling** checkbox will let you change the densities based on a factor. If you want to double the density, set the dropdown to '**Twice**'. To reduce the density by half, set the dropdown to '**Half**', and click **Apply** to modify the densities. The **Reset** button resets all the modifications which are done in the dialog

box. Note that it won't be able to reset the modification once the dialog box is closed and reopened.

In case a higher order grid is to be created, then the dropdown under the **Note: For Higher Order Elements** can be used. Specifying the type of higher order grid to be created, and clicking on **Apply** automatically updates the density to the closest multiple of the order of the higher order mesh to be created.

Note - The greater the density of the topology, the longer the smoother, Ggrid, will take to complete. For two-dimensional grids, every time you double the density of the topology edges, you increase Ggrid's processing time by a factor of 4 ( $=2^2$ ). For three-dimensional grids, every time you double the density of the topology edges, you increase Ggrid's processing time by a factor of 8 ( $=2^3$ ).

Clicking on the **List** button opens a dialog box, as shown. The first section lists all the edges whose densities are modified from the default density. When you select an edge id in the list, the corresponding edge and its parallel edges would be highlighted for your reference. You can choose to modify them by clicking on the **Edit**  icon, which will enable the density value as editable input. You can also reset the density to default density by clicking on the **Delete**  icon. Note that if you modify anything under the density section, make sure to click the **Apply** button at the bottom to make these changes in the topology; else, everything will be revert to the state before opening the dialog box.

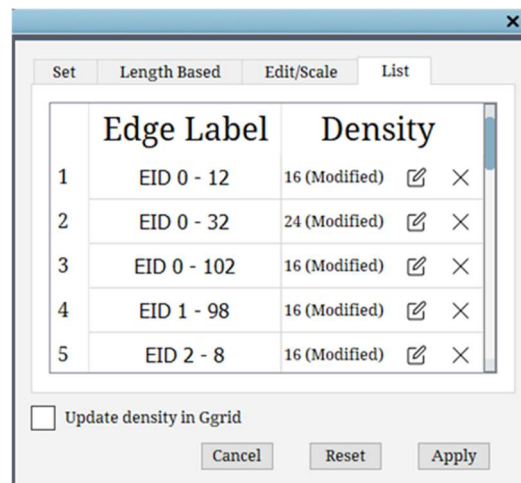


Figure 4.160: List panel

In addition to this, there is a density visualisation bar chart which will be switched ON whenever the density button is clicked. This legend shows the range of densities which are used in the current topology. Each density is shown with a colour code which is a gradient from blue to yellow. You can view all the edges which are having the same density by clicking on the desired colour bar.



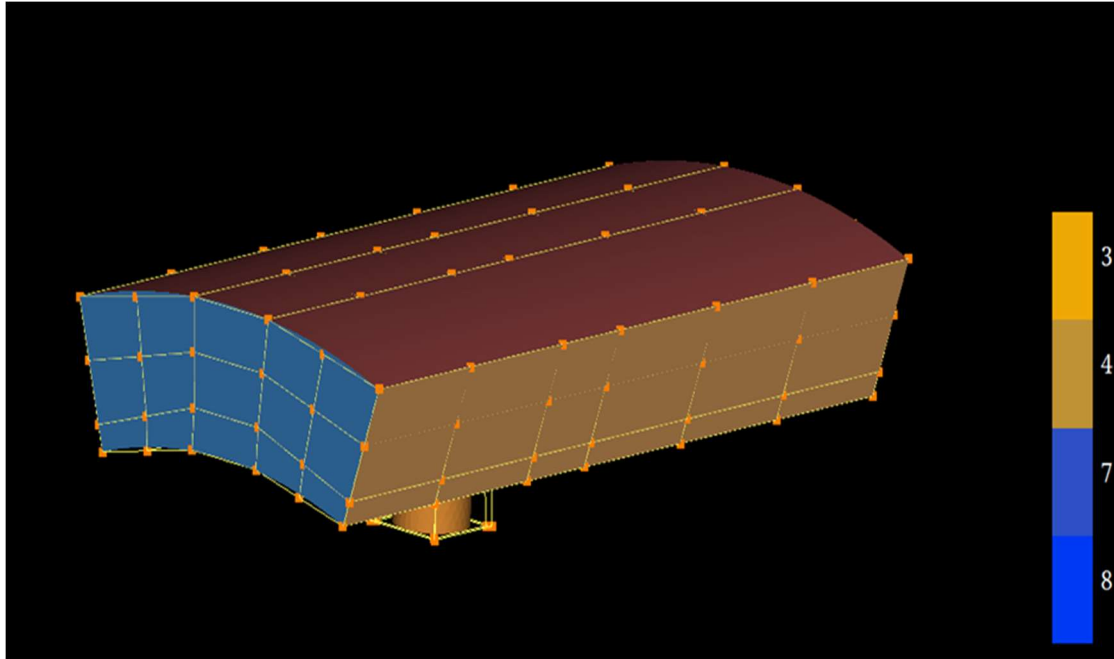


Figure 4.161: Density visualisation bar graph

## 4.9.2 Nest

*Nest* is very useful when you are handling geometries with large-scale differences. It automates the creation of topology to achieve nested refinement in a simple and efficient way.

Conventional meshing techniques on geometries with scale differences create grids with large aspect ratio, especially near the smaller features. Improvement of the aspect ratio can be done by increasing the density throughout the geometry to capture the geometry of a smaller scale. The refinement near the geometry is necessary but limiting because refinement propagates to the far-field. This limitation can be overcome by using nesting. It helps to increase the number of grid points near the geometry locally, with one to one connectivity, without increasing the points elsewhere. The nested structure loops back blocks near the geometry in the defined region.

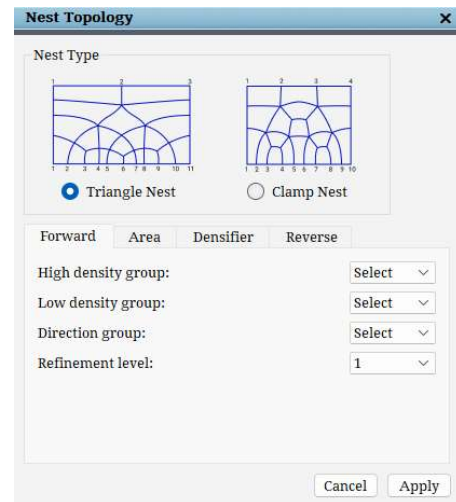



Figure 4.162: Nest dialog box

The *Nest* button  is located in the *Edit* section of the right-side panel in the **Topology** tab. Clicking on the button opens a dialog box, as shown. The **Nest Type** section showcases two

options showing the block structure for each nest type – **Triangle Nest** and **Clamp Nest**. Triangle Nest is used to fill up large volumes, and is an aggressive nesting pattern, ie, it leads to a steep rise in cell size. Clamp nest is generally used in internal regions when it is required to have nesting in internal flows. This type of nest leads to a gradual growth in cell size.

There are four tabs below this. The first three tabs deal with the nesting, and the reverse nest works opposite to the nesting, which helps coarsen the grid as it moves towards the far-field. This nest utility is further divided into 4 different tools having a dedicated tab, each working in a different way.

- **Forward Nest**

Forward nest can be achieved only on a complete topology. You need to provide *high-density group*, i.e. where the density should be more *low-density group*, i.e. where the density should be retained as original, *directional group* which guides the nesting tool along which direction it has to be nested, which is mostly along the wrap direction and finally the *number of refinements* which decides the number of layers of nesting.

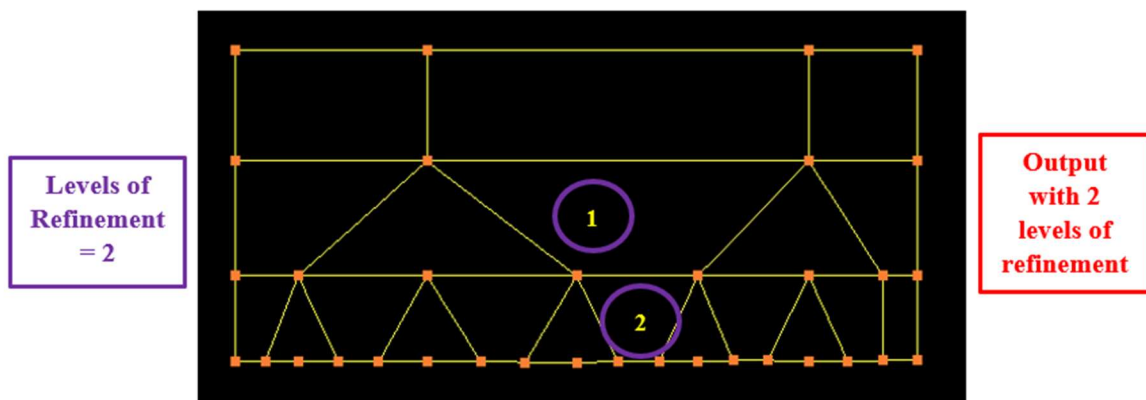


Figure 4.163 shows how the nesting is done on a topology locally. You need to provide the group number in which each group has been grouped into the respective fields. As you can see, for each layer, three corners are substituted for each corner on the previous layer.

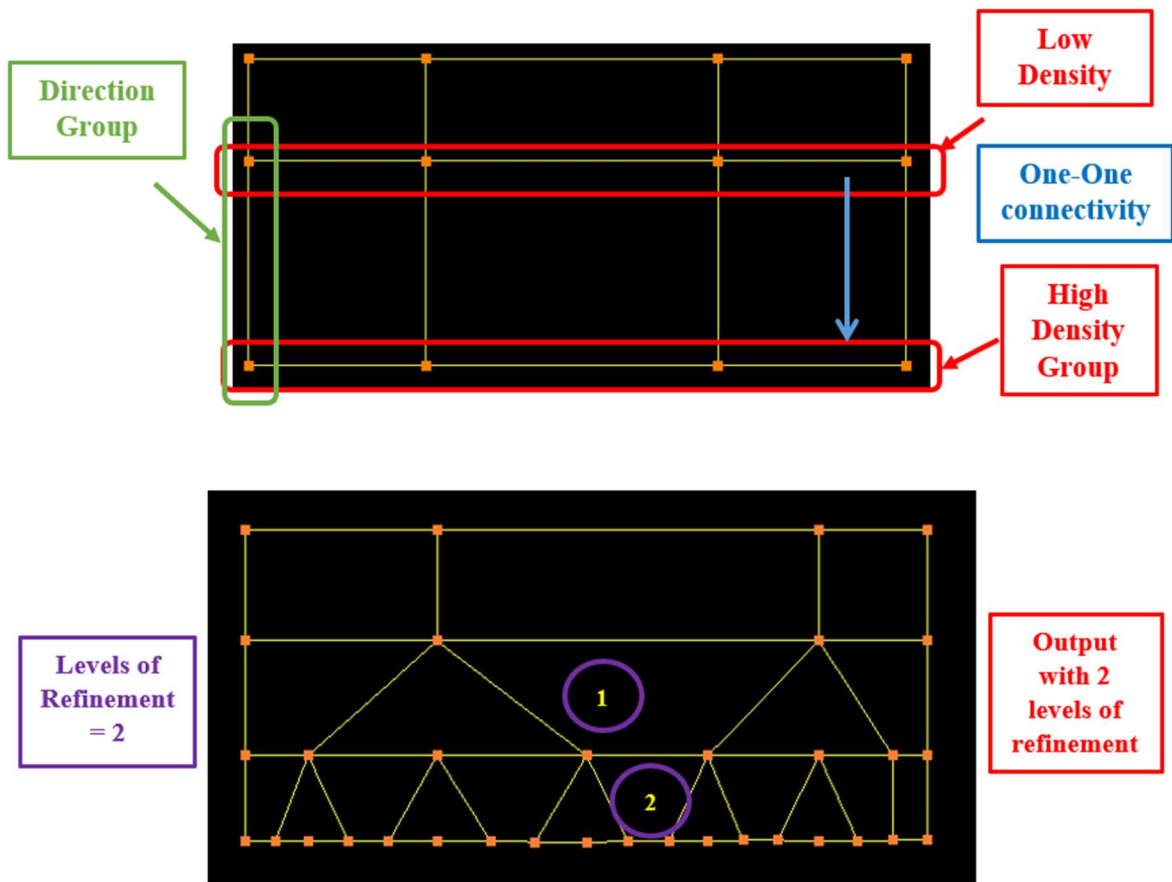


Figure 4.163: Nesting structure – Input & Output

There are few requirements that needs to be taken care in order to do the nesting efficiently. They are,

1. There should be one-to-one connectivity between Low-Density Group and the High Density Group, as shown in

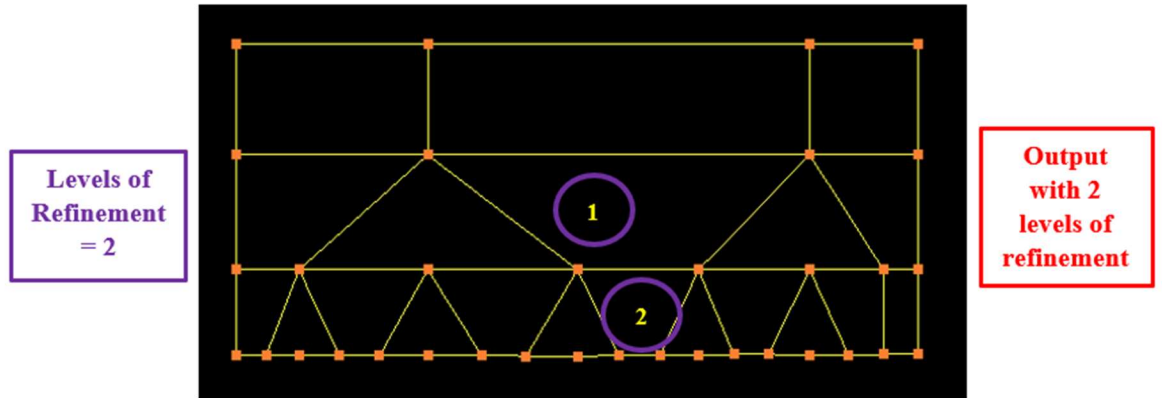


Figure 4.163.

2. Nesting cannot be applied for 3D as it may encounter a singular point. As shown in Figure 4.164, it cannot be applied along 3D sheet.
3. Nesting can be performed along 2D sheet only, and it is applied in 1D, as seen in Figure 4.165.
4. The topology structure of the high and low-density groups should be same at every cross-sectional sheet, as explained in Figure 4.166.

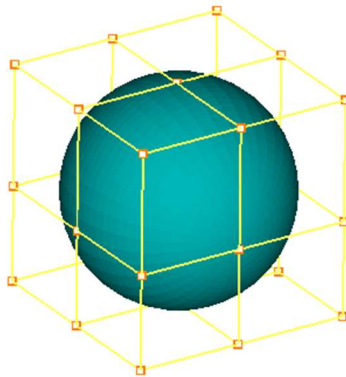


Figure 4.164: Topology where nesting cannot be applied

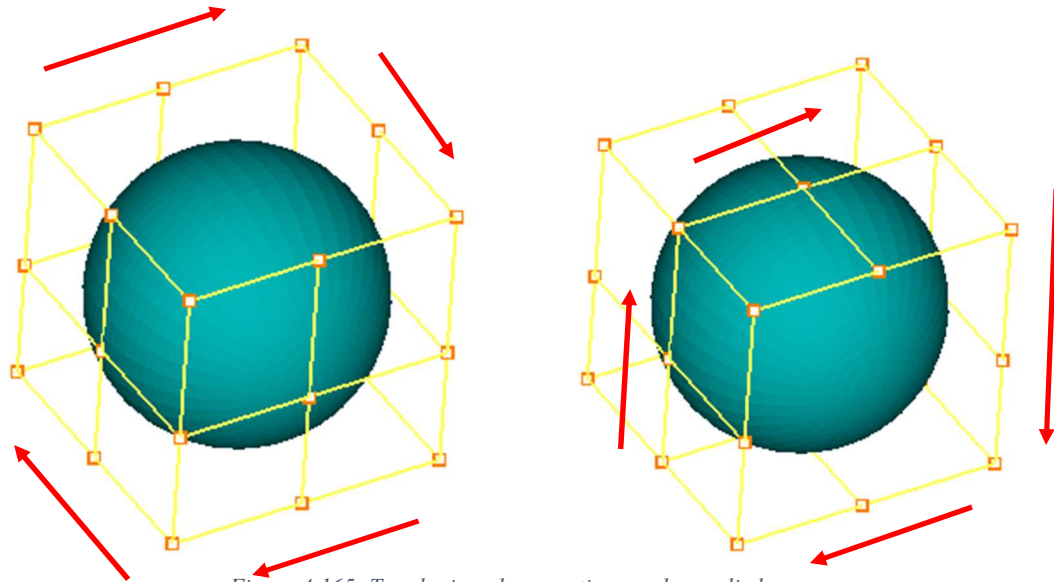


Figure 4.165: Topologies where nesting can be applied

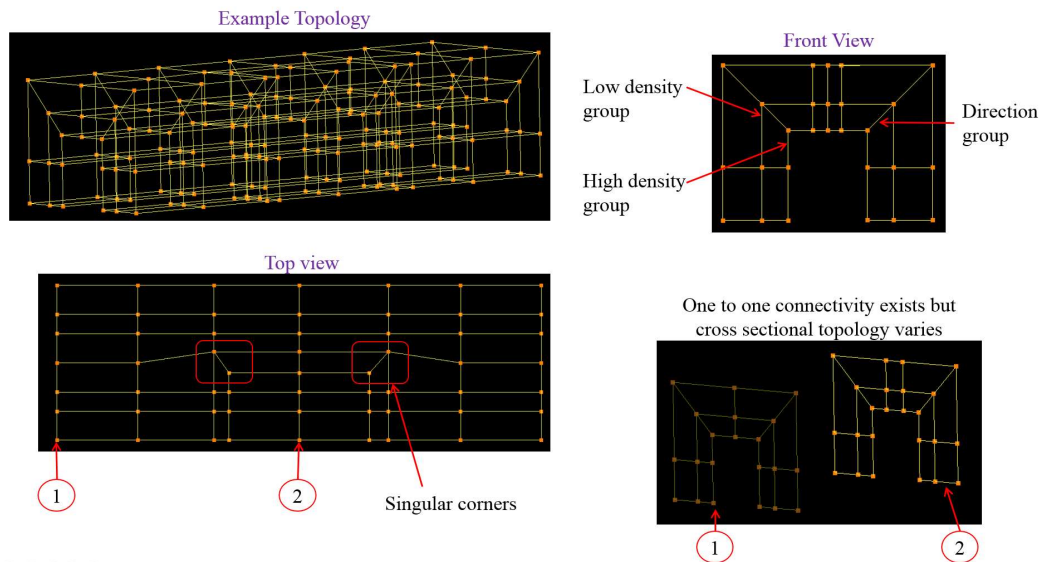


Figure 4.166: Topology structure of high and low-density group

- **Area Nest**

Area nest is similar to the forward nest tool where the nest is applied along one direction but in the area nest; it is applied in 2 directions simultaneously. Figure 4.167 shows the dialog box of area nest tab. Unlike forward nest, **Area nest** requires only 3 inputs such as **high**, **low-density group** and the **number of refinements**. Since the nesting is on 2 directions, it does not require direction group. It first applies nested refinement in one direction and takes those corners, and applies the next refinement in the other direction. Hence it is named as **Area nest**. All the requirements mentioned in the nest are applicable for the area nest as well, except for the last one since area nest is applied in 2 directions. Figure 4.168 shows an example of how area nest is being done.



explained above. For the area nest, it assumes the wrapped corners as the high-density group and the inserted sheet as low-density group and applies the area nest using the **number of refinements** provided in the dialog box. The **wrap percentage** is used to do the wrap. This is a more sophisticated tool compare to the **forward nest** as it is automated the grouping needs to be done for the forward nest/area nest. If the operation did not output the refinement level in the expected direction, undo the operation and enable '**Flip direction**' checkbox & try again.

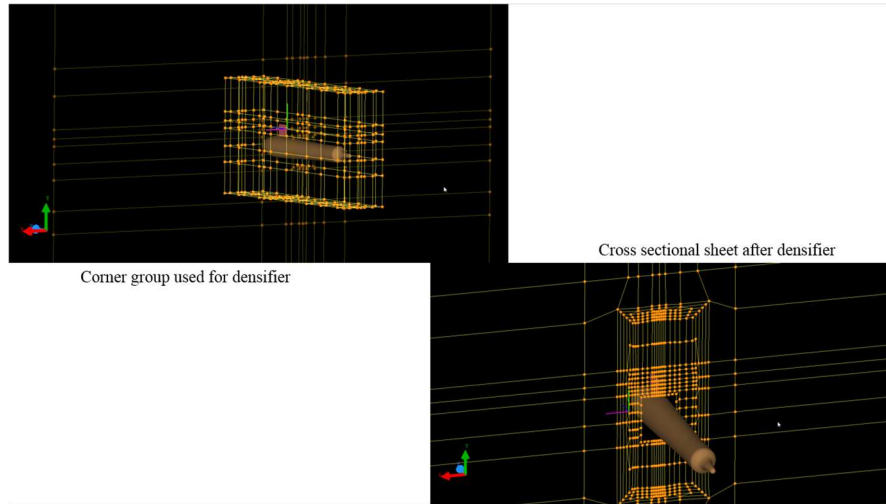


Figure 4.170: Topology Densifier

- **Reverse nest**

The **forward nest** creates a high-density topology from a low-density topology, whereas **reverse nest** works in the opposite way, which creates a low-density topology from a high-density topology. A complete topology is required to apply **Reverse nest**. This helps in coarsening as the topology moves from the geometry to the far-field. Figure 4.171 shows the dialog box of the reverse nest.

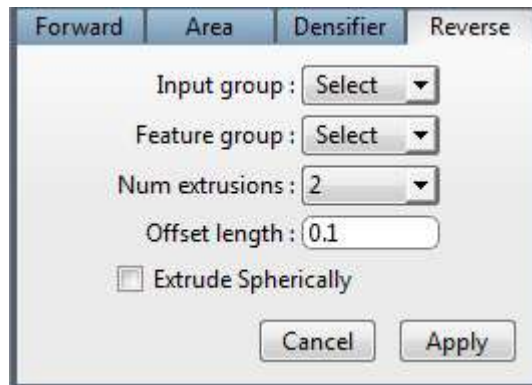


Figure 4.171: Reverse nest dialog box

The **input group** contains the corners in the outermost layer of the topology (assigned to the outer boundary). The **feature group** contains the corners where the reverse nest should not be



done. The *number of extrusions* is the number of layers the reverse nest has to be done. The *ratio* is the ratio of the distance at which each extrusion should be placed. It is the distance between each successive layer given in terms of ratio. Other than the feature group corners, general reverse nest omits the corners on the intersection of two boundary surfaces from reverse nesting. Enabling the checkbox of *extrude spherically* make the reverse nest to account the corners at the intersection of two surfaces.

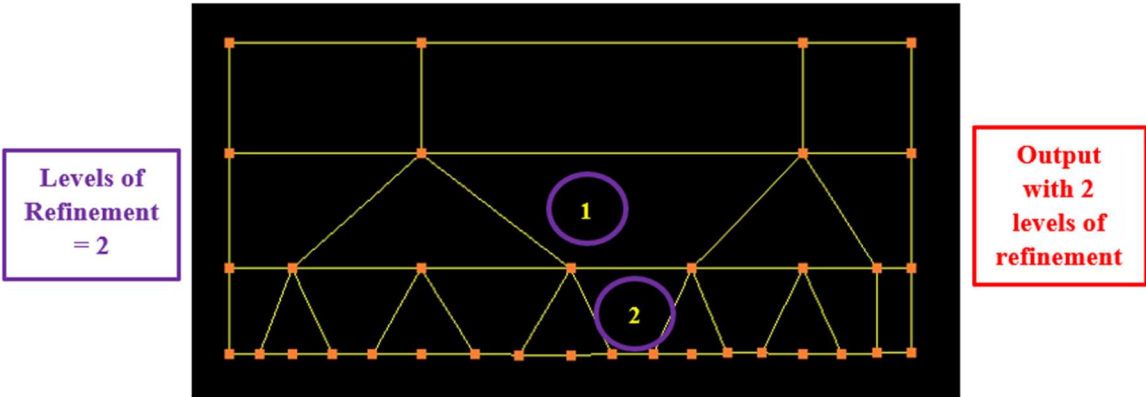


Figure 4.163 shows how the forward nest works, whereas the reverse nest works in the opposite direction and in 2 directions similar to the area nest.

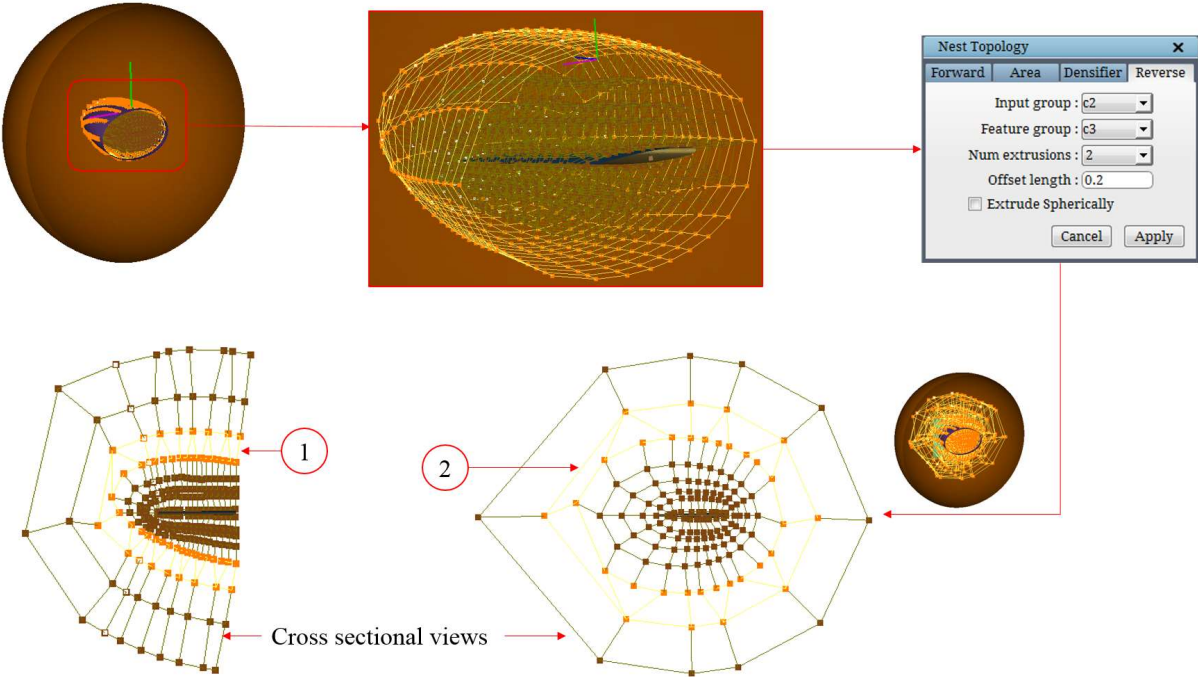


Figure 4.172: Reverse Nest topology

### 4.9.3 Compact enrichment

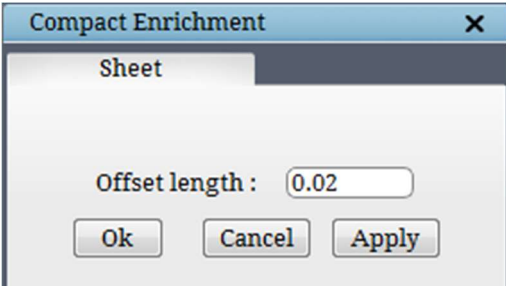



Figure 4.173: Enrich dialog box

Whenever the grid has to be refined, increasing the density is the tool that comes to mind first. But increasing the density also affects the far-field, which is not desirable. So *GridPro*

provides a tool called **Enrich**  that simply adds more topology corners in a complete topology without affecting its validity and without propagating to far-field. This can be applied on a string of corners which will create a buffer layer around it using the offset ratio. This buffer layer is nothing but a special case of the internal wrap. Figure 4.173 shows the dialog box of the compact enrichment. You need to provide the **offset ratio**, which gives the distance at which the internal wrapped corners should be placed. By default, it takes *0.02* as offset ratio. The corners where the compact enrichment has to be applied have to be grouped in a group number which should be provided in the **Input corner group** field.

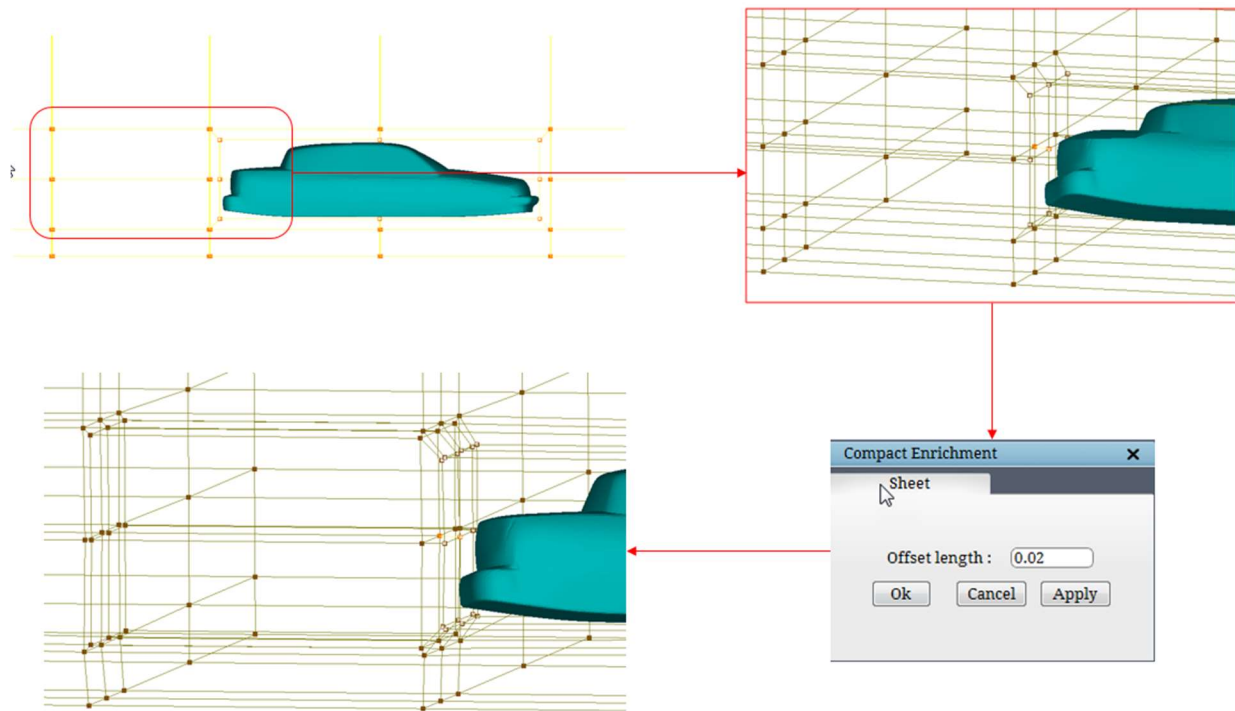


Figure 4.174: Enrich Topology

## 4.10 Surface Assignment Utilities

Unlike other grid generation programs, *GridPro* automatically generates a multi-block grid within a region bounded by surfaces after the wireframe topology is given. However, to complete the topology, you must tell *GridPro* which collection of corners is to be assigned to each surface. Even if you are gridding a region bounded by only one surface, you have to assign the boundary corners of your inputted wireframe to that surface.

Corners can be assigned to zero, one, two, or more surfaces. If a corner is assigned to each of two intersecting surfaces, then the corner will go to the intersection between the two surfaces. For example, when gridding a rectangle, the double assignment to both a horizontal and a vertical side will attract the corner to the vertex of the rectangle formed by the intersection. If it were the case of a box in three dimensions, then the attraction would be to a box edge. A triple assignment, however, would send the corner to a box vertex.

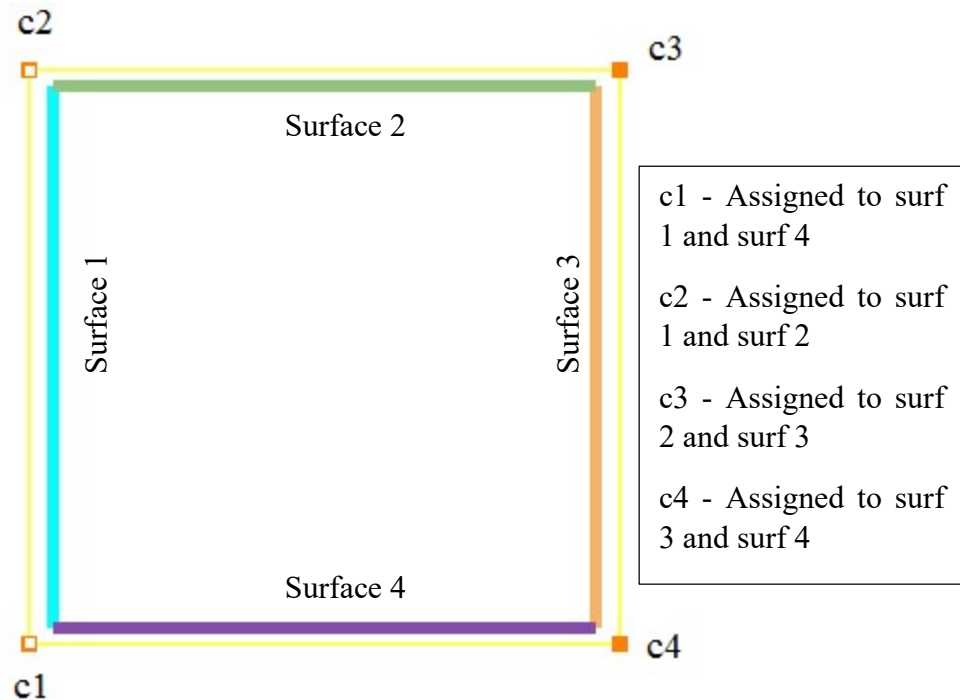


Figure 4.175: Rectangle topology with double assignment

Corner assignment to the surfaces of the topology is an easy and important step for building a valid topology. There are many tools available in the GUI to collect the appropriate corners in a separate corner group so that it is easy to assign these corners to the current surface of the topology. All those tools for collecting the corners in different corner groups are described in *section 4.4*. This section describes how the corners in the current corner group of the topology can be assigned to the current surface. Figure 4.176 shows the **Assignment** section in the right-side panel of the topology tab. Note that all these operations are in relation to the *current* surface. So the appropriate surface should be made *current* before assigning/un-assigning the corners. Refer *section 2.5* to know how to make a surface *current*.

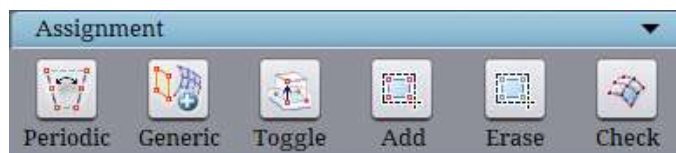


Figure 4.176: Assignment section

### 4.10.1 Toggle



The **Toggle** button is used to assign/un-assign corners to the current surface on an individual basis. The corners of the topology can be individually picked up and assigned/unassigned to the *current* surface of the topology. A corner is eligible for picking only if it is appended to the current corner group. If none of the corner groups is active, all the corners are eligible for selection. If a corner which is already assigned to the *current* surface is picked, it is unassigned, and similarly, when an unassigned corner is picked, it is assigned to the current surface. This is very helpful when the user wants to pick the missing assignments manually. If there are no surfaces in the topology, selected corners are just ignored. So making the desired surface *current* is the foremost operation in surface assignments, next comes the corners to be assigned has to be grouped. Once the corners to be assigned is grouped, click on the **toggle** button and pick the corners to be assigned one by one. Each corner assigned to the *current* surface will have a white dot at its centre. This white box provides you with a visual indication of confirmed surface assignment, which proves to be useful when many surfaces and many corners are involved in a topology. When the current surface is switched, the white dot appears automatically at each of the corners assigned to the new current surface. If you wish to deactivate a surface assignment, click on **Toggle** button and select the assigned corner. The small, white internal box will disappear.

Though special tool buttons are provided for assignment and un-assignment of corners, the shortcuts ‘S’ and ‘Shift+S’ are more comfortable and handy to apply these operations. The keyboard shortcut ‘S’ can be used to assign, and similarly, the shortcut ‘Shift+S’ can be used to un-assign a corner which is assigned to the *current* surface.

### 4.10.2 Add



To assign multiple corners to the *current* surface simultaneously, use **Add** button. When **Add** button is active, user can drag a box (by pressing the mouse right-click button) to select the corners from the graphics window. The selected corners which are not assigned to the *current* surface are assigned to it, and all other selected corners retain their assignment. Hence it makes sure that all the selected corners are assigned to the *current* surface. Note that only the corners appended to the current corner group are eligible for selection. If none of the corner groups is active, all the corners of the topology are eligible for selection. Also, note that if there are no surfaces in the topology, this operation is ignored. Like the Toggle button, the **Add** button can also be achieved using the shortcut ‘S’ key. By holding the ‘S’ key, in addition to picking corner one by one, you can also drag a box with right-click mouse button around a group of corners. Note that these shortcuts can be modified using the keyboard settings described in *section 2.2*.

### 4.10.3 Erase

There are two ways to erase or un-assign corners. Once the **Erase** button is clicked, a dialog box pops up, as shown in Figure 4.177. **Screen Select** lets you un-assign multiple corners that are selected by dragging the mouse. While **Clear** removes all assignments for the *current* surface.

- **Select**

To un-assign, multiple corners to the *current* surface simultaneously, use **Screen Select** button under **Erase**. When **Screen Select** button is active, user can drag a box (by pressing the right-click mouse button) to select the corners from the graphics window. The selected corners which are assigned to the *current* surface are unassigned. Note that only the corners appended to the current corner group are eligible for selection. If none of the corner groups is active, all the corners of the topology are eligible for selection. Also, note that if there are no surfaces in the topology, this operation is ignored. This operation can also be achieved using the shortcut '**Shift+S**' key. By holding the '**shift+s**' key, in addition to picking corner one by one, you can also drag a box with the right-click mouse button around a group of corners. If the corners are assigned, it becomes unassigned; if it is not assigned, then the operation is ignored on those corners.

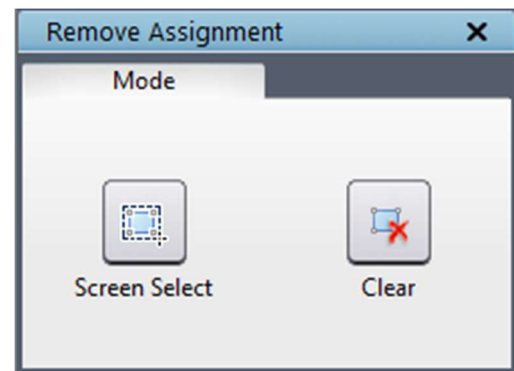


Figure 4.177: Erase dialog box

- **Clear**

All the corners assigned to the *current* surface can be unassigned with a single click using **Clear** button. This ensures that none of the corners is assigned to the *current* surface. When there are no surfaces in the topology, this operation is ignored. This operation is internally used when a surface is deleted from the topology.

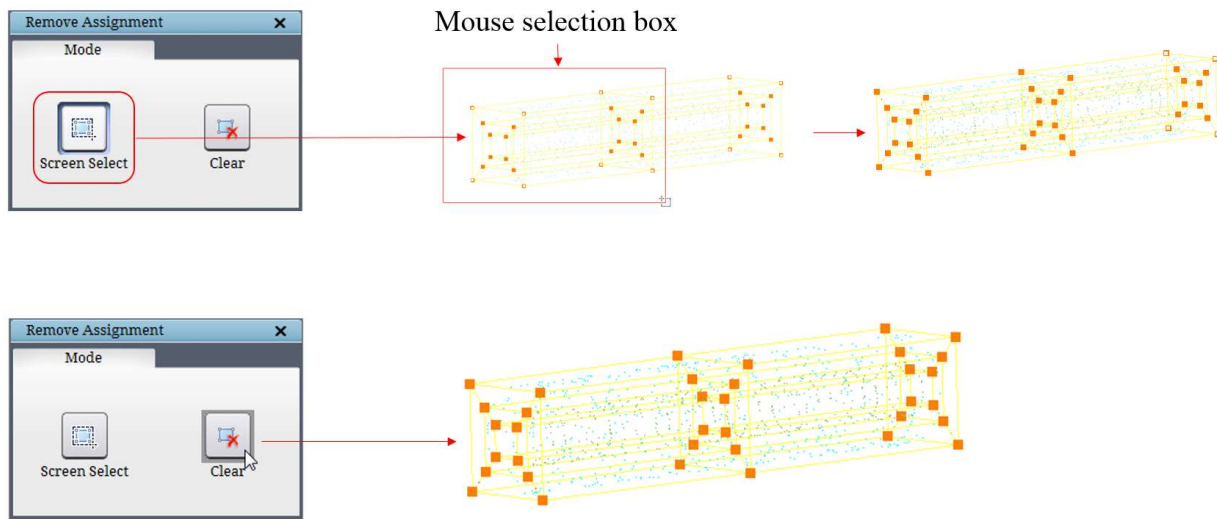


Figure 4.178: Various Erase modes output

#### 4.10.4 Periodic

As discussed in *section 3.1*, similar to any other surface, you must create and assign corners to the periodic surfaces. Unlike other surfaces, you have to use **Periodic** button in the assignment section. Just as the periodic surface is on the periodic border of an object, the corners assigned to the periodic surface is on the periodic border of the wireframe. Clicking on the **periodic** button opens a dialog box, as shown in Figure 4.179. Both the surface creation and assignments are done at the same time for a periodic surface. But you can also modify it later using the same button. Once it is opened, select the **periodicity type**, which can be either rotational or translational. Next, the **Axis** has to be defined, which has the default 3 axes under the dropdown. It is preferable to use any one of these axes. Hence rotate the geometry such that it has one of these axes as periodic axis instead of rotating the periodic surface to suit the geometry axis. Then **periodicity** has to be defined, which should be a factor of 360. This helps in identifying the periodic boundaries. Depending on the other surface assignments in the topology, block structure and no. of faces on either side of the boundaries, it calculates all possible periodic pairs in the UI and lists them under periodic pair column. Clicking on the pair displays the faces in red colour, as shown in Figure 4.180. Then click on pair to create a periodic surface and assign the selected faces to it. This automated pair detection eases assigning the corners to the periodic surface operation a lot. In some cases, if it is not able to detect the pairs, then there might be some topological errors.

