

# **GridPro GUI Manual**

**Version 2.3**



Program Development Corporation

300 Hamilton Ave., Suite 409

White Plains, NY 10601

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 6.7**

July 15, 2018

© Copyright Program Development Company, 1992-2018 – 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>	1
1.1    What is <i>GridPro</i> ?	2
1.2    How <i>GridPro</i> works?	3
<b>Chapter 2 – Getting Started With GUI</b>	6
2.1    Introduction to GUI	7
2.2    Top Menu Bar	11
2.3    Global Dock	22
2.4    Workplane Toolbar	47
2.5    Toggle Surface Section	54
2.6    Toggle Display Section	55
<b>Chapter 3 – Surface</b>	61
3.1    Surface Types	62
3.2    Surface Orientation	77
3.3    Surface Probing Utilities	79
3.4    Surface Reconstruction Utilities	80
<b>Chapter 4 – Topology</b>	96
4.1    Topology Building	97
4.2    Topology Creation Utilities	102
4.3    Automatic topology creation utilities	117
4.4    Topology Grouping Utilities	121
4.5    Topology Modifying Utilities	129
4.6    Topology Transformation Utilities	137
4.7    Topology Enhancing Utilities	142
4.8    Corner Projection Utilities	152

4.9	Topology Optimization Utilities	156
4.10	Surface Assignment Utilities	165
4.11	Surface Building Utilities	171
4.12	Gridding Process	187
<b>Chapter 5 – Grid</b>		<b>191</b>
5.1	Grid Viewing Utilities	192
5.2	Preprocessing Utilities	198
5.3	Post Processing Utilities	199
5.4	Grid Probing Utility	211
5.5	Quality Check Parameters	212
5.6	Grid Toggling Section	214
5.7	Grid Trimming Utilities	217
<b>Chapter 6 – Properties</b>		<b>222</b>
6.1	Properties Section	223
6.2	Solver Section	228
6.3	File Format	231
<b>Chapter 7 – CAD</b>		<b>233</b>
7.1	Geometry Faces Section	234
7.2	Tools Section	237
<b>Appendix A</b>		<b>242</b>
<b>Appendix B</b>		<b>245</b>
<b>Appendix C</b>		<b>250</b>
<b>Appendix D</b>		<b>252</b>

# CHAPTER 1

## INTRODUCTION

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

- 1) Quick and easy to setup typical complex gridding problems: The considerations include the first gridding turn-around time, the subsequent parametric design turnaround time, the modular parametric design (adding and subtracting features) turnaround time and the clustering capabilities (for CFD use).
- 2) Good grid quality: such as the 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 is having 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. To ensure that the highest quality grids are generated, the software engine incorporates a unique algorithm using advanced mathematical techniques. 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.

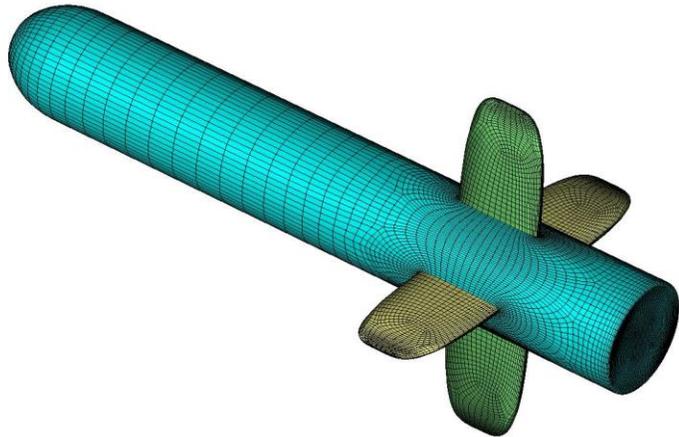


Figure 1.1: Missile configuration

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 both the initial setup time and more drastically the subsequent setup time for configuration modifications; the second translates into a higher solution accuracy, and faster convergence for the CFD solvers; and the third reduces the demand for computer resources in terms of both the CPU time and the amount of RAM usage.