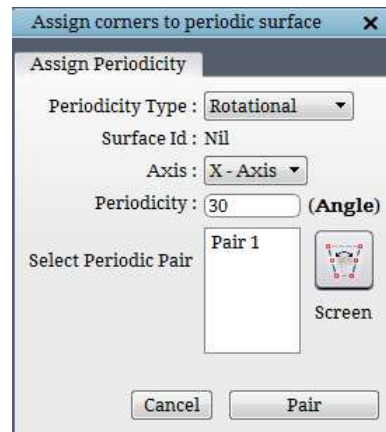


Figure 4.179: Period dialog box



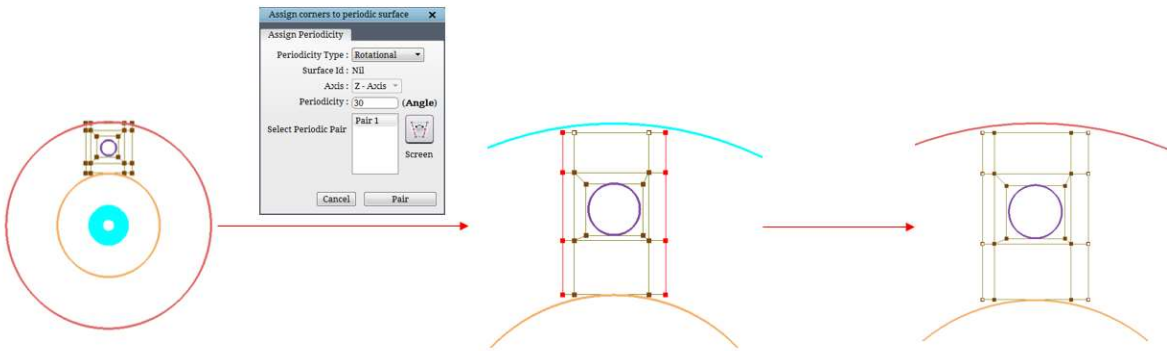
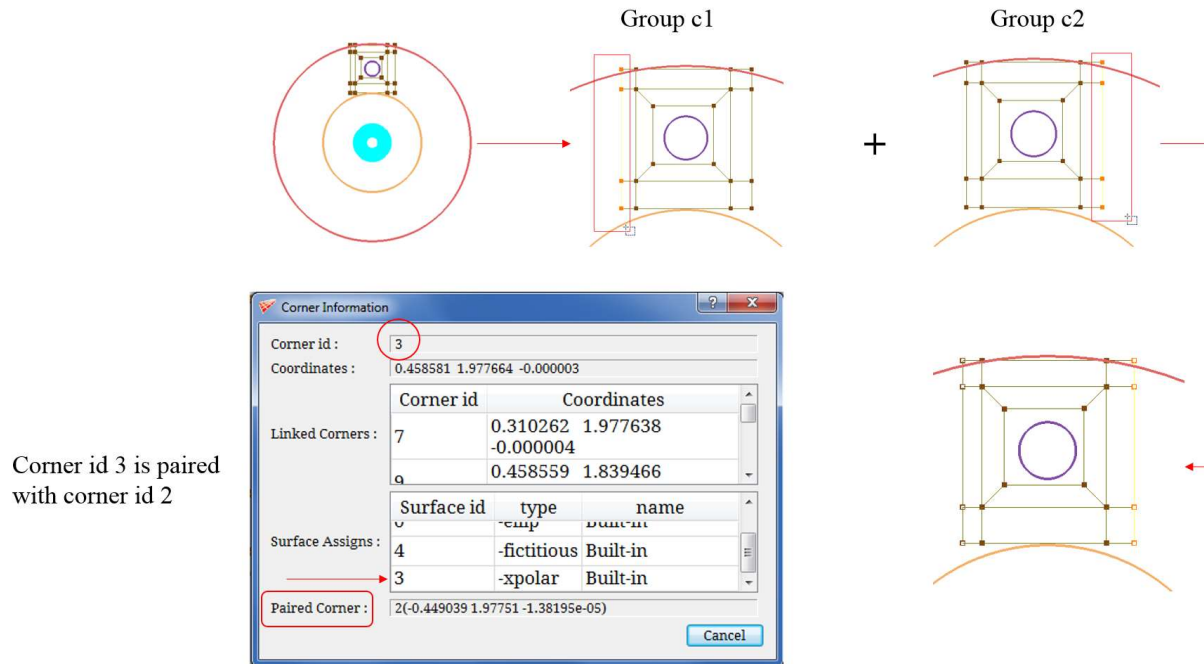


Figure 4.180: Automated periodic pairs detection

You are also provided with the manual assigning button through which the periodic pairs can be assigned manually, as explained below. Before you start assigning the corners, you have to group the corners on the periodic boundary in two different groups or in the same group, i.e. corners represent one end of the periodic boundary in one group, and the corners represent the other end of the periodic boundary in the second group or both the corners in the same group.



For Non-periodic surfaces, paired corner information would be 'NONE'

Figure 4.181: Manual periodic surface assignments


Then click on the **Screen** button and select all the corners which represent the one end of the periodic surface and then select the corners which represent the other end of the surface separately if it is in the same group. If the corners are grouped in different groups, switch on the



**Screen** button, select the first corner group corners, and then switch on the other group and select all the corners. The corners of one group will be mapped onto the other when reflected about the periodic axis and list them as a pair under the periodic pair column. After which, select that pair and click on ‘pair’ option to assign.

#### 4.10.5 Check



**Check**  button allows you to check the assignments of the *current* surface. It displays only the corner assigned to the *current* surface. All the corners except the corners assigned to the current surface are hidden. Assigning corners to a surface is a process where you are prone to make mistakes by assigning corners that are not supposed to be assigned. As you know that the surface assignment corners of a *current* surface is highlighted with a white dot, you switch on this button and scroll through the surfaces and check which corners are assigned to each surface.

#### 4.10.6 Generic

Generic option is the latest addition to the Assignment section. Once clicked, it opens a dialog box, as shown in Figure 4.182. There are two columns and two buttons in the dialog box. The two buttons are nothing but **Add** and **Erase** button. The two columns represent the Face Groups and Surface List. Face groups column lists all the possible external face groups. According to the rule of surface assignments, all these face groups must have one or more surface assignments. If it is assigned, the face groups are named as “<surf id>-assign”, and if it is unassigned, then they are named as “Unassigned-<face group no.>”. The second column is the surface list which lists all the surfaces in the topology. This is to make the assignment process easier. The workflow goes as follows. First, you select one of the face groups in the list, which will be highlighted in green colour in the UI. Then you select the surface ID of the surface to which this face group has to be assigned. Instead of scrolling through the surfaces, you can select the ID directly from the list if you know it, which will make that surface current and once it is current, click on the add button to assign the selected face groups to the current surface. To unassign, any assigned face group, follow the same workflow.

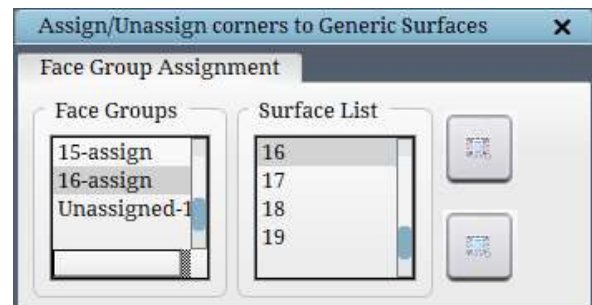


Figure 4.182: Generic dialog box

## 4.11 Surface Building Utilities

Many times you need to create a surface (internal surface most often) in *GridPro* using the corners and edges. The use of internal surface is to capture the slope discontinuity in the geometry or the sharp features in the geometry. To do so, the internal surface should exactly pass through the slope discontinuity of the geometry. And this surface should also cut the geometry orthogonally. These surface building utilities, which are grouped under ***Surfaces Tools*** located in the Menu bar, help you to satisfy the conditions in creating the internal surface.

### 4.11.1 Feature edges

***Feature edges*** is the foremost tool to be used while creating the surface. This creates corners on the *current* surface slope discontinuity using the surface nodes, otherwise called as *surface triangulation nodes*. Once the **feature edges** are selected from the dropdown list, a dialog box opens, as shown in Figure 4.183. You need to input the **threshold angle** to calculate/find out where the slope discontinuities are. By default, it takes 40 degrees. The feature angle will be calculated at every node on the *current* surface; wherever it exceeds the given feature angle, a corner will be created on that node which will be later linked with the corners created adjacent to it and also assigned to the *current* surface. You can also include/exclude the boundary edges of the surfaces from using this utility. Including the boundary, edges will make the utility calculate the feature angle on the surface boundaries also and create corners if it exceeds. This can be done by checking/unchecking the ***include boundary edges*** check box.

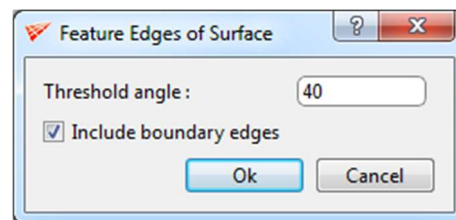


Figure 4.183: Feature edges dialog box

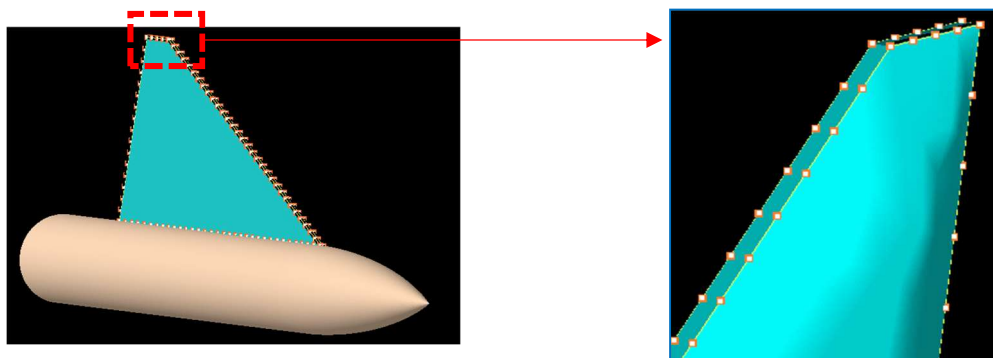


Figure 4.184: Feature edges created on a blunt edge wing

### 4.11.2 Intersection of Surfaces

*The intersection of surfaces* option under *Surface tools* dropdown helps in creating corners (all connected by edges) on the intersection of two or more surfaces. Once selected, a dialog box opens where you need to input the **surface group**, which contains the intersecting surfaces. Every corner created is automatically assigned to each of the intersecting surfaces. Also, there are no restrictions on the type of intersections you can contour with corners. If you are not able to capture any intersection correctly, switch ON the pick button and click at the intersection of the two desired surfaces to capture them.

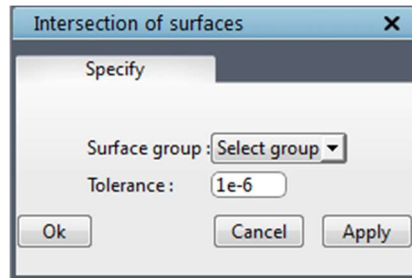


Figure 4.185: Intersection of surfaces dialog box

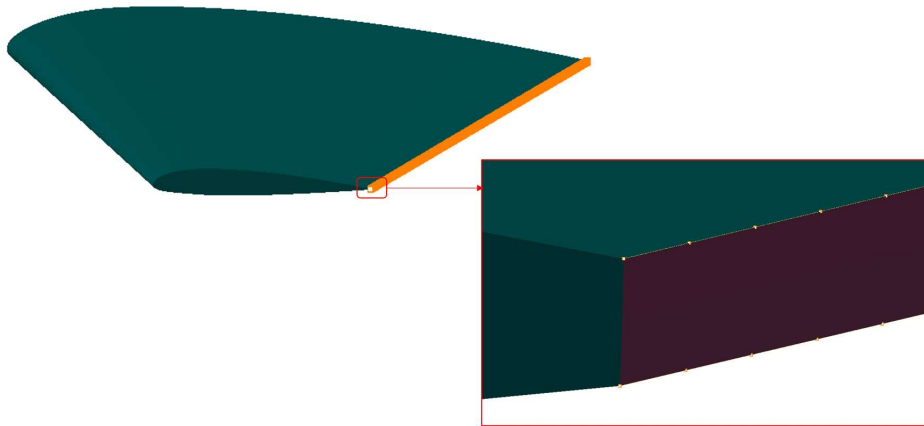


Figure 4.186: Corners created at the intersection of two surfaces

### 4.11.3 Ribbon

Once the corners are created on the slope discontinuity of the surface or the intersection of surfaces, it needs to be extruded normally inwards or outwards that will satisfy the condition of orthogonality for the surface created on the slope discontinuity. **Ribbon** creates a layer of corners, either normally inwards or outwards to the given set of corners with the given width based on the given corners and its assignments. Once selected, it opens the Generate Ribbon dialog box, as shown in Figure 4.187. The **path group** is where the corners created by the feature edge/intersection are grouped.

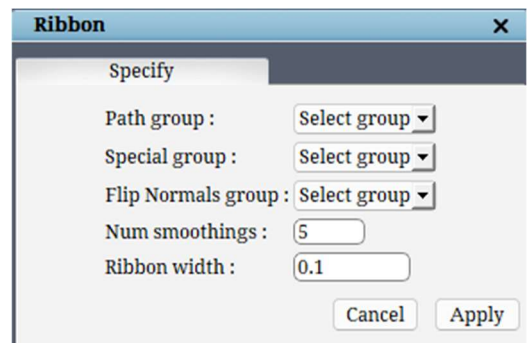


Figure 4.187: Ribbon dialog box

This corner group is used to create the ribbon layer out of it. The ***special group*** is the corner group which should contain the corners whose ribbon corner need not be created normal to the surface intersection. The ***inverted surface group*** is the corner group which contains the corners whose normal should be reversed before creating the ribbon corner. The ribbon corner created from those inverted surface group will be using the negative normal direction. Out of these 3 groups, the path group is a must while the other two are optional. Similarly, the corners in the path group should be assigned at least to one surface in order to find the normal. The ***number of smoothing*** is the number of times the smoothing algorithm has to be run on the ribbon layer. The ***ribbon width*** is the distance between the input corners and the ribbon layer corners.

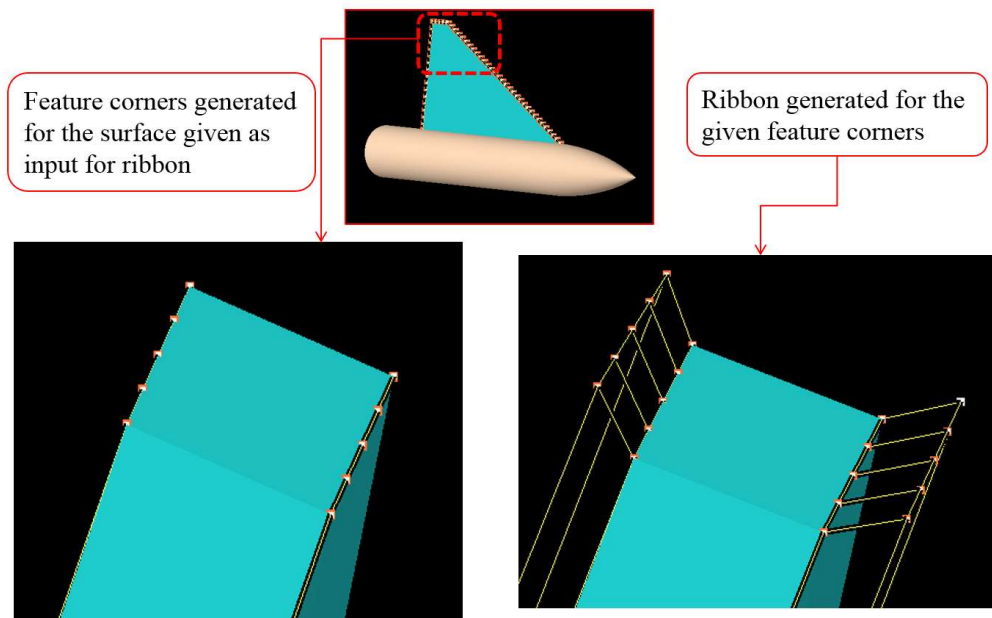


Figure 4.188: Ribbon generated from the feature edge corners

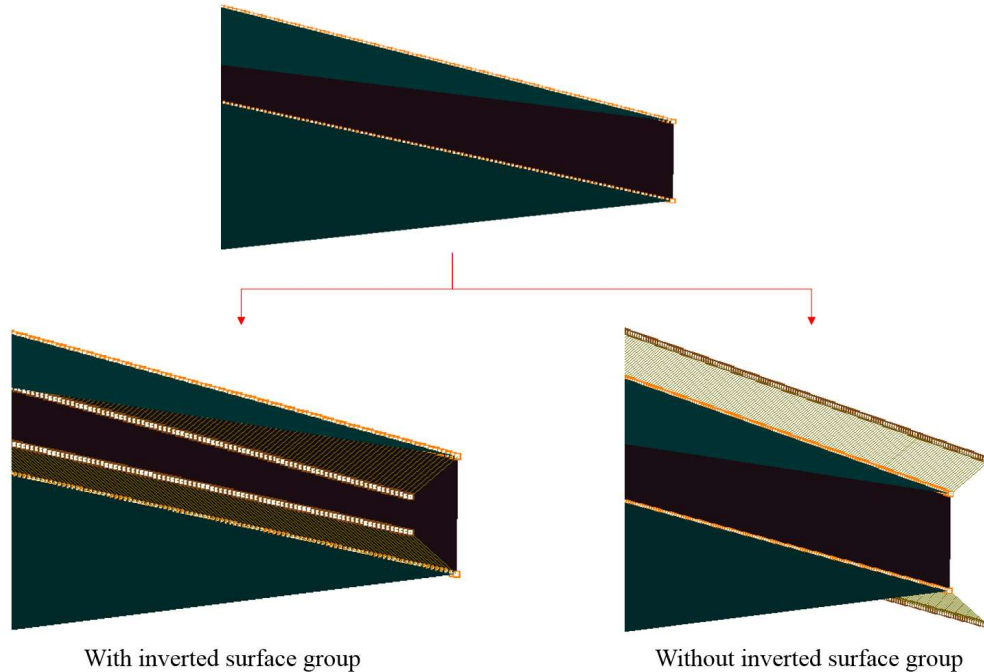


Figure 4.189: Ribbon – Inverted surface groups

#### 4.11.4 Ribbon nest

Many times, the nodes on the surface are more to get the shape of the surface correct, which in turn ensures good quality grid. So the number of corners created by feature edges and ribbon would also be more. Working with more number of corners to build the wireframe for the surface would be difficult and also a time-consuming process. So it is advisable to reduce the number of corners. To do so, GridPro provides a function, **Ribbon nest**, which creates a given number of layers with a reduced number of corners in each layer without affecting the normal of the ribbon generated. This process of reducing the number of corners is also called **reverse nest**. Reverse nest is used in case of a 2-dimensional reverse nest, while ribbon nest is used for a 1-dimensional reverse nest. To do ribbon nest, two layers of corners are required; one is the layer on which the nesting has to be applied and the other layer which is created using the ribbon to retain the normal. Once selected, Surface Nest dialog box opens, as shown in Figure 4.190.

The **nesting group** is the corner group which contains the layer of corners from where the nesting should be applied. The **ribbon group** is the corner group which contains both the nesting group corners as well as the corner layer next to it (i.e. corners generated using the ribbon). The **special group** is the corner group which contains the corners where the nesting should not be applied. The number of layers that needs to be created can be provided in the **number of extrusions** field. The number of corners will be reduced with each layer. If the corners cannot be reduced further, the same number of corners will be created in each layer until it satisfies the given number of extrusions. The distance/length between each layer can be provided in the **ratio** field in the form of a ratio. Grouping the corners is always an important task in GridPro to do

any operation. In order to continue building the wireframe after nesting, you need to group the outermost layer in a group and start working on it. You can group the outermost layer by selecting a group number from the **outer group** dropdown list. Ensure the group that you provide in the outer group is an empty group.

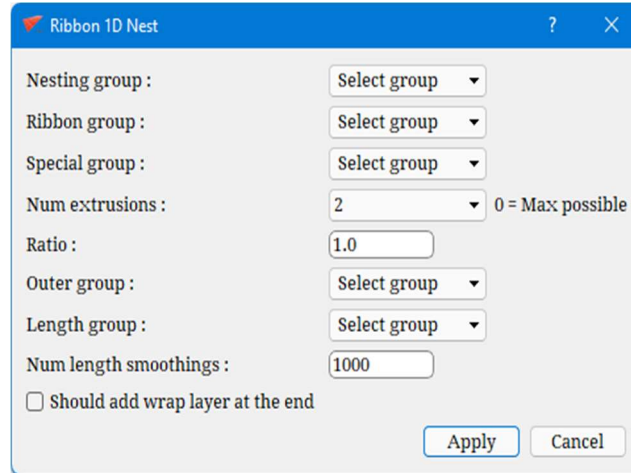


Figure 4.190: Ribbon nest dialog box

The **length group** is the corner group where the length between each nesting layer will not be changed due to smoothing. The **number of smoothings** is the number of times the smoothing algorithm has to be run on the resulted ribbon nest layer. You can also add a wrap layer after creating the ribbon nest by enabling **should add wrap layer** checkbox. This helps to avoid skewed cells created in the surface because once the ribbon nest is created, the outer layer will have corners shared by 4 edges. If you start creating a wireframe from that corner, it becomes a corner shared by 5 edges. In order to get a smooth transition on the surface, it is better to add a wrap layer after creating the ribbon nest.

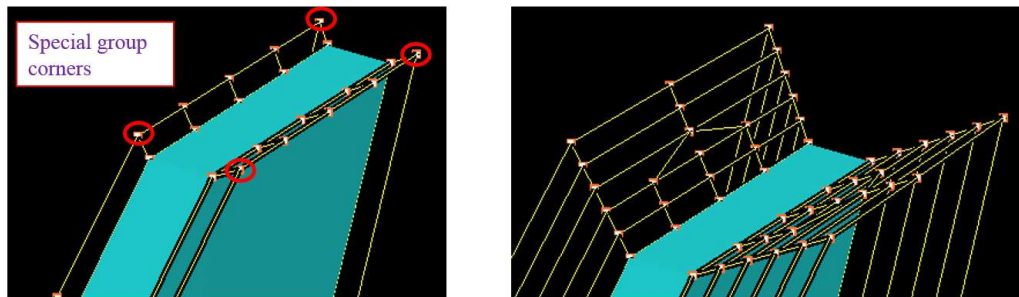


Figure 4.191: Ribbon nest generated from the ribbon corners with one level of refinement

### 4.11.5 Controlnet Surface



In *GridPro*, the quad surfaces are generated using the **Controlnet Surface** tool. In order to create a controlnet surface, click on the **Controlnet Surface** button in the dropdown list of the **Surface Tools** menu in the menu bar. Figure 4.192 shows the controlnet surface dialog box. Controlnet surfaces are created based on a group of topology corners you input. Select the input corner group next to the **Input group** field. Two optional features are also provided, such as feature corner group and smooth corner group, which is the part of input corner group. **Feature corner group** creates the controlnet surface, which passes through the given feature corner group. **Smooth corner group** smooths the surface in that given smooth corner group region using the

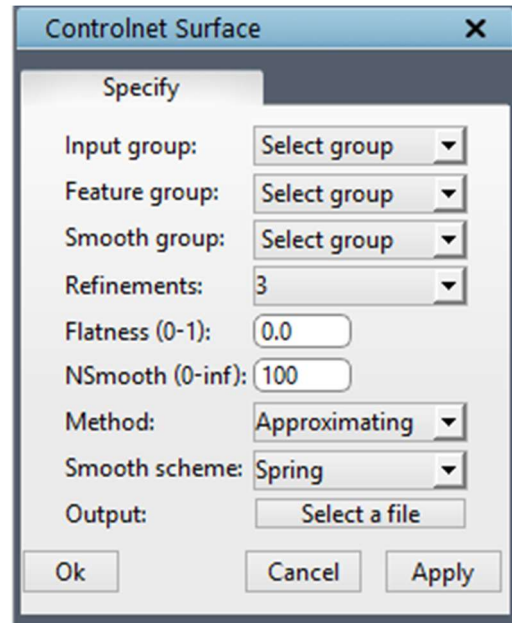


Figure 4.192: Controlnet Surface dialog box

smoothing scheme, which is inputted in the '**Smooth Scheme**' field. You can select the smoothing scheme from the dropdown button next to smooth scheme. In addition to this, smoothing or refining the whole surface can be done by inputting a refinement level in the **Refinements** dropdown. Surface created using high number of refinements is heavy in terms of higher number of cells. Depending upon your requirement, it should be selected. Usually, refinements levels are either kept as 3 or 4. Refinement also tries to smooth the whole surface irrespective of the smooth corner group. You can define the **flatness** and **number of smoothing** required for your surface. These parameters will apply to the whole input corner group. Similar to the smoothing scheme, you can select one of the three methods available to create the controlnet surface. In **approximating**, controlnet surface tries to create surface passing approximately through all the input corners; it may not be passing through all the input corners. In **interpolating** method, it tries to ensure that controlnet surface is passing through all the input corners. **Sphere like method** is similar to interpolation, where surfaces pass through all the corners, but between the two adjacent corners, it tries to create a sphere-like shape. The surface will always pass through the end corners irrespective of the method used. The output file can be named by clicking on the push button next to the Output and mentioning the desired file name. The extension for this file should be \*.quad. You can change the location of the output file if desired. Click on **Apply** after defining the output file name to create a control net surface.

There are various ways in which you can control the creation of a surface using controlnet utility. They are

1. Default Input
2. Face exclusion
3. Inserts



4. Feature group
5. Smooth group
6. Different Algorithms
7. Refinement
8. Flatness

All these options are explained below with an example image.

- **Default values**

Consider a topology that looks similar to an elbow, as shown in Figure 4.193. The surface is created using default values such as input group with 3 levels of refinements using the approximating method and spring smooth scheme.

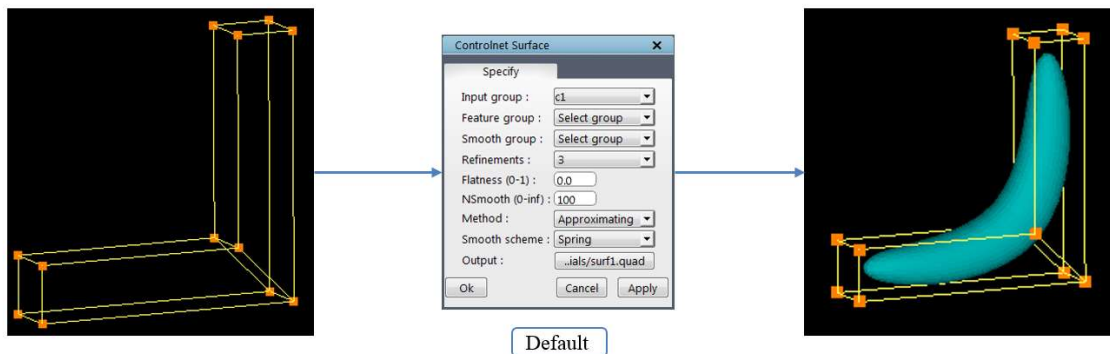


Figure 4.193: Controlnet surface using default values

- **Face exclusion**

Notice that the output in Figure 4.193 is closed on both ends since the structure at the ends represents a perfect quad. If you want an open surface, either the quad face has to be excluded, or a sheet needs to be inserted and make it open similar to the steps that are followed for wrapping a quad as in 'Medium Tutorial 4 – Y-Exhaust'

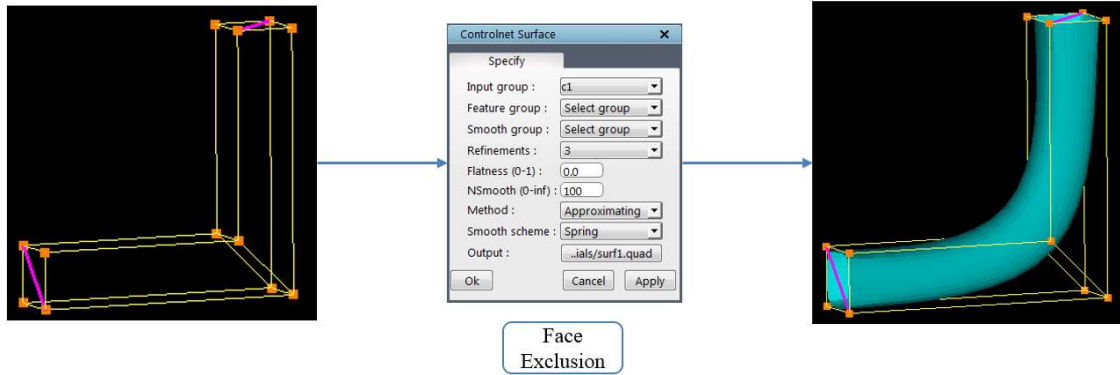


Figure 4.194: Controlnet surface using face exclusion

- **Inserts**

You can add more inserts to capture the curvatures in the wireframe as accurate as possible.

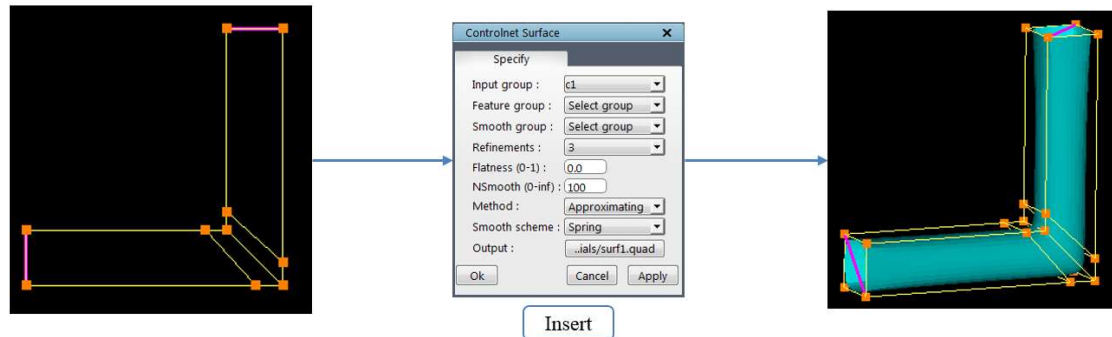


Figure 4.195: Controlnet surface using Inserts

- **Feature group**

As mentioned in the above description, feature group ensures that the surface is passing through the given group of corners

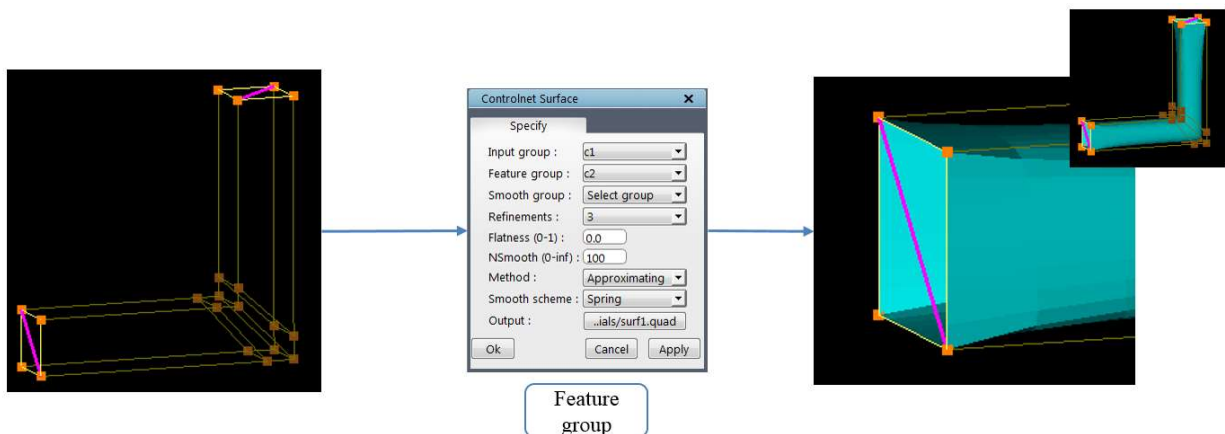


Figure 4.196: Controlnet surface using feature group

- **Smooth groups**

Smooth groups help in smoothing the surface in the input region.

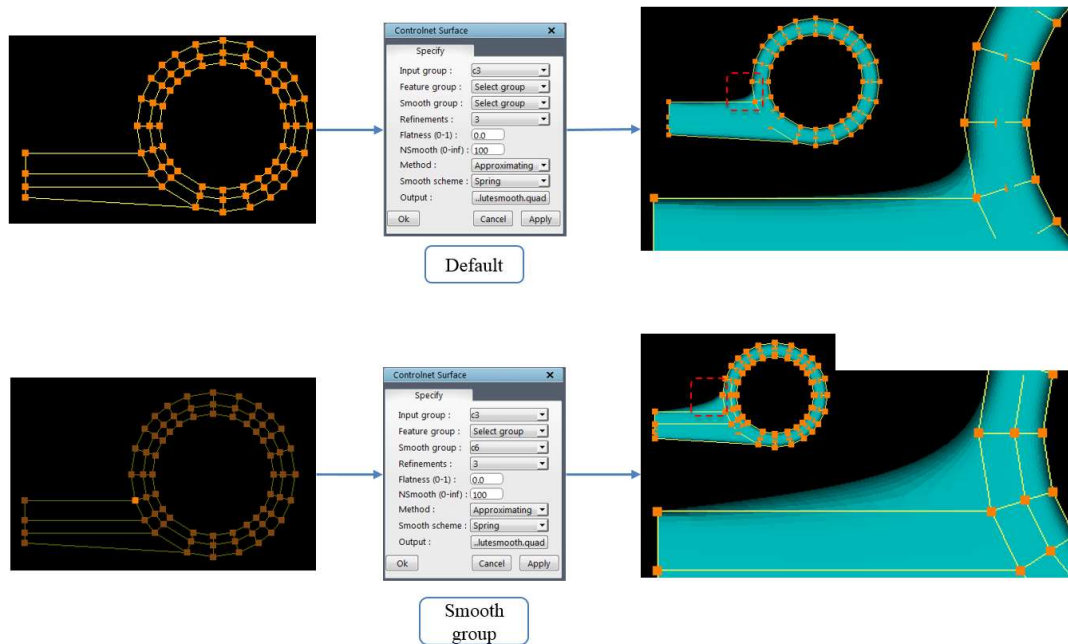


Figure 4.197: Controlnet surface using smooth group

- **Different Algorithms**

Three different algorithms can be used to create the surface in various shapes, such as Approximating, Interpolating and Sphere-Like.

Approximating → Generates a surface following the general shape of the wireframe

Interpolating → Generates an interpolating surface through the given wireframe with tight curvature

Sphere-Like → Generates an interpolating surface through the given wireframe with gentler curvature.

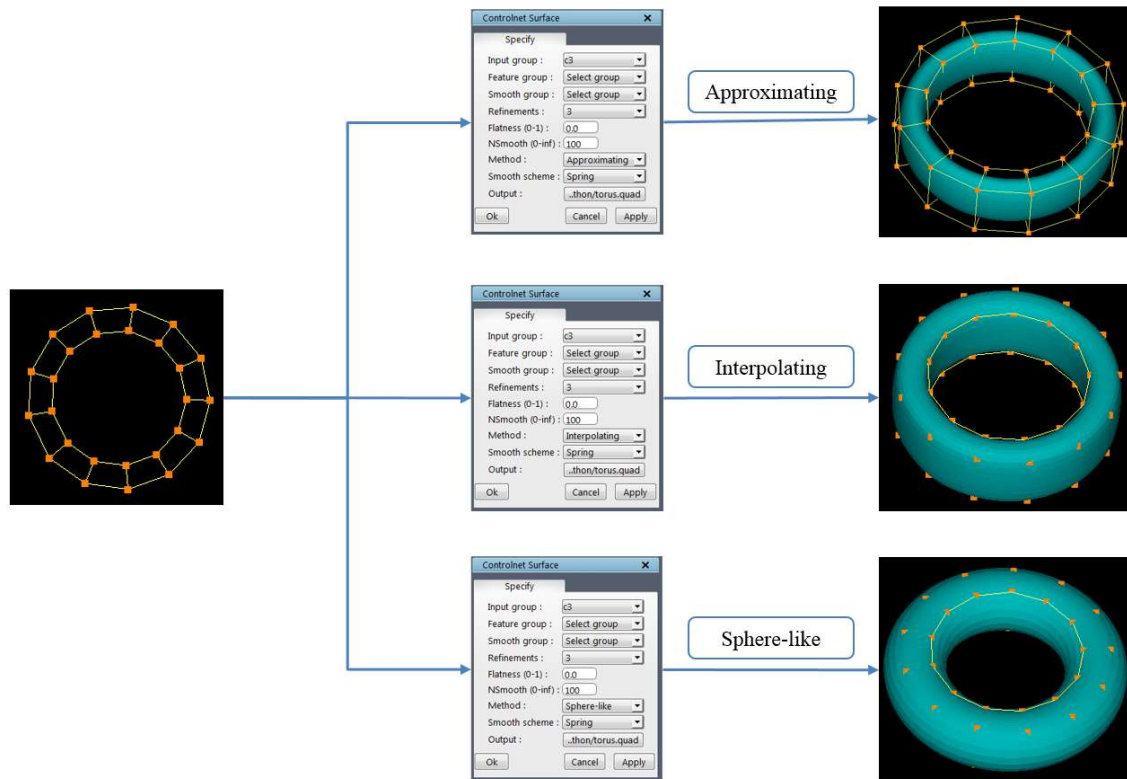


Figure 4.198: Controlnet surface using different algorithms

- **Refinement**

Refinement level ensures the smooth capture of the surface in the curvatures

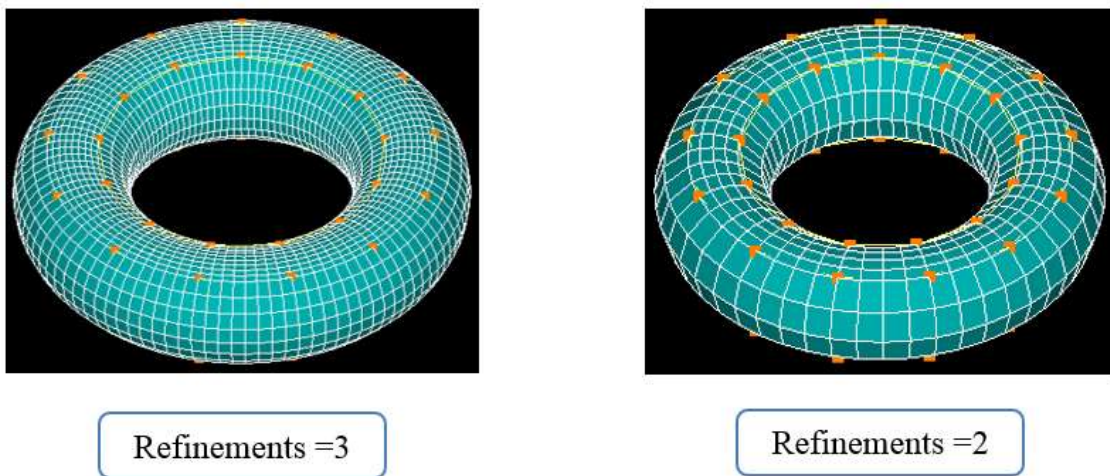


Figure 4.199: Controlnet surface using different refinement levels

- **Flatness**

The value range for flatness varies from 0 to 1, where 0 is curvy, and 1 is flat.

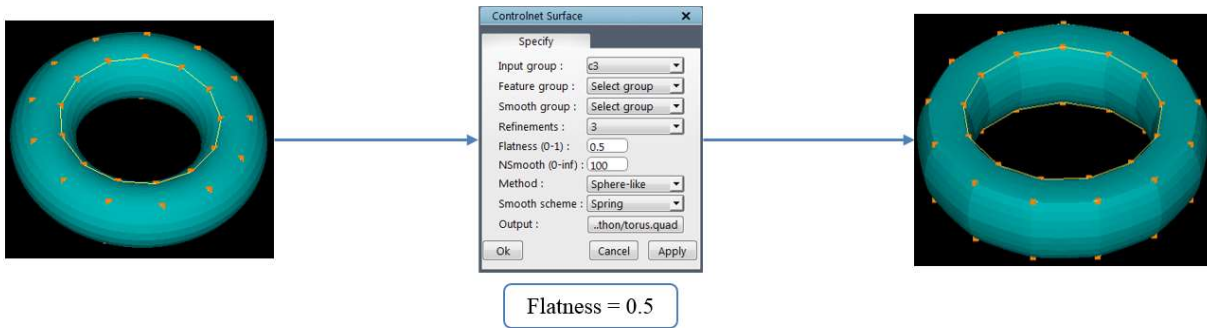


Figure 4.200: Controlnet surface using flatness

### 4.11.6 Controlnet Curves

To create a curve, click on the **Controlnet Curves** option in the Surface Tools menu dropdown list in the menu bar. Figure 4.201 shows the controlnet curves dialog box. Similar to controlnet surfaces, the controlnet curves are created based upon the input topology corner group. Select the corner group next to the **corner group** field for which you want to create curves. The number of levels of refinements can be defined in the **Refinements** field. The greater the number of refinements, the greater the number of points in the output curve and hence smoother the curve. If the number of refinements is zero, the output curve is just a copy of the corners provided. Two different schemes are provided for the generation of these curves, which can be selected from the dropdown provided next to '**methods**'. If the **Interpolating** scheme is used, the final curve passes through all the input corners of the topology. The **Approximating** scheme results in a curve which may not pass through the input corners. The input corners are used mainly to guide the curve. Note that the curve will always pass through the endpoints irrespective of the scheme used. The corners in the input corner group are automatically assigned to the output curve. The name of the output curve can be provided in the '**prefix**' field.

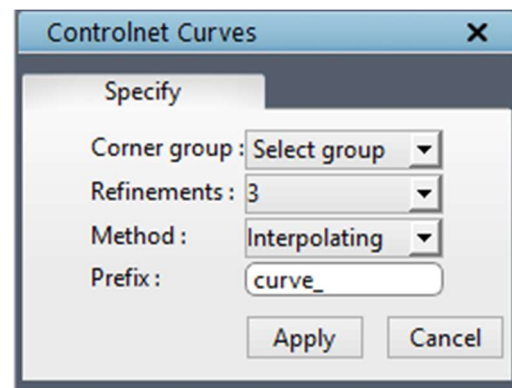


Figure 4.201: Controlnet Curves dialog box

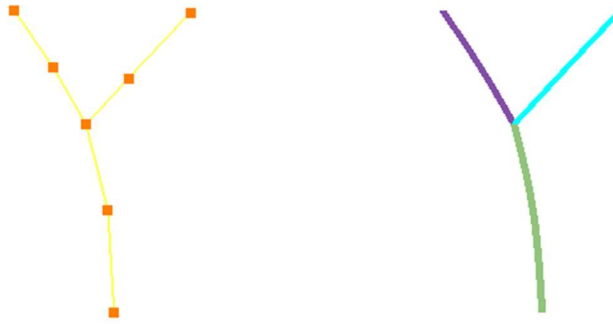


Figure 4.202: Branching wireframe and its corresponding controlnet curves

When there is branching (three different edges emerging from a given corner), it is not possible to output a single linear surface, and hence the input corner group is split into different non-branching groups, and a curve is generated for each of these non-branching groups. All the curves are generated using the above-mentioned parameters (the number of refinements and scheme). All these curves are saved as *prefix1.lin*, *prefix2.lin*, based upon the prefix you give. Figure 4.202 shows the branching input controlnet and output curve.

#### 4.11.7 Tube

A **tube** surface can be defined by the revolution around a centre curve. The revolution for each point on the centre curve is performed in the normal plane of that point with respect to the tangent of the curve. Hence, a **tube** surface is represented by a set of points on the curved centre line or axis and a variable/constant radii at all the cross-sections of the tube.

A **tube** surface can be created using the topology corners and edges. In order to create a tube, click on the Tube option in the dropdown list of the **Surface Tools** menu in the menu bar. Figure 4.203 shows the dialog box of the tube. The **axis** of the tube surface can be represented in two ways, either by a group of corners added to a corner group or by providing an origin and a direction. Similarly, the **radius** can also be represented in two ways, either by a corner group or by providing a constant radius value. Hence a tube can be created by four different input combinations such as group-group, group-value, orientation-group, orientation-value for axis and radius, respectively. When you are using group-group combination, please make sure both the groups have an equal number of corners.

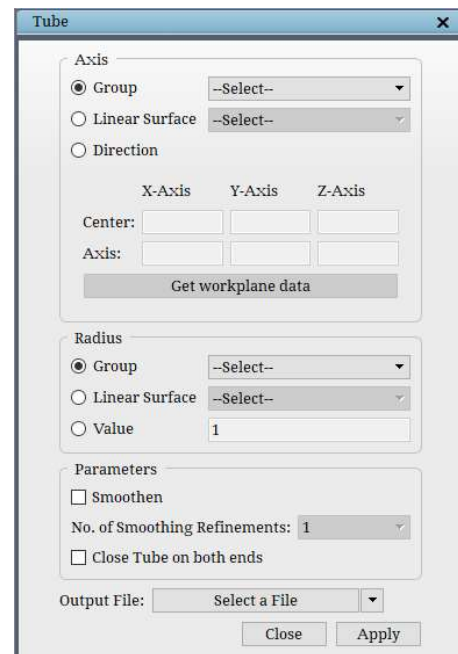


Figure 4.203: Tube dialog box



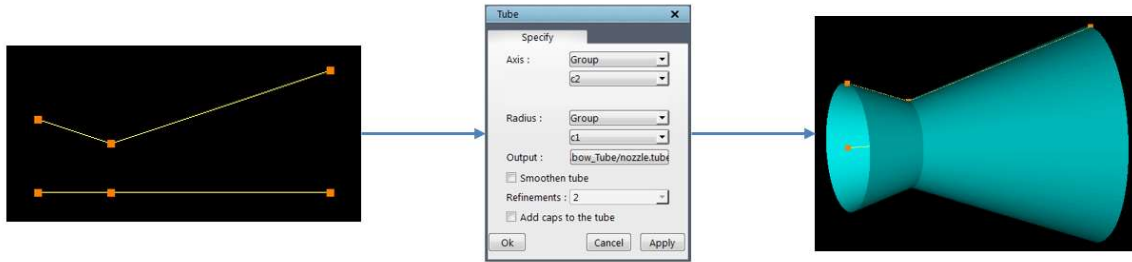


Figure 4.204: Tube surface with group-group combination

In group-value combination, the radius of the tube would be constant as you are providing a value.

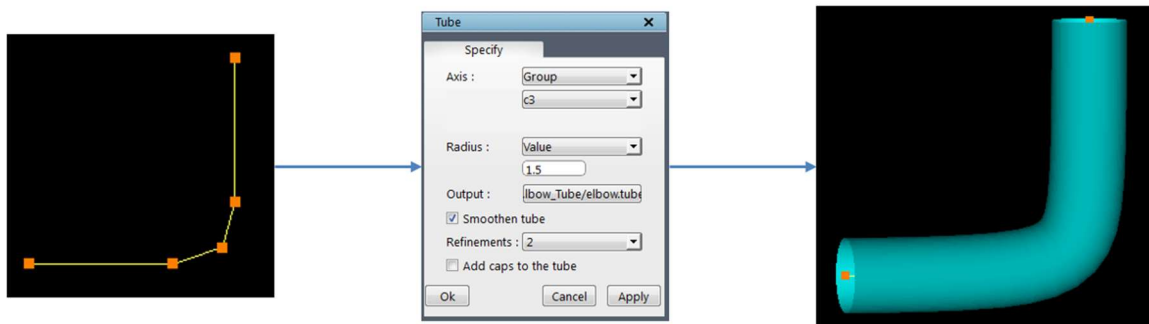


Figure 4.205: Tube surface with group-value combination

In orientation-group combination, if all the corners in the radius group are planar, then the axis should not be perpendicular to this plane.

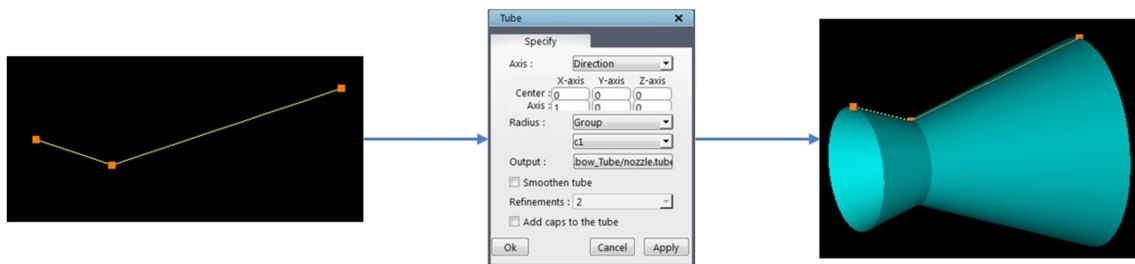
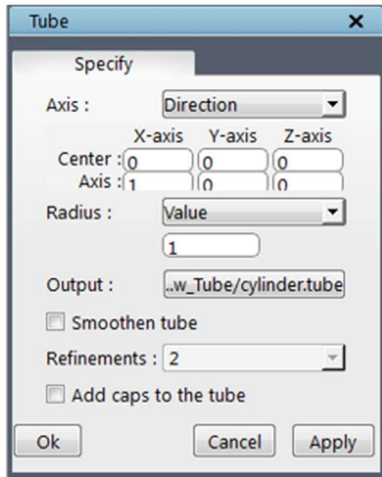


Figure 4.206: Tube surface with orientation-group combination

Orientation-value combination gives the same output as the cylinder option, so a cylinder is recommended for use in this combination.





Cylinder



Figure 4.207: Tube surface with an orientation-value combination

The file name for the resultant tube surface can be provided using the push button provided below the axis and radius fields. A file selection dialog box is pops-up to input the file name to save the output surface. Give the desired name for the tube surface file with which you want to save.

The *tube* surface can be smoothed using the approximation scheme, which can be enabled by checking the checkbox next to the **smoothen tube**. If the smoothen tube is enabled, then the number of levels of refinements need to be selected from the dropdown list provided next to the **refinements**. This dropdown is disabled until smoothen tube is checked. Sometimes it is desirable to have the tube surface closed on both the ends; this can be achieved by checking the checkbox, '**add caps**'. It will add caps to both sides.

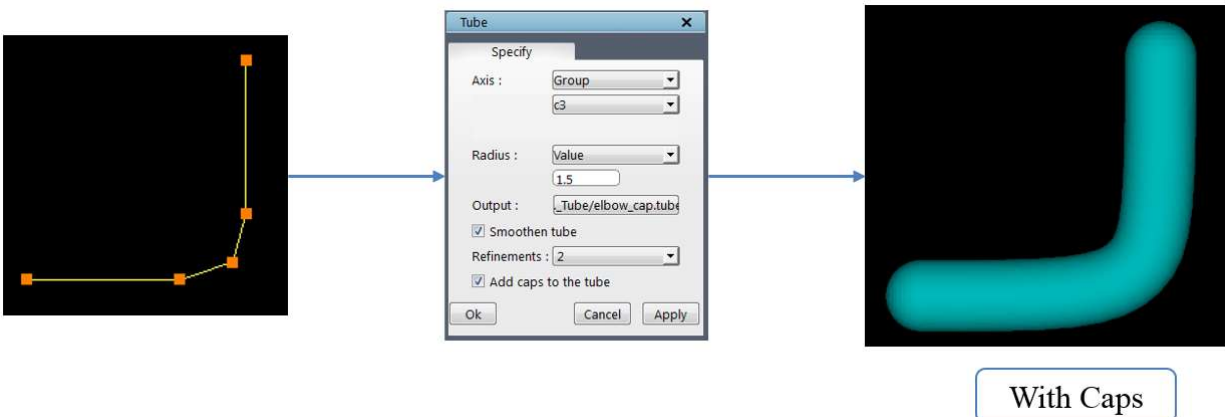


Figure 4.208: Tube surface with caps


Note that the radius of the last point on the cap of the tube surface should be zero. But a tube surface, by definition, cannot have zero radii at any cross-section. Hence, a very small

radius of  $1e-5$  is used at the end of the cap resulting in a small hole. However, this hole will not affect the quality of the grid.

From v7.0, you can also input a linear curve for both axis and group. The option has to be input in terms of surface ID, so the linear curve has to be loaded into the UI first. It also saves the previously created tube file names under the dropdown next to output file name option.

### 4.11.8 Internal surface

*The internal surface* is another tool which creates a quad surface at the intersection of surfaces. Figure 4.209 shows the dialog box of the Internal surface. You can choose to create it based on the surface or corner groups. By default, corner group option is selected.

If Surface group is enabled, then the desired group should be selected from the dropdown; it runs the intersection of surfaces tool internally and gets the list of corners at each intersection. It will be listed under the 'Accepted' section of the dialog box. Clicking on each item will highlight the corresponding intersection group in the UI. If you do not want to create any particular intersection, the same can be moved to the 'Rejected' section of the dialog box using the arrow buttons. You can also clear the lists by clicking on the delete  button. If multiple surfaces are created in a single execution, the output surface name becomes the prefix for all the surfaces.

To create the internal surface using corner group option, the *Intersection* tool should be used first to obtain the corners at the intersection. Unlike surface group, corner group option can only execute one group at a time.

The surface would be created using the normal at the intersection; so if some corners normal has to be changed, then it has to be grouped separately, and it must be input it in the *Flip Normals group* option. There are two kinds of algorithm that can be used to create a surface, such as *Algebraic* and *Variational*, which can be selected under *Type* dropdown list. These three (Input corner group, Flip normal group and Algorithm) are the basic options that a user can be easily used to create a surface. There are also advanced options available which can be enabled by clicking on the *Advanced* options checkbox. The first option is the method, which is

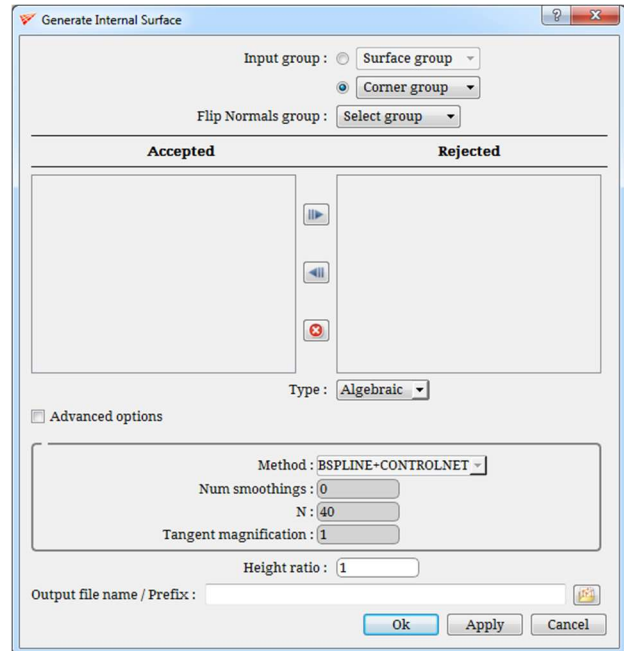


Figure 4.209: Internal surface dialog box

a separate algorithm for creating the surface which will be applied after the algebraic/variational type. A list of 6 different methods are available under the **method** dropdown list. Next, is the **number of smoothings** to be applied on the surface created. Next is the  $N$ , which is the number of points in the other direction. That is the direction opposite to the surface created. This ensures that the surface passes through the intersection. Fourth one is the **tangent magnification** which helps in controlling the height till which the orthogonality has to be maintained. The last option under the advanced option is the **height ratio** which controls the maximum height of the surface.

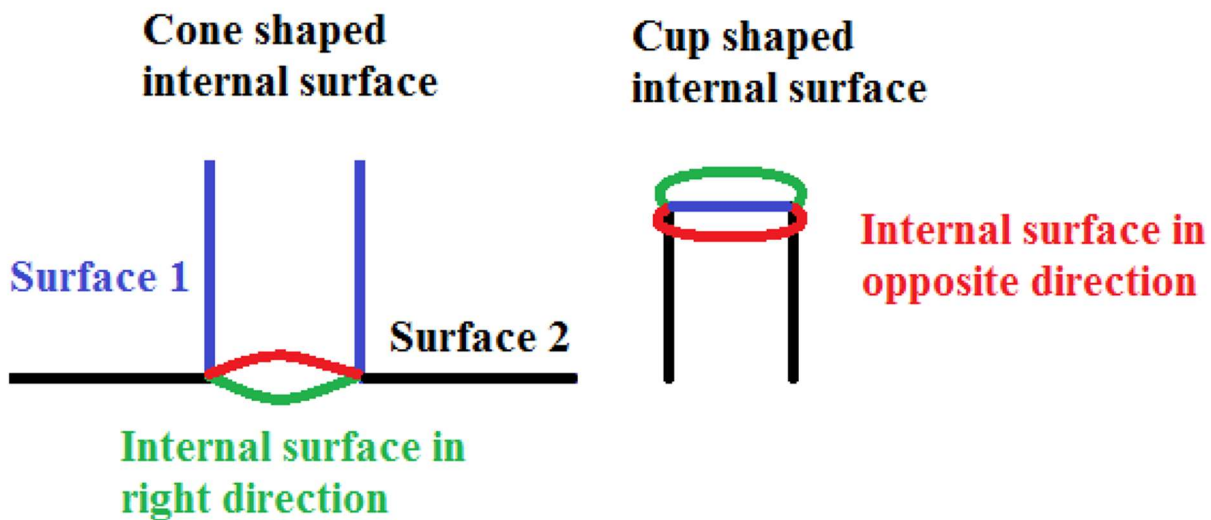


Figure 4.210: Four different ways on internal surface creation

At any given intersection, four different internal surfaces can be created. The default inputs may not create the desired surface always. The four possible internal surfaces are cone and cup-shaped surface on either side of the intersection, as shown in Figure 4.210.

*Scenario 1: Surface shape is right, but direction is wrong*

In such cases, input the surface group under ‘Inverted surface group’ option, which has both the intersection surfaces are grouped into.

*Scenario 2: Both Surface shape and direction is wrong*

In such cases, input a surface group with either one of the intersection surfaces. It may create a right surface in the right direction. Sometimes it will create a right surface shape, but the direction may not be right; during such scenario, input the surface group which has the other intersection surface.

## 4.12 Gridding Process

*GridPro* is well known for its revolutionary ability to generate grids from a topological basis, one that is not encumbered by detailed geometric demands, decisions or judgements. *GridPro* requires no manual construction of geometric pieces such as block edges or faces, or their associated pointwise distributions. All gridding is done in one automated process; there is no separate process for surface grid generation or surface to surface intersections. Even within the context of a wireframe, there is no projection algorithm required to send wireframe edges or faces onto the geometric boundary of a physical region. Such projection algorithms require tight constraints upon the placement of wireframe corners - the antithesis of true topology, which relies upon a much looser and more flexible structure.

Grid generation takes place when the user starts ***Ggrid***, a transformation process, to the topology code file (with extension \*.fra). The resulting grid is optimized to be smooth and nearly orthogonal throughout an entire region and to be properly clustered to a region's boundary curvature. The optimization is, of course, relative to the constraint of topology choice (usually a rather mild constraint). Generally, you will get smooth, nearly orthogonal grids that are appropriately clustered to both concave and convex boundary curvature. The resulting grid is so natural, you will become accustomed to high-quality grids every time you use *GridPro*. With the interactive GUI, the entire gridding process is purely visual. The ***Ggrid*** launch can be started, stopped, and re-launched graphically, which is explained in *section 2.3.24*.

The ***Ggrid*** process is tightly coupled with interactive topology generation; topology errors are automatically detected and highlighted in red when you try to launch an incorrect TIL code. The highlight appears as an overlay on top of the corners and edges with different colours based on the error. The error and highlight display point precisely to the problem area that prevents ***Ggrid*** from running. Concurrently, a more technical error message is also displayed in the console window. Once you know the location of an error, the error and highlight display that helped you find the error will sometimes be in the way of your corrective actions. To remove the highlight, uncheck ***error*** under ***Topology*** menu in ***Toggle display*** section. If the topology is valid, a confirmation dialog box pops up. When you click ***ok***, the ***Ggrid*** process will commence. However, you will not explicitly see the gridding process itself. Instead, *GridPro* will display the progress in the console window that will dynamically give information about your run.

The numbers displayed in the console window convey information about the quality and convergence of the grid being processed. The first three letters are shown; ***swp*** form an abbreviation for *sweep*. The number proceeding *sweep* illustrates the number of times *GridPro* has visited your topology in its attempt to optimize the grid in the corresponding region. Every subsequent sweep relies on information processed in previous sweeps in order to improve the quality of the grid. In other words, the more sweeps you allow *GridPro* to perform, the better

your resulting grid. In order to numerically measure convergence for the grid optimization process, look at the residual number found in the status window. This number follows the *r* expression found in every sweep and indicates the disparity between the actual grid and an ideal grid. An ideal one would require a limitation on generality and an infinite number of points because this ideal is unobtainable, *GridPro* compromises by creating the best possible grid for the chosen topology, a grid that is consistent with the number of points and specified parameters of the topology. As the grid generation process converges, the grid's *residual number* becomes *constant*. As the number of sweeps increase, the residual number will often oscillate and then converge to a constant. Often, however, the residual does not completely converge. Still, the resulting grid quality is far better than what you would expect from traditional methods.

Sometimes, if the gridding process is stopped too quickly, it will produce a grid whose coordinate sheets fold in on themselves at some location or locations. Therefore, you must allow *GridPro* to sort out all surfaces. In the status window, a stream of consecutive sweep messages is periodically interrupted by the reading, *fold count*. When the number following fold count reaches zero, all surfaces have been sorted out. The fold count gives the user a coarse means to judge the convergence of the process. Generally, in the convergence, non-zero fold counts appear only in the early stages, well before it is worthwhile to inspect the residual; the residual becomes the main focus of your monitoring once you are at zero folds. Your attention is then redirected to seeing when the residual settles down to a constant value. While the grid is converging, you have the option to take the grid for use at that intermediate stage or wait until a more full convergence of the optimizer has taken place.

The gridding process is controlled by four **Ggrid** operations found in the **global dock**, which is already discussed in *section 2.3.24*.

During the gridding process, after every 100 sweeps, *GridPro* saves the grid to the file, *blk.tmp*. The **Ggrid** process really represents the running of a schedule file, with extension '.sch'. Thus, when transferring topology data from one terminal to another, be sure to include the schedule file with the '.fra' (topology) file. If there is no '.sch' file, *GridPro* automatically creates one with the default value. A sample schedule file is shown in Figure 4.212. The schedule can be modified in the GUI itself by modifying the parameters in the grid schedule pop-up, which is explained in *section 2.3.24*.

Like topologies, grids can be saved under user-defined file names. Because *GridPro* saves grids to the default file, *blk.tmp*, you must alter the name *blk.tmp* when you retrieve your grid if you wish to save it under a username. To assign a user name to a grid, access the **save grid as** an option in the grid tab (explained in *section 2.3.3*). You can arbitrarily choose the extension name for any grid file. By altering the file name, you will be able to retrieve your file whenever you wish because all other grids produced will be saved under *blk.tmp*. Each grid file is accompanied by a connectivity file. This connectivity file specifies how the grid blocks are

connected and has the extension '.conn'. GridPro automatically runs the connectivity file when you access *blk.tmp*.

```
>Scheduling at step 1(swp 0) with 3 actions..
+action: (-g all 1.000000x) markCnt=1. mark e (eid den eCnt):
  (0 8Xd 4) (1 8Xd 4)
  total points needed: points=192x96+500x8 = 0 MB
  allocated : b0/1 points=192x96+500x8 = 0 MB
  no change.
+action: (-S 100) new sweep interval = 100
swp 0 0(6{0},0) max(r0.139,c 0(0.00),s7.76e+029(0.0019),a1.5),act(B100,U100,F25)
  phase 0 1 2 3 4 5 -> 1.
swp 1 1(6{0},0) max(r0.0931,c 0(0.00),s1.14(0.0035),a1.5),act(B100,U100,F25)
  phase 0 1 2 3 4 5 -> 2.
swp 2 max(r0.0517,c 0(0.00),s1.06(0.0032),a1.5),act(B100,U100,F25)
swp 3 max(r0.0436,c 0(0.00),s1.05(0.0030),a1.5),act(B100,U100,F25)
swp 4 max(r0.0193,c 0(0.00),s1.06(0.0028),a1.5),act(B100,U100,F25)
swp 5 max(r0.0164,c 0(0.00),s1.05(0.0027),a1.5),act(B100,U100,F25)
set static ctrl funcs... (0.4) (0)
swp 6 max(r0.108,c 0(0.00),s1.04(0.0026),a1.5),act(B100,U100,F25)
swp 7 max(r0.0608,c 0(0.00),s1.03(0.0025),a1.5),act(B100,U100,F25)
swp 8 max(r0.0213,c 0(0.00),s1.12(0.0024),a1.5),act(B100,U100,F25)
swp 9 max(r0.0181,c 0(0.00),s1.16(0.0023),a1.5),act(B100,U100,F25)
swp 10 max(r0.0143,c 0(0.00),s1.18(0.0023),a1.6),act(B100,U100,F25)
set static ctrl funcs... (0.4) (0)
swp 11 max(r0.0103,c 0(0.00),s1.19(0.0022),a1.6),act(B100,U100,F25)
swp 12 max(r0.00862,c 0(0.00),s1.21(0.0022),a1.6),act(B100,U100,F25)
swp 13 max(r0.00625,c 0(0.00),s1.21(0.0021),a1.6),act(B100,U100,F25)
swp 14 max(r0.00490,c 0(0.00),s1.22(0.0021),a1.6),act(B100,U100,F25)
swp 15 max(r0.00417,c 0(0.00),s1.22(0.0021),a1.6),act(B100,U100,F25)
set static ctrl funcs... (0.4) (0)
swp 16 max(r0.00375,c 0(0.00),s1.22(0.0020),a1.6),act(B100,U100,F25)
swp 17 max(r0.00281,c 0(0.00),s1.22(0.0020),a1.6),act(B100,U100,F25)
swp 18 max(r0.00252,c 0(0.00),s1.22(0.0020),a1.6),act(B100,U100,F25)
swp 19 max(r0.00200,c 0(0.00),s1.22(0.0019),a1.6),act(B100,U100,F25)
  fold count: 0
swp 20 max(r0.00169,c 0(0.00),s1.23(0.0019),a1.6),act(B100,U100,F25)
set static ctrl funcs... (0.4) (0)
swp 21 max(r0.00153,c 0(0.00),s1.23(0.0019),a1.6),act(B100,U100,F25)
swp 22 max(r0.00143,c 0(0.00),s1.23(0.0019),a1.6),act(B100,U100,F25)
swp 23 max(r0.00114,c 0(0.00),s1.23(0.0019),a1.6),act(B100,U100,F25)
swp 24 max(r0.00107,c 0(0.00),s1.23(0.0018),a1.6),act(B100,U100,F25)
swp 25 max(r0.000956,c 0(0.00),s1.23(0.0018),a1.6),act(B100,U100,F25)
set static ctrl funcs... (0.4) (0)
swp 26 max(r0.000917,c 0(0.00),s1.23(0.0018),a1.6),act(B100,U100,F25)
swp 27 max(r0.000856,c 0(0.00),s1.23(0.0018),a1.6),act(B100,U100,F25)
swp 28 max(r0.000770,c 0(0.00),s1.23(0.0018),a1.6),act(B100,U100,F25)
swp 29 max(r0.000646,c 0(0.00),s1.23(0.0018),a1.6),act(B100,U100,F25)
swp 30 max(r0.000592,c 0(0.00),s1.23(0.0017),a1.6),act(B100,U100,F25)
set static ctrl funcs... (0.4) (0)
swp 31 max(r0.000548,c 0(0.00),s1.23(0.0017),a1.6),act(B100,U100,F25)
swp 32 max(r0.000533,c 0(0.00),s1.23(0.0017),a1.6),act(B100,U100,F25)
```

Figure 4.211: Gridding process

```
#This is a sample schedule file. You may edit it.
step 1: -g all 1.000000x
step 10000: -c all 1.0 0 -C all 1.0 24 -r -S 100 -w
write -f blk.tmp
write -D 0 -f dump.tmp
```

Figure 4.212: Sample schedule file

### 4.12.1 Ggrid Groups

From version 6.6, ggrid can be run on part of topology. i.e. on any given input group. The given group should have its boundary faces assigned to any internal or external surface. Only if all the boundary faces are assigned to a surface ggrid will run else it will pop\_out invalid messages similar to Validity button. This is helpful when you work on complex geometry, and you would like to position the corners in a better way in some regions of the geometry. You can run ggrid on part corner group and align the corners to the grid using Align→Group2Grid. Refer section 2.3.24 for more information on ggrid groups.



## CHAPTER 5

# GRID

Grid visualization is carried out in the **Grid** tab. Once the Ggrid is run and the grid is generated, you would like to check the grid. In the **Grid** tab, you will find grid viewing utilities, preprocessing utilities, post-processing utilities, grid probing utilities, grid quality check utilities and tools to toggle grid and grid sheets. All these utilities are placed under two sections, namely the *Grid toggling* and *Tools* section at the right-side panel of the **Grid** tab.

Once the grid is generated, you can load a *blk.tmp* (default name saved by schedule file) file from the working directory. Any number of grids can be loaded in the **Grid** tab at any time. You can work independently on each grid. The GUI allows you to use the tools on different grids without affecting other grids in the UI.

A grid can be split up, sliced and represented in many different ways. The block traces composing a grid contain edge outlines, a centre of mass, and skeletons (links from the centre of mass to the centre of each face). These various features can be hidden or activated in various ways to display different parts of a grid, located in the **Grid** tab.

## 5.1 Grid Viewing Utilities

Because meshes are used extensively for analysis testing, *GridPro* allows you to visualize them differently. First and foremost, to understand meshes, you must understand the difference between blocks, faces, slices, sheets, and shells. When the ‘Ggrid’ process creates a grid, *GridPro* carves up the region inside each *block* into IJK coordinates (index coordinates) to create a smooth grid. The IJK coordinates of each point are integers; each node thus lies on a lattice point. The maximum and minimum values of these IJK coordinates appear on the boundary of the block. Note that IJK coordinates differ for each block (within one grid).

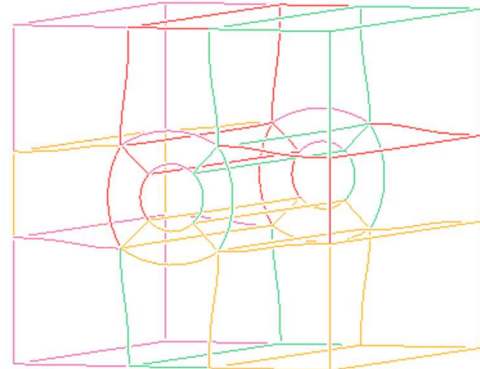


Figure 5.1: Grid blocks

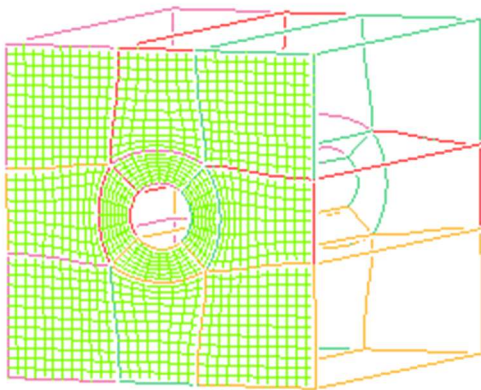


Figure 5.2: Grid blocks with one face highlighted

A *face* is a grid about a boundary of a block. More technically, a face is a network of links along with indexed IJK lattice points such that I, J, or K is a maximum or minimum.

A *slice* is a grid that cuts through a block. More technically, a slice is a network of links along with indexed IJK lattice points such that I, J, or K is any constant integral value between a certain maximum and minimum. A *sheet* is simply a collection of linked faces or a collection of linked slices.

A *sheet* composed of linked faces is called a *face sheet*, and a sheet composed of linked slices is called a *slice sheet*.

A *face sheet* composed of the faces of the outer boundaries of all outer, activated blocks is called a *shell* (which is a sheet). A face sheet composed of the faces of the outer boundaries of some outer, activated blocks is called a *partial shell* (which is also a sheet). Notice that sheets can take on many shapes.

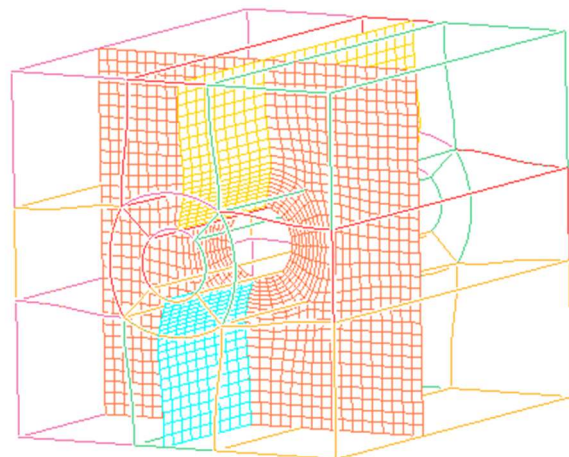


Figure 5.3: Grid blocks with a sliced sheet

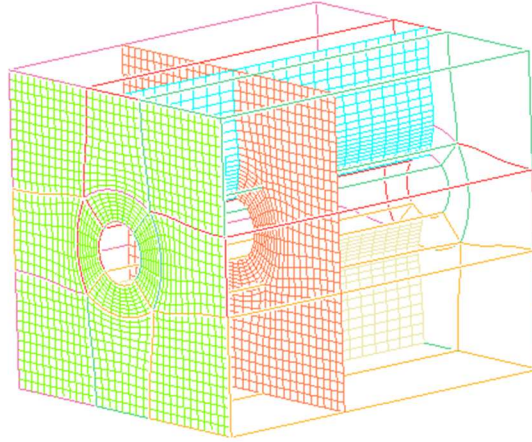



Figure 5.4: Grid block with face sheet and slice sheets

Let us summarize: Distinct blocks have distinct faces and distinct slices. Many linked faces or linked slices comprise a sheet (either a face sheet or a slice sheet). A face sheet can be either a shell or a partial shell. Manipulating blocks, faces, slices, and sheets will help you look inside and analyze parts of grids. For complex grids, you will want to check numerous cross-sections to observe specific features such as the density of the mesh lines or the number of singularities in a sub-region.

### 5.1.1 Mesh

To view the overall grid or mesh, click on  **Mesh** button located in the **Display** section in the right-side panel. All grid points, which are displayed when the **Mesh** button is depressed, define ‘mini blocks’ or ‘cells’. Corners are to blocks as grid points are to cells.

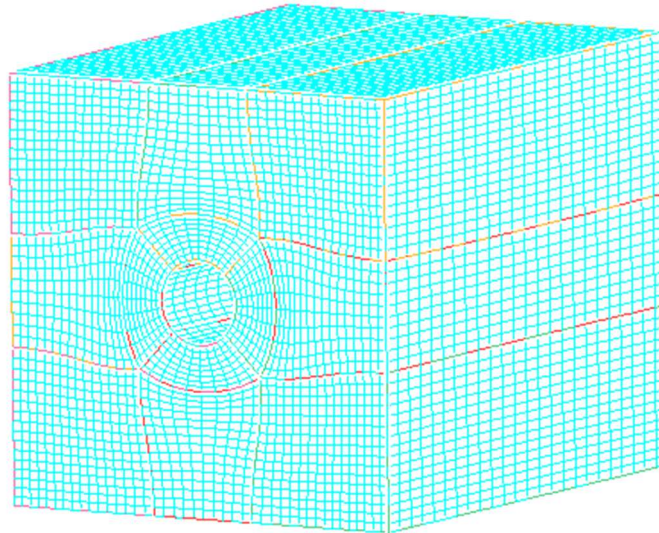



Figure 5.5: Grid blocks with mesh

## 5.1.2 Faces

**Faces**  button is used to create a sheet consisting of all block faces on all the external grid surfaces. It creates grid sheets for all surfaces, displayed in different colours. Clicking on the Faces button opens a dialog box in the pop-up section, as shown in Figure 5.6. To create grid sheets on external surface/boundary surfaces, use the **Boundary** button in the **Sheets** tab of the **Display Faces** pop-up, similarly for internal surfaces, use the **Internal** button. By default, the **Boundary** button is depressed.

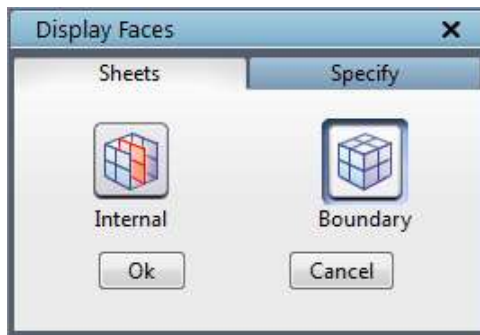


Figure 5.6: Faces dialog box

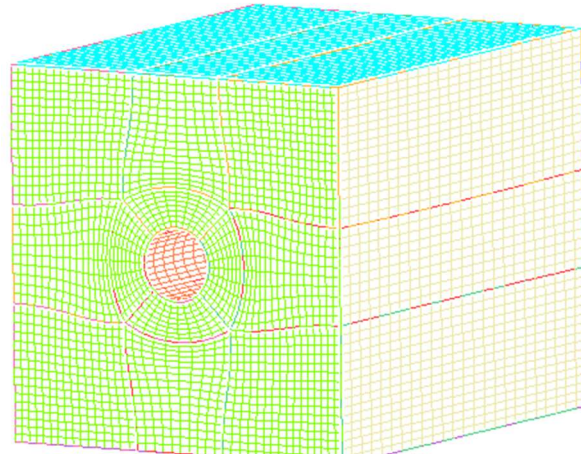


Figure 5.7: Grid blocks with boundary grid sheets

You can also create a grid sheet based on surface ids. Switch to **Specify** tab and enter the desired surface ID to create its corresponding grid sheet. By default, it displays the last surface id for the **Current** grid.

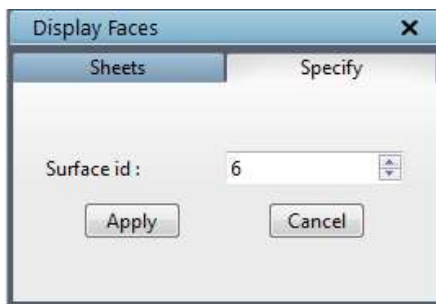


Figure 5.8: Faces dialog box for specify tab

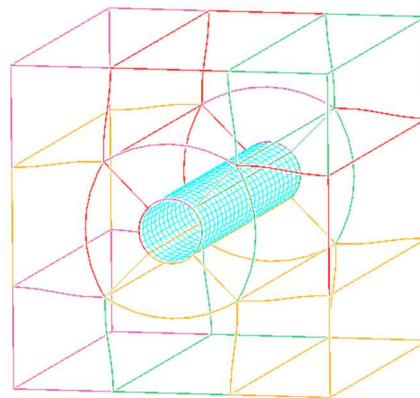



Figure 5.9: Grid blocks with specific surface grid sheet



### 5.1.3 Slice

Aside from creating sheets about surfaces, you can also create sheets at any desired IJK setting. By pressing the **Slice**  button, then clicking any point on an edge, you can create a sheet composed of slices. The sheet will appear probably close to where you click on the edge, at the closest IJK constant index. After creating a sheet, you can use the button's operation in the **grid sheets** section on the right-hand side to step forward or backwards on a sheet. The stepping process of the grid sheets is explained in *section 5.6*.

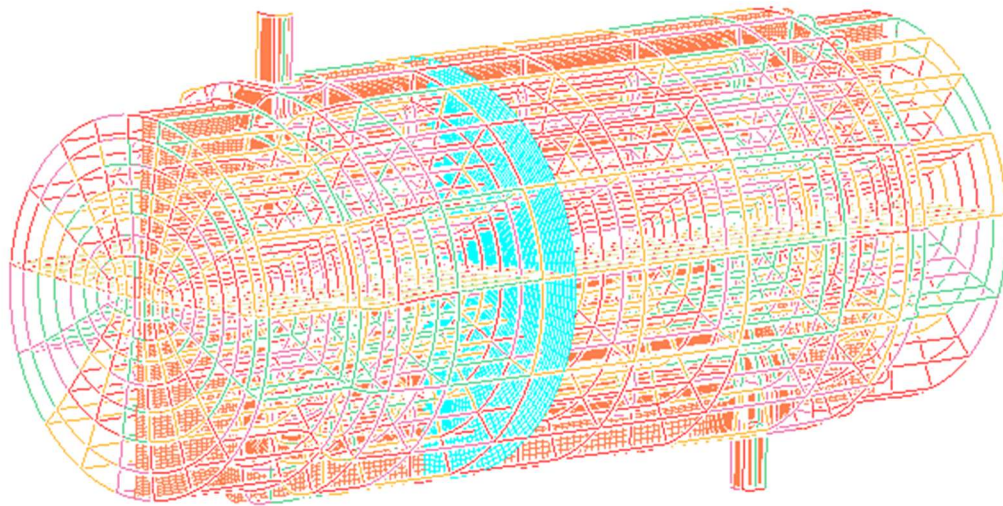


Figure 5.10: Cross-sectional grid sheets at arbitrary IJK

In the case of staggered periodic grids, the default cross-sectional grid sheet will be created only for that one block propagation. It is difficult to visualize how the cross-sectional sheet is propagating throughout the grid in such a scenario. To view all the staggered cross-sectional sheets of the grid, select the staggered sheets icon in the dialog box as shown in Figure 5.11 and click on the desired region, which will create cross-sectional grid sheets as shown in Figure 5.12.

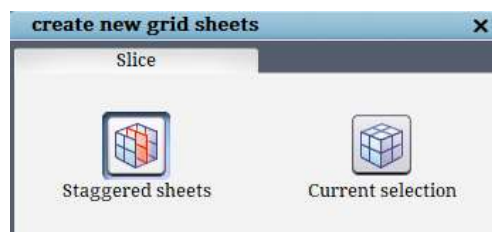


Figure 5.11: Slice dialog box

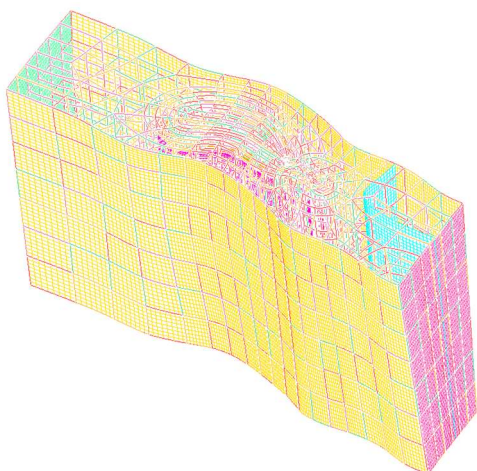


Figure 5.12: Grid with default cross sectional sheet

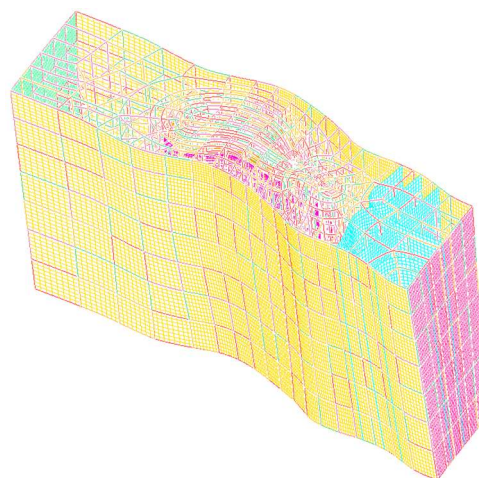


Figure 5.13: Grid with staggered cross sectional sheets

From v8.1, a new feature is implemented to make the workflow easier. It is always cumbersome to create a desired cross-sectional grid when you have many blocks in the grid. Creating a cross-sectional grid sheet often becomes a trial and error because we end up selecting the wrong block edge. This happens because it is very difficult to identify the desired block edge when there are ‘n’ number of edges crossing over in front of your eyes. A new feature that highlights only the block edges of the current grid sheet is

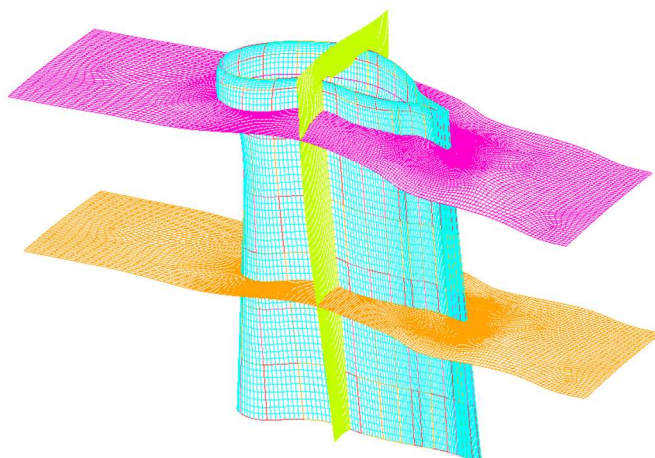


Figure 5.14: Grid sheet with blocks highlighted for current sheet

implemented to resolve this issue. When you click on the slice button, a GL widget with two buttons pop up on the screen. One is the default way of representing the blocks for creating the cross-sectional grid sheet, and the other one displays the block structure only on the current grid sheet. You can select the desired grid sheet through which the cross-sectional grid sheet has to be created and click on the location on the highlighted block structure to create the cross-sectional grid sheet, as shown in Figure 5.14. This mode is persistent until you change it manually.

In general, when you create a cross-sectional grid sheet, the newly created grid sheet is made as a current grid sheet, but in this mode, the selected grid sheet stays current to display the block structure. Also, note that the block structure will only be displayed for the surface grid sheets and not for any of the cross-sectional grid sheets which are created manually.



- **Right-click menu**

When you *right-click* on the edge of a grid block, a menu appears, as shown in Figure 5.15. The first option is

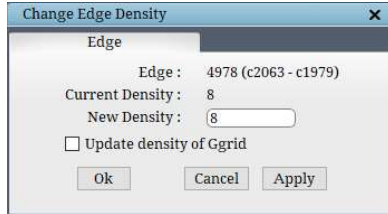


Figure 5.16: Change Edge Density dialog box

**centre of rotation**; you can pick a particular point of that edge of the grid block as the centre of rotation by selecting the **centre of rotation** option after right-clicking. Like the **slice** button, you can add a grid sheet at that particular point on the grid edge by selecting the **add grid sheet** option. The

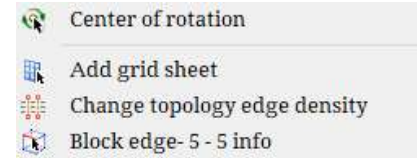


Figure 5.15: Right click menu of grid blocks

next option is **change topology edge density** which switches on the Topology in the Grid panel and opens the ‘Change Edge Density’ dialog box as shown in Figure 5.16 to change the density of the selected block edges. The last option shows the **grid block edge information**. It will show block index number, edge index number, along with edge density, as shown in Figure 5.17.

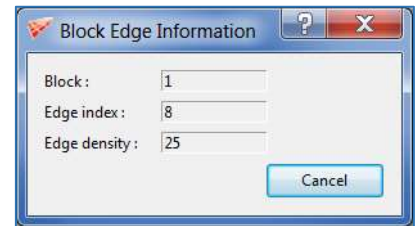


Figure 5.17: Grid block edge information

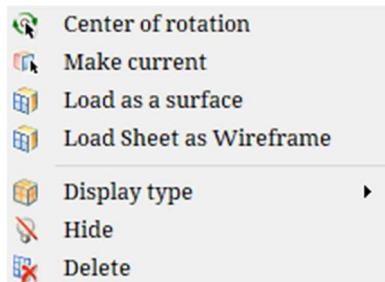


Figure 5.18: Right click menu of grid sheets


You can also *right-click* on a grid sheet that shows the menu, as in Figure 5.18. Similar to picking the centre of rotation on a surface, wireframe, and grid blocks, you can also select it on the grid sheet by selecting the **Centre of rotation** option from the right-click menu on the grid sheet. You can also make any grid sheet as current by clicking on **Make current** option. In general, while converging the grid, we often use a grid sheet as an internal surface for faster convergence. **Load as a surface** option helps in saving the current sheet as Quad surface and

loads it back into the UI with a click of a button. **Load sheet as Wireframe** option helps in generating the topology faces of the selected grid sheet and loads it into the GUI. This topology faces primarily helps in building the wireframe for the Controlnet surface. The grid sheets can also be viewed using different display types by selecting the **Display type** from the right-click menu. The different display types can be seen in figure 2.68 in section 2.3.15. Similar to hide and delete a surface, you can hide/delete a grid sheet using the options **Hide** and **Delete**. Once it is hidden, it can be shown by checking the corresponding checkbox of the grid sheet in the **Show/Hide** section, which is explained in section 5.6.

## 5.1.4 Update

The **load** option in the global dock opens a window that allows you to find and access the file, *blk.tmp*. Note: you need not stop the gridding process to access the grid. In fact, since the



file, *blk.tmp*, is being updated after every 100 sweeps (by default), you can reload an updated version of *blk.tmp* by accessing the **Update**  button in the **Edit** section. Clicking on the **Update** button updates the grid in the GUI and the updated grid will be seen.

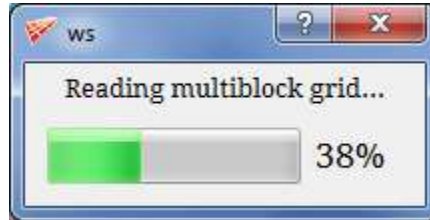



Figure 5.19: Loading/Updating grid pop-up

### 5.1.5 Colour

When you have one or more sheets on display, you can select a colour scheme that will fit your needs. You can colour each sheet with slices or faces or colour each sheet differently.

**Colour** button  pops -up a dialog box that contains the **colour by block**, **colour by sheet**, **colour by IJK** and **colour by quality** operations. The **colour by sheet** operation colours each displayed sheet differently (there is one colour on each sheet). The **colour by block** operation colours every block face in each sheet differently. Each face or slice is created when I, J, or K is a constant. **Colour by IJK** operation allows you to colour faces and slices depending on which coordinate is a constant. **Colour by Quality** allows you to view the sheets depending on the qualities such as Skewness, Aspect ratio and Warpage. By default, the **colour by sheet** option is selected as shown in Figure 5.20.

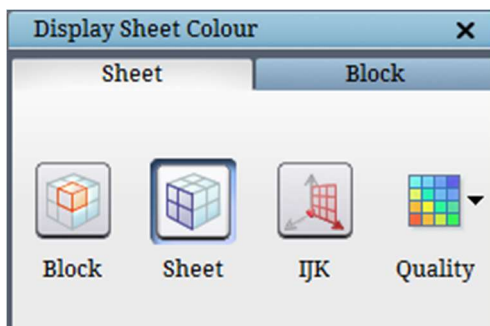


Figure 5.20: Colour dialog box with Sheet tab



Figure 5.21: Colour dialog box with Sheet tab

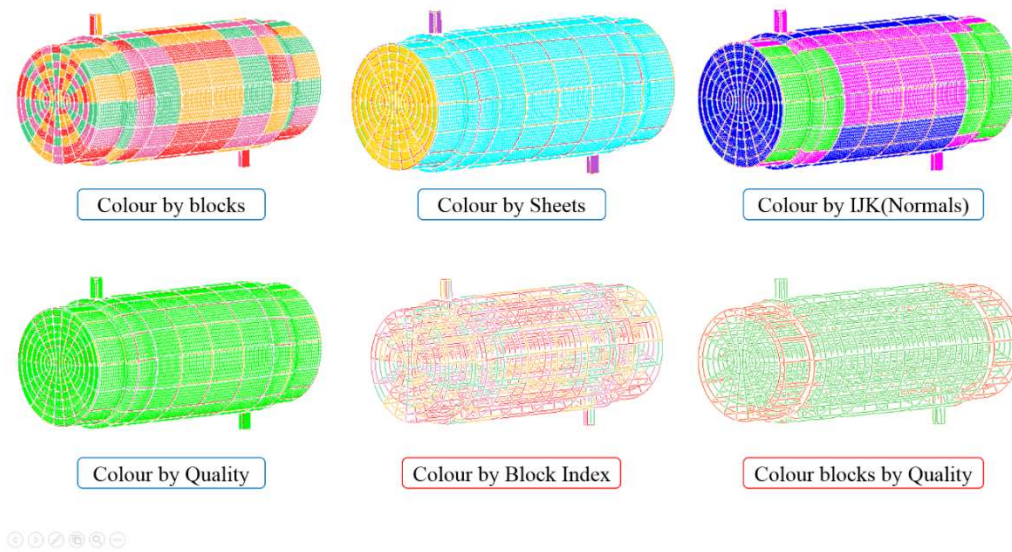



Figure 5.22: Different colour by sheets and blocks options

Similarly, you can also display the blocks with different colours either by **index** or **quality**. Selecting the **index** will display all the blocks with four different colours, which is the default option. If colour by **quality** is selected, all the good blocks would be displayed in green shade and the bad blocks in red. The reference values for the good quality would be assumed from the default values of the **Quality** button.