The mathematical underpinnings that reduce the user effort also account for *GridPro*'s substantially high grid quality. This quality comes in the form of 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 cases 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 normally 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 basically no limit to the number of blocks.

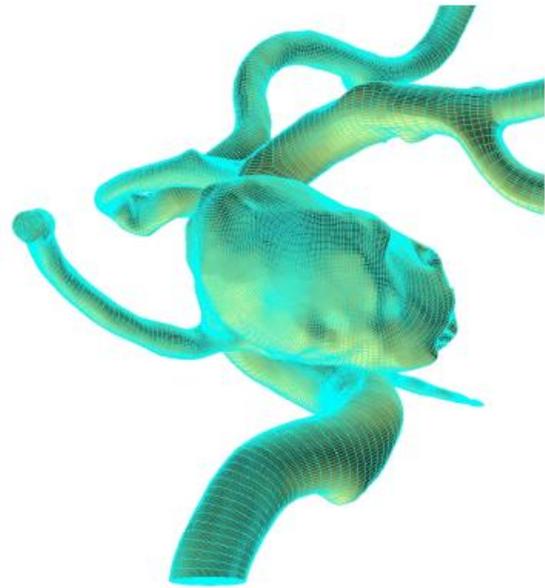


Figure 1.2: Brain aneurysm

## 1.2 How *GridPro* Works?

*GridPro* is 'automatic' grid generator, which accesses an automatic process earlier than any other 'automatic' grid generator. Included in *GridPro*'s automation are such tasks as 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 and/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)

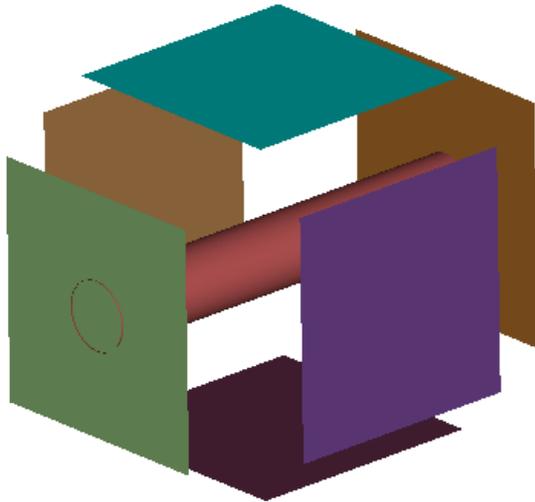


Figure 1.3: A collection of surfaces bounding a region to be gridded

can also be imported. It also has a certain capability of automatic surface restructuring and repair, 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 from CAD, a topology for the region bounded by those surfaces has to 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; while 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 properly 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, first, define the wireframe through the creation of 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.

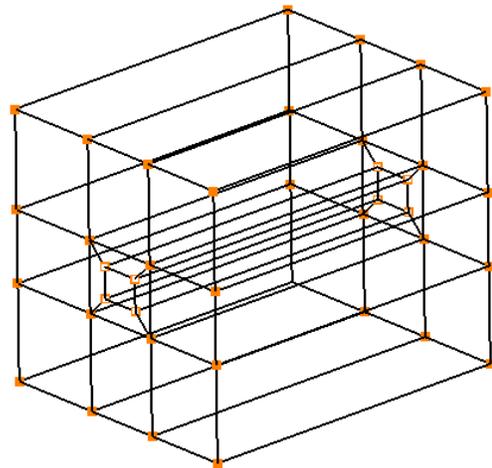


Figure 1.4: Wireframe for the region enclosed by the surfaces in figure 1.3

The two functions, which help in building wireframe or can be said as 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, simply, a plane in space that is used for interactive topology generation and for graphical viewing. Second, the groups are the most sophisticated tool to use while working on the complex geometries. Groups are, simply, 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 helps the user to generate topology effectively and efficiently. Apart from these tools, there is an important term, which will be often 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 color of the surface, it is always highlighted in sea blue color; the second method is using the surface id which is 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 simply 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 conformity with the bounding geometry, generating a multi-block grid. The process is driven by the algorithms that ensure the highest possible level of grid quality 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 make changes in region boundaries to achieve the desired physical attributes, *GridPro* dynamically tracks the evolving boundary shapes. Traditional approaches to grid generation are labor 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 a variety of 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, an environment that allows 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 four 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.