### 5.1.6 Preview

**Preview** button  helps in visualizing the grid on a group of faces without running the gridding process. It is located at the global dock next to the Error button. Switching it ON will display all the active faces as grid sheets based on the edge densities. Note that this is just a visualization of the block faces as a grid sheet, and no smoothing algorithm is applied. Hence, the grid visualization would be rigid, as shown in Figure 5.23 and not curvy or smooth. It also gives instant feedback by updating the visual if the density is modified on the active edges or repositioning any active corner.

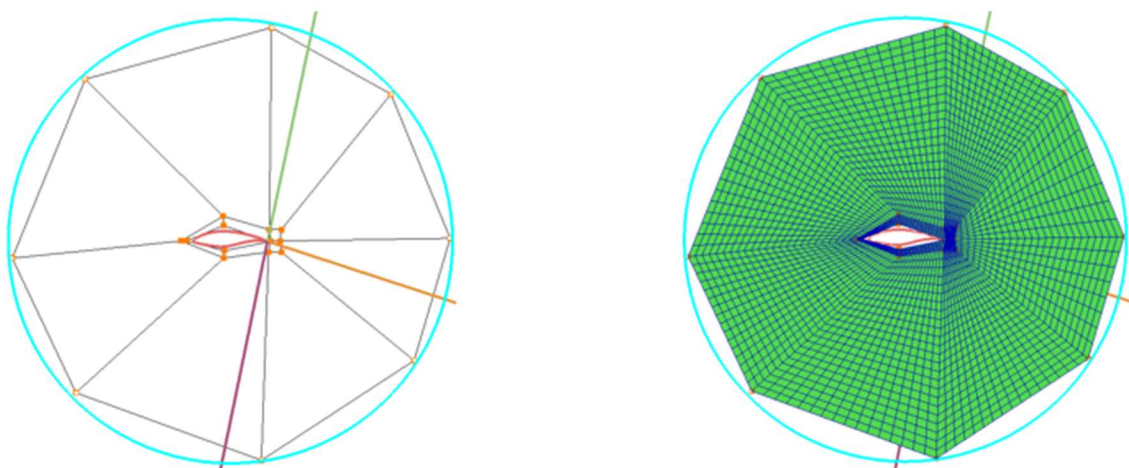


Figure 5.23: Topology and Grid Preview

## 5.2 Preprocessing Utilities

### 5.2.1 Extract

In addition to visualizing the grid, you can also generate the topology from the grid in the **Grid** tab. **Extract** button in the **Edit** section located in the right-side panel of the **Grid** tab is used when you have a structured grid without the corresponding TIL code and still want to use *GridPro* to relax it. **Extract** allows you to generate topology (wireframe + surfaces) from the available structured grid. The necessary condition is you must also have a grid connectivity file (\*.conn) along with the grid. Once the button is clicked, a confirmation dialog box pops up where you need to click on ‘**Ok**’ to confirm the execution of the process.

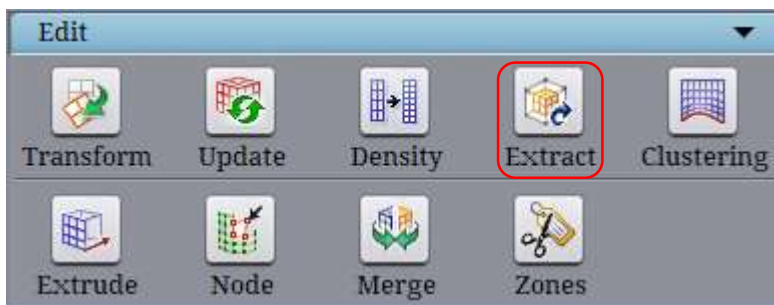


Figure 5.24: Edit Section

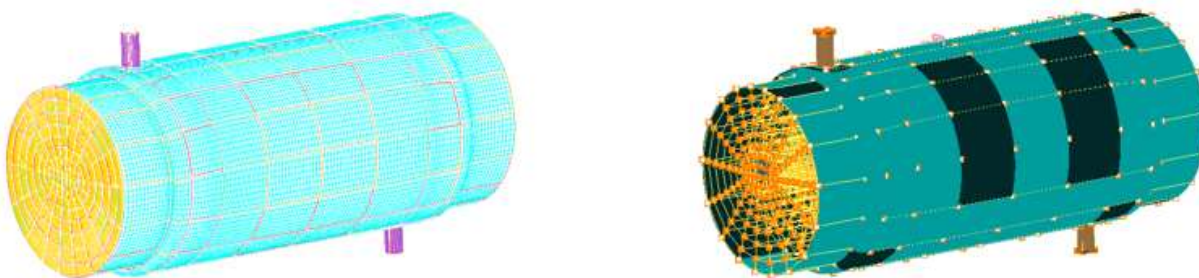


Figure 5.25: Grid and its corresponding topology extracted using Extract


The Extract button calls an external function called ‘**grid2til**’, a command-line utility. It offers more control in extracting the topology from a grid. Please refer ‘User guide for TIL’ manual for more information on the utility.

## 5.3 Post Processing Utilities

Sometimes you might like to make minor modifications in the grid, say changing the density in a few edges. Once the grid is generated, *GridPro* post-processing utilities are used.

### 5.3.1 Density

Similar to changing the number of cells for particular parallel edges or whole topology before a grid is generated, you can also change it after a grid is generated

using the **Density**  button in the **Edit** section in the right-side panel under the **Grid** tab. It is a post-processing tool where the changes are limited to the grid file only, and it is not updated in the topology file.

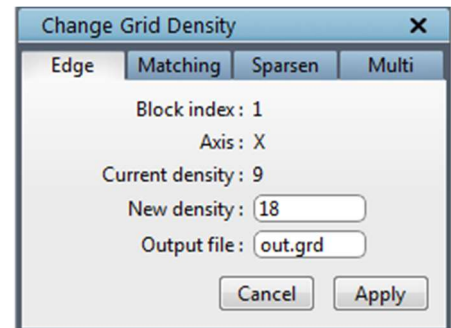


Figure 5.26: Density dialog box

**Density** button has four tabs, each performing different operations. Once you click on the Density button, the Change Grid Density dialog box opens, as shown in Figure 5.26. By default, the tab is set to **Edge**. The density of a particular edge group can be changed using the **Edge** tab. Open the dialog box and click on the desired edge whose density has to be changed. The selected block index will be updated in the dialog box. Enter the new density for the selected edge and click on *Apply* to see the changes. When the *Apply* button is clicked, a new grid file is created with the name specified in the *Output file* input field and automatically loaded into the UI. Once the new grid is loaded, the old grid will be switched off, which can be switched on again from the **Grid Toggling** section. The density of an edge of a block can also be changed by using the **Change density** option from right-click menu. Selecting the **Change density** option from the right-click menu of a grid block edge will open the same dialog box as shown in Figure 5.26.



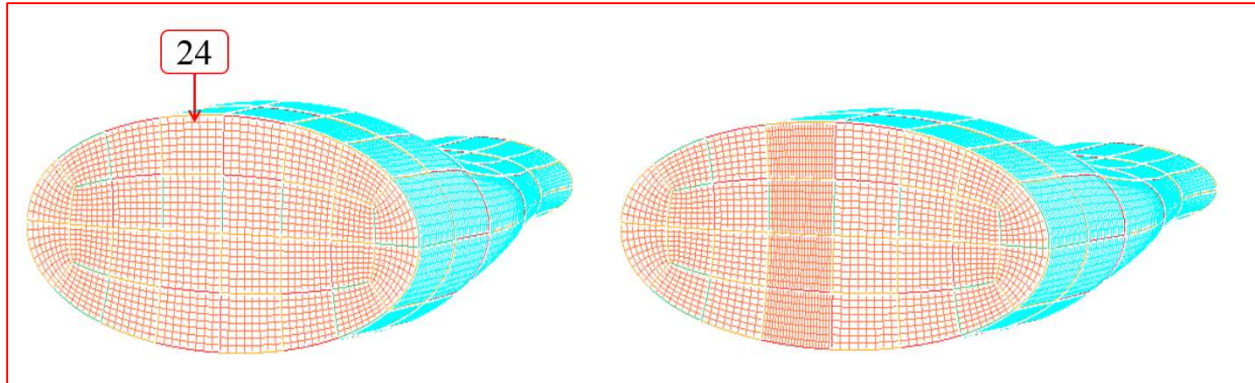


Figure 5.27: Before and after changing Edge density

The second tab in the Density dialog box is **Matching**, whose dialog box is shown in Figure 5.28. A particular density of a whole grid can also be modified using this option. It will modify the density of the edges which had the edge density as mentioned in the existing density field. Input both the existing and new density along with the output file name in the corresponding fields and click on *Apply* to see the changes. All the edges which have the density as mentioned in the **Existing density** field would be replaced with the **New density** value. A new grid will be created and loaded into the UI. Once the new grid is loaded, the old grid will be switched off, which can be switched on again from the **Grid Toggling** section.

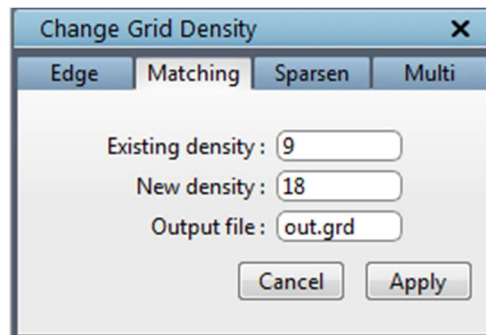


Figure 5.28: Matching tab dialog box

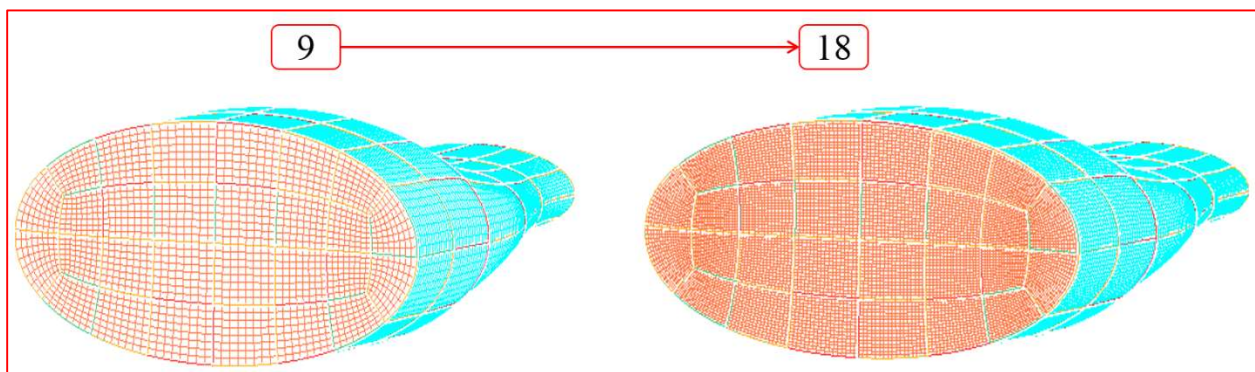


Figure 5.29: Before and after changing the density

If the density of the whole grid has to be increased or decreased by a ratio, the **Sparsen** option can be used, which is the third tab in the Density dialog box. The dialog box of the same is shown in Figure 5.30. Provide the ratio by which the density of the whole grid has to be changed along with the output file name. Click on *Apply* to load the new grid. Use a ratio less than 1 to sparse the grid and more than 1 to refine the grid. Similar to the **Matching** option, **Sparsen** will create and load a new grid file and switch off the old one.

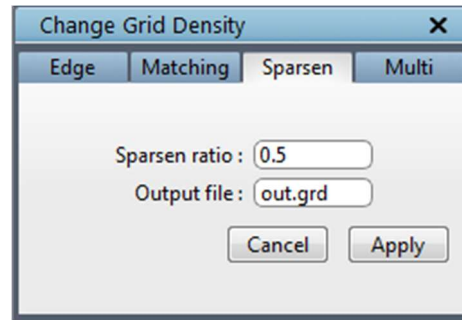


Figure 5.30: Sparsen tab dialog box

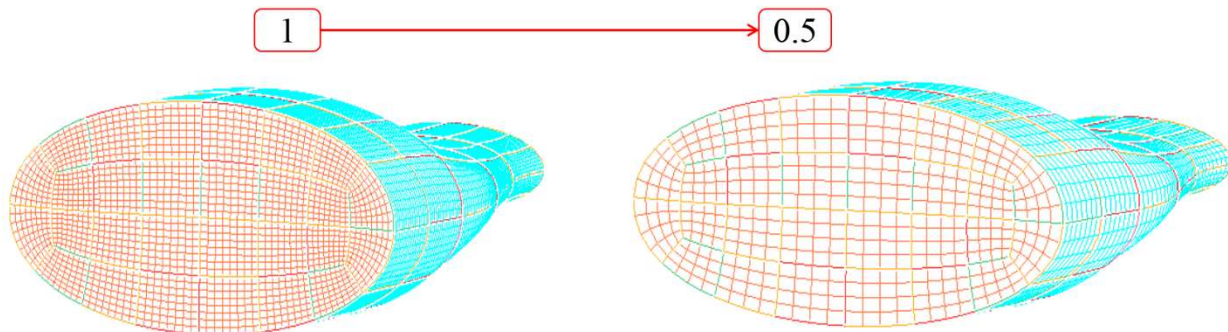


Figure 5.31: Before and after changing the density

Multiple grid files with arithmetic or geometric increment/decrement in ratio can also be generated using the Multi option, the last tab in the **Density** dialog box. The dialog box of the same is shown in Figure 5.32. Provide the ratio and number of grids that need to be created and select one of the increment methods, i.e. arithmetic or geometric. Based on these inputs, it automatically calculates the sparsen ratio for each grid. Input the prefix of output grid files. The output file name will have a nomenclature '*prefix\_<individual sparsen ratio>.tmp*'. You need to load the newly created grid manually.

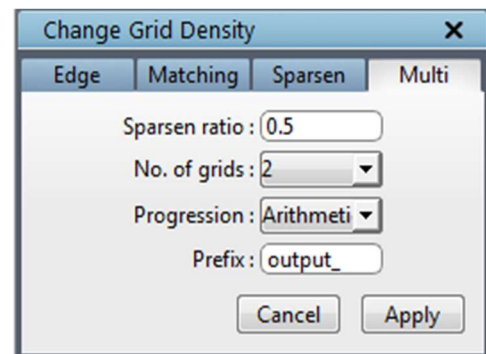


Figure 5.32: Multi tab dialog box

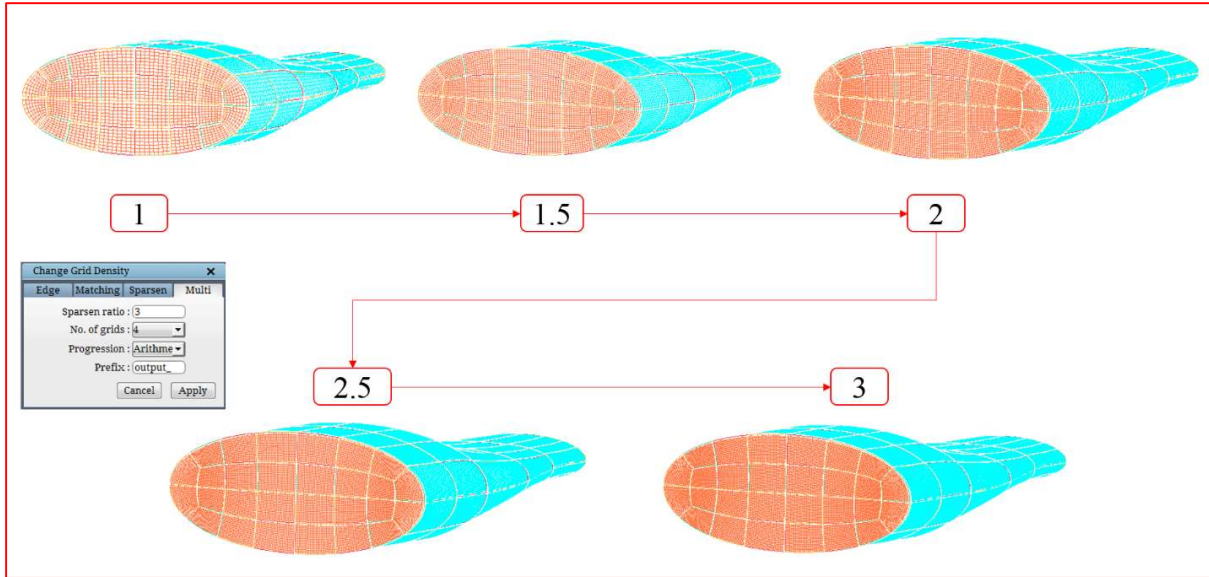


Figure 5.33: Multi-Grids output

Note: By default, the first three tabs have the output file name as ‘*out.grd*’. After the first execution, the *out.grd* becomes the *current* grid which may not be visually clear in the workspace. Please check the grids tab to find which grid is the current grid. If you are using the Density button for the second time simultaneously, ensure that you have selected the desired grid as *current*.

### 5.3.2 Transform

**Transform** is a transformation tool that can translate, rotate, mirror or scale the current grid in the UI. It can be found in the **Edit** section of the **Grid** tab, as shown in Figure 5.24. Like the Transform button in the topology tab, the Grid tab transform button also has separate buttons for each transformation option, such as translate, rotate, mirror and scale. Apart from this, it also has a separate button called General. Let us have a look at the buttons one by one.

- **Translate**



**Translate** button helps in translating the grid linearly along any XYZ direction. Clicking on the Translate button opens the dialog box, as shown in Figure 5.34. By default, it just translates the grid in the UI visually. Once the grid is translated to the desired location, it must be explicitly saved using **Save→Grid→Elementary Grid**. It can also be copied by checking the **Copy** checkbox. Enabling the copy check box enables the **Number of Copies** input box. The existing grid is copied and translated based on the given

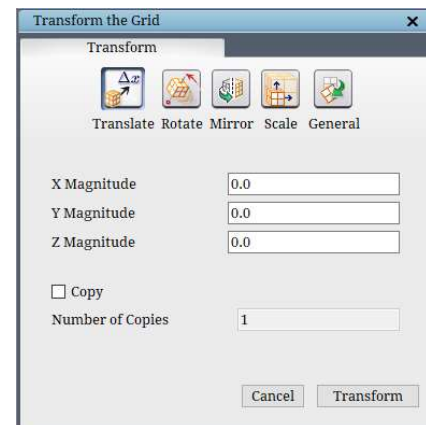


Figure 5.34: Translate dialog box



input if the Copy check box is enabled. Suppose if the no. of copies is given as 2, first the input grid is copied and translated once, next the translated copy would be copied and translated based on the input parameters. It goes on like an arithmetic progression until the input for no. of copies is satisfied. In other words, if the no. of copies is 3, it will create 3 copies of the input grid, each with an increment of the given translation values as shown in Figure 5.35. Note that all the copies should be explicitly saved to the working directory.

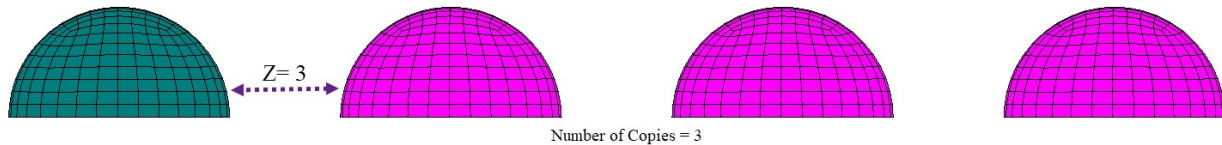


Figure 5.35: Translated grids with 5 no. of copies

- **Rotate**



**Rotate** helps in rotating the grid with the given angle using the centre and axis of rotation. Clicking on the Rotate opens the dialog box, as shown in Figure 5.36. By default, the centre and axis of rotation are used as the origin and X-axis. You can choose to change the **centre of rotation** to a work plane location or a user-defined centre. Similarly, the **axis of rotation** can be chosen from the dropdown list such as X, Y, Z and user-defined axis. Next, the **angle of rotation** has to be given. Once the ‘Transform’ button is clicked, the existing grid will be rotated and displayed in the UI. The rotated grid has to be saved explicitly if the desired rotation is achieved. Similar to translate, rotate has an option to copy the input grid. Enabling the **Copy check box** enables the number of copies input. For e.g., if the no. of copies is selected as 3 with an angle of rotation 90, then the input grid is copied 3 times with an increment angle of 90 degrees for each copy, thus completing a full 360-degree grid. Once it is copied, all the copied grids have to be saved explicitly. You can also merge the grids with each other to form a complete 360-degree grid using the Merge→Grid tool as explained in section 5.3.6.

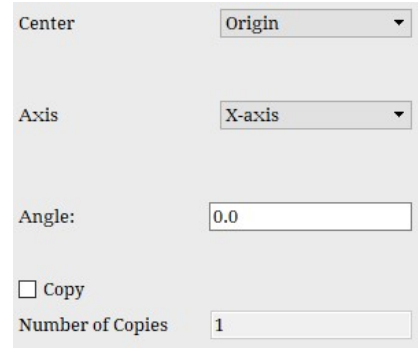


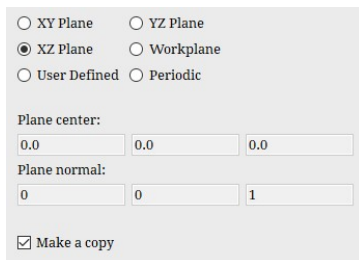
Figure 5.36: Rotate grid dialog box

- **Mirror**



Another feature that can create reflective or periodic copies of a grid is **Mirror**. If clicked on, the **Mirror** option will open a dialog box in the right-side panel, as shown in Figure 5.37. Specify the plane along which the copy has to be created. It creates a new mirrored grid and loads into GUI in addition to the original. If periodic is selected, it will create the copies based on their periodicity and merge them into a single grid file name with .tmp as an added extension to the original grid file name. The new grid file will be loaded into GUI in addition to

the previous grid file. If the **workplane** is selected, the centre and normal of the workplane is assumed as the mirror plane, and the grid will be mirrored with respect to that. If **user-defined** is selected, you can input the centre and normal of the plane along which the grid has to be mirrored. By default, the user-defined option is selected. There will be two grids existing in the UI after executing the command: the original grid and a newly created mirrored grid. The new grid must be explicitly saved to the working directory for further processing. Note that the mirrored grid can not be used to merge or any other post-processing tools unless it is saved and loaded manually into the UI.



☐ XY Plane    ☐ YZ Plane  
☒ XZ Plane    ☐ Workplane  
☐ User Defined    ☐ Periodic

Plane center:

0.0    0.0    0.0

Plane normal:

0    0    1

☒ Make a copy

Figure 5.37: Mirror Grid dialog box

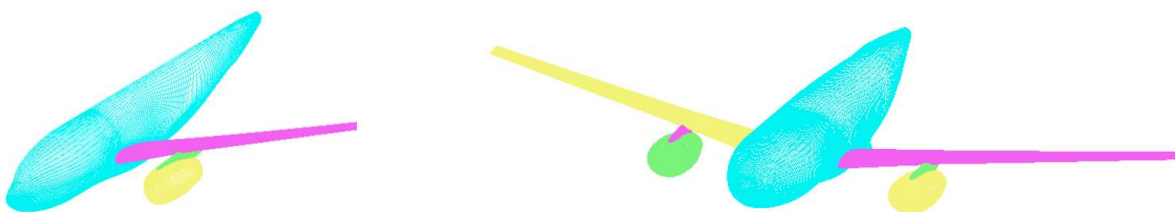


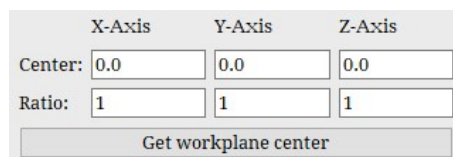
Figure 5.38: Before and after Mirror grid

Mirror grid calls an external function ‘extconn’, a command-line utility. It offers more control, such as mirrors, and merges the grid if the default axis is used. Please refer to ‘User guide for TIL’ for more information on the utility.

- **Scale**



**Scale** helps in scaling the grid using the given centre and scaling ratio. If you want to scale w.r.t the work plane location, reposition the work plane to the desired location and click the ‘**Get workplane centre**’ button to find out the workplane centre and fill it up in the Centre input box.



	X-Axis	Y-Axis	Z-Axis
Center:	0.0	0.0	0.0
Ratio:	1	1	1
<b>Get workplane center</b>			

Figure 5.39: Scale grid dialog box

- **General**



**General** tab allows the user to execute all the transformation options in one execution. Clicking on the **General** button opens the dialog box, as shown in Figure 5.40. In addition to the regular input options, you can also specify the transformation option using the Matrix form. Enabling the **Matrix form** check box enables the input boxes for the matrix and disables all other inputs. i.e. you can either transform using all other options except matrix or matrix form alone and not both. All the transformation options listed under General execute themselves in a series like fashion. i.e. it first executes translation 1, then scaling, next rotation and finally translation 2. By default, all the options values indicate no transformation, so if you do not provide any values for scaling or rotation, theoretically, it does not affect the input grid and skips to the next transformation option.

Figure 5.40: General grid transform dialog box

### 5.3.3 Extrude

Extrude is a simple extrusion tool, which can either linearly extrude the given block faces based on the inputs or extrude a 2D grid to 3D with a given no. of cells in the normal direction. It can be found in the **Edit** section of the **Grid** in the right side tab. Clicking on the **Extrude** button pops up a dialog box, as shown in Figure 5.41.



Figure 5.41: Extrude dialog box

- **3D Extrusion**

The input parameters can be given in two ways,

- i. Property ids
- ii. Surface ids

Which can be selected from the ‘**Select Mode**’ dialog box.

Selecting the ‘**property ids**’ option opens a table, as shown in Figure 5.42.

	Property ID	First Cell Spacing (*)	Growth Ratio	Length	Cells
1	1002				
2	2002				
3	3002				
4	4002				
5	5002				
6	6002				
7	7002				

Note : Length takes precedence over number of cells

Figure 5.42: Property id tabular form

**Property ID** is an id that is assigned when the pty file is written. When the grid is loaded, the pty file will be written to the working directory based on the surface Ids. The same can be visualized in the show/hide section of the Property panel. To know the property id of each property listed, you will have to open the \*.pty file in a text editor and scroll to the end of the file. The properties would be listed as shown in Figure 5.43. The number prior to the property labels (surface\_1, etc.) is called property id. The property ID of the new grid is inherited from the old grid upon extrusion.

```

12 Blocks
#B bid b-pty b-lbid f_i-pty f_i-lbid f_I-pty f_I-lbid ...
B 1 1 -1 2002 -1 1 -1 5002 -1 4002 -1 1 -1 6002 -1
B 2 1 -1 1 -1 1 -1 5002 -1 4002 -1 1 -1 6002 -1
B 3 1 -1 1 -1 3002 -1 5002 -1 4002 -1 1 -1 6002 -1
B 4 1 -1 5002 -1 4002 -1 1 -1 2002 -1 1 -1 1 -1
B 5 1 -1 1 -1 2002 -1 5002 -1 4002 -1 1 -1 7002 -1
B 6 1 -1 4002 -1 5002 -1 1 -1 3002 -1 1 -1 1 -1
B 7 1 -1 1 -1 1 -1 5002 -1 4002 -1 1 -1 7002 -1
B 8 1 -1 3002 -1 1 -1 5002 -1 4002 -1 1 -1 7002 -1
B 9 1 -1 5002 -1 4002 -1 1 -1 1 -1 1002 -1 1 -1
B 10 1 -1 1 -1 1 -1 5002 -1 4002 -1 1002 -1 1 -1
B 11 1 -1 1 -1 1 -1 5002 -1 4002 -1 1002 -1 1 -1
B 12 1 -1 4002 -1 5002 -1 1 -1 1 -1 1002 -1 1 -1
0 labels
10 0 2D properties (used, max-id)
0 unused (pdc:DEFAULT)
1 ws.Interblock (pdc:INTERBLK)
2 ws.Wall (pdc:WALL)
1002 surface_1 (pdc:WALL)
2002 surface_2 (pdc:WALL)
3002 surface_3 (pdc:WALL)
4002 surface_4 (pdc:WALL)
5002 surface_5 (pdc:WALL)
6002 surface_6 (pdc:WALL)
7002 surface_7 (pdc:WALL)
2 0 3D properties (used, max-id)
0 unused (pdc:DEFAULT)
1 ws.Fluid (pdc:Fluid)

```

Figure 5.43: Pty file

**First cell spacing** is the cell spacing of the first cell from the given grid face. It should be given the same value as the last cell space of the grid so that the grid will have a smooth transition. **Growth ratio** represents the ratio at which the cell spacing has to increase. **Length** defines the maximum length to which the

extrusion should be executed. **Cells** represent the no. of cells to be used for the extrusion. Note that either length or cells should be given as input; if both are given, length takes precedence. Once set, click on the ‘set parameters’ button and click on Apply in the main dialog box, as shown in Figure 5.41.

Similarly, if the ‘**surface ids**’ option is selected, a dialog box, as shown in Figure 5.44, pops up. Here, it will list all the surface Ids used in the topology file for generating the grid. All other options are the same as the ‘property id’ dialog box.

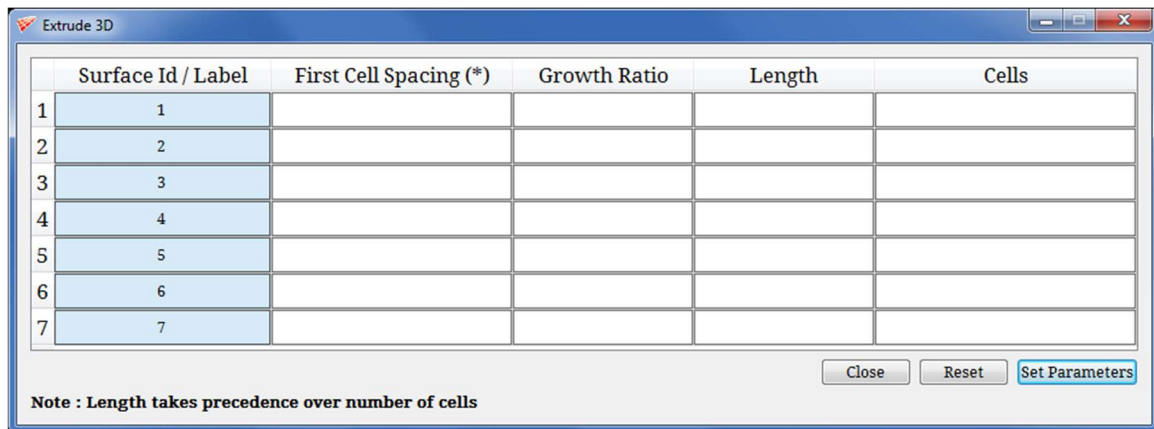


Figure 5.44: Surface id tabular form

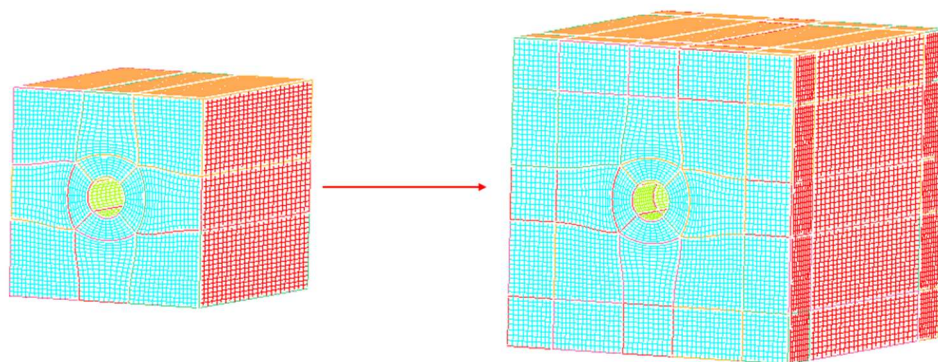


Figure 5.45: Before and After Extrude on all 6 sides



- **2D → 3D**

A 2D grid can be extruded to generate a 3D grid using the 2D→3D dialog box as shown in Figure 5.46. Since the 2D grids are always generated on the XY plane, they can only be extruded either +ve Z or -ve Z-axis. **Length** represents the length of the block in 3D which starts from Z = 0 and ends at the given length. **Number of sections** is the no. of cells that need to be generated within the block. This is similar to the New Density option under the Density button. Once the required input and output file name is given, it generates a 3D grid and loads it into the UI.

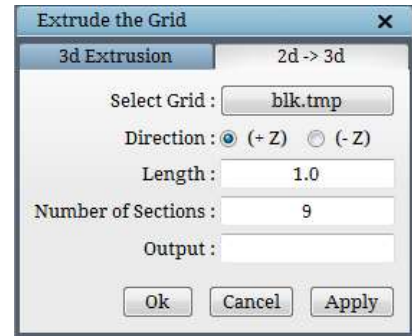


Figure 5.46: 2D → 3D dialog box

This operation is primarily helpful when your geometry has the same cross-section along its lengths. For e.g. analyzing an infinite wing geometry where the cross-section of the wing has the same airfoil profile along its span.

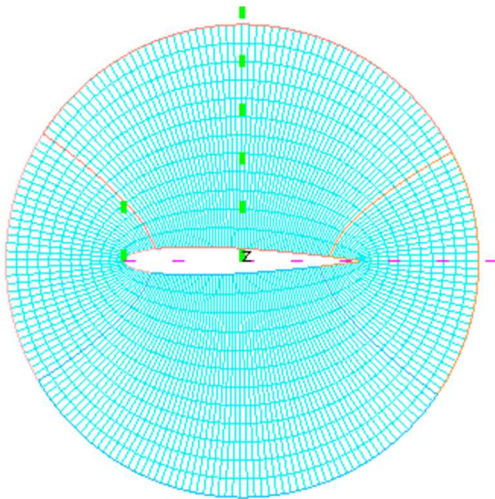


Figure 5.48: 2D grid

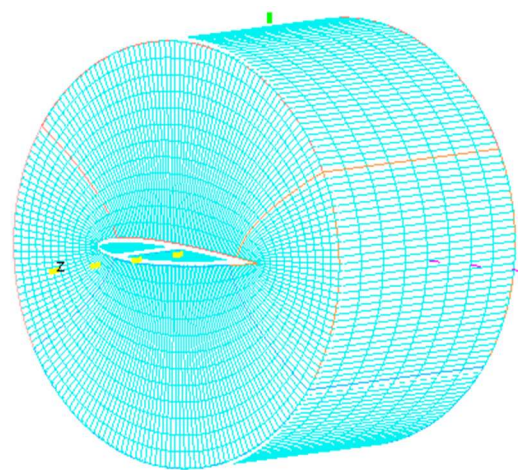



Figure 5.47: 3D grid after extrusion

### 5.3.4 Clustering

Boundary layer clustering  is one of the essential steps in grid generation. It was accomplished using two ways; either set it to the surfaces before grid generation or run it as a post-processing tool on the generated grid from the command line. Now the same is implemented in the UI with a tabular form, as shown in Figure 5.50.

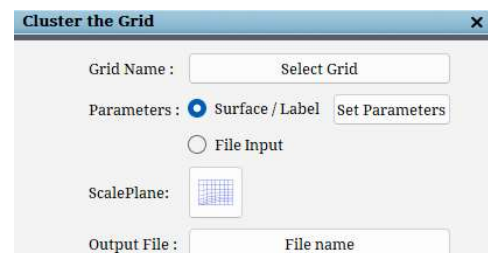


Figure 5.49: Clustering dialog box



If clicked on the **Clustering** button, a dialog box opens in the pop-up section, as shown in Figure 5.49. Grid Name will have the list of grid file names loaded into the UI. If the UI has only one grid, it automatically reads the same. The clustering parameters can be provided in two ways.

1. Using Surfaces or labels
2. Using a text file as input

By default, the **Surface/Label** option is selected. To set the parameters, click on the **Set Parameters** button, which will open a tabular form dialog box, as shown in Figure 5.50.

The Clustering Tool dialog box contains a table with 7 rows and 6 columns. The columns are: Enable (checkbox), Surface Id / Label, Direction (dropdown), Spacing (\*), Growth Ratio, and Cells. All checkboxes are checked, and all Direction dropdowns are set to 'Auto'. Below the table is a section for Global Values with fields for Algorithm (2 (slowest)), Multi Grid (1), Growth Ratio (1.2), and Precision (Double). There is also an Import Values dropdown set to 'Select a mode'. At the bottom are buttons for Close, Reset, and Set Parameters. Two notes are present: 'Note : A surface id / label row must be enabled for it to be clustered and off wall spacing cannot be empty.' and 'Note : The values are retained only for the rows which are enabled.'

Enable	Surface Id / Label	Direction	Spacing (*)	Growth Ratio	Cells
<input checked="" type="checkbox"/>	1	Auto		1.2	AUTO
<input checked="" type="checkbox"/>	2	Auto		1.2	AUTO
<input checked="" type="checkbox"/>	3	Auto		1.2	AUTO
<input checked="" type="checkbox"/>	4	Auto		1.2	AUTO
<input checked="" type="checkbox"/>	5	Auto		1.2	AUTO
<input checked="" type="checkbox"/>	6	Auto		1.2	AUTO
<input checked="" type="checkbox"/>	7	Auto		1.2	AUTO

Global Values

Algorithm : 2 (slowest) Multi Grid : 1 Growth Ratio : 1.2 Precision : Double

Import Values : Select a mode

Close Reset Set Parameters

**Note :** A surface id / label row must be enabled for it to be clustered and off wall spacing cannot be empty.  
**Note :** The values are retained only for the rows which are enabled.

Figure 5.50: Clustering tabular form

The table will automatically read the Surface Id's and Labels, if any, from the grid connectivity file. Enabling the checkbox across each of the ids will enable the corresponding input parameters box. **Direction** is applicable only for the 2 sided surfaces, and it will only be enabled for the same.

**Spacing** is the first cell spacing of the boundary layer clustering, and the growth ratio represents the stretching ratio at which the clustered cells have to grow. **Cells** option is the alternate option for the growth ratio. Under the Growth ratio, the ratio is mentioned, and the no. of cells for the block is being determined internally. Under cells, the no. of cells are provided, and the growth ratio is calculated internally. So for any surface id/label, only either one of them can be provided.

There are four global values: Algorithm, Multi-Grid, Growth ratio, and Precision. **Algorithm** is the smoothing parameter for the clustered cells, which can vary from 0 to 3.

Where,

0 = Linear algorithm with an average spacing

- 1 = Smoother algorithm with an average spacing
- 2 = Smoother algorithm with uniform spacing
- 3 = Linear algorithm with uniform spacing

**Multi-grid** defines that the number of cells in the clustered layer should be a multiple of the given input. For e.g., If the multi-grid is given as 4, then no. of clustered cells in a block will be a multiple of 4 such as 8, 12, 16, etc.

**Growth ratio** in the global values tab defines the global growth ratio. Modifying the global growth ratio will modify all the default growth ratios except the locally modified values under each surface id.

**Precision** defines the number of digits to be written for each coordinate. Single represents 8 digits, and double represents 16. Once the parameters are set, click on the '**set parameters**' button at the bottom and click '**Apply**' in the main dialog box as in Figure 5.49.

The parameters can also be imported into the table from a fra file or text file using the Import values dropdown. It can also be imported by selecting the '**File Input**' option in the Clustering dialog box as in Figure 5.49.

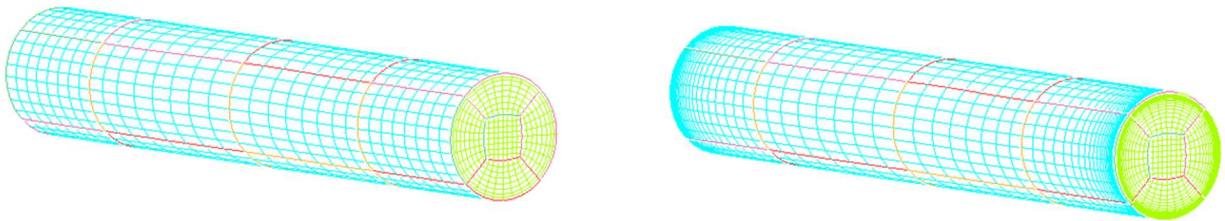



Figure 5.51: Before and After Clustering

There is another option called **Scale Plane** which can be opened using the button  in the clustering dialog box. It is used to scale the boundary layer cluster on the given region of interest by defining imaginary planes. The tool will achieve the given spacing at every scale plane and use interpolation for the cells in between two scale planes.

Note: The first cell spacing also depends on various parameters such as the length of the boundary block, first cell size of the adjacent block, etc. Hence it may not be exactly the same as the given value.

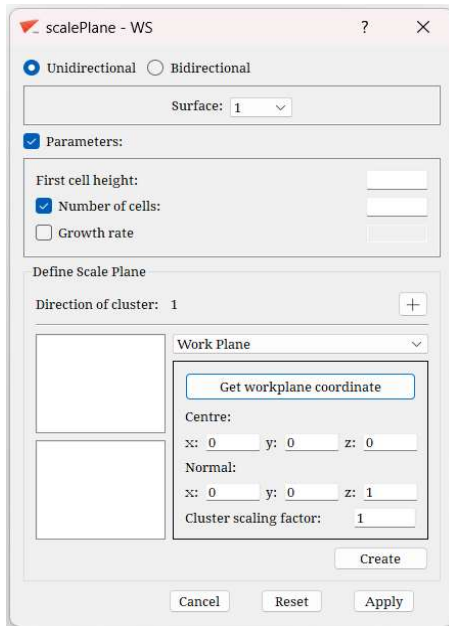



Figure 5.52: Scale Plane dialog box

The scale plane can either be **unidirectional** or **bidirectional**, depending on the direction of the scaling. The **Parameters** section must be enabled if the boundary layer clustering needs to be done on the grid surfaces. **First cell height** needs to be specified for the given surface. Only one option from among the **Number of cells** and **Growth rate** can be used.

In the **Define Scale Plane** section, the **Direction of cluster** only indicates to the user if the direction of scaling is unidirectional or bidirectional. The **+** button is used to add the required scale planes, which are represented by the letters **U** and **V** followed by their respective number. **U** represents the plane in one direction and **V** represents the plane in another direction. The **V** plane is only shown if the

bidirectional option is selected. The added planes are shown in the window below and also on the grid itself. The  button can be used to toggle the visibility of the planes in the grid.

The **Centre** and **Normal** of the scale plane can either be defined using the **Work Plane** or can be defined by the user using the **User Defined** option from the dropdown. In case the Workplane is used, it can be positioned to the required scale plane's location and the **Get workplane coordinate** button can be clicked to automatically get the Centre and Normal. The user will have to manually enter the Centre and Normal in case the User Defined option is used. The **Cluster scaling factor** defines the factor by which the boundary layer is to be scaled at the given scale plane. Clicking on **Create** updates the Centre, Normal and Cluster scaling factor for that plane.

The usage of the scale plane is shown with the help of a mesh on a cube. The scale planes are used to enclose the regions between which the scaling is to be done. The first scale plane, where the scaling begins from (left plane), has a scaling factor of 1, whereas the plane on the right has a scaling factor of 2. Note that in the image, it is the lower surface which is being clustered and subsequently scaled. The scale plane must never be parallel to the surface being clustered and scaled. Also, the scaling can be performed on an unclustered grid, but the output grid must be clustered. This means that the unclustered grid will first be clustered and then the boundary layer will be scaled.

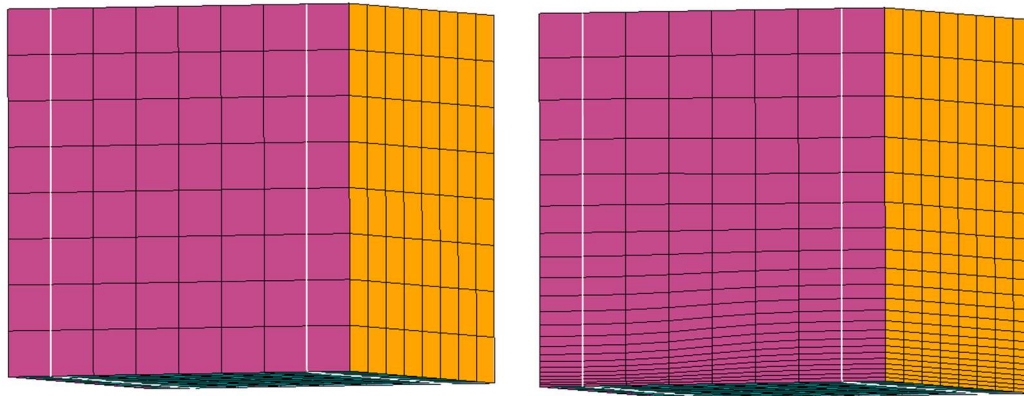


Figure 5.53: Unidirectional boundary layer scaling

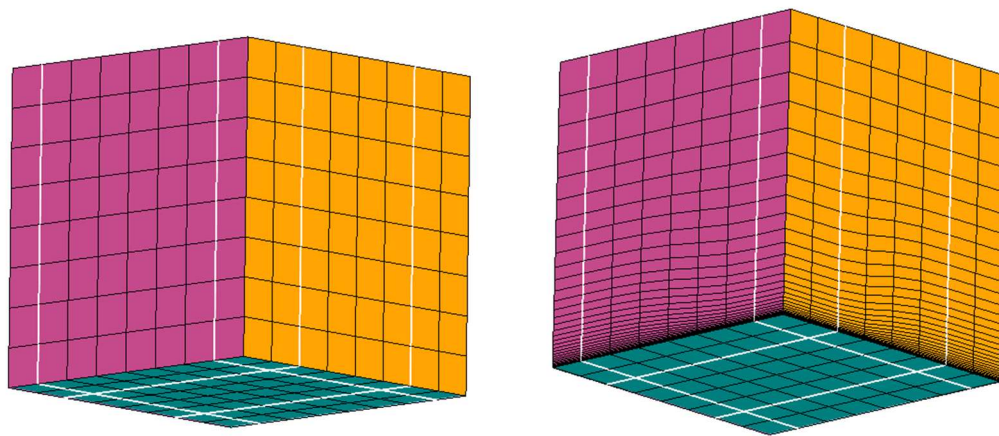



Figure 5.54: Bidirectional boundary layer scaling

### 5.3.5 Merge

Structured multi-block grid with many blocks is not always preferred because of the computation time. **Merge**

button  can help in reducing the no. of blocks. Clicking on the Merge button opens a dialog box, as shown in Figure 5.55.

You can either merge the blocks within a grid using the Merge→Blocks tab or merge two different grids into a single grid using the Merge→Grids tab.

By default, the dialog box opens with the **Merge→Blocks** tab, where the input grid has to be selected.

If multiple grids are loaded into the UI, a dialog box will pop up with the list of grids in the UI;

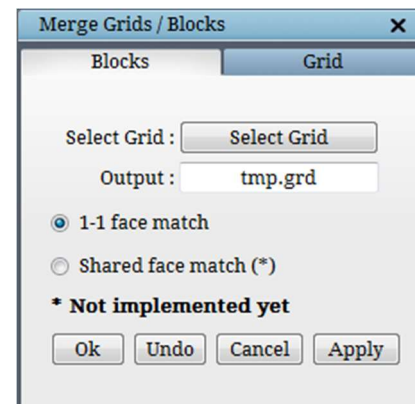


Figure 5.55: Merge→Blocks tab

otherwise, it directly chooses the grid. The output grid file name cannot be defined by the user in this version; however, it will be improved in the upcoming releases. Currently, it only merges the blocks with one to one face match.

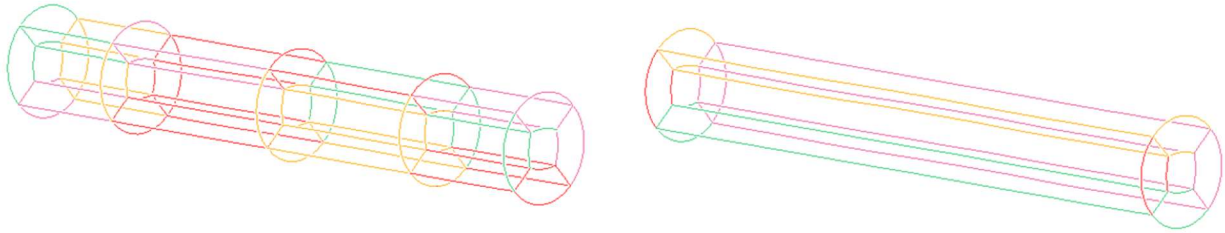


Figure 5.56: Before and After merging

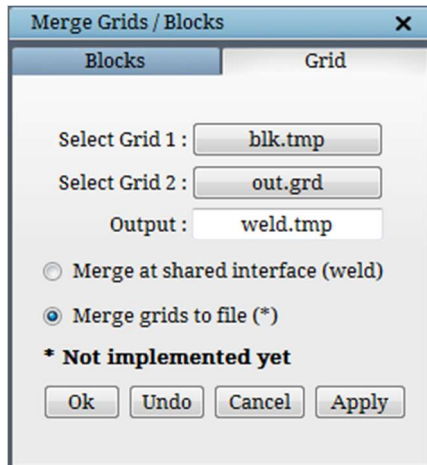


Figure 5.57: Merge →Grid tab

**Merge→Grids** tab helps in merging two different grids into one by welding the interface. It is helpful when you are working on a huge case. You can split the geometry into pieces with common boundaries and run Ggrid separately. Later the grids can be welded together to get a single full grid for analysis. It should have the same block pattern and no. of cells at all the welding interfaces. Currently, it only executes basic welding. Refer to the command line tool called ‘weld’ for advanced merging options. Similar to Merge→Blocks, users also need to select the grids from the loaded ones.

### 5.3.6 Node

**Node** button helps improve the quality of the grid by moving the grid nodes. Many times you may end up having various singularities while building topology. Only the topology which is free of singularities will output a grid without any folds. Resolving the singularities is a time-consuming process; hence we came up with a tool that can move the nodes at the grid level. Once you generate a grid with folds, create a cross-sectional sheet across it and move the nodes manually to the desired position such that it is not folded anymore. Clicking on the Node button will pop up a dialog box and highlight the grid nodes of the current grid sheet, which aids the cursor to move the desired nodes, as shown in Figure 5.58.



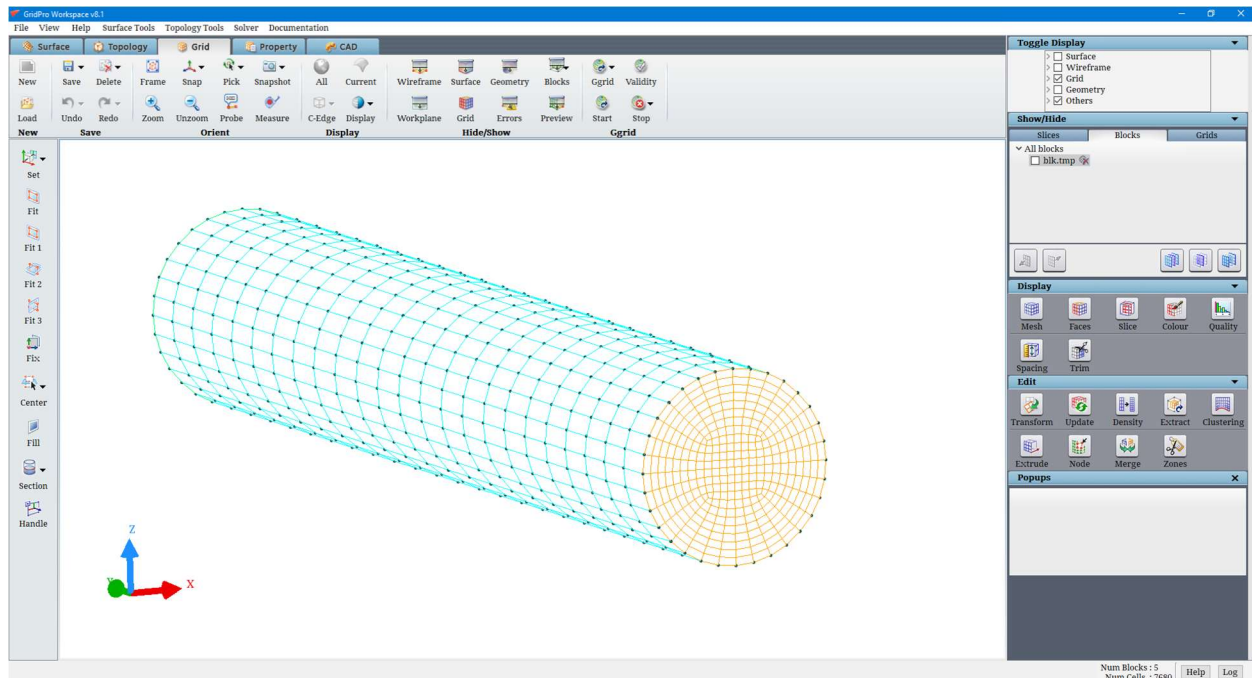


Figure 5.58: Node dialog box

Please note that the nodes can only be moved on the Screen Z-axis of the workplane, so you will have to be cautious while moving. If any nodes are moved by mistake, you can undo the step by clicking on the **Undo** button in the dialog box. Similarly, all the nodes which are moved manually are listed under the '**List**'. Any node movement can be reverted using this list.

### 5.3.7 Zones

Boundary conditions are an integral part of a grid. There are many ways to set a 2D boundary condition like inlet, outlet, wall, etc., to a grid sheet in GridPro. When it comes to 3D boundary conditions or volume property, you can only set it manually using the available tools in the Property tab. It becomes cumbersome when you have to pick the block centres manually in a complex grid like aircraft. To ease the process of setting the volume properties to a complex grid, a new function called '**Zones**' is implemented in v6.7.



Figure 5.59: Edit section



**Zones** is located under the **Edit** section of the **Grid** tab, as shown in Figure 5.59. Using this, you can now set the volume property with a click of a button. Based on the given grid sheets, it splits the grid blocks internally and sets a default boundary condition to each piece. These default boundary conditions can be renamed later using the Solver section in the property tab, as explained in section 6.2.

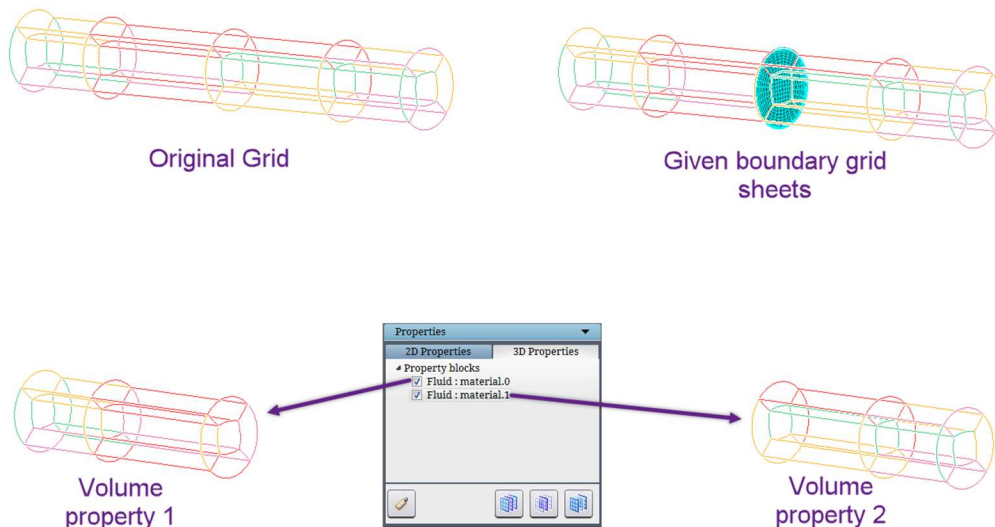


Figure 5.60: Create 3D property

If the input sheet does not have any internal boundary sheet or volume sheets, then the UI will report a warning message.

### 5.3.8 Smooth Grid Blocks

The **Smooth Grid Blocks** option is located under the **Grid Tools** dropdown in the Menu bar. It is used as a post processing step after the grid has been generated. This tool smoothes the interior nodes in the grid and tries to eliminate any folds in the grid. This option is best used when there are a few folds in the grid generated which the grid smoother Ggrid is unable to eliminate, and thus its usage is very case specific. There are two options to be selected from - **TFI** (Transfinite Interpolation) and **PDE** (Partial Differential Equation). TFI based smoothing has a

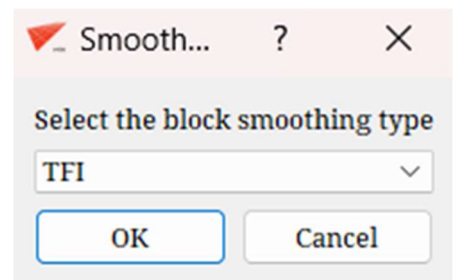


Figure 5.61: Smooth Grid blocks dialog box

low computational cost. PDE based smoothing is well suited for regions of high curvature, however at a higher computational cost. Note that only the interior nodes are smoothed by this tool, block faces and boundaries remain fixed. The **PDE** based smoothing will result in less folds in regions of high curvature.

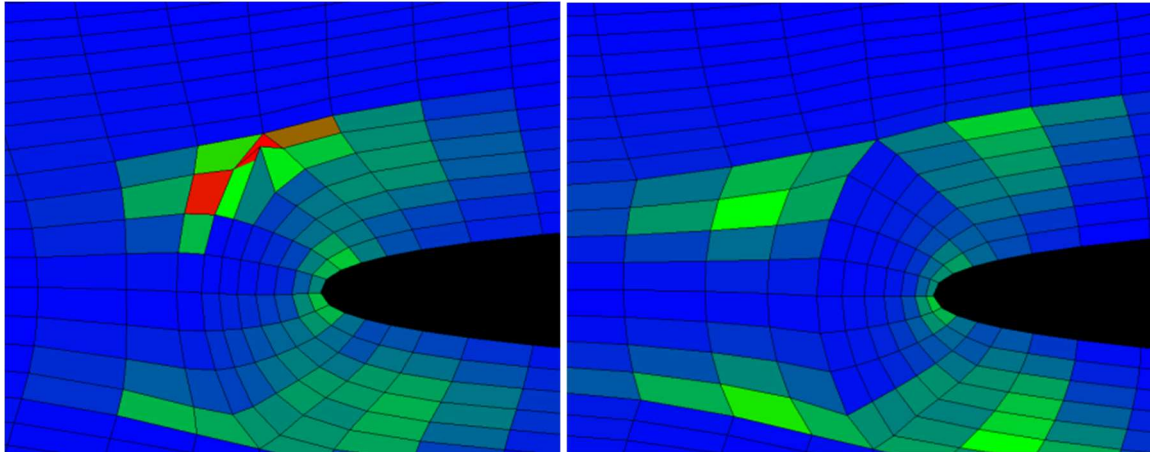


Figure 5.62: Before and after smoothing the grid blocks

## 5.4 Grid Probing Utility

### 5.4.1 Spacing

Sometimes, the user will need to test the smoothness of a grid or measures the precise dimensions of a grid. The distance between two nodes can be measured precisely by first



pressing the spacing button found in the **Display** section in the right-side panel in the **Grid** tab, then moving your cursor to the line connecting two nodes and clicking the left mouse button. Colour display mode must be in **colour by sheet** mode to use **spacing** because it works only on the *current* grid sheet. A green line connecting two nodes will be shown when a grid line is clicked after pressing the spacing button. The length of the line, or the node spacing, is displayed next to the line as shown in Figure 5.63. The **spacing** button proves helpful when clustering is used in the grid. It allows to precisely check the off-wall spacing near a boundary.

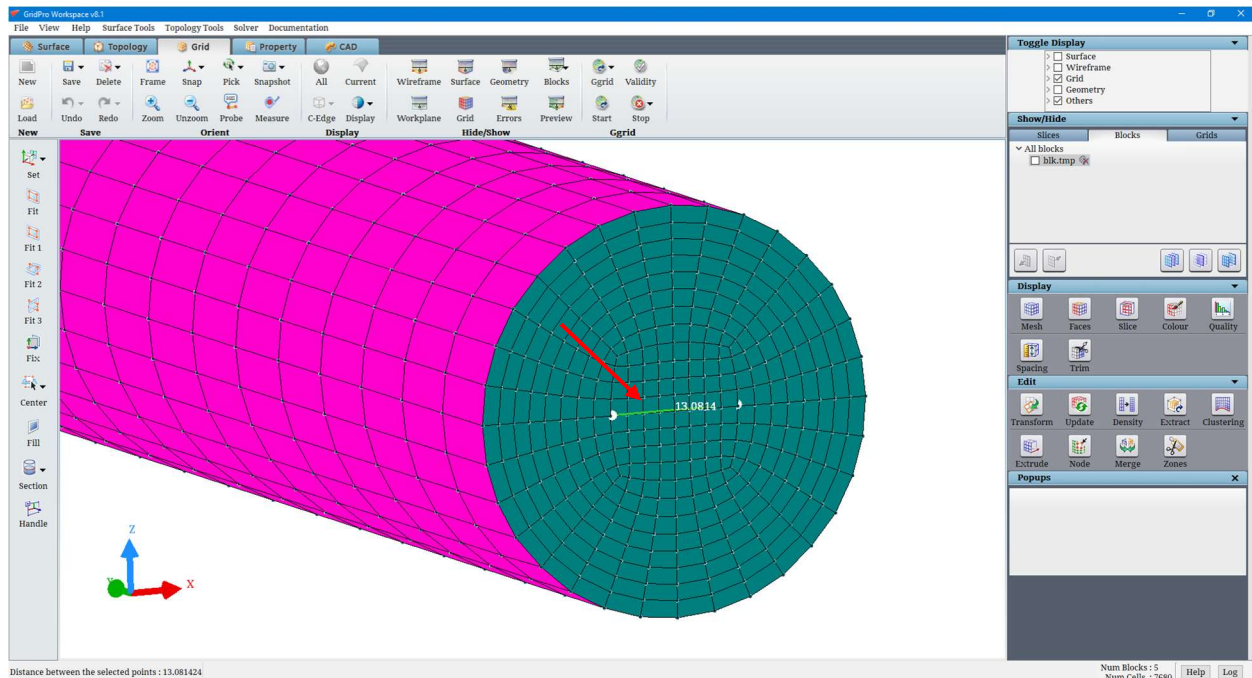


Figure 5.63: Spacing line with the distance pop\_up

Note: The spacing between any two nodes will be calculated in a straight line, and it will not follow the path of the grid lines. So the value will not be accurate in curvy regions.

## 5.4.2 Higher Order Mesh Quality

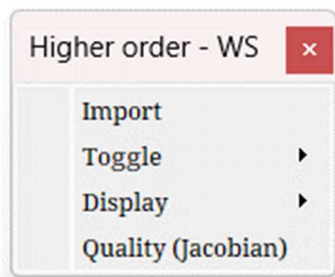


Figure 5.64 Higher Order dialog box

GridPro has the ability to export its grids into higher order formats. Viewing the created grids can be done using the **Higher Order** option under **Grid Tools** in the Menu bar. The **Import** option allows you to directly import a higher order mesh (.hex) into the UI and view it. The **Toggle** option is used to **Show** or **Hide** the higher order mesh. Note that any details about the higher order mesh do not show up in the **Show/Hide** section in the Grid tab. The **Display** section allows us to view the grid with **Only Edges**, **Edges with corners**, **Edges with nodes** and **Edges with all nodes**.

The **Quality (Jacobian)** option can be used to check the higher order mesh quality. A grid loaded in the UI, in *GridPro*'s native .tmp format is directly converted into a higher order grid of required order and checked for its quality using this option. Any elements whose Jacobian lies between 0.5 and 1 are displayed in the GL along with the native mesh present.

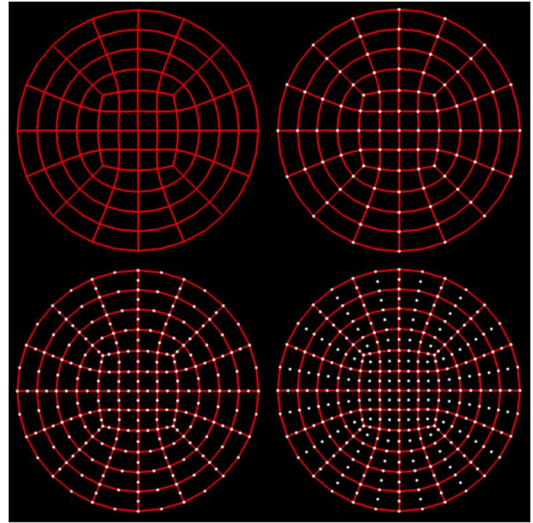


Figure 5.65 The various Display types

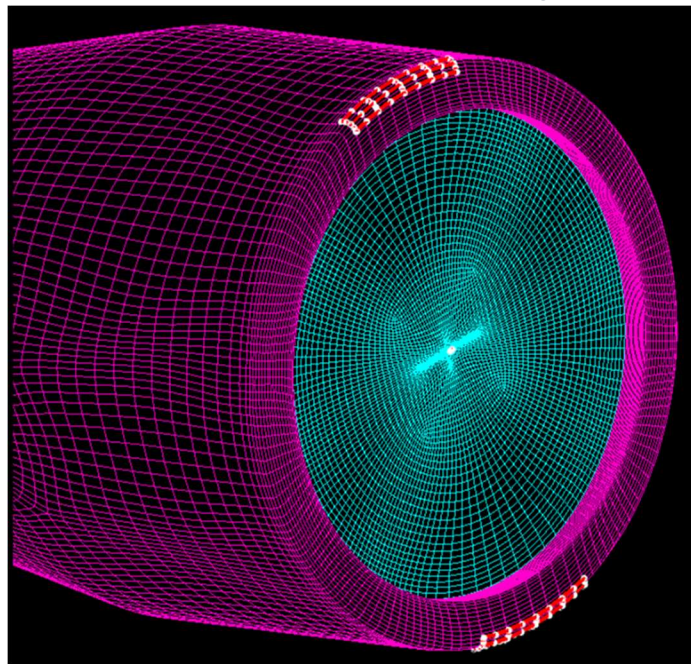


Figure 5.66 The native mesh and the bad quality higher order elements



## 5.5 Quality Check Parameters

Quality check is one of the most critical post-processing tools in mesh generation. Using a good quality grid will give more accurate results from the solver. GridPro has a quality checking tool called **Qcheck**, accessible from both command line and GUI. To check the quality of the grid, click on the **Quality** button in the **Display** section of the right-side panel of the **Grid** tab. Once clicked, the Quality parameters dialog box opens, as shown in Figure 5.67. Clicking on Evaluate button will give a report on each parameter. The individual parameters can also be viewed in a histogram by clicking on the graph icon next to each parameter. In addition to quality check, it can also check if all the blocks are oriented to right-handed, display the folded cells (i.e. negative volume cells), check mismatched blocks, periodicity of the grid, first cell height and also add the topology faces of the folded cells to QFAIL group.

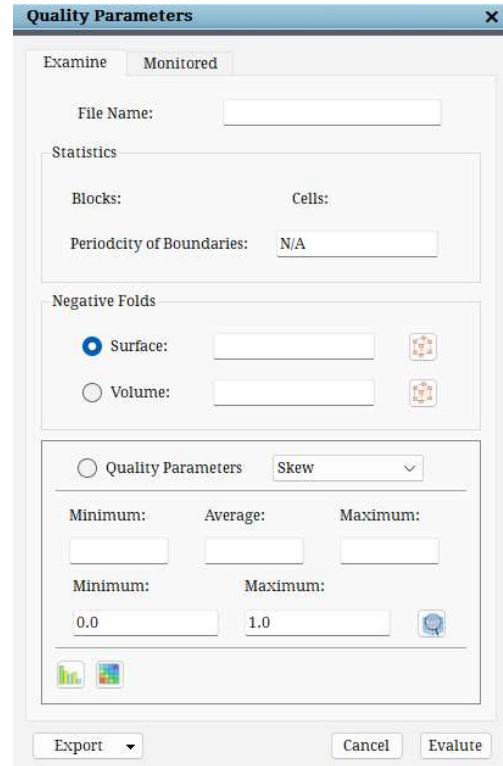


Figure 5.67: Quality dialog box

The first section shows the grid statistics, i.e. Grid file name, number of blocks & cells, and number of surface & volume folds. Clicking on the radio button next to the surface & volume folds highlights the corresponding folded cells in the grid. Clicking on the icon groups the corresponding topology blocks to its respective group button, i.e. SF and VF.

- **Folds**

Folds are defined as the inconsistency in the positioning of the grid nodes of a cell with respect to the neighbouring cells. Display folded cells options load all the cells folded in the grid as blocks for better visualization.

- **Skewness**

The first option in the Quality Parameters section is Skewness. Skewness is always considered the best measure of mesh quality. Skewness determines how close a cell or face is to the ideal one, i.e. Equilateral. Skewness is measured from 0 to 1. A skewness of 0 indicates an equilateral cell (best), and a value of 1 indicates a completely

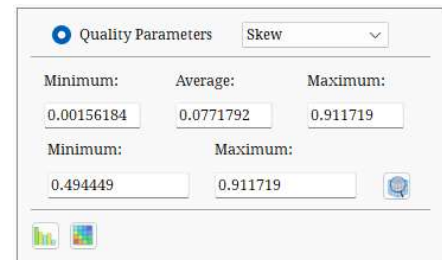



Figure 5.68: Skewness section

degenerate cell (worst). A skewness up to 0.9 is widely accepted for computation.

Once the Evaluate button is clicked, all the parameters are calculated, including skewness, and displayed in their respective columns. You can choose to view the skewed cells of a grid by giving a certain range. For e.g. 0.8-0.9. Enter the desired minimum and maximum values and click on the  icon. The skewed cells of the given range will be displayed in the UI.

- **Warpage**

The next option is Warpage, a measure of the angle of a face, which is defined as the angle of the normal vectors of the two planes generated by cutting through the diagonal corners of the face. **Qcheck** uses 12 possible measures for a cell's face to define the warpage angle of the cell. The warpage angle is measured from 0 to 180 degrees. Like skewness, warped cells can also be viewed based on a given range of values.

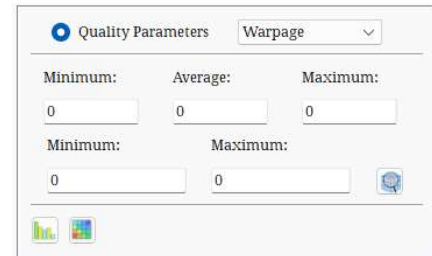


Figure 5.69: Warpage section

- **Aspect ratio**

Next option is the Aspect ratio, which is a measure of the ratio of the longest to the shortest side in a cell. It is a kind of local measuring done individually on each block, irrespective of the neighbouring blocks. The aspect ratio value can range from 1 to infinity. Ideally, it should be equal to 1 to ensure the best results. By default, it takes the range from 1 to 100000.

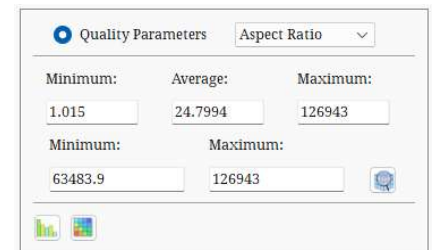


Figure 5.70: Aspect Ratio section

- **Orientation**

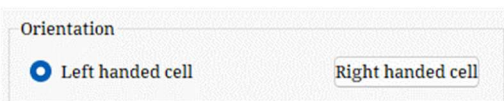


Figure 5.71: Orientation section

Most of the CFD solvers prefer a grid with right-handed orientation blocks. Hence, a different option is provided for the same. To check if the grid has left-handed blocks, open the **Monitored** tab and click on the **Left handed cell** radio button. If the grid has left-handed blocks, a pop up will be shown as in Figure 5.72. Clicking on the icon changes all the



left-handed orientation blocks to right-handed. Note that the new grid has to be saved explicitly to the working directory, otherwise the modifications would be lost.

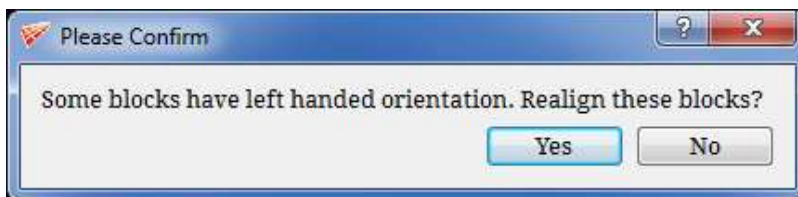


Figure 5.72: Confirmation dialog box

- **Mismatched Blocks**

Sometimes the cells of the adjacent blocks would be overlapping, and the pattern may not be the same at the interface; this happens due to boundary layer clustering. In such scenarios, the quality of the grid may not be suitable for CFD. An option is provided under the quality button to figure out such mismatched blocks and resolve them. Clicking on the eyeball icon highlights the mismatched blocks, and clicking on the Fix button will resolve the mismatched blocks.

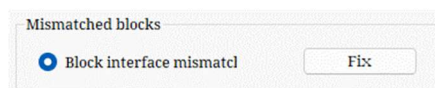


Figure 5.73: Mismatched blocks section

- **Periodicity**

If the grid is periodic, the periodicity of the grid can be verified using the periodicity section in under **Statistics**. Clicking on the Evaluate button figures out the periodicity of the grid and fills in its corresponding box.

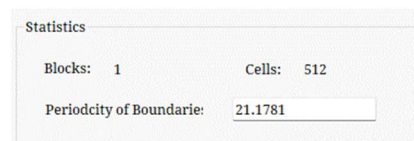


Figure 5.74: Periodicity section

- **First Cell Height**

The last option under Quality parameters is the First cell height. Though we give a specific value as first cell height while clustering, it is not respected throughout the grid. It always deviates slightly due to various reasons. A separate option is provided under the quality button to find out the average first cell height around a certain surface grid sheet. The first cell height can be calculated for any given surface by selecting the surface grid sheet id from the dropdown list and clicking on the calculate button, which will evaluate the average height and enter the value in the corresponding box.



Figure 5.75: First cell height section

## 5.6 Grid Toggling Section

The **Grid** tab can hold many grids in the same space. If the previous grid is not removed from the **Grid** tab, the *current* grid which is loaded will overlap the previous grid. Just as the

surfaces are selected as *current*, grids can also be selected as *current*. The user can toggle between grids, grid blocks and sheets from the **Show/Hide** section located in the right-side panel below the **Toggle display section**, as shown in Figure 5.76. Local toggle section has three tabs, one for **Grids**, one for **Grid blocks** and the other for **Grid slices**. By default, the grid slices tab is visible; click on the **grid/blocks** tab to see the grid toggle tab/blocks toggle tab. The names of all the grids loaded into the GUI can be seen in both blocks and grids tab.

The grid name, highlighted with a blue shade, is the *current* grid. All the operations performed in the **Grids** tab are executed on the *current* grid blocks and their respective sheets. While in the **blocks** tab, it is performed on the blocks of the *current* grid and not on its respective sheets and in the **sheets** tab, it is performed only on the sheets of the *current* grid and not on its blocks/grid. Any grid can be selected as *current* by clicking on that grid name. To hide a particular grid, uncheck the box located before the respective grid name. If a box is checked, the grids will be visible. To delete a particular grid from GUI, click on the delete icon next to the grid name. Once the grid is deleted, it cannot be retrieved. If it has to be retrieved, it must be loaded again using the **Load** option in the **Global dock**.

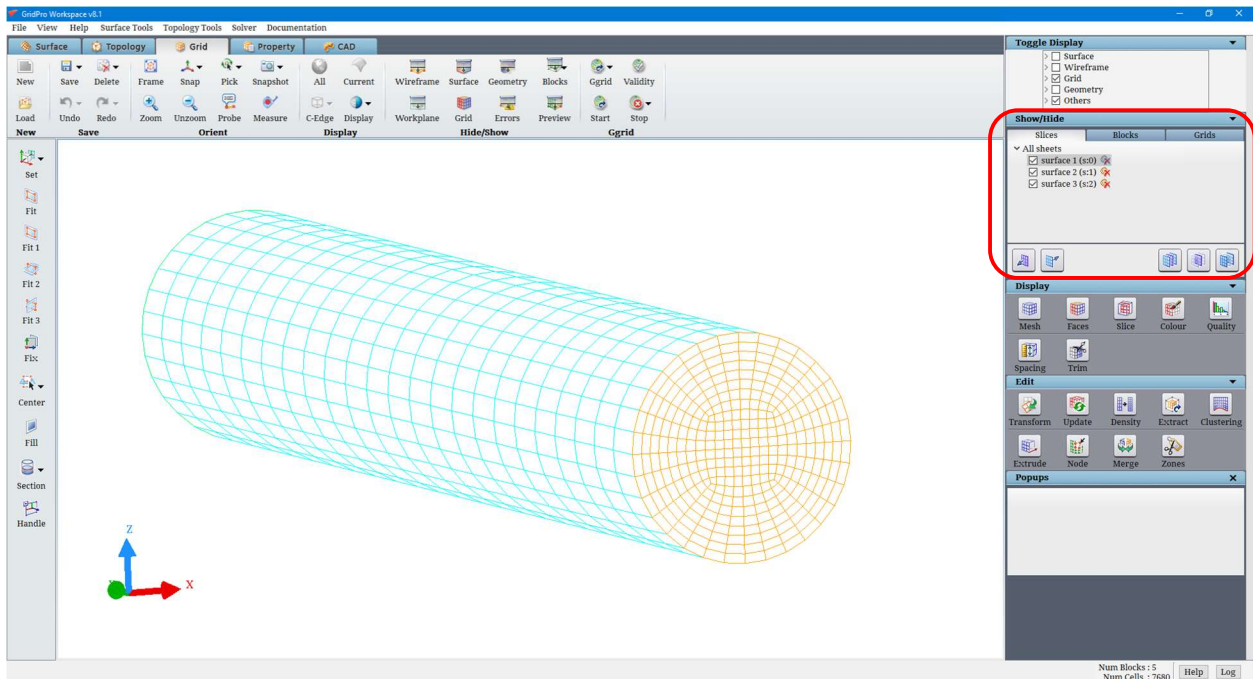





Figure 5.76: GUI highlighting the Show/Hide section

As discussed before, the *current* grid can be deleted using the **Current grid** option under the **Delete** button in the **Global dock**. Below the grid names list, there are five buttons placed, as shown in Figure 5.77. The first two buttons are enabled for only **Grid slices**. The rest three are enabled for all three tabs. With respect to the **Grids** tab, the third button, i.e. **Show all Grids**  button switches on display for all the grids loaded into GUI. The fourth button, i.e. **Show only Current Grid**  button, hides all the other grid and displays only the current grid. The fifth button, i.e. **Toggle all grids**  button, flips the display of the grids in the GUI, i.e. grids that are displayed will be hidden, and hidden grids will be visible. Once the **Current** grid is selected, the user can create as many grid sheets as desired. The grid sheets can be toggled from the **Slices** tab. It will show only the sheets belonging to the *current* grid. Similar to the grid tab, grid *slices* have the same options. A sheet can be selected as *current* by clicking on it. The checkbox can be used to show or hide the respective sheets. The respective grid sheet can be deleted using the **delete** icon next to the grid sheet name.

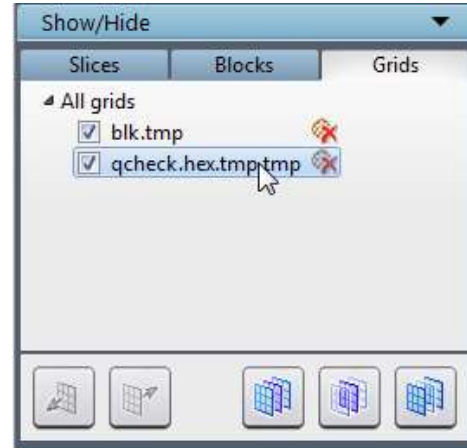







Figure 5.77: Grids tab

The first two buttons, i.e. forward step and backward step, below the grid sheet list allow the user to scroll the *current* sheet through all the positions within that block. Forward  button steps the sheet forward, while backward  button steps backwards. It is useful when a sheet is created using the **slice** button, and the user wants to check how the *current* grid sheet is moving across that block. The rest of the three buttons function similar to grids. Third button, i.e. **show all grid sheets**  will turn the display on for all the grid sheets in the above list for the

*current* grid. The fourth button, i.e. **show only current sheet**  will hide all other sheets in the list for the current grid and will show only the *current* sheet of the *current* grid. The fifth button **toggle all grid sheets**  will flip the visibility of all the grid sheets.

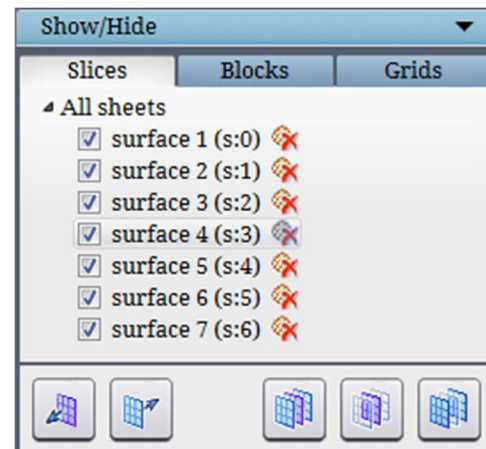





Figure 5.78: Slices tab

The **Blocks** tab displays the grids that are loaded into the GUI. As discussed before, the *blocks* tab works similar to the *Grids* tab, but the only difference is, in the *grids* tab, every operation is performed on both blocks and sheets of a grid whereas, in the *Blocks* tab, it is performed only on the blocks of a grid. The three buttons which are enabled in *Blocks* tabs work as follows, the third button, i.e. **show all grid blocks**  will turn the display on for all the grid blocks in the GUI. The fourth button, i.e. **show only current grid blocks**  will hide all other grid blocks (not sheets) in the GUI and show only *current* grid blocks. The fifth button is **toggle all grid blocks**  will flip the visibility of all the grid blocks.

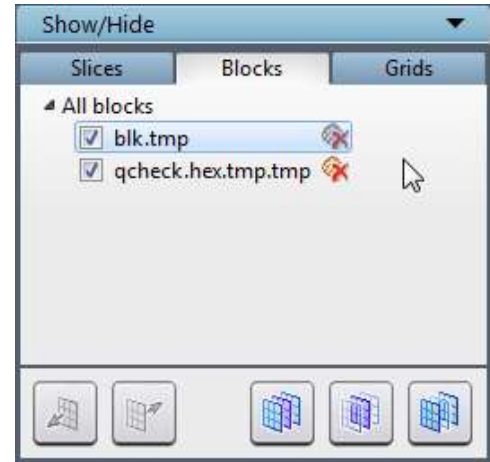


Figure 5.79: Blocks tab

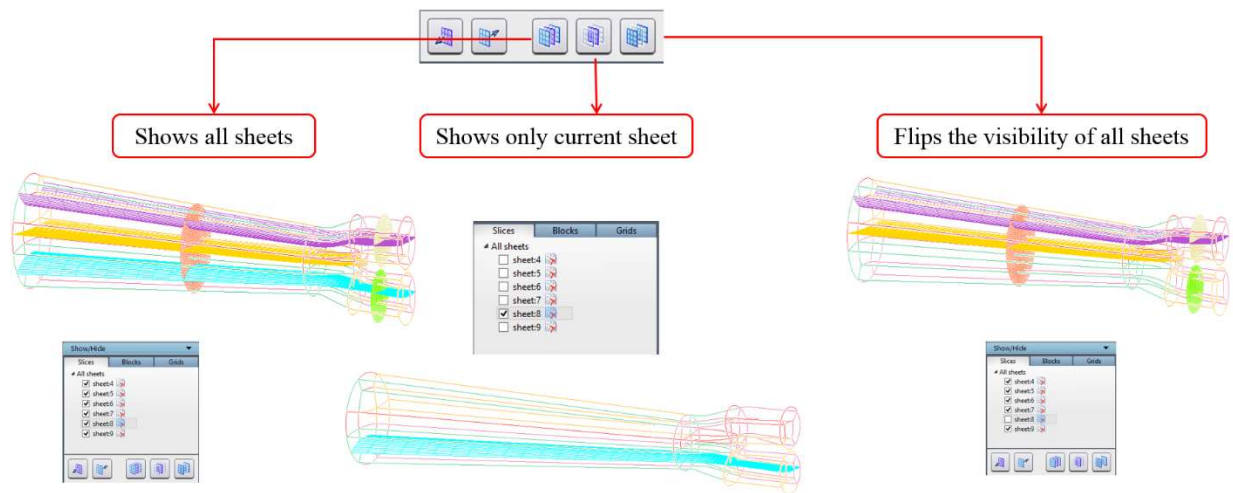


Figure 5.80: Toggling options

NOTE: Switching off the grid sheets or grid blocks in the toggle display section will also switch off the corresponding object in the **Grid sheets** section, but vice versa is not possible.

## 5.7 Grid Trimming Utilities

Besides viewing the whole grid, GUI has an option to view a desired portion of the grid and save it with the desired name. Trimming feature helps to achieve this. Both the blocks and sheets can be trimmed as per the requirement. Operations related to trimming are grouped in the



**Trim** button located in the **Tools** section. Clicking on the **Trim** button will open a dialog box, as shown in Figure 5.81. Users can select the trimming tools for blocks or faces by selecting one of the tabs in the **Trim** dialog box. Select **Blocks** for trimming blocks and **Faces** for trimming sheets. By default, **Blocks** have been selected for trimming.

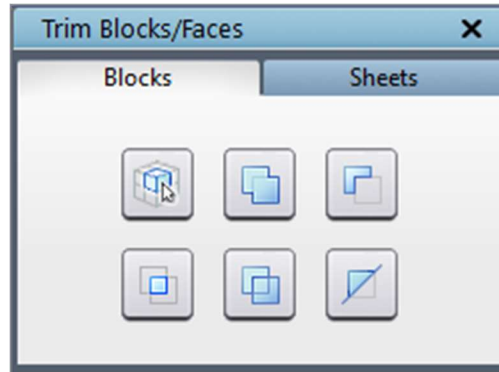


Figure 5.81: Trim dialog box

### 5.7.1 Blocks

Each grid block has a centre, an outline, and a skeleton. The outlines of blocks are the first visual features of a grid that can be seen. The outline of each block is displayed once the grid, *blk.tmp* file is loaded. The outline of a block is simply the boundary of the block formed by its edges. The centre of each block is the centre of mass of each block and is represented by a small square. The skeleton of a block is the lines connecting the indexed centre of the block with the centres of the block's bounding edges (2D) or faces (3D). Each block skeleton is drawn in its own colour. Each block skeleton forms a skeleton for the entire multi-block grid when collected. The skeleton represents the minimal amount of information required to show a broad representation of a multi-block grid. Also, the skeleton illustrates the flow of the block structure better than the outline. In three dimensions, the number of lines produced by a skeleton is 6, while the number of edges of a block is 12. Figure 5.82 displays the broader physical representation provided by the skeleton. The flow is the block connectivity that is witnessed as simple curves in the skeleton. In the case of the circle with a hyper quad topology, the skeleton shows one circular loop and four radial like lines. The grids can be viewed as a collection of outlines or skeletons. However, only one of these display forms can be used at a time to avoid screen clutter.



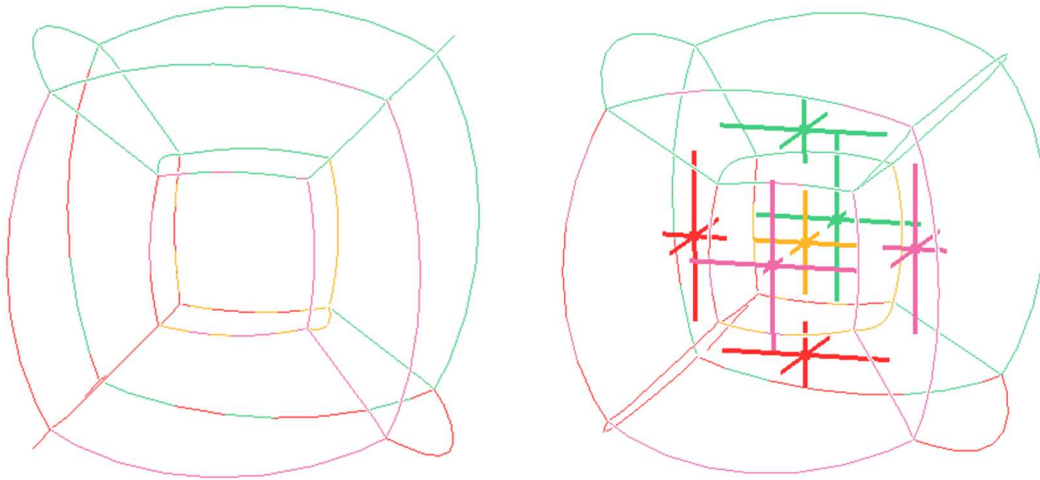


Figure 5.82: Blocks outline and skeleton of a sphere

Grid skeleton will appear whenever the user tries to trim the blocks. If add or subtract or intersection in **Trim**→**Blocks** operation is selected, the skeleton will appear in addition to the outline. Once trimming blocks operation is done, the skeleton will disappear.

All the operations under **Trim**→**Blocks** deal with activating and deactivating blocks. To deactivate a block, deactivate the skeletal centre of the block. By default, all the blocks will be on. If the block's centre is deactivated, the outline of the block will not be shown. Visually, the cutout blocks will allow the user to view the inside region of (3-dimensional) the grids as solid objects or to view grid slices that do not propagate beyond the active blocks. If a block skeletal centre is a filled, coloured square, the centre is activated. If a centre is a hollow, coloured square, the centre is deactivated. The **Mesh** button places a mesh on the exterior boundaries of the activated blocks. The result of pressing **Mesh** is the display of the active blocks as if they were solid objects. The propagation is limited to the active blocks.

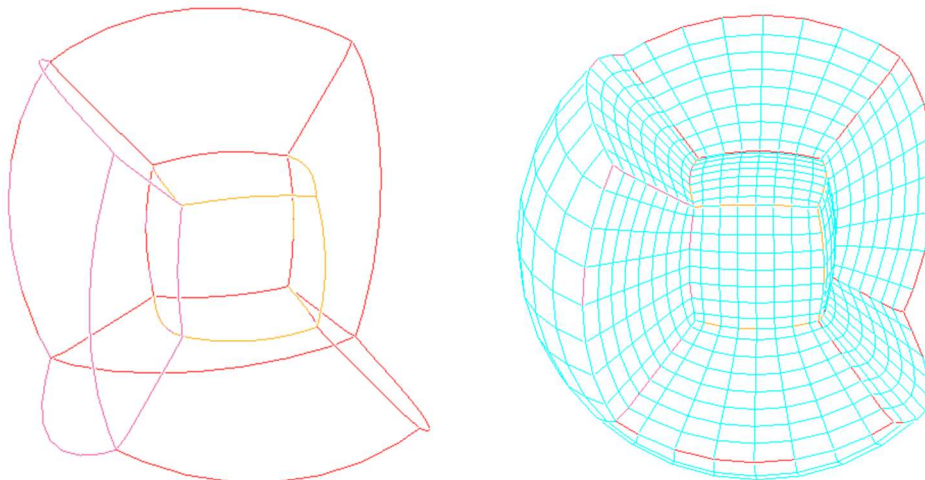













Figure 5.83: Trimmed blocks and trimmed blocks with grid cells



Because most grids will contain numerous blocks, the user will often need to activate or deactivate many blocks at a time. If block by block, activation or deactivation is required, depress the  button (first button located in Trim panel) and click on whatever centres have to be activated or deactivated. The  button is an addition operation, which, when depressed it allows the user to create a purple box with the right mouse button and enclose the box around all the centres that have to be activated. The  button complements the  button; When depressed, the  button allows the user to create a purple box with the right mouse button and enclose the box around all centres that have to be deactivated. The  button is an intersection operation. Like the  and  operations, the  operation allows the user to create a purple box and enclose it around centres. Any centre which is not enclosed by this box will be deactivated. Activated centres enclosed by this box remain activated, and deactivated centres enclosed by this box remain deactivated. The  button, when depressed, simply activates all the centres. The  button, when pressed, activates all the deactivated centres and deactivates all the active centres.

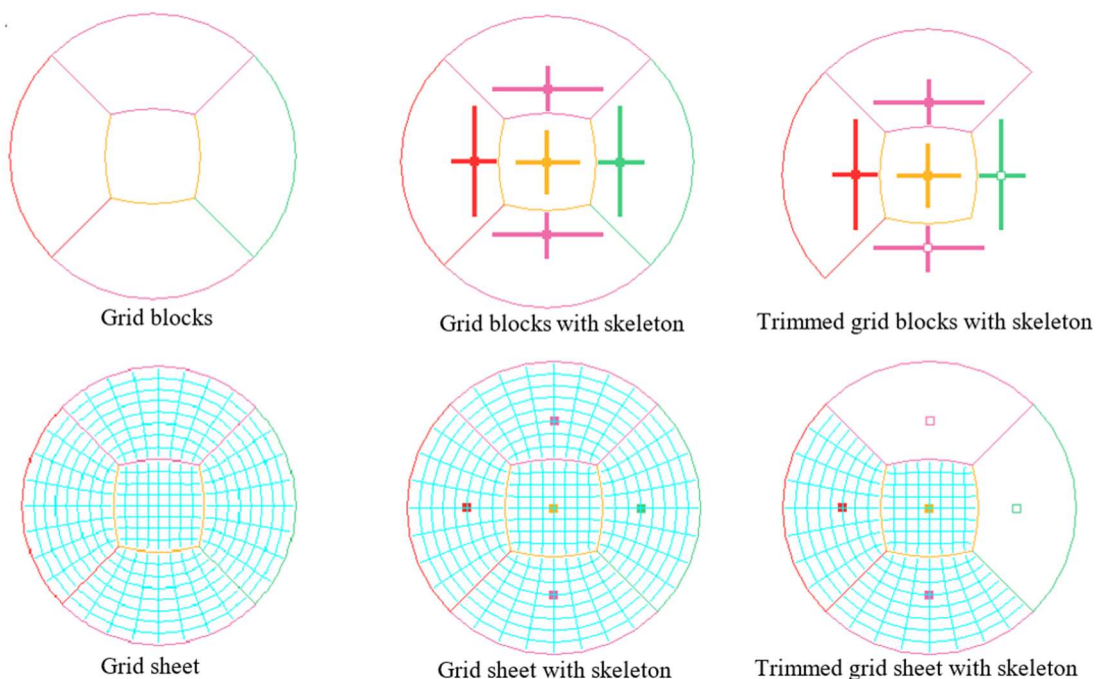




Figure 5.84: Grid trimming

## 5.7.2 Sheets

Like blocks, ***Sheets*** can also be activated and deactivated. Also, like blocks, grid sheets can be activated and deactivated using the available buttons, as shown in Figure 5.85. When the trim mode is set to ***Sheets***, all the trimming buttons work for grid sheets, just like how it works for the blocks when the trim mode is set to ***Blocks***. Before the buttons in the Sheets tab are used, however, a grid sheet must be already created; the buttons in the ***Sheets*** tab can manipulate only the faces on the *current* grid sheet. Similar to the  button for trimming blocks one by one, the  button will allow the user to trim the faces of each block one by one by picking the face centres of the given sheets. The other buttons are the same and work the same way for the sheet faces instead of blocks depending on the mode set in the trim section.

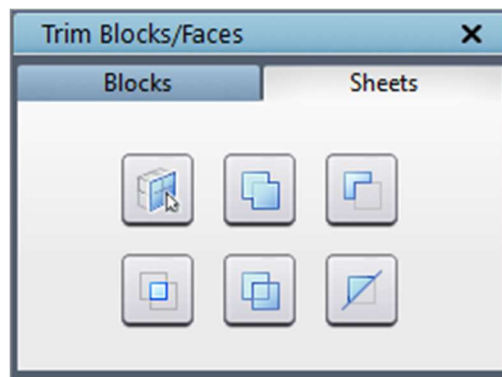


Figure 5.85: Sheets tab dialog box

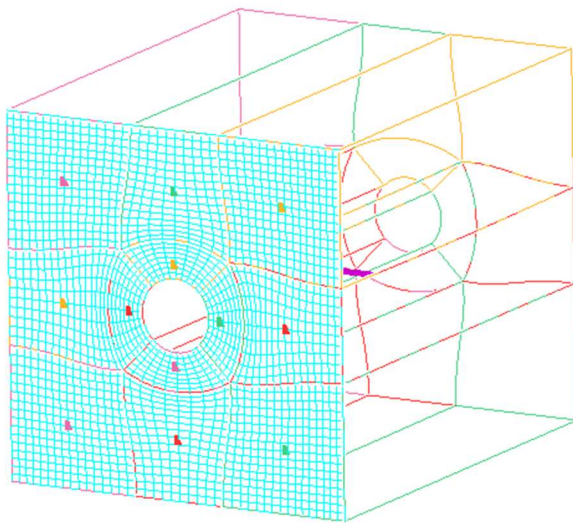


Figure 5.87: Skeleton for the grid sheet

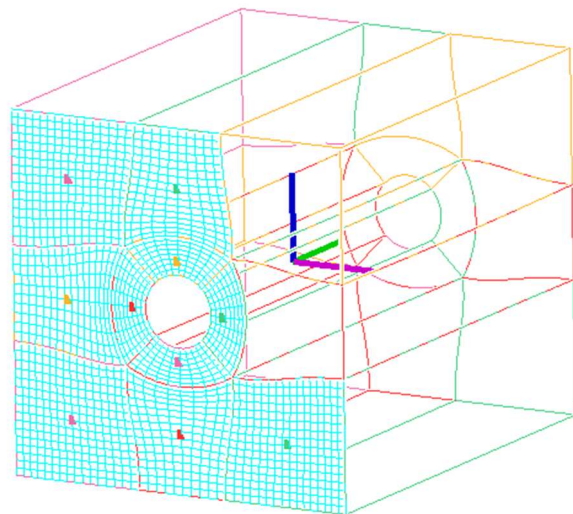


Figure 5.86: After trimming two faces of the grid sheet

# CHAPTER 6

## PROPERTY

Before exporting grids for CFD solvers, you may want to set properties/boundary conditions on these grids. You can set properties in two different ways by either setting it on the surface, which will be reflected in the grid or directly setting it on the grid once it is generated. *GridPro* enables you to set two dimensional and three-dimensional properties. That is, you can assign properties to each face of each grid block and to the volume of each grid block itself. You can set properties in the **Property** tab. Figure 6.1 illustrates the operations associated with the **Property** tab.

Property tab allows you to assign different properties to a grid based on the solver selected under the **Solver** menu in the top menu bar. Each solver has a separate list of properties. Depending on the selected solver, the corresponding property list can be seen in the property dropdown list. Properties are assigned to faces and blocks separately.

Additionally, the **Property** tab contains operations similar to those in the **Grid** tab. The user should be aware of what each operation does and what each operation displays in the drawing area.

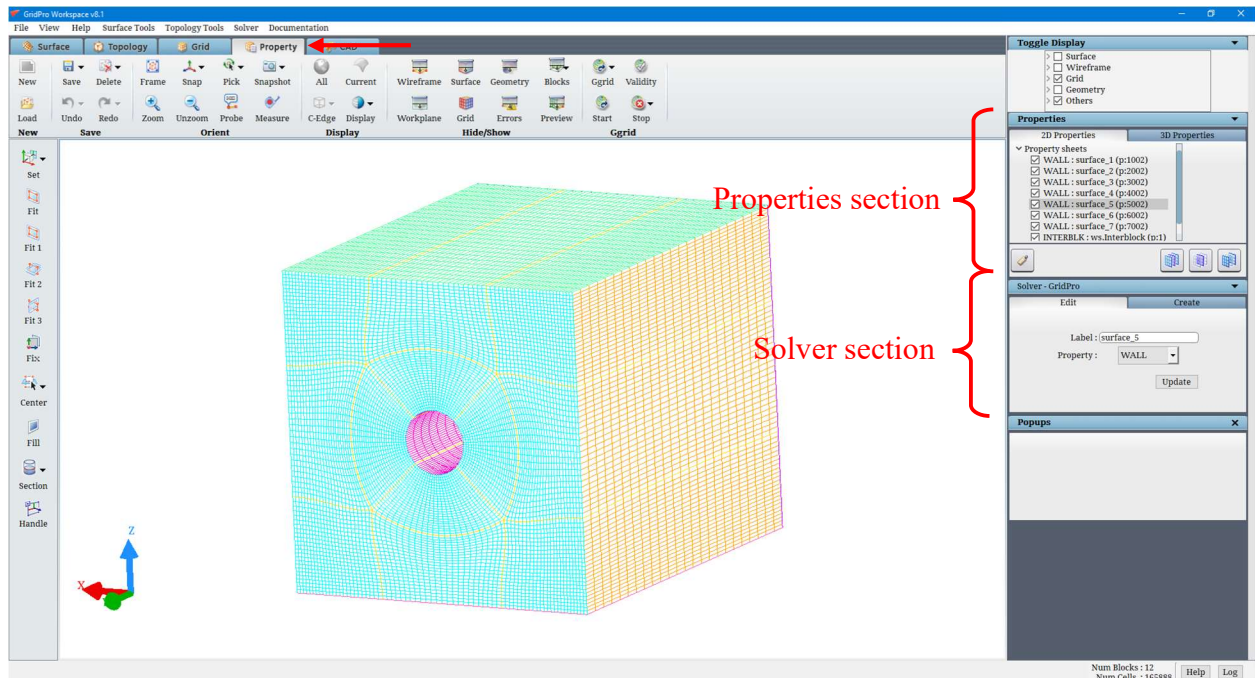


Figure 6.1: Property tab

## 6.1 Properties Section

In the *properties* section, which is located below *the toggle display* section in the **Properties** tab on the right-side panel, the properties assigned to the grid sheets and grid blocks can be viewed. The **Properties** section is split into two tabs, i.e. 2D and 3D properties, as shown in Figure 6.2 & Figure 6.3.

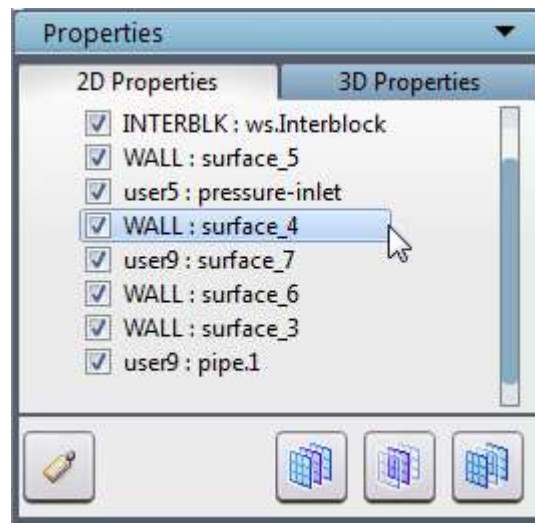


Figure 6.2: 2D properties tab

In the **2D properties** tab, assigned property sheets are shown, and in the **3D properties** tab, assigned property blocks are shown. It also gives access to toggle on/off the property. By default,

each grid has two 2D properties: interblock (INTERBLK) and boundary (WALL) and one 3D property, i.e. Fluid, independent of solver format. 2D properties are listed based on the number of surfaces assigned in the topology, and in addition to that, interior block cells property is also listed in it.

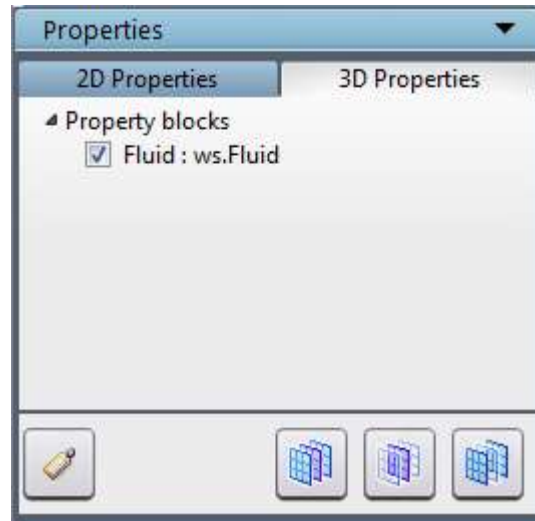


Figure 6.3: 3D properties tab

There are two names for every property sheet in Figures 6.2 & 6.3. The first one defines the property of the grid sheet, and the second one defines the label/surface ID/default name.

Depending on the tab selected, the user can assign the properties either to the grid sheets or grid blocks. If the **2D properties** tab is selected, properties are assigned to grid sheets, and in the **3D properties** tab, properties are assigned to grid blocks.

In the **2D properties** tab, only grid sheets will be visible and in **the 3D properties** tab, only blocks will be visible. Similar to toggling between grid sheets in the **Grid** tab, you can toggle between the property sheets.

The property name, highlighted with blue shade, is the *current* property displayed as a sea blue colour grid sheet in the graphics window. Any property can be selected as the *current* property by clicking on that property name. To hide a property, uncheck the box before the property name. If the box is checked, the property will be visible.



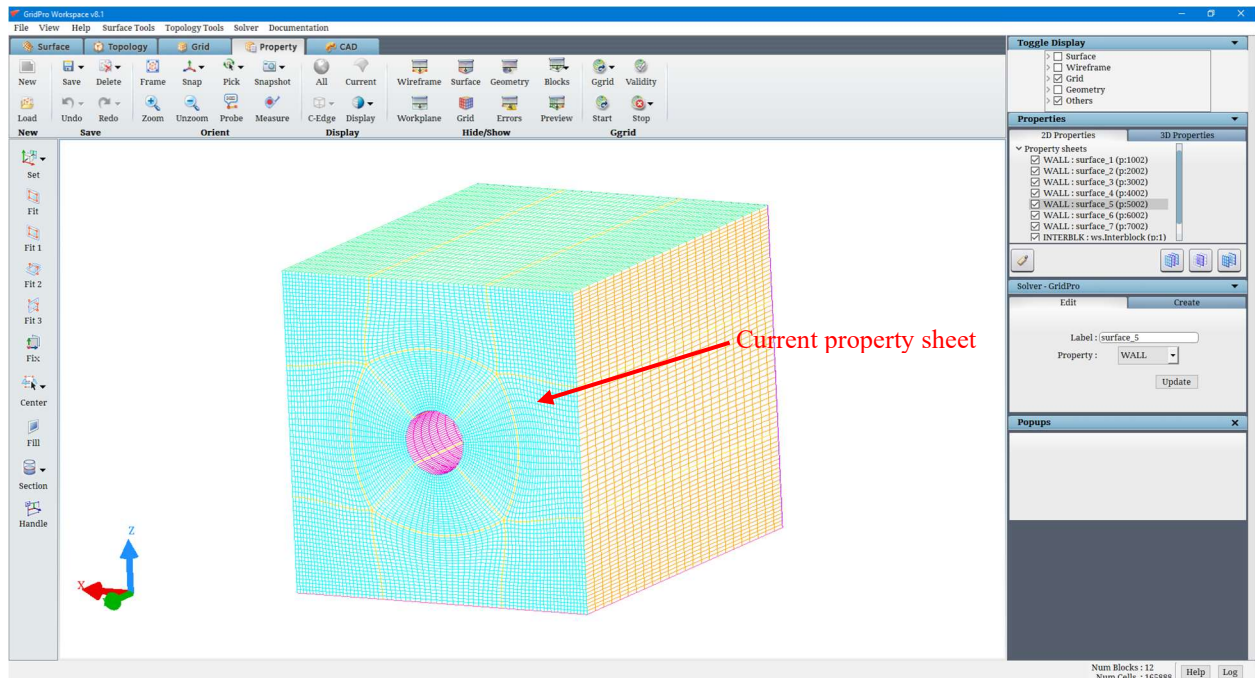


Figure 6.4: 2D properties section with Inlet as current property

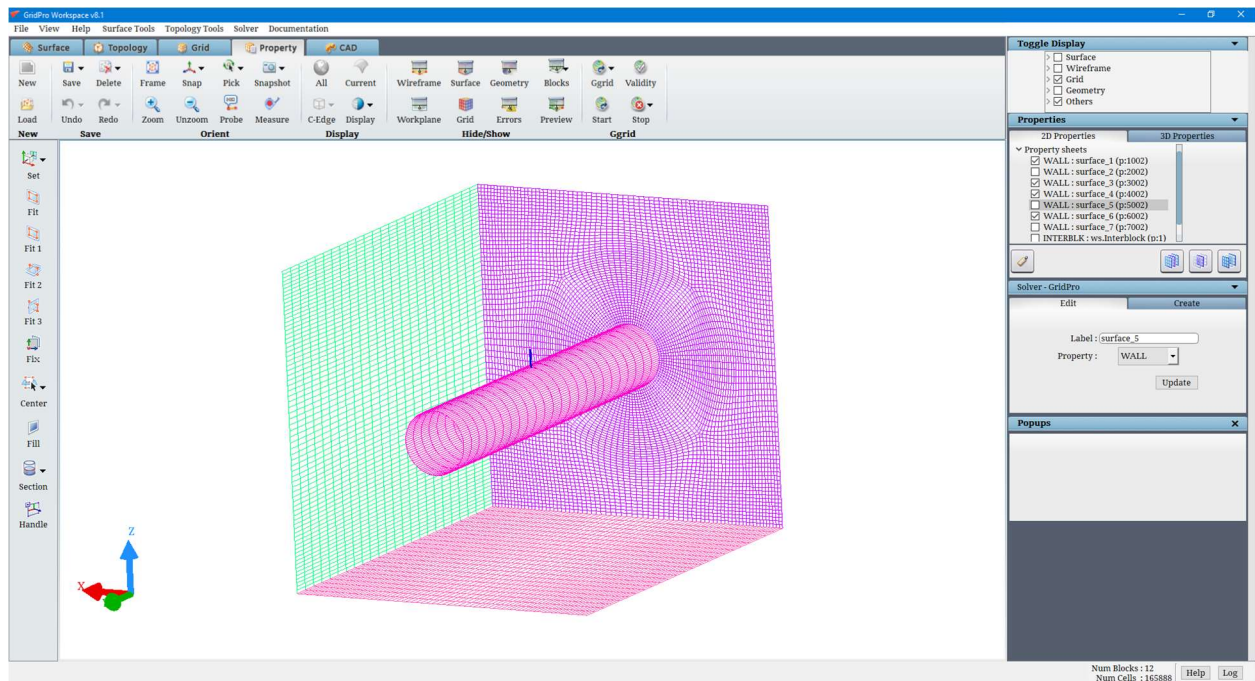



Figure 6.5: Property sheets switched off in the 2D properties section

The first button located at the bottom on the right side of the *Properties* section, i.e. **Show All Property**  button, will switch on the visibility of all properties (sheets/blocks) in the list respective to the tab.



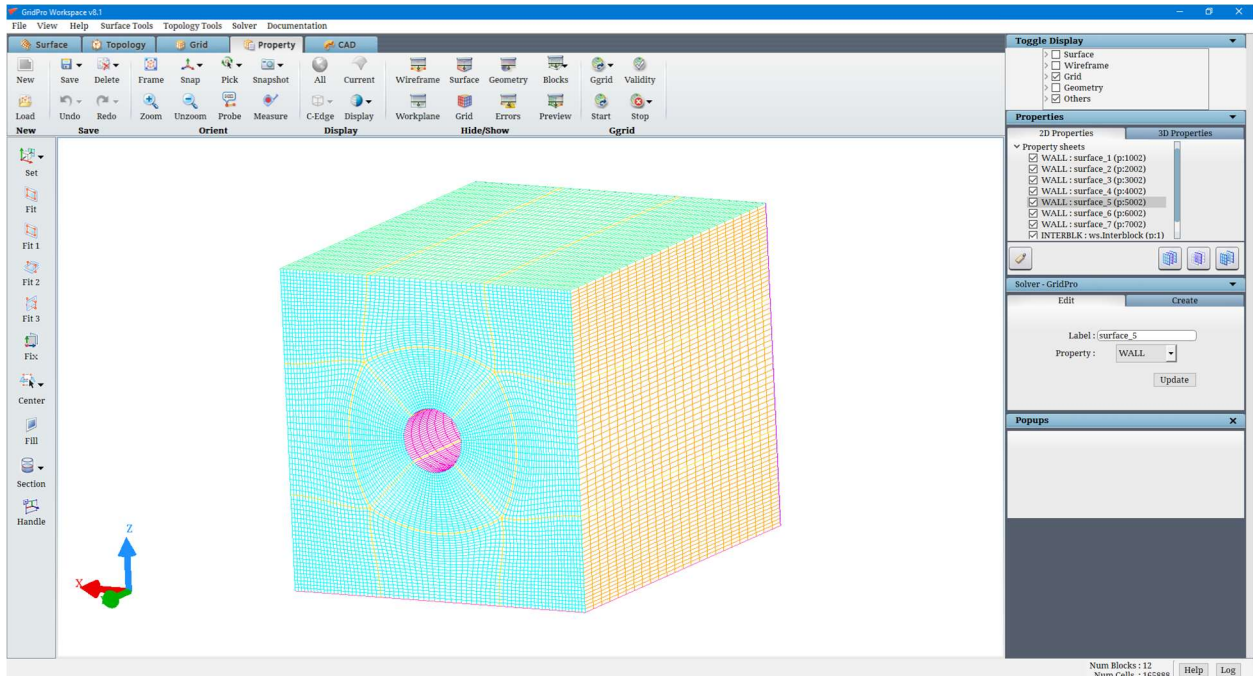


Figure 6.6: After clicking on Show all property button

The second button, i.e. **Show only Current**  button, will hide all other properties except current property.

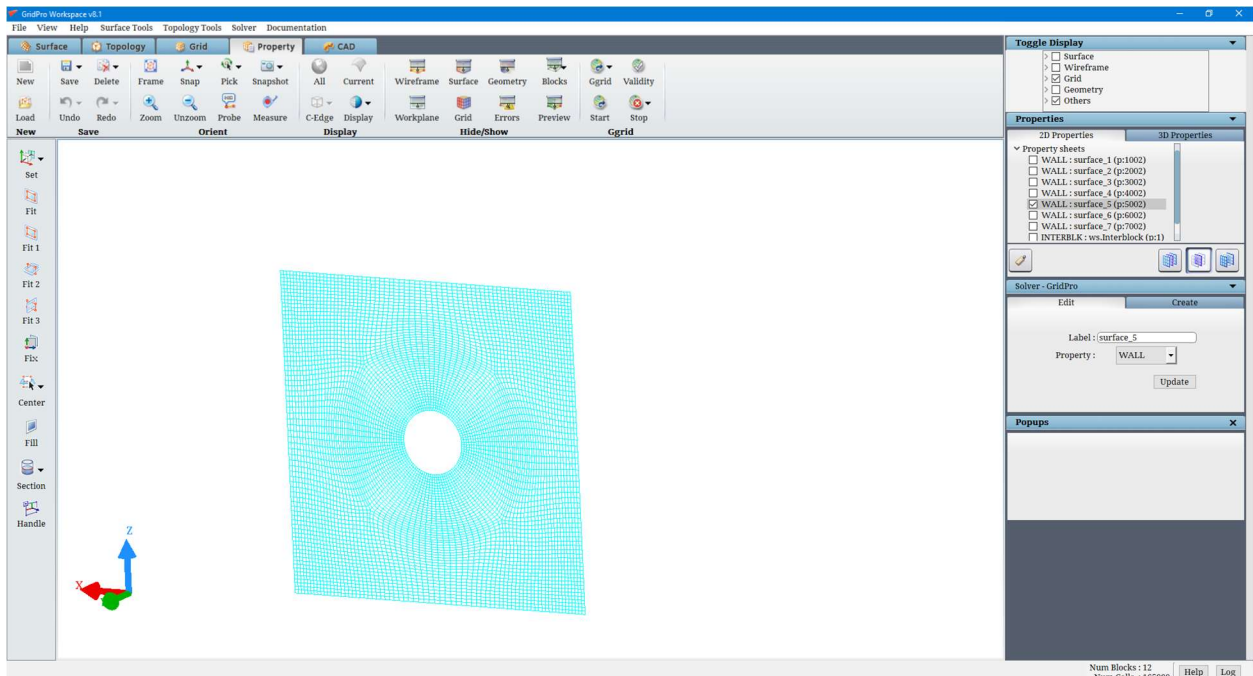


Figure 6.7: After clicking on Show only current property button

The third button, i.e. **Toggle All Property**  button, will flip the visibility of all the properties in that tab.

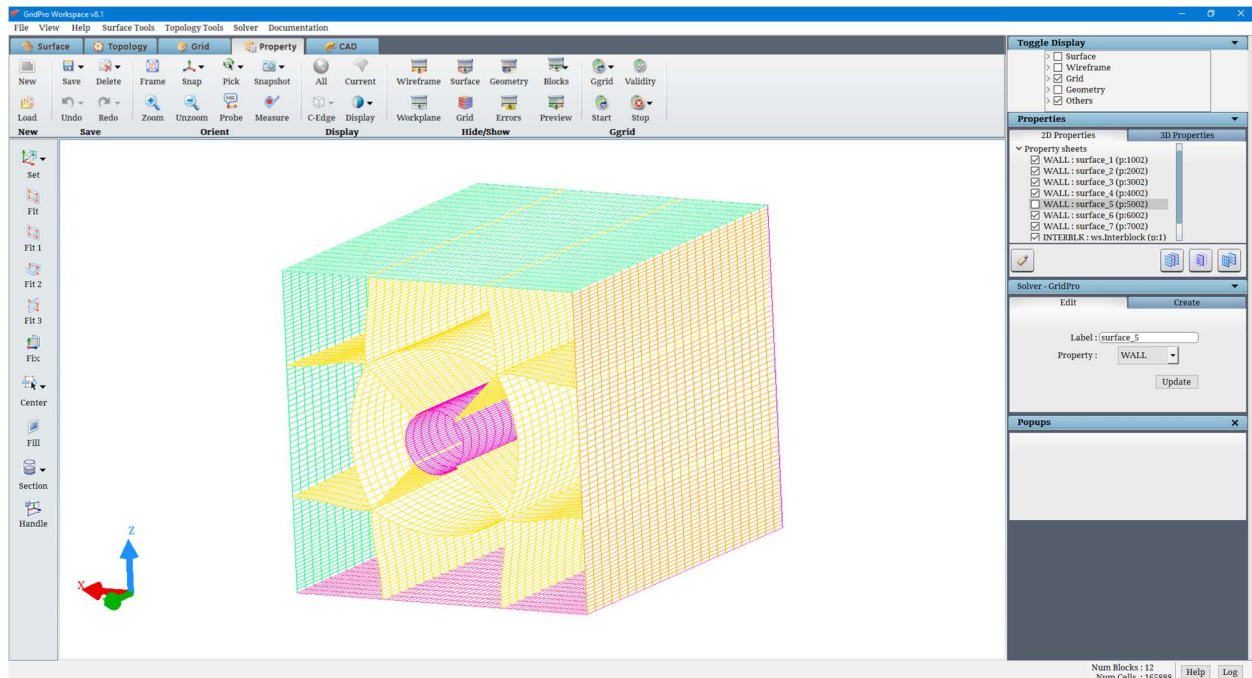


Figure 6.8: After clicking on Toggle all property button

In the **3D properties** tab, all these buttons work similarly for the grid blocks. All actions are independent of other tabs, i.e. actions performed for the **2D properties** tab will not affect the visibility of properties in the **3D properties** tab and vice versa.

The button on the left side bottom is called '**Labels**'. You can tag the properties using surface Ids. Each surface sheet will be tagged using its ID, and a default property will be assigned to it. Clicking on **Labels** will open a confirmation dialog box, as shown in Figure 6.9, where the user has to confirm the operation.

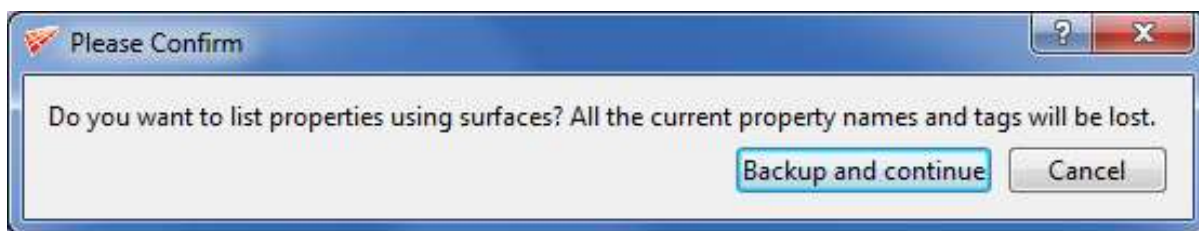


Figure 6.9: Labels Confirmation dialog box

Clicking on 'Backup and continue' will backup the existing property file. For e.g., blk.tmp.pty will be backed up to "blk.tmp.pty.~1~". Up to four files can be backed up in any working directory; beyond that, the four files will be overwritten one after the other, starting from 1.

## 6.2 Solver Section

Below the **Properties** section, the **Solver** section is located, which goes hand in hand with the **Properties** section. First, a CFD solver format has to be selected. Various solver formats are available in the dropdown of the **Solver** menu in the menu bar, as shown in Figure 6.10. By default, the GridPro format is selected.

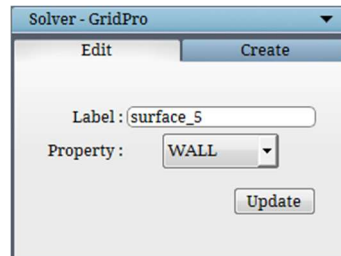


Figure 6.11: Solver section

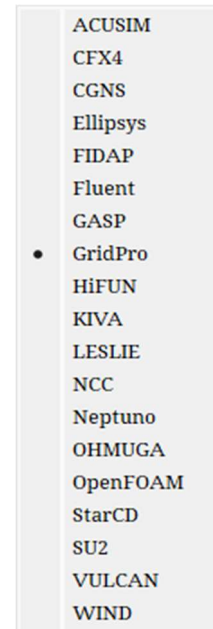


Figure 6.10: List of solver formats

Each format has unique properties. The list of properties for each solver can be viewed in the field next to **the property** in the **solver** section. Figure 6.12 provides an example of properties listed under the property dropdown list.

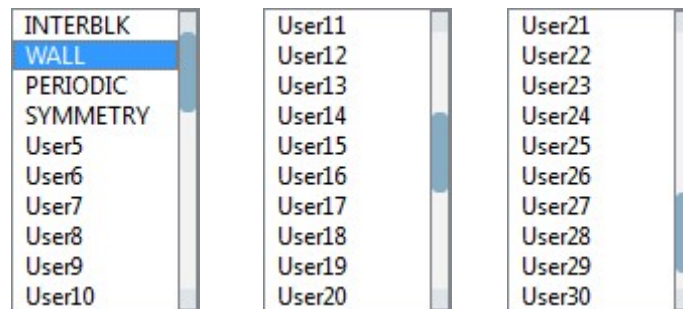


Figure 6.12: List of properties for the GridPro format

The **Property** tab makes extensive use of grid skeletons for efficient property assignment. As previously mentioned, the skeleton of a block is the lines connecting the indexed centre of the block with the centres of the block's bounding edges (2D) or faces (3D). Each block skeleton is drawn in its own colour and represents the minimal amount of information required to show a broad representation of a multi-block grid.

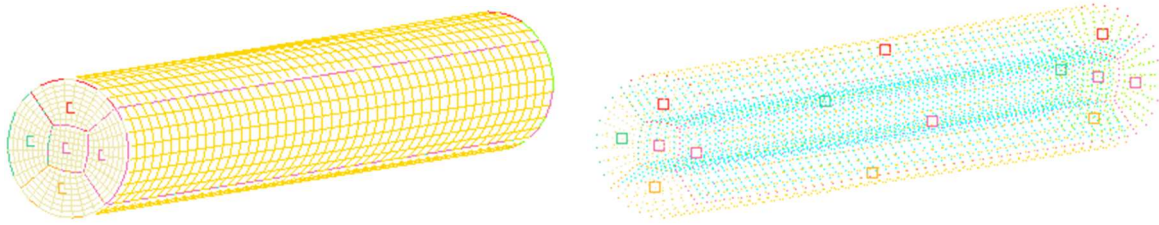


Figure 6.13: Block skeleton

Switch to the **2D properties** tab in the property section after selecting the desired solver format in the ***solver*** menu. Remember, in the **2D properties** tab; a property can be assigned to each face of each grid sheet, even if the faces intersect.

The property name highlighted in the ***Property*** section can be modified in the ***Edit*** tab of the ***Solver*** section, as shown in Figure 6.14. You can update the desired label and change the property of the Current highlighted property in the ***Edit*** tab. You can update the changes by clicking the ***Update*** button in the ***Edit*** tab. The newly updated changes can be viewed in the property list.

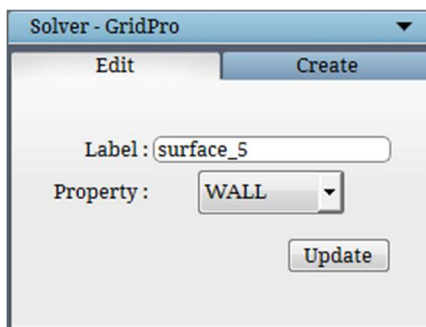






Figure 6.14: Edit tab






Figure 6.15: Create tab

New properties can be created using the options available in the ***Create*** tab under the ***Solver*** Section, as shown in Figure 6.15.

Select the property to be assigned in the property field in the ***Create*** tab of the Solver section. Give the desired label for the property. The four buttons in the selection row (the ***Create*** , ***toggle*** , ***add*** , ***subtract***  buttons) helps in assigning the new properties. This row has operations relevant to the activation and display of the skeleton of faces. If the face is hidden, the centre of the face skeleton will be deactivated. Conversely, if the face skeleton centre has to be activated, the face must be displayed first. First, select the desired property, add its corresponding label, and click on the ***Apply*** button to add it to the property list. The newly added property becomes the *current* property. Assigning the *current* property to a face involves the centre of that face's skeleton.



Depressing the  button will display the skeletons of all the faces and allow the user to assign/un-assign the current property to any one of the faces by clicking (with the left mouse button) on the centre of the face's skeleton. Assigned face centres will be displayed as solid blocks, and unassigned face centres will be displayed as transparent blocks. Depressing the  button will display the skeletons of all faces which do not contain the current property and allow the user to assign the current property to many faces by dragging a red box (with the right mouse button) over their skeletons centres. Depressing the  button will display the skeletons of all faces which contain the current property and allow the user to un-assign the current property from many faces by dragging a red box over their skeletons centres. Once the property is unassigned from the faces, the un-assigned faces will automatically be added to the **wall** property. When one button is depressed, the others un-depress.

The process of assigning properties to blocks is identical to that of assigning properties to faces. The **2D properties** tab has to be switched to the **3D Properties** tab. The chosen property format determines the list of 3D properties displayed when the dropdown next to **properties** in the **solver** section is clicked.

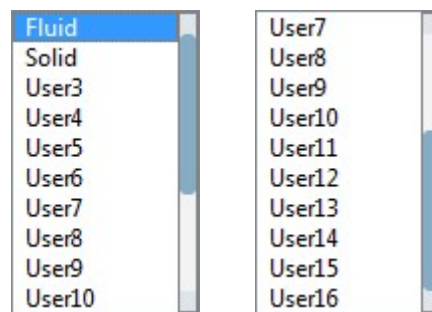


Figure 6.16: 3D Property list of PDC format

- **Right-click menu**

When you *right-click* on the edge of a property block, a menu appears similar to the right-click menu of an edge of grid block in the Grid tab, as shown in Figure 6.17. All the options are discussed under section 5.1.3.

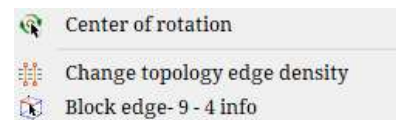


Figure 6.17: Right-click menu of grid blocks

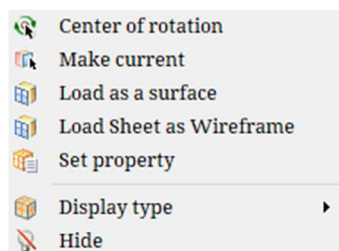


Figure 6.18: Right-click menu of grid sheets

Similarly, the right-click menu for the property sheet is similar to the right-click menu in the grid tab, as shown in Figure 6.18. All the options except **Set Property** are explained in section 5.1.3. The Set Property option can set the property on any grid sheet defined by the block boundary. Once selected, it will pop up a box with a list of all properties available for the selected solver, as shown in Figure 6.19.

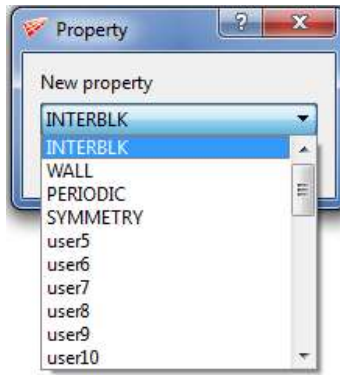


Figure 6.19: Set property dialog box

Create 2D/3D Property	Edit 2D/3D Property
<ol style="list-style-type: none"> <li>1. Select the desired Property from the dropdown</li> <li>2. Enter the Tag/Label for the selected property</li> <li>3. Click 'Apply' to create the selected property</li> <li>4. Use the selection buttons to assign the grid sheet to the selected property</li> </ol>	<ol style="list-style-type: none"> <li>1. Select the property to be edited as current</li> <li>2. Modify the Tags/Property as desired</li> <li>3. Click 'Update' to update the modifications</li> </ol> <p>Note: To add new grid sheets to the existing property → Select the property, Go to Create tab and use the selection buttons.</p>

Table 6.1: Steps to create/edit a property

## 6.3 File format

In the new UI, WS, the way the property file is written has been modified considerably. To start with, the old file format does not have the concept of property labels. Property labels help in identifying the grid sheets in the respective solvers. For e.g., many grid sheets will have the same boundary conditions, so property labels were introduced recently to differentiate each.

Another significant modification is the property Id. In the old file format, it used to assign 1 to n. But in the new file format, in addition to old property Id's, new property Id's are introduced, such as 1002, 2002, etc. This is primarily used when the property labels are defined in terms of the surface Ids.

Please refer to chapter 7 in the User\_Guide\_for\_TIL manual for more information on how the block information is written.



```

2 Blocks
#B bid b-pty b-lbid f_i-pty f_i-lbid f_l-pty f_l-lbid ...
B 1 1 -1 2002 -1 1 -1 5002 -1 4002 -1 1 -1 6002 -1
B 2 1 -1 1 -1 1 -1 5002 -1 4002 -1 1 -1 6002 -1
B 3 1 -1 1 -1 3002 -1 5002 -1 4002 -1 1 -1 6002 -1
B 4 1 -1 5002 -1 4002 -1 1 -1 2002 -1 1 -1 1 -1
B 5 1 -1 1 -1 2002 -1 5002 -1 4002 -1 1 -1 7002 -1
B 6 1 -1 4002 -1 5002 -1 1 -1 3002 -1 1 -1 1 -1
B 7 1 -1 1 -1 1 -1 5002 -1 4002 -1 1 -1 7002 -1
B 8 1 -1 3002 -1 1 -1 5002 -1 4002 -1 1 -1 7002 -1
B 9 1 -1 5002 -1 4002 -1 1 -1 1 -1 1002 -1 1 -1
B 10 1 -1 1 -1 1 -1 5002 -1 4002 -1 1002 -1 1 -1
B 11 1 -1 1 -1 1 -1 5002 -1 4002 -1 1002 -1 1 -1
B 12 1 -1 4002 -1 5002 -1 1 -1 1 -1 1002 -1 1 -1
0 labels
10 0 2D properties (used, max-id)
0 unused (pdc:DEFAULT)
1 ws.Interblock (pdc:INTERBLK)
2 ws.Wall (pdc:WALL)
1002 surface_1 (pdc:WALL)
2002 surface_2 (pdc:WALL)
3002 surface_3 (pdc:WALL)
4002 surface_4 (pdc:WALL)
5002 surface_5 (pdc:WALL)
6002 surface_6 (pdc:WALL)
7002 surface_7 (pdc:WALL)
2 0 3D properties (used, max-id)
0 unused (pdc:DEFAULT)
1 ws.Fluid (pdc:Fluid)

```

1002  
2002  
3002  
4002 → Property id's  
5002  
6002  
7002

surface\_1  
surface\_2  
surface\_3  
surface\_4 → Property labels  
surface\_5  
surface\_6  
surface\_7



(pdc:WALL)  
(pdc:WALL)  
(pdc:WALL)  
(pdc:WALL) → Boundary conditions/Properties  
(pdc:WALL)  
(pdc:WALL)  
(pdc:WALL)

Figure 6.20: Property file format

# CHAPTER 7

## CAD

CAD tab has the ability to view, repair and mesh the geometry. There are two sections that help in achieving these capabilities. Figure 7.1 illustrates the operations associated with the **CAD** tab.

CAD tab allows you to import geometries in IGES, STEP, Brep and Linear formats. Once it is imported, the geometric faces associated with the geometry are loaded in the ***Geometry Faces*** section. The faces can be toggled using the toggle button  provided. You can also view the face/body information using the  button.

Under the Tools section, you are provided with the tools to create, repair and mesh the geometry. Section 7.2 explains the tools in detail.

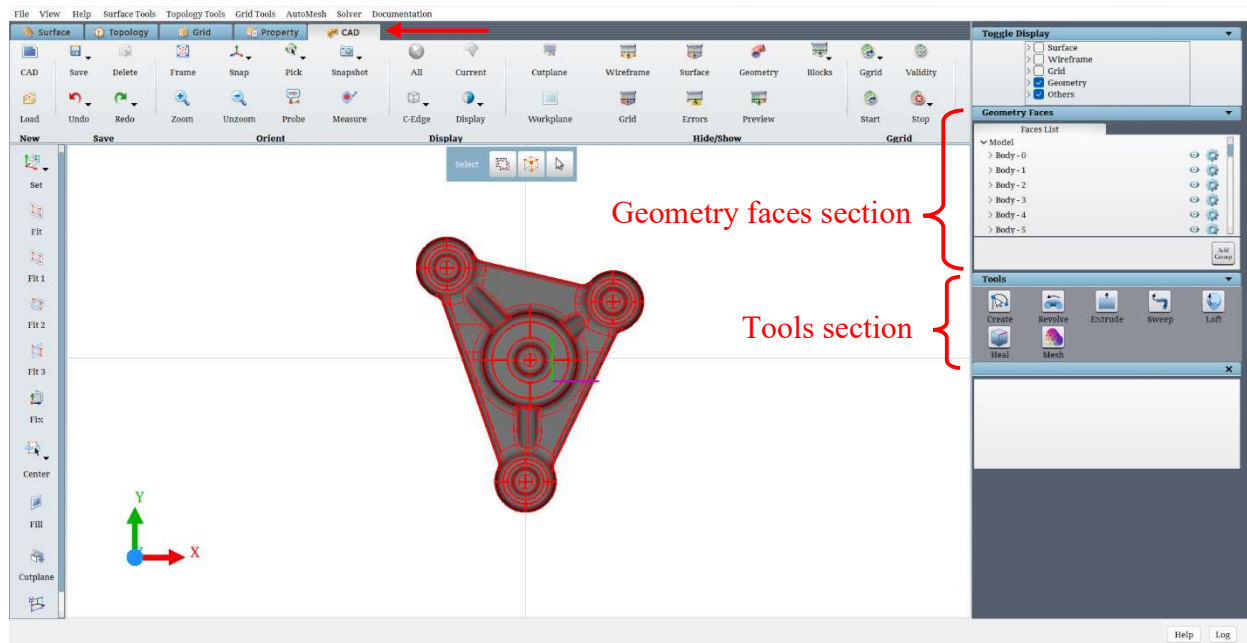




Figure 7.1: CAD tab

## 7.1 Geometry Faces

In the *geometric faces* section, which is located below *the toggle display* section in the CAD tab on the right-side panel, the individual body and face definitions of the geometry can be viewed. Figure 7.2 shows the Geometric Faces section.



Figure 7.2: Geometry Faces section

To hide a face/body, click on the toggle button  next to the face/body name. The button icon will subsequently change to , and the face/body becomes hidden. Clicking on the toggle button again shows the unhides the face/body.

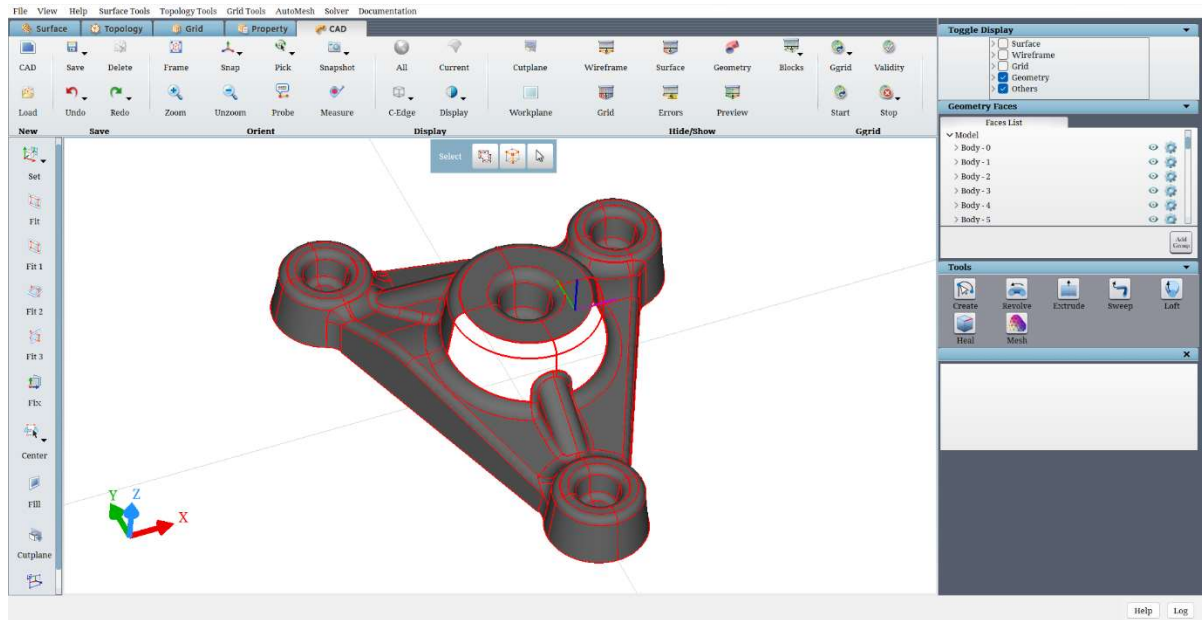




Figure 7.3: After a few of the Geometry faces are switched off

The **Add new face group**  button located on the bottom right in the Geometric Faces section, is used to group the CAD faces. The grouped CAD faces can be automatically meshed and imported as a single surface in GridPro. To group the faces, click on the **Selection button**  and choose the faces to be grouped. A custom **Group Name** can also be given. Clicking on **Create** creates the group of faces.

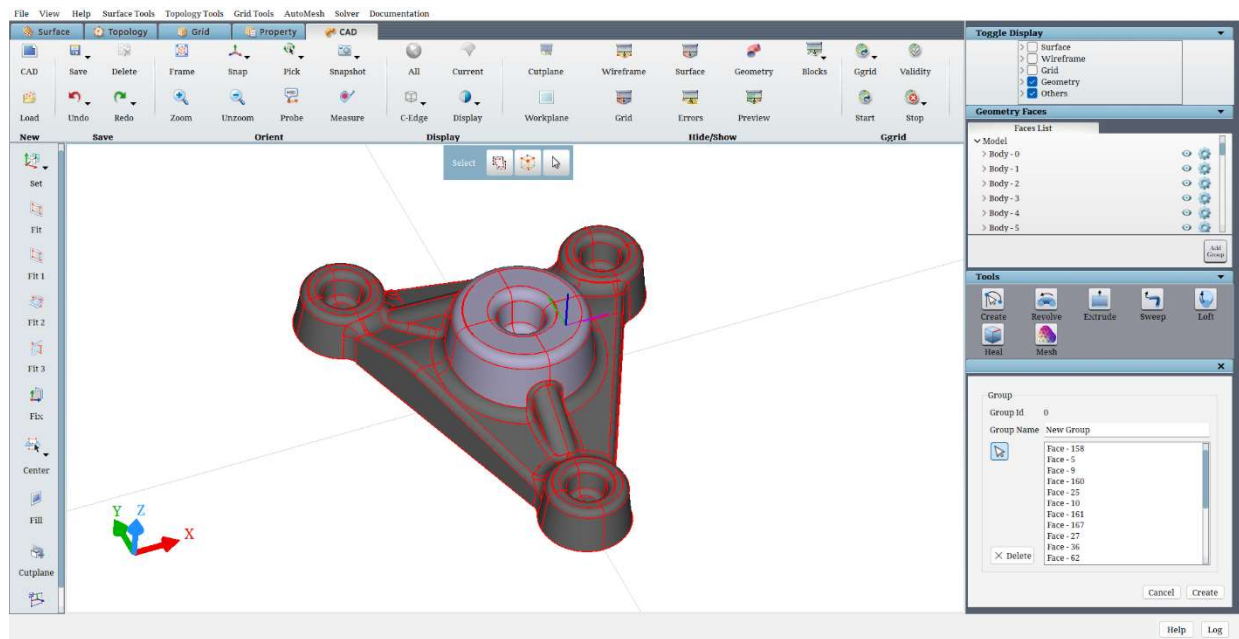



Figure 7.4: Grouping the CAD faces

## 7.2 Tools Section

There are seven buttons provided under the Tools section; **Create**, **Revolve**, **Extrude**, **Sweep**, **Loft**, **Heal** and **Mesh**. The different options are explained below.

### 7.2.1 Create

The **Create**  tool is used to create CAD features such as a **Node**, **Line** or **Arc**. try.

- **Node**

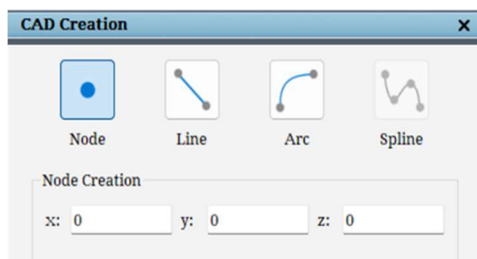



Figure 7.5: The Node creation dialog box

A **node** is a point where CAD **lines** or **arcs** intersect and connect. It is represented by a small, yellow coloured sphere. To create a node, only its cartesian coordinates are required.

To create a node, firstly, the **Node** option  is selected. Enter its x, y and z coordinates in the text boxes provided in the Node **Creation** section, and then click on **Create**. This should create a node which can be seen in the GL.

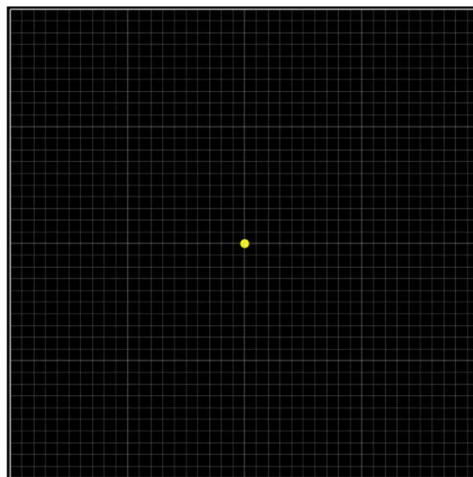




Figure 7.6: The created Node

- **Line**

The **line** tool is used to create a straight line segment between two nodes. It is represented by a green line joining the two nodes. To create a line, the **Line** option  is selected. The two nodes are selected using the respective **Selection** buttons  in the **Node Selection**

section. After clicking on the button and then clicking on the node in the GL, the node which is selected is specified in the dialog box, along with its coordinates. Clicking on **Create** creates the line in the GL. It should also be noted that the colour of the nodes changes from yellow to dark blue, indicating that the node is not a stray node.

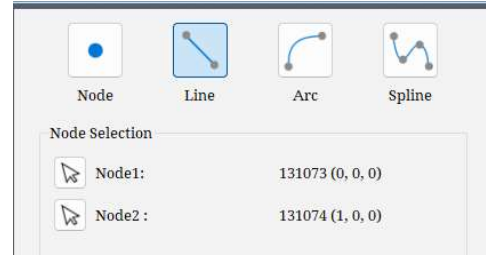


Figure 7.7: The Line creation dialog box

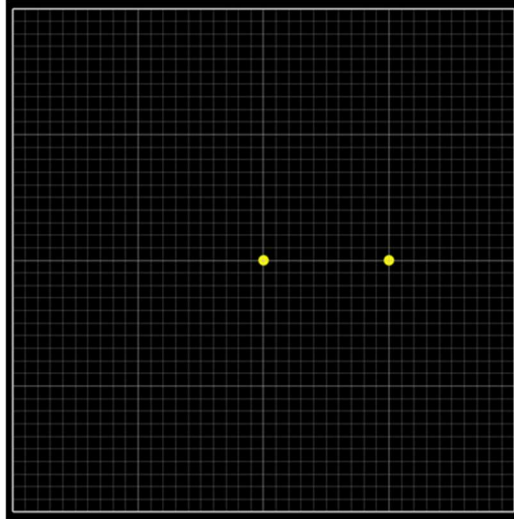


Figure 7.8: The Nodes before the line is created

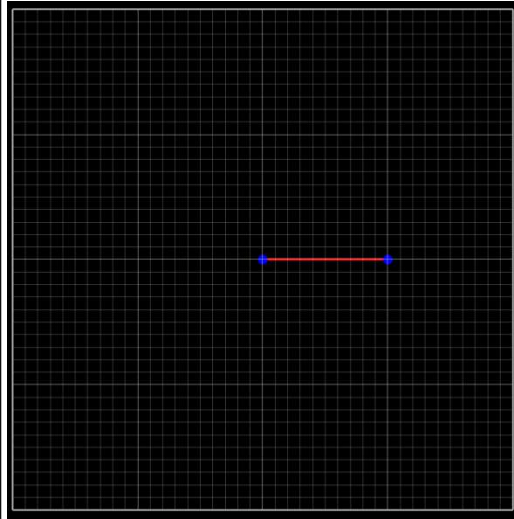



Figure 7.9: The Nodes after the line is created

- **Arc**

The **arc** command is used to create a curved line segment. An arc is a portion of a circle, and is represented by a green line similar to a **line**. To create an arc, the **Arc** option  is selected. An arc is created using its **Center**, which represents the center of the circle which it is a part of, and by its initial location, which is given by the **Initial Node**. Both the center node and initial node are selected

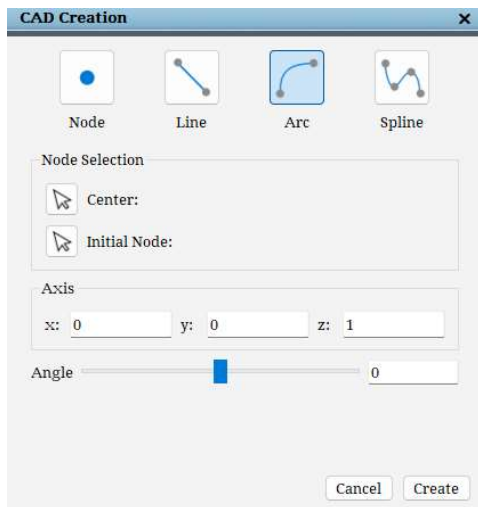



Figure 7.10: The Arc creation dialog box

using their respective **Selection** buttons . The axis of the arc must also be defined, which is done by specifying the direction of the axis in the **Axis** section of the dialog box. Once all these inputs have been provided, a preview of the arc to be created can be seen by dragging the **Angle** slider in either direction. Clicking on **Create** creates the previewed arc.



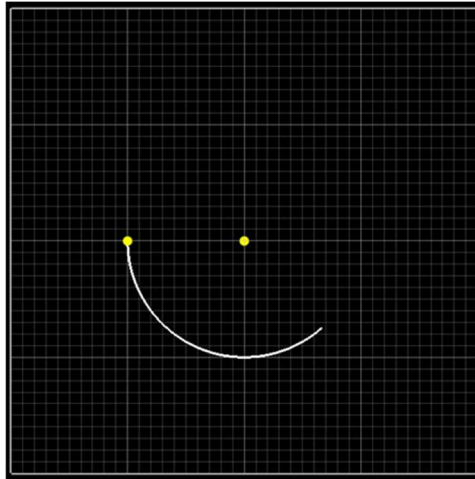


Figure 7.11: The preview of the arc

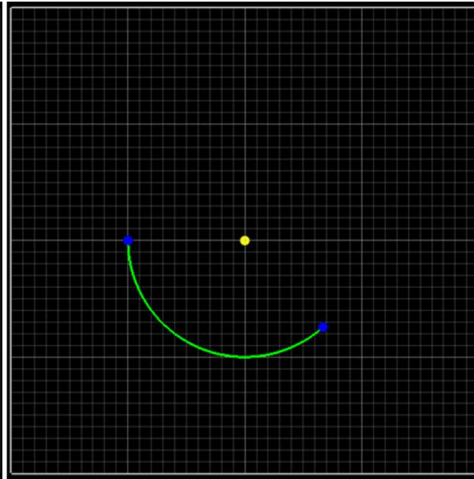


Figure 7.12: The created arc

## • Spline

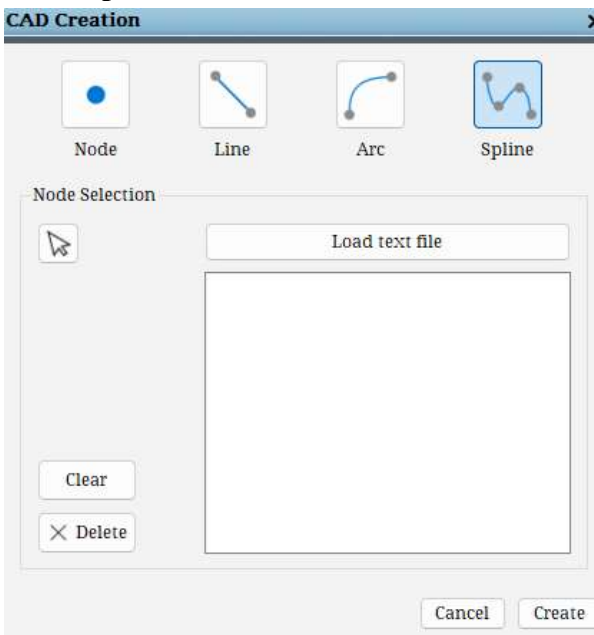



Figure 7.13: The Spline creation dialog box

The **spline** command is used to create a smooth curve passing through or close to a set of fixed nodes. To create the spline, select the **Spline**  option. Under the **Node Selection** section, the **Selection** button can be used to select the respective nodes through which the spline is going to be created. Alternatively, using the **Load text file** button, a .dat or .txt file containing the node coordinates can be loaded. Clicking on **Create** creates the spline. The **Clear** button deletes all the coordinates that have been selected, and the **Delete** button is used to delete individual nodes.

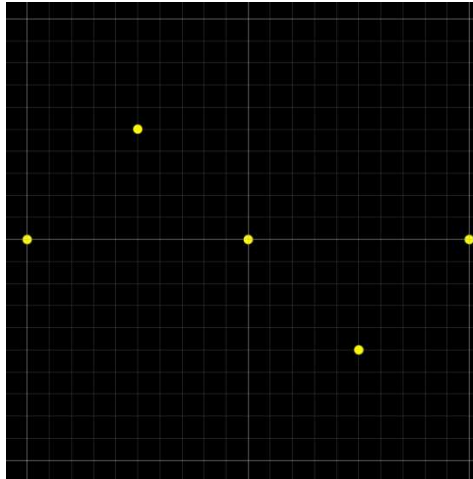


Figure 7.14: The nodes which will be used to create the spline

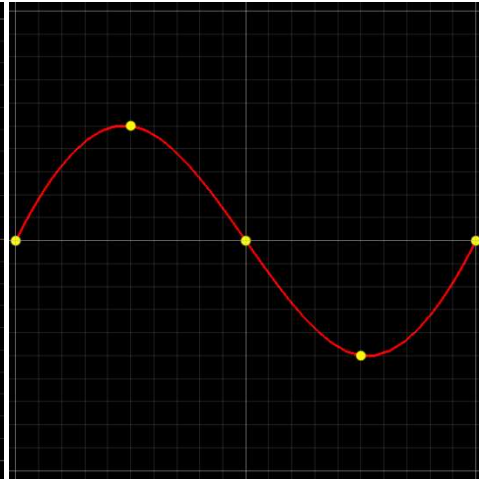



Figure 7.15: The created spline

## 7.2.2 Revolve

The **Revolve**  tool is used to create a face by revolving a curve about an axis through a predefined angle.

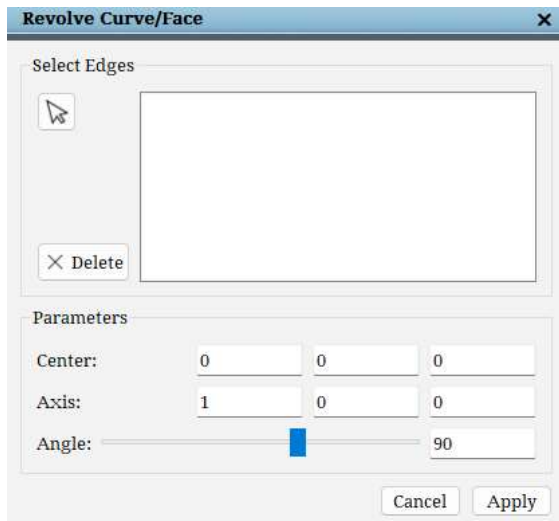



Figure 7.16: The Revolve dialog box

To revolve the curve, select the edge representing the curve by clicking on the **Selection** button  in the **Select Edges** section, and then selecting the curve. In case any edge has been selected incorrectly, click on the **Delete** button and click on the edge either on the GL or in the list of edges shown in the **Select Edges** section. The curve is revolved about a **Center** about a predefined **Axis**, through a particular **Angle**. The **Center** coordinates are entered in the **Parameters** section. Similarly, the direction of the **Axis** and the **Angle** is also entered here. Once the parameters are entered, a preview of

the revolve operation is shown. Clicking on **Apply** creates the revolved face.

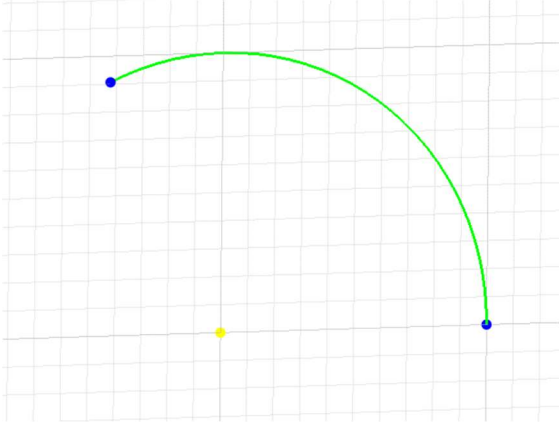


Figure 7.17: The curve which is revolved

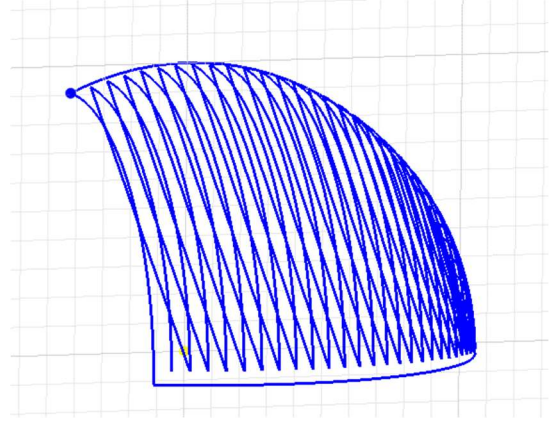


Figure 7.18: The preview of the Revolve operation

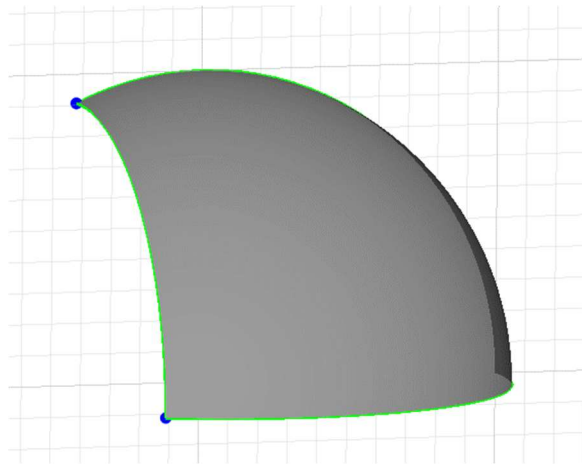




Figure 7.19: The face generated after using Revolve

### 7.2.3 Extrude

The **Extrude**  tool is used to create a Face or Body by extruding a Curve or Face along an axis.

To extrude the face or curve, it is first selected by clicking on the **Selection** button  in the **Select** section and then clicking on the face or curve to be extruded. After this, the **Distance** of extrusion is entered in the **Parameters** section. The **Axis** direction is also entered. A preview of the extrusion is shown to visualise what the extrusion would produce. Clicking on **Apply** creates the extruded surface.

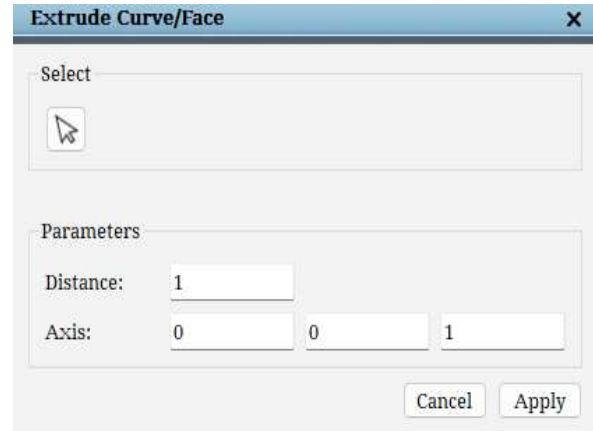


Figure 7.20: The Extrude dialog

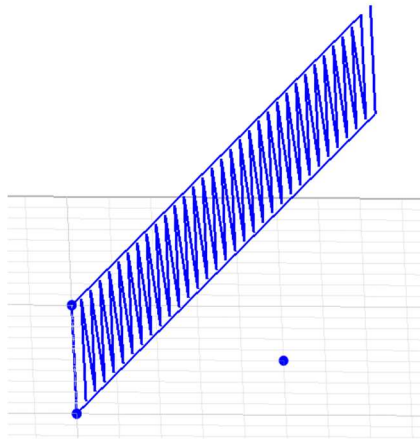


Figure 7.21: The preview of the face

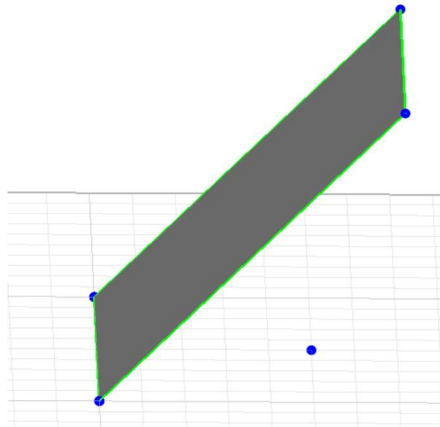


Figure 7.22: The face created after extrusion

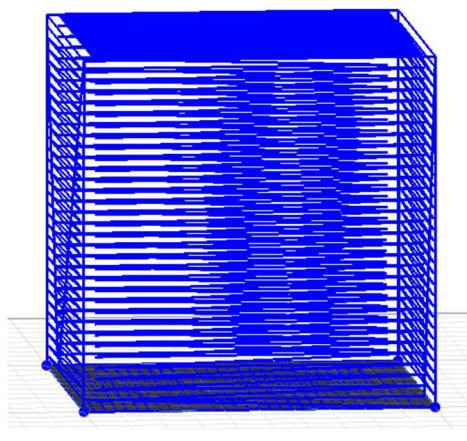


Figure 7.23: The preview of the body

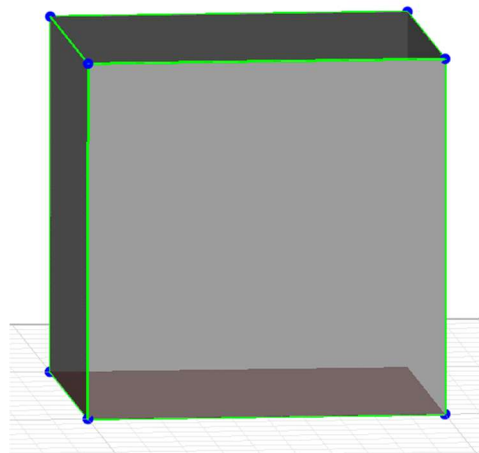




Figure 7.24: The body created after extrusion

## 7.2.4 Sweep

The **Sweep**  tool is used to create a Face or Body by sweeping a curve or Face along a single path or multiple paths. There are two tabs under the Sweep tool - **Spine** and **Rail**.

- **Spine**

Using the **Spine** tab, a Curve or Face **Source Object** can be swept along a spine curve to create an object.

To do this, first select the **Source Object** by clicking on the **Selection** button , after which the Source Object, which can be an edge or a face, is selected. The selected object is highlighted in white and the name of the object is also displayed in the dialog box. Similarly the **Spine** curve is selected. The Spine curve is a curve such that the swept Source Object always lies on a plane perpendicular to the Spine curve.

There are also several modes that can be used to perform the sweep. These are different mathematical algorithms that are used to perform the sweep. Once all the inputs have been provided, click on **Apply** to generate the swept body.

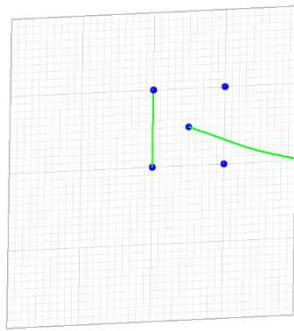


Figure 7.25: The curve to be swept along the spine

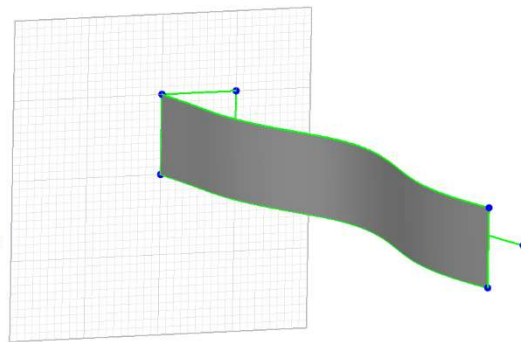


Figure 7.26: The face created after sweeping

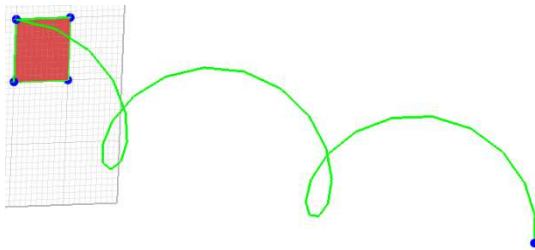


Figure 7.27: The curve to be swept along the spine

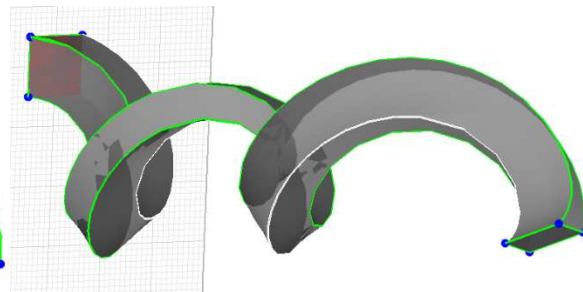



Figure 7.28: The curve to be swept along the spine

- **Rail**

Using the **Rail** tab, a **Profile Curve** is swept using two **Guide Curves**, to create an object. The **Profile Curve** may be a closed or open loop. The **Profile Curve**, **Guide Curve 1** and

**Guide Curve 2** are all selected using the **Selection** button . Once all the inputs have been provided, click on **Apply** to generate the swept body.

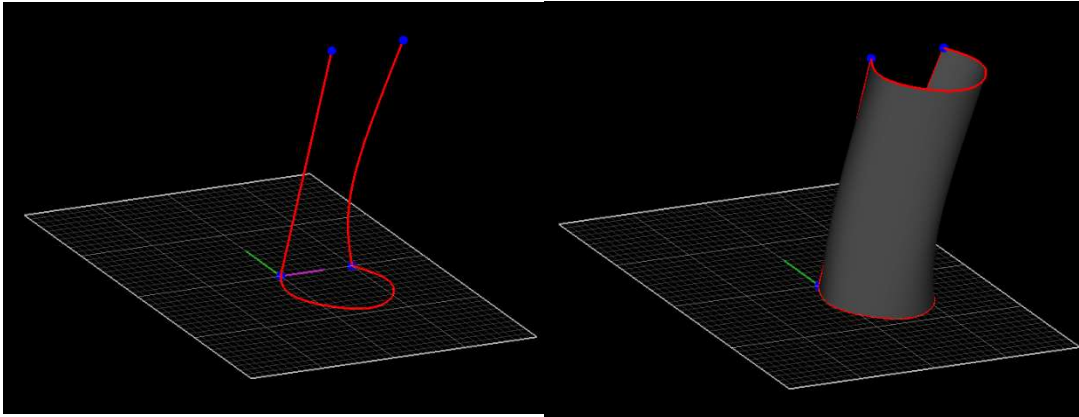


Figure 7.29: The profile and guide curves

Figure 7.30: After Rail sweep

## 7.2.5 Loft

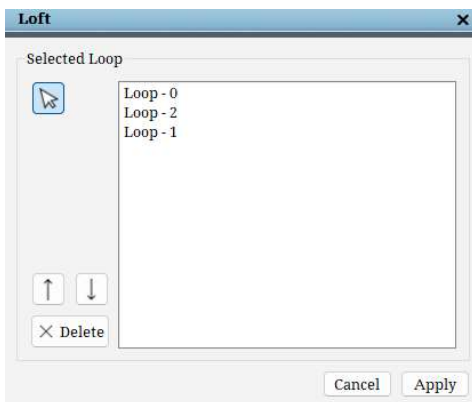



Figure 7.31: The Loft dialog box

The **Loft**  tool is used to create a smooth face between two or more profile curves.

To loft multiple profile curves, we need to first select the curves to be lofted. This is done using the **Selection** button which is clicked and then the curves to be lofted are selected. The selected loops are shown in the dialog box.



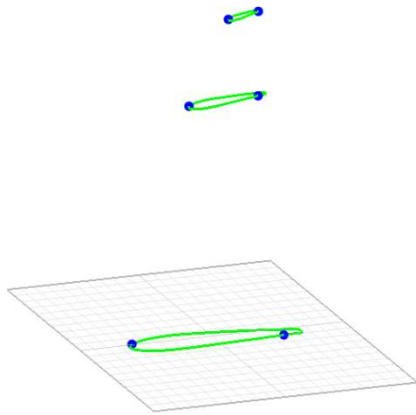


Figure 7.32: The curves to be lofted

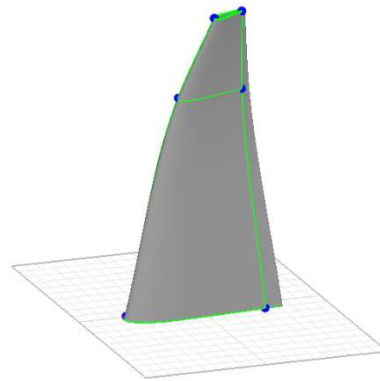



Figure 7.33: After loft

Any incorrectly selected loop can be deleted by clicking on the **Delete** button and clicking on the name of the curve to be deleted. Clicking on **Apply** creates the lofted surface.

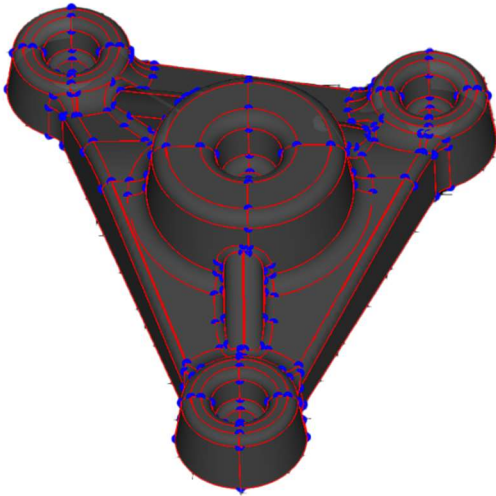
## 7.2.6 Heal



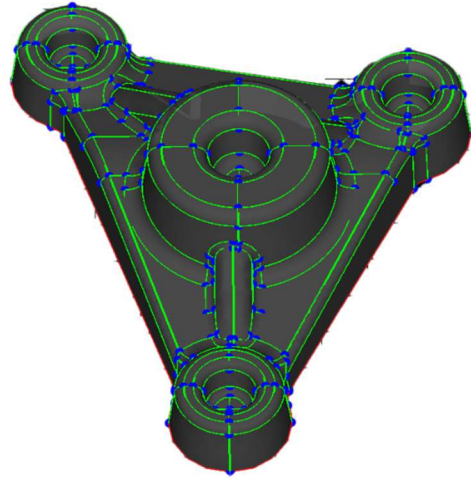
Figure 7.34: The Heal dialog box

The **Heal**  tool sews disconnected surfaces under the given **Tolerance**. If two disconnected faces are separated by a distance less than a tolerance they will be sewed together.

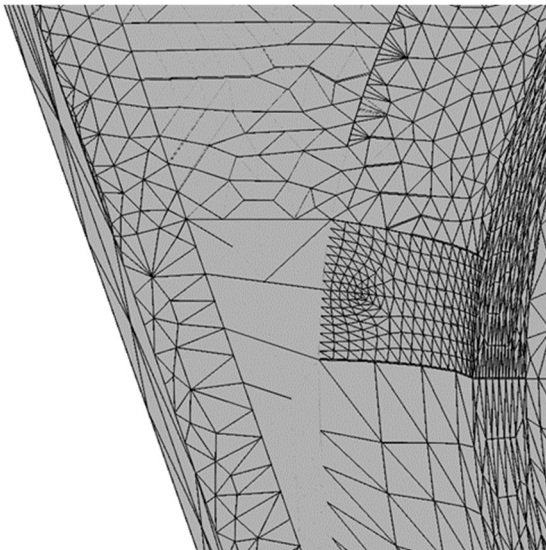
Also, all edges and vertices not connected to any face will be removed. The colouring of the edges changes based on the C-Edge colour scheme after healing.



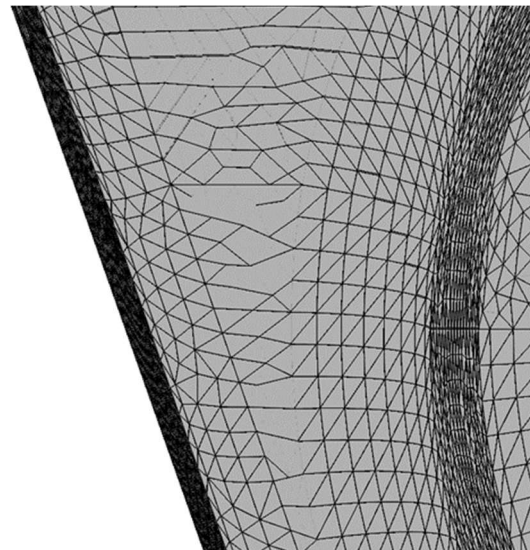
*Figure 7.35: Before Heal*



*Figure 7.36: After Heal*



*Figure 7.37: Surface mesh before heal  
(Triangulation not matching)*



*Figure 7.38: Surface mesh after heal  
(Matching triangulation)*

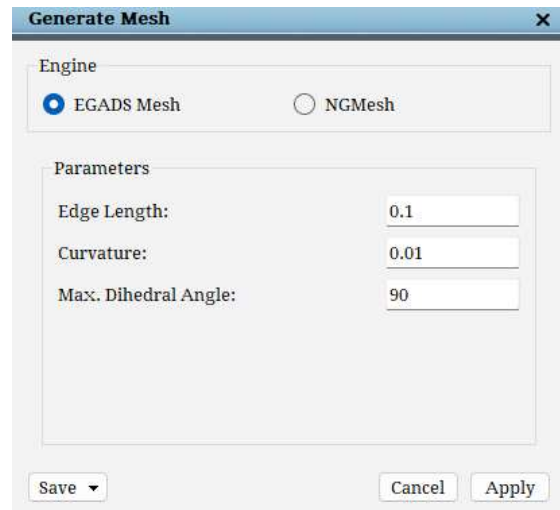
### 7.2.7 Mesh

There are two meshing engines that have been implemented in GridPro. They are **EGADS** and **NetGen**.

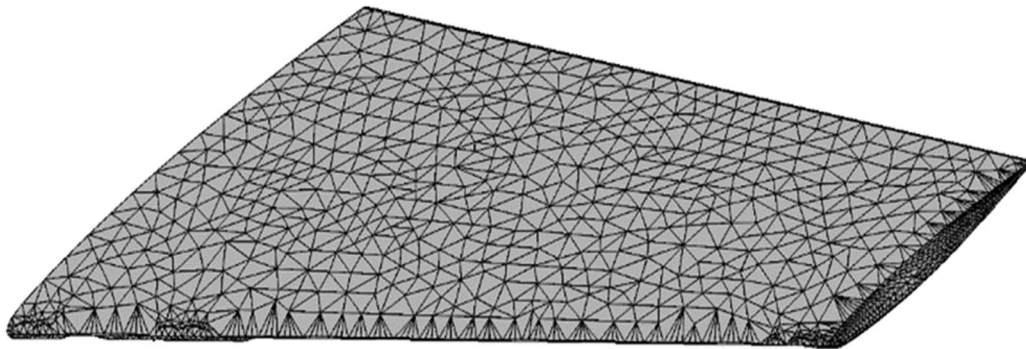
- **EGADS**

Clicking on the **Mesh** button opens the **EGADS** dialog box by default. The EGADS meshing engine is able to mesh a given CAD geometry and produce triangular (**Tria**) elements.

There are certain parameters which can be controlled by the user and drive the edge discretization and face triangulation in the **Parameters** section. The **Edge Length** represents the maximum length of an edge segment or triangle edge. The **Curvature** is a curvature-based value that looks locally at the deviation between the centroid of the discrete object and the underlying geometry. Any deviation larger than the input value will cause the tessellation to be enhanced in those regions. These two parameters are relative to the maximum dimension of the geometry which is being meshed. The **Max. Dihedral Angle** is the maximum interior dihedral angle (in degrees) between triangle facets.



*Figure 7.39: The EGADS Mesh dialog box*



*Figure 7.40: A .tria mesh created using the EGADS Mesher*

- **NetGen (NGMesh)**

Clicking on the **NGMesh** radio button opens up the NetGen mesher.

The **Parameters** section has advanced options to control the mesh quality. A brief explanation of all the options is as follows. **Fineness ratio** is the ratio which defines how fine the mesh should be. There are five different options listed under the dropdown. Each

option will automatically fill in the pre-defined values for the other parameters.

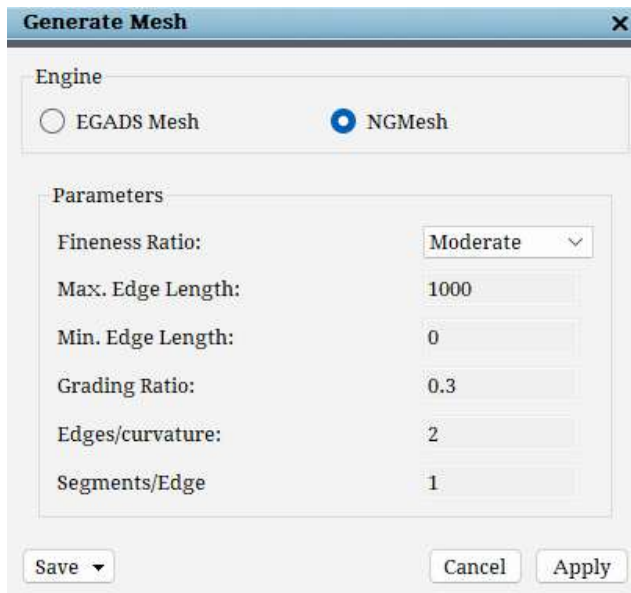


Figure 7.41: The NGMesh dialog box

**Maximum edge length** represents the maximum length of a triangle edge.

**Minimum edge length** represents the minimum length that a triangle edge can take. The minimum edge length will not create edges lower than the specified value.

**Grading ratio** defines how slow or fast the fine mesh at the edges should be transformed into a coarser mesh at the middle of a face.

**Edges/Curvature** defines the number of edges to be used per radius of curvature.

**Segments/Edge** allows defining the minimum number of mesh segments in which edges will be split. Size of elements computed using this value is trimmed between Min Size and Max Size bounds.

Once the desired mesh is created, it can be exported using the **Save** button at the bottom of the dialog box.

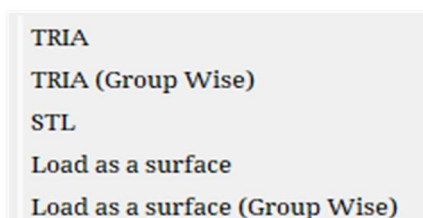



Figure 7.42: Mesh saving options

Saving the file as a **TRIA** will save the entire model as a single .tria file. Even if there are disconnected surfaces, that will be stored in a single file. When the user has grouped different faces using the **Add new**

**face group**  option (refer Section 7.1), saving using the option **TRIA (Groupwise)** will save multiple TRIA files each appended with the group name at the end. Saving as an **STL** will save the mesh file in the .stl format.

**Load as a surface** will load the mesh as a single file in the topology, whereas **Load as a**

**surface (Group Wise)**, will load the model and each group will be loaded as a separate surface in the topology.

### 7.3 Right Click menu

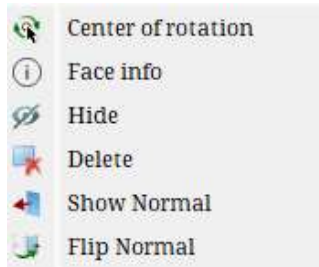


Figure 7.43: The Right Click menu

When you right-click on a geometry face, a menu appears, as shown. **Face info** displays information about the face such as Face ID, Body ID, Loops and Child Edges of the face. **Hide**, as the name suggests, hides the selected face. To make the face visible again by using the **Geometry Faces** section explained earlier (*Section 7.1*) . **Delete** deletes the selected CAD face. **Show Normal** displays the normal of the CAD face with a blue arrow. **Flip** flips the direction of the normal of the face by 180 degrees. This is used as sometimes the CAD may not create bodies if the normal are not oriented along one direction. **Center of rotation** picks the right-clicked position on the geometry as the centre of rotation of the screen.

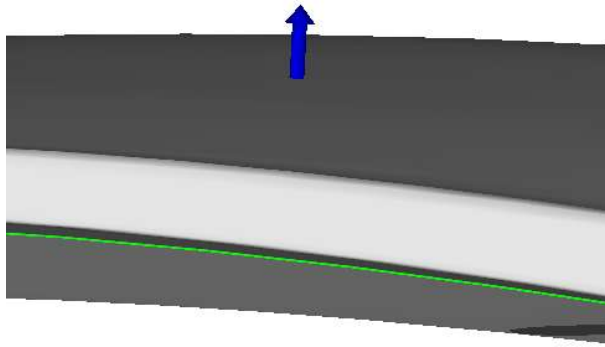


Figure 7.44: Before flipping the normal

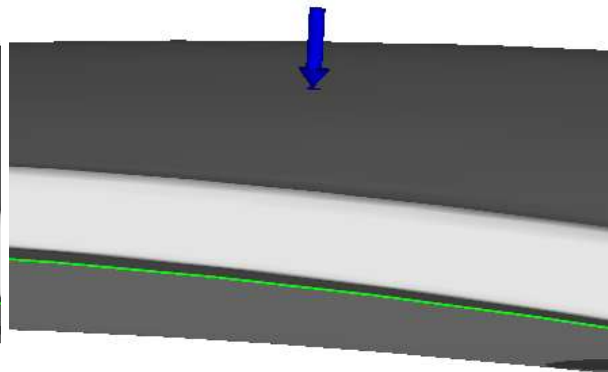


Figure 7.45: After flipping the normal

## APPENDIX A

### A1. Adding your Solver name to the Solver menu:

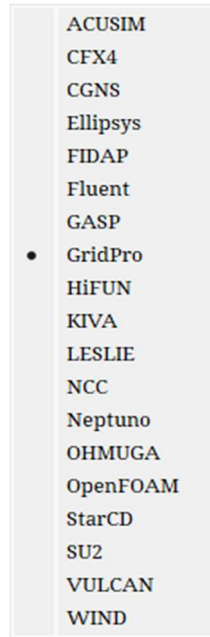


Figure A1: List of Solver formats

The solver names are listed under `ws.ptymap.menu` file. The file can be found inside the `az_mngr` folder of the `GridPro` installation directory.

**File path = GridPro/az\_mngr/ws.ptymap.menu**

Open the file in a text editor. The file looks like as shown in Figure A2.

```
ACUSIM &pty=ACUSIM & ws_ptymap.acusim
CFX4 &pty=CFX4 & ws_ptymap.cfx4
CGNS &pty=CGNS & ws_ptymap.cgns
# DTF &pty=DTF & ws_ptymap.dtf
Ellipsys &pty=Ellipsys & ws_ptymap.ellipsys
FIDAP &pty=FIDAP & ws_ptymap.fidap
Fluent &pty=FLUENT & ws_ptymap.fluent
GASP &pty=GASP & ws_ptymap.gasp
GridPro &pty=PDC & ws_ptymap.default
HiFUN &pty=HiFUN & ws_ptymap.hifun
KIVA &pty=KIVA & ws_ptymap.kiva
LESLIE &pty=LESLIE & ws_ptymap.leslie
NCC &pty=NCC & ws_ptymap.ncc
Neptuno &pty=NEPTUNO & ws_ptymap.neptuno
OpenFOAM &pty=OPENFOAM & ws_ptymap.openfoam
StarCD &pty=STAR & ws_ptymap.starcd
SU2 &pty=SU2 & ws_ptymap.su2
#USER &pty=USER & ws_ptymap.user
VULCAN &pty=VULCAN & ws_ptymap.vulcan
WIND &pty=WIND & ws_ptymap.wind
```

Figure A2: `ws.ptymap.menu` file



The format for your solver entry is,

MySolver      & pty=MySolver      & ws\_ptymap.MySolver

Where,

'MySolver' = Your solver name

'& pty=MySolver' = How your solver name should be displayed in the UI.

'& ws\_ptymap.MySolver' = ws\_ptymap.MySolver is a file that contains the boundary conditions of your solver. This file should exist in the GridPro/az\_mgr folder; otherwise, it may lead to a UI crash.

## A2. Adding a boundary conditions file for your solver:

The boundary conditions of the solver are defined in the "ws.ptymap.MySolver" file. Figure A3 shows the boundary conditions of the 'OpenFOAM' solver.

```

32      # 2d-ptys
#map-id& pty-name & default-name #comments
1      & Interior    #PDC generic
2      & Wall        #PDC generic
3      & Cyclic      #PDC generic
4      & Symmetryplane
5      & Patch
6      & Wedge
7      & Empty
8      & user8       & OF.User8
9      & user9       & OF.User9
10     & user10      & OF.User10
11     & user11      & OF.User11
12     & user12      & OF.User12
13     & user13      & OF.User13
14     & user14      & OF.User14
15     & user15      & OF.User15
16     & user16      & OF.User16
17     & user17      & OF.User17
18     & user18      & OF.User18
19     & user19      & OF.User19
20     & user20      & OF.User20
21     & user21      & OF.User21
22     & user22      & OF.User22
23     & user23      & OF.User23
24     & user24      & OF.User24
25     & user25      & OF.User25
26     & user26      & OF.User26
27     & user27      & OF.User27
28     & user28      & OF.User28
29     & user29      & OF.User29
30     & user30      & OF.User30
31     & user31      & OF.User31
32     & user32      & OF.User32

#-----
16     # 3d-ptys.
#map-id& pty-name & default-name #comments
1      & Fluid
2      & Solid #means 0x11=17
3      & user3       & OF.User3
4      & user4       & OF.User4
5      & user5       & OF.User5
6      & user6       & OF.User6
7      & user7       & OF.User7
8      & user8       & OF.User8
9      & user9       & OF.User9
10     & user10      & OF.User10
11     & user11      & OF.User11
12     & user12      & OF.User12
13     & user13      & OF.User13
14     & user14      & OF.User14
15     & user15      & OF.User15
16     & user16      & OF.User16

```

Figure A3: ws.ptymap.openfoam file

There are three columns in the file, and the names of the same are explained on the top. The first column is the map-id, which helps map the properties from *GridPro* to other solvers. The second column is the respective solver's property name or boundary conditions. The third column is the default name or default tags; this is used when the user defines no labels/tags for any given property. If no default name is defined in this file, then the property name is taken as the default name. For e.g., in Figure A3, the default name for the property 'Wedge' will be Wedge (Since there is no default name defined).

Note that there are '32' 2D and '16' 3D properties. This is a must for every solver. In case if your solver does not have the mentioned number of properties, please create dummy properties as shown in Figure A3 like user8, user9, etc.

### A3. Adding your solver format in Export options:

The export options list is defined in `ws_gridfmt.menu` file under `GridPro/az_mngr` folder.

**File location: GridPro/az\_mngr/ws\_gridfmt.menu**

Figure A4 shows the `ws_gridfmt.menu` file.

```
#menu lbl & head & elem b script to run & super b script to run & unst script to run
Acusin/Acusolve & Unstructured & outU_acusin.script
CFD++ & Elementary & outE_cfdpp.script
CFX4 & Elementary & outE_cfx4.script & Merged & outM_cfx4.script
CFL3D & Elementary & outE_cfl3d.script
CGNS & Elementary & outE_cgns.script & Merged & outM_cgns.script & Unstructured & outU_cgns.script
CGNS (Higher Order) & Quadratic & outU_cgns_quadratic.script & Cubic & outU_cgns_cubic.script & Quartic & outU_cgns_quartic.script
# DTF & Elementary & outE_dtf.script & Merged & outM_dtf.script
Ellipsys & Elementary & outE_ellipsys.script
FIDAP & Unstructured & outU_fidap.script
Fluent & Unstructured & outU_fluent_new.script
FluentOld (for US3D users) & Unstructured & outU_fluent.script
GASP & Elementary & outE_gasp.script & Merged & outM_gasp.script
GridPro & Superblock & outM_pdc.script & Unstructured & outU_pdc.script
HiFUN & Unstructured & outU_hifun.script
HOMUSIC & Linear & outU_ho_linear.script & Quadratic & outU_ho_quadratic.script & Cubic & outU_ho_cubic.script & Quartic & outU_ho_quartic.script
KIVA & Unstructured & outU_kiva.script
LESLIE & Elementary & outE_leslie.script
Nastran & Unstructured & outU_nastran.script
NCC & Unstructured & outU_ncc.script
Neptuno & Elementary & outE_neptuno.script
OFLOW & Elementary & outE_oflow.script & Merged & outM_oflow.script
OHMUGA & Elementary & outE_ohmuga.script
# OpenFOAM & New cyclic (>= v2.0.0) & outU_openfoam.script & Old cyclic (< v2.0.0) & outU_openfoam.script
Patran & Unstructured & outU_patran.script
PLOT3D & Elementary & outE_p3d.script & Merged & outM_p3d.script
STARCD & Unstructured & outU_starcd.script
SU2 & Hexa & outU_su2_hexa.script & Tetra & outU_su2_tetra.script
VULCAN & Elementary & outE_vulcan.script & Merged & outM_vulcan.script
WIND & Elementary & outE_wind.script & Merged & outM_wind.script
```

Figure A4: `ws_gridfmt.menu` file

The format for your solver entry is,

MySolver      & Output grid type      & Script file to be run

The script file to be run should be kept inside `GridPro/az_mngr` directory to access it while exporting the grid to the user-specific solver. The script file should contain the information on how the *GridPro* grid should be converted into the user-specific solver. The script file can also use the commands available in *GridPro*, such as `mrgb`, `chfmt`, `change_format`, `trf` etc.

Following is an example of the OpenFOAM solver script.

```
rem #!/bin/csh -fe
chfmt %1 -f sync -O %1.grd
gp_utilities change_format -ifn %1.grd -outfn %2.foam -iif -dn %2
del %1.grd %1.grd.conn %1.grd.ptx
```

*Figure A5: outU\_OpenFOAM.script file*

Since the conversion code is implemented within the command "change\_format", the same is used in the script file as a function call. In the case of your solver, you can add the code itself in the script file.

## **APPENDIX B**

### **B.1 Export to CFX format/Import into CFX-Pre**

#### **How to export the GridPro grid format to CFX format?**

There is no direct export for the CFX format like other formats. Please follow the procedure to successfully import a GridPro grid into CFX UI without losing any parameters or properties.

Step 1: Generate the GridPro grid and assign properties if required.

Step 2: Open CFX Pre-processor UI

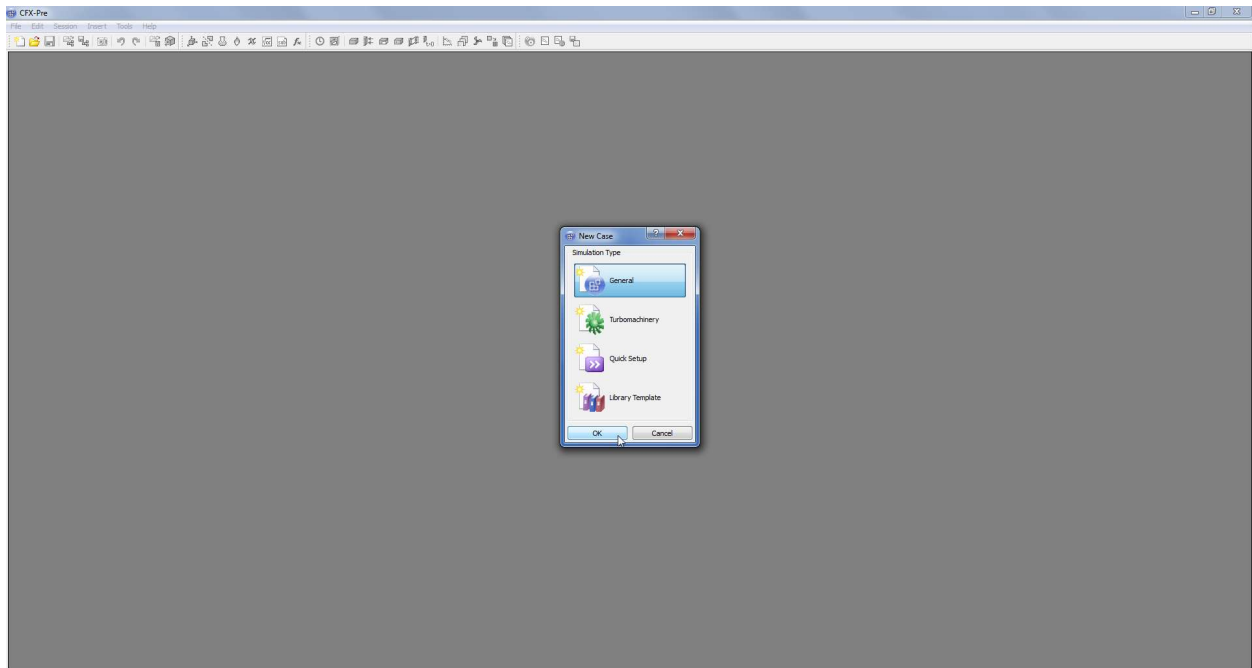


Figure B1: CFX-Pre UI

Step 3: Go to File→Import→Mesh

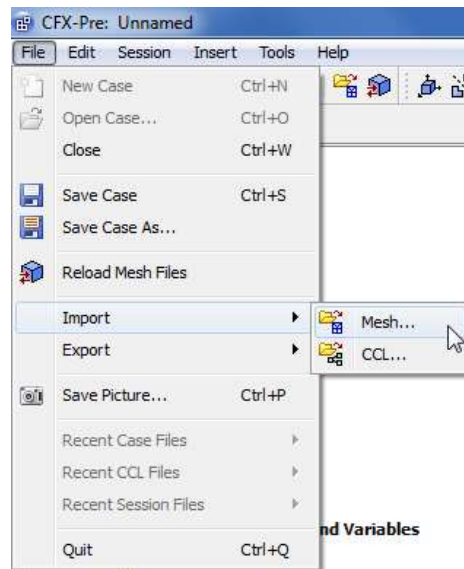


Figure B2: File menu

Step 4: Navigate the folder to the working directory and select the File type to “User Import(\*)”, as shown below.

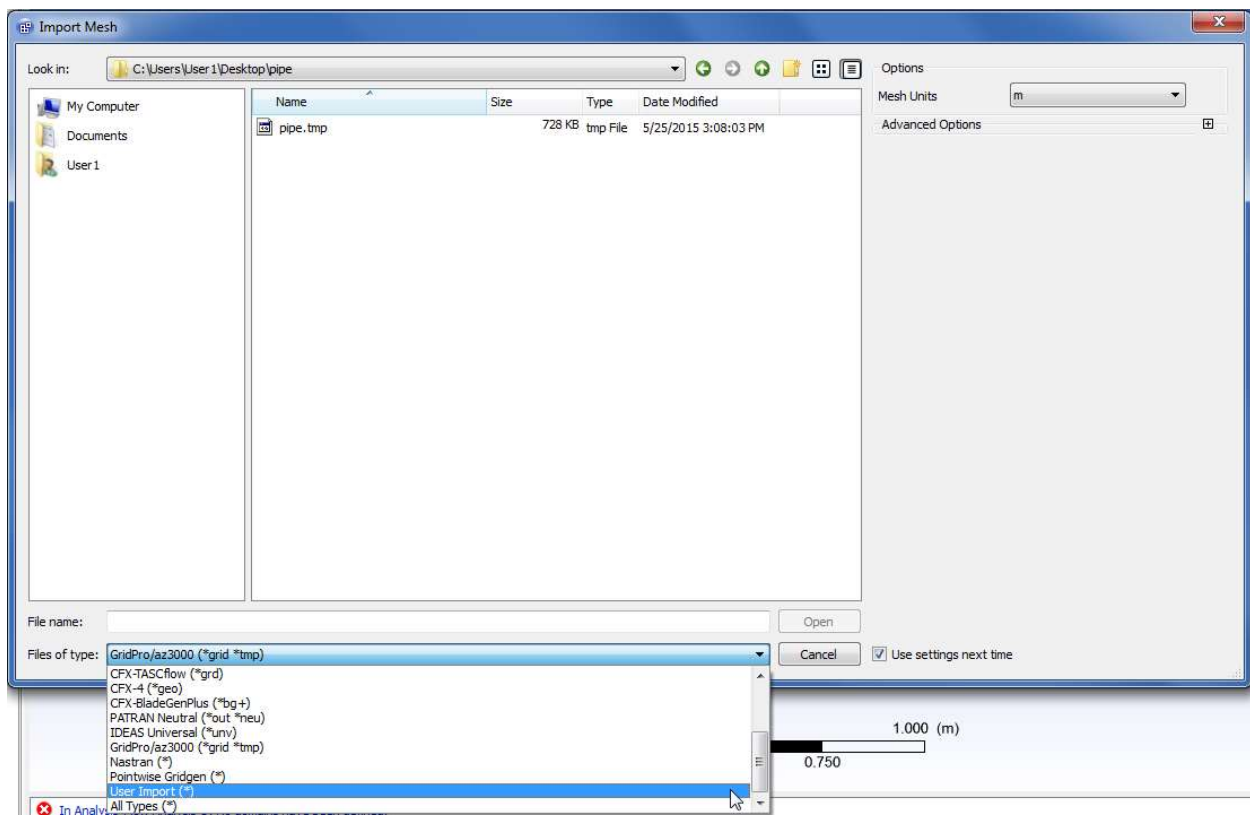


Figure B3: Import Mesh dialog box

Step 5: Navigate the “Exec Location” field to the GridPro installation directory and select the ‘ws.exe’ file.

For Windows users: C:/Program Files/GridPro/bin/ws.exe

For Linux/Mac users: /HOME/GridPro/az3000/ws

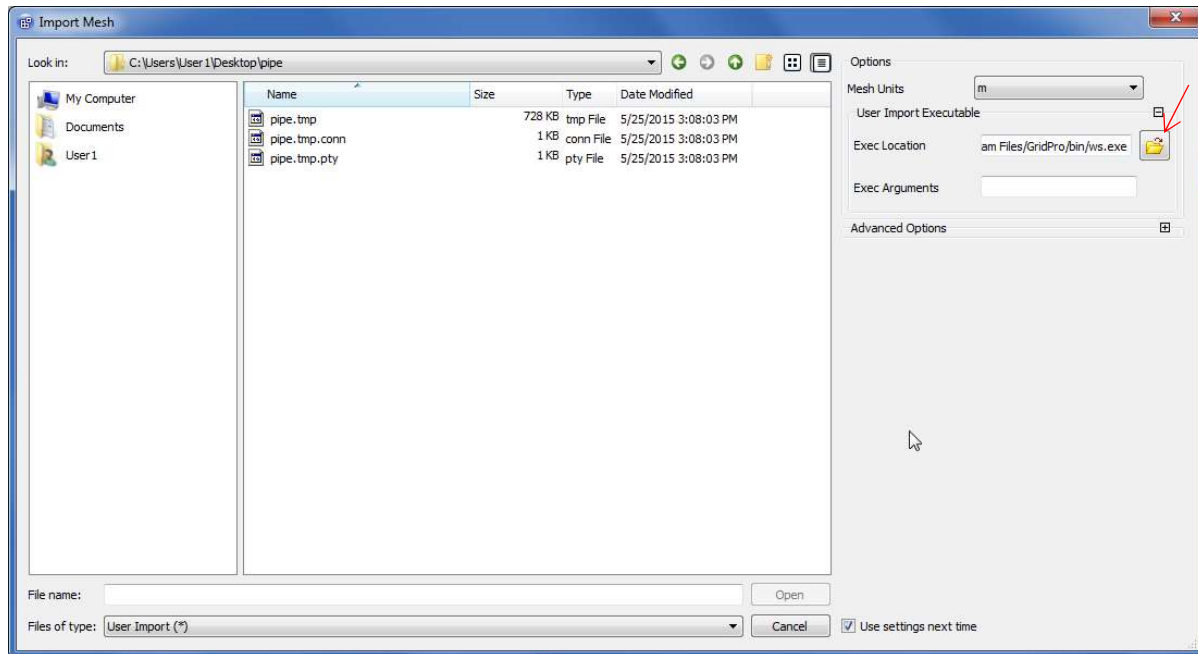


Figure B4: User Import options

Step 6: Enter the following syntax in the “Exec Arguments” field.

“change\_format -ifn”

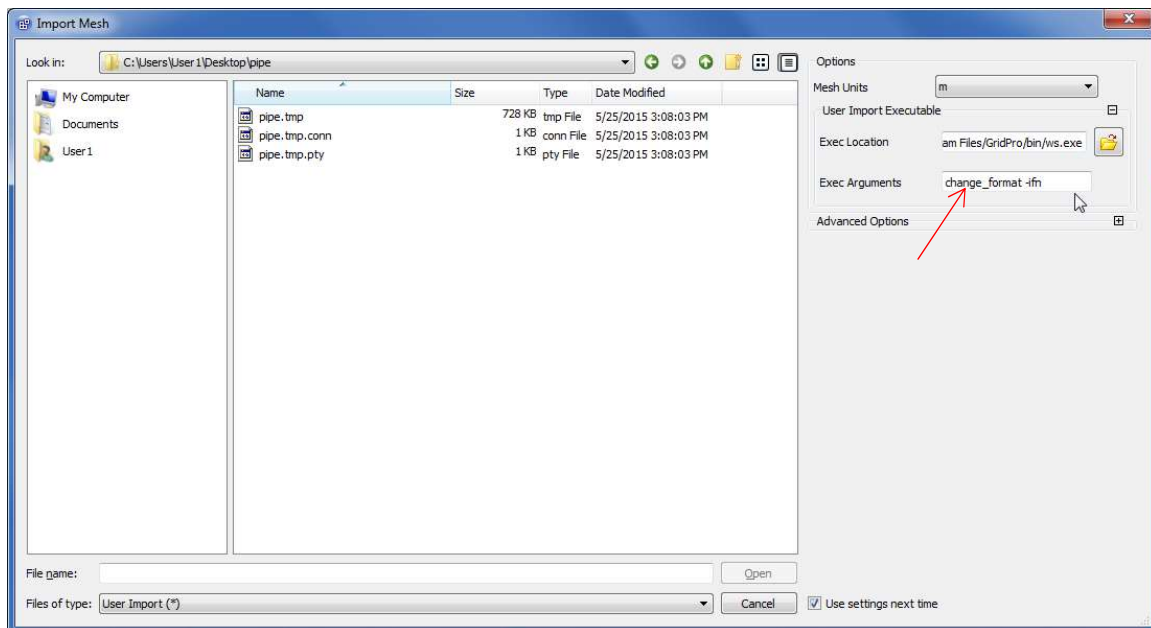


Figure B5: Input parameters



Step 7: Select the *GridPro* grid in \*.tmp or \*.grd format and click 'Open'

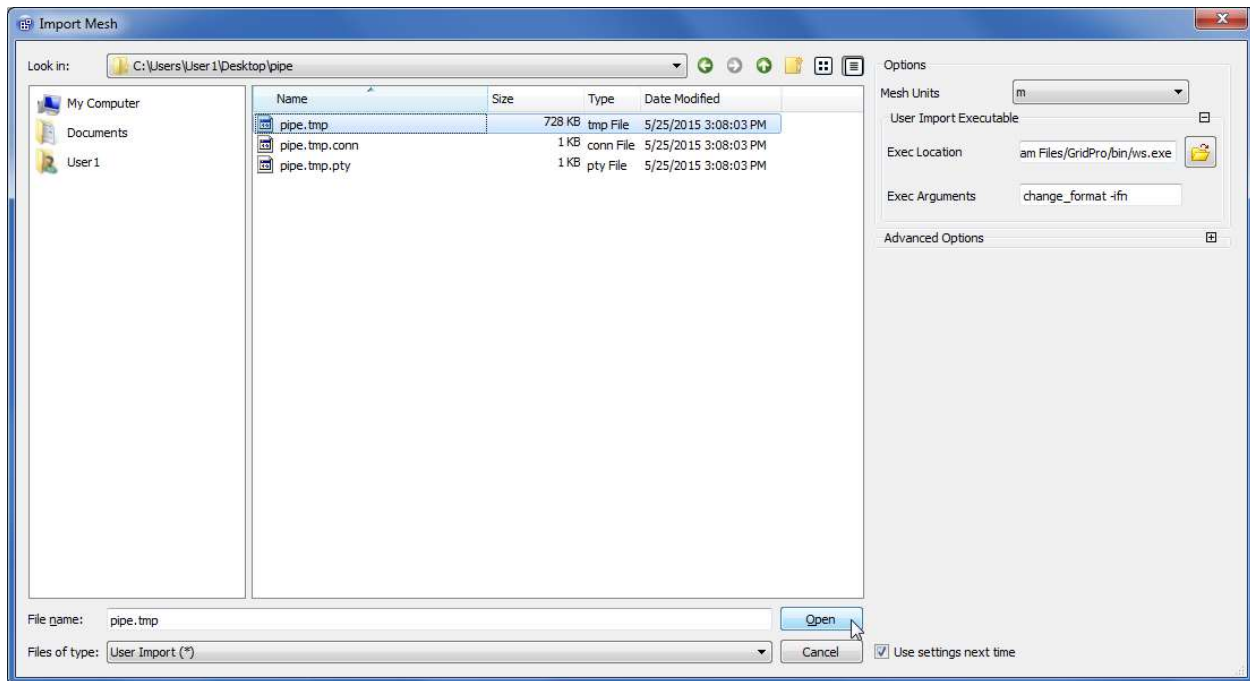


Figure B6: File selection

The grid will be converted and loaded into the UI.

NOTE: If there is an error message, as shown in figure B7 below, decrease the relative tolerance value and retry.

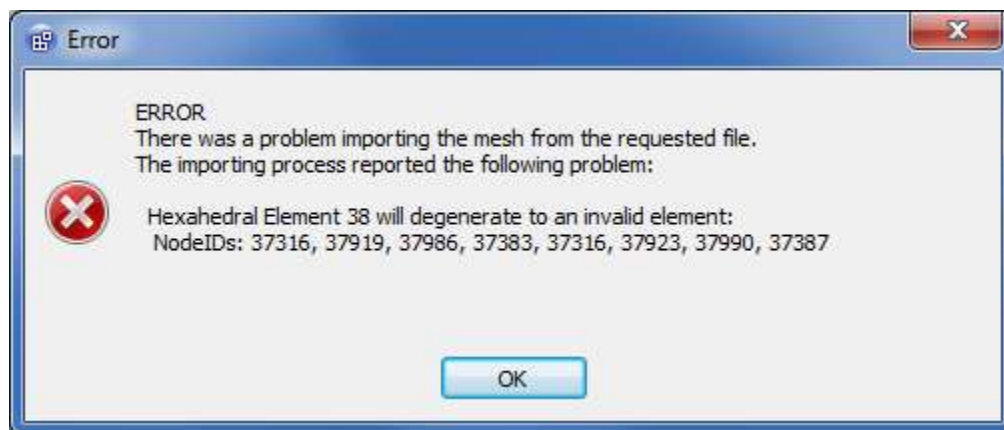


Figure B7: Error message

## How to decrease the relative tolerance?

Step A: Go to “Duplicate node checking” under Advanced Options.

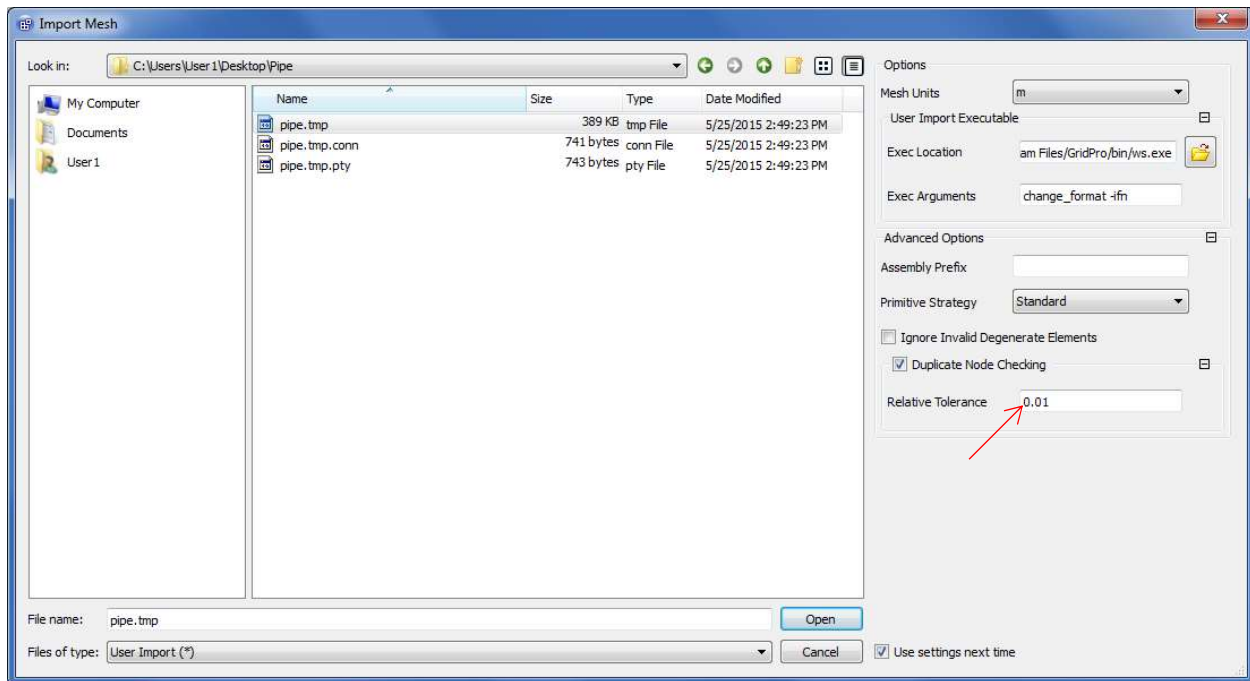


Figure B8: Advanced options

Step B: Decrease the relative tolerance value and retry

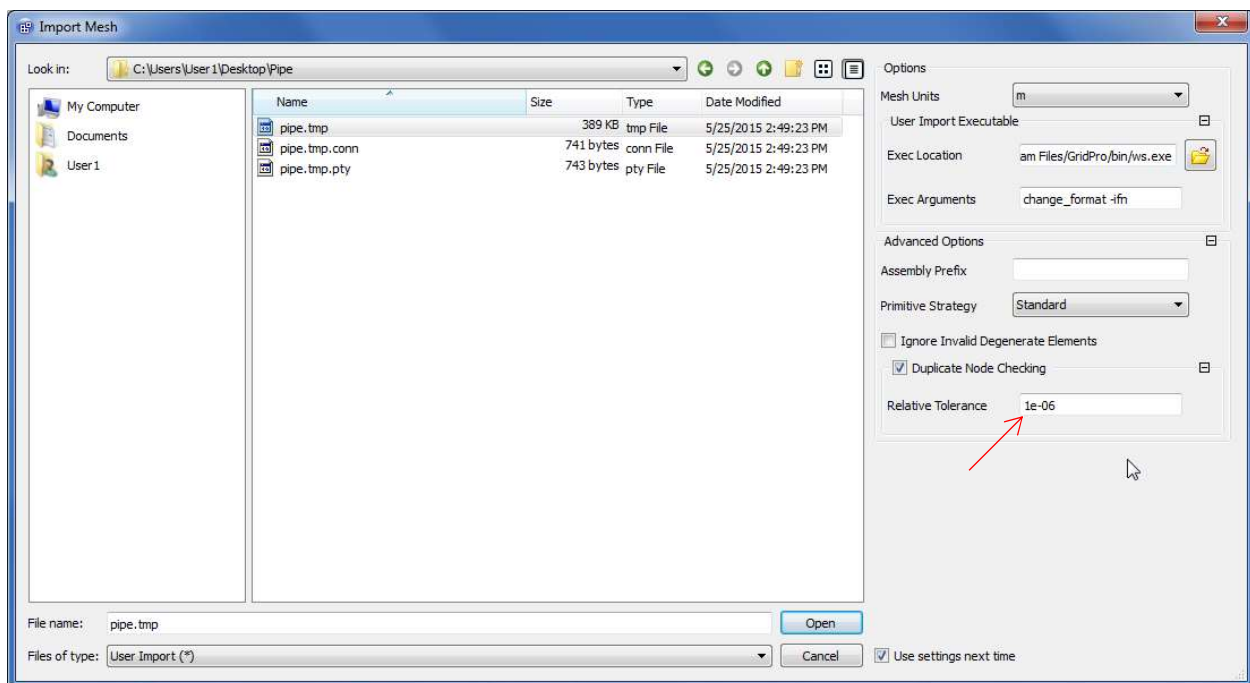


Figure B9: Relative tolerance

*If you are trying to import a clustered grid, the relative tolerance value should be lesser than the clustering parameter. For e.g., If the clustering parameter is  $1e-05$ , the relative tolerance should be lesser than  $1e-05$ , say  $1e-06$  or  $1e-07$ .*

## APPENDIX C

### Schedule file

A schedule file \*.sch controls the gridding process. As mentioned in section 2.3.25, the basic controls of a schedule file can be modified in the UI. For advanced controls, the schedule file has to be edited manually. Figure C1 shows a default schedule file which is written every time the Ggrid is started.

```
#This is a sample schedule file. You may edit it.  
step 1: -g all 1.000000x  
step 10000: -c all 1.0 0 -C all 1.0 24 -r -S 100 -w  
write -f blk.tmp  
write -D 0 -f dump.tmp
```

Figure C1: Default schedule file

The schedule file is composed of a sequence of steps executed one by one.

*step 1* → Represents the sequence of steps.

*-g all 1.0x* → All the edges in the topology takes the corresponding densities as defined. To run half density grid, input 0.5x instead of 1.0x

*-c all 1.0 0* → Include boundary layer clustering while running Ggrid if any surface is defined with. To run the Euler grid, remove this syntax from the schedule file.

*-C all 1.0 24* → Internal clustering for cell smoothing. It is not recommended to modify any of its values.

*-r <x>* → Readjust surfaces with radius = x. If no value is given, the default value 1 is taken into account.

*-S <num>* → Represents the number of sweeps per step.

*-w <num>* → Writes the output grid file every <num> sweeps. Without <num>, it writes out after every step. For e.g. in this case, it writes out every 100 sweeps. i.e 1 step = num (in -s <num>)

`-f blk.tmp` → Writes the ASCII grid file to blk.tmp. To write the grid in binary format, a flag `-B` must be added before `-f`.

`-D <num>` → Writes the binary format dump file for num = 0 → For restart purpose, 2 → For 2D grids, 3 → 3D grids

`-f dump.tmp` → Writes the grid file to dump.tmp

Often, the schedule file is edited to converge the grid faster by running the grid with half density initially and output the grid with full density at the end. Following is an example that runs the grid with half density for 2000 sweeps and runs with full density for 3000 sweeps by writing the output every 500 sweeps. The final grid is obtained after 5000 sweeps and stops the Ggrid automatically.

```
#This is a sample schedule file. You may edit it.
step 1: -g all 0.5x      #Set half density
step 5: -c all 1.0 0 -C all 1.0 24 -r -S 500 -w #Run Ggrid for 2000 sweeps in total(step 2,3,4,5)
step 6: -g def 0.5x      #Set all default edge density(default =8) to half density
step 7: -g all 2.0x      #Set twice the previous density(which is half of original density)
step 13: -c all 1.0 0 -C all 1.0 24 -r -S 300 -w #Run Ggrid upto 5000 sweeps(step 8,9,10,11,12,13)

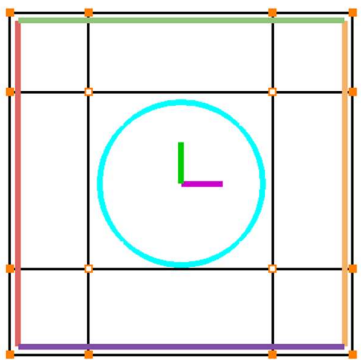
write -f blk.tmp
write -D 0 -f dump.tmp
```

Figure C2: Example schedule file

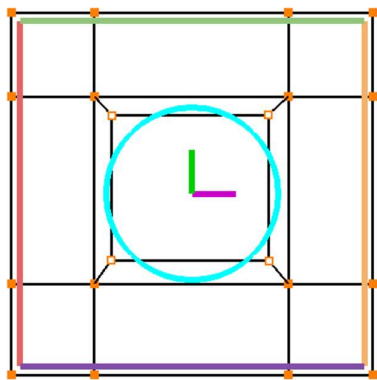
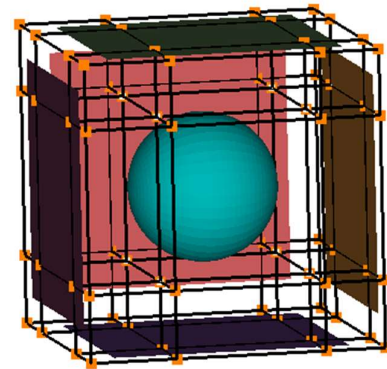
## APPENDIX D

### D.1 Surface Assignment Rules

1. If three or more blocks emerge from an edge in 3D (or corner in 2D), it cannot be assigned to a boundary surface.



Incorrect Assignment



Correct Assignment

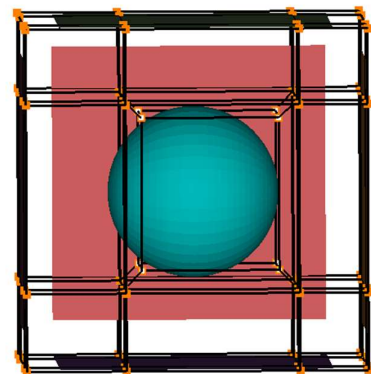
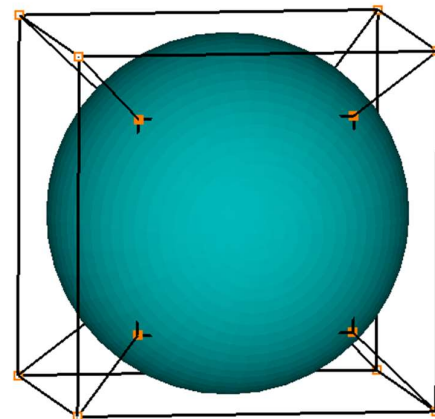
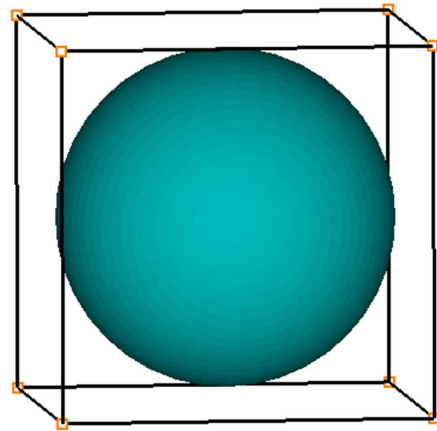
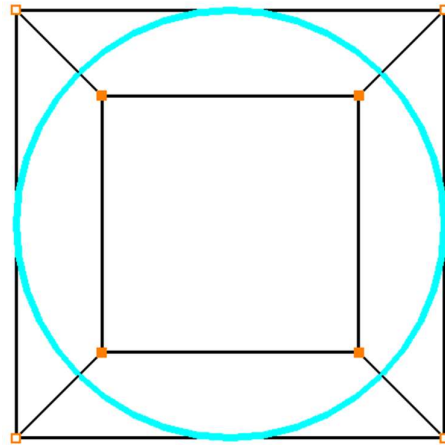
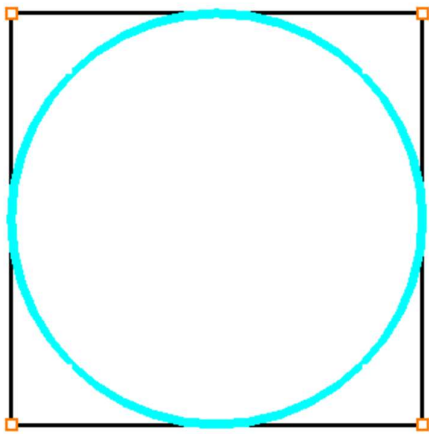


Figure D1



- Two edges/faces of the same block cannot be assigned/associated with a single surface.

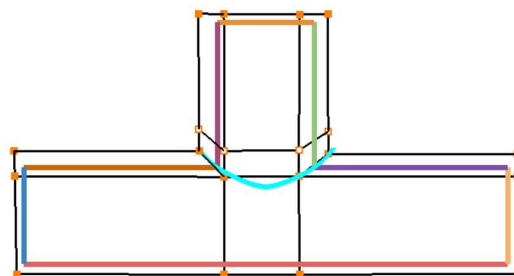
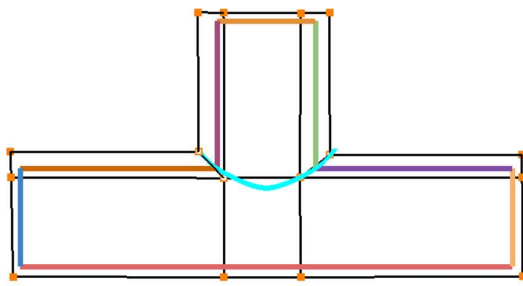


Incorrect Assignment

Correct Assignment

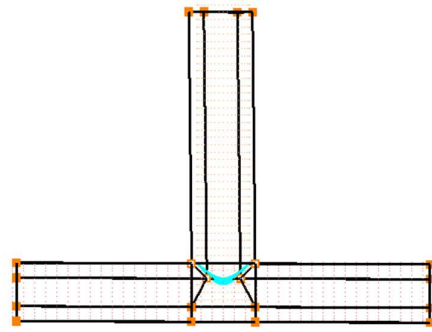
Figure D2

- The corners/edges assigned to an internal surface should share two blocks on either side.

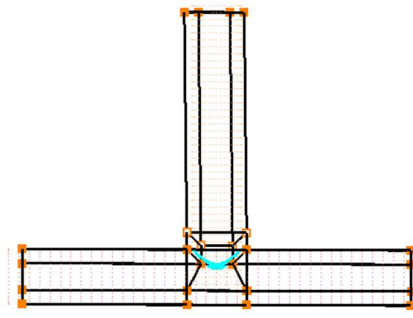


Incorrect Assignment

Correct Assignment



Incorrect Assignment



Correct Assignment

Figure D3

4. The slope discontinuity in a single surface should be represented by two distinct surfaces.

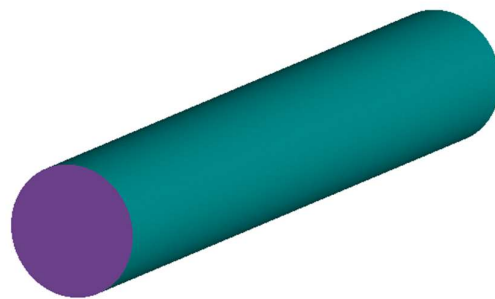
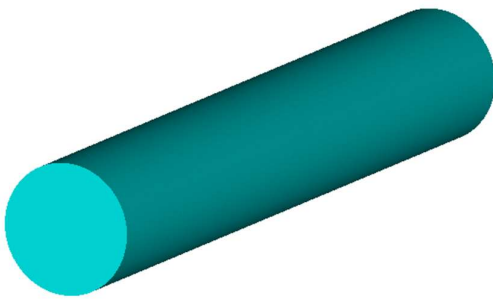
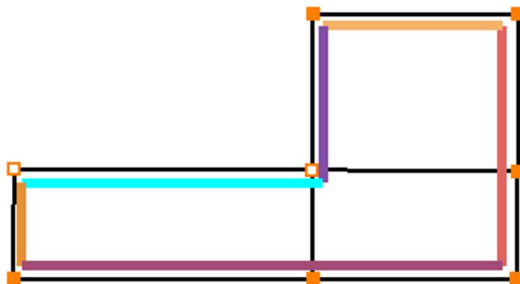
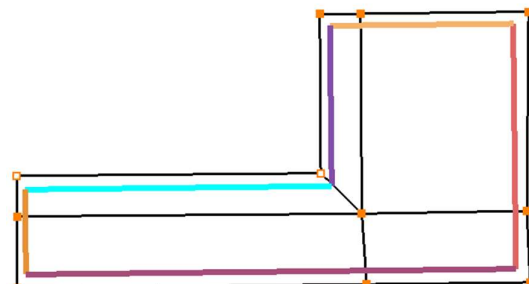


Figure D4

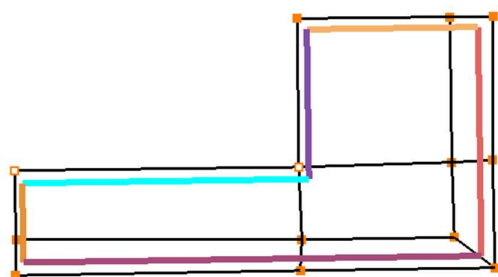
5. When the surface slope discontinuity is convex, the blocks emerging out of an edge (or corner in 2D) should be 2, and if it is concave, the blocks emerging out of an edge (or corner in 2D) should be 1.



Incorrect Topology



Correct Topology



Incorrect Topology

Figure D5