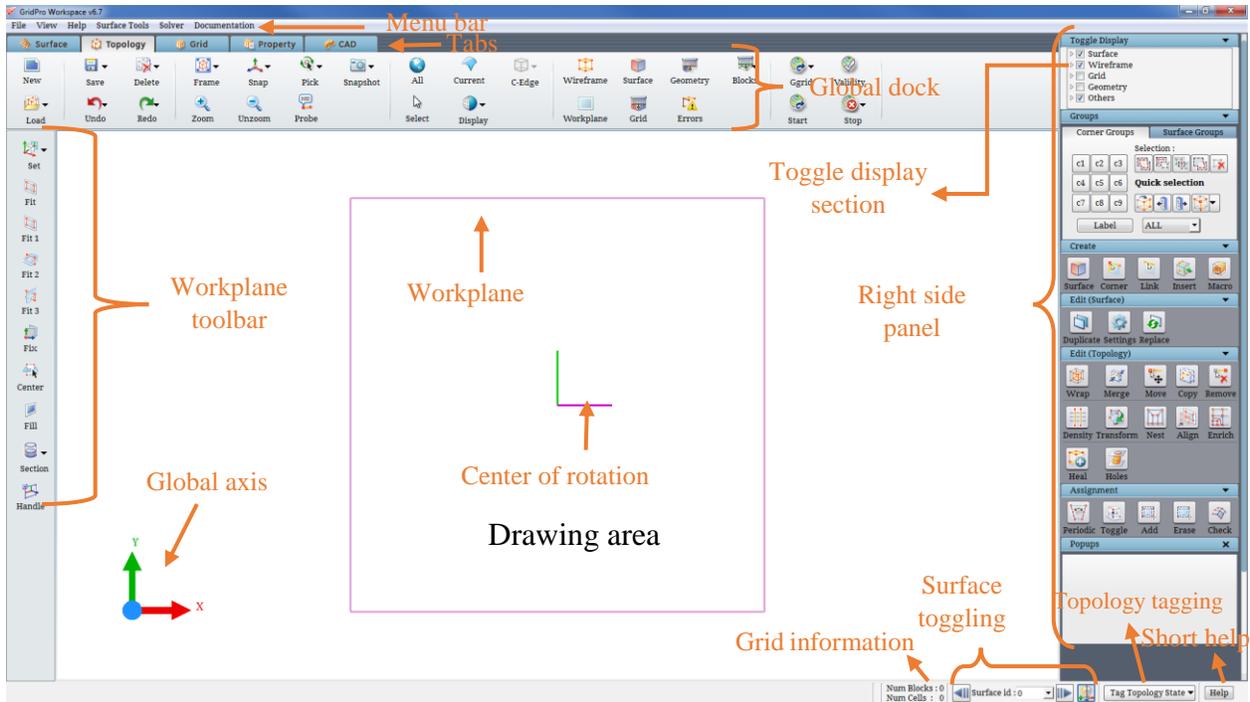


Figure 2.1: GridPro GUI

The empty space in GUI is referred as *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 five menu’s i.e. File, View, Help, Surface Tools and Solver. Then below the menu bar, 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, toggle display section is also located in the global dock. All the tools related to each tab are located in the right side panel of each tab. All tools are categorized based upon their use. Below the tools section, there is ‘popups’ section, where the dialog box of each tool opens. In the left side, the workplane toolbar is located. Bottom bar shows status messages, details of the grid, option 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 gridding process, you firstly deal with surfaces which lead to 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 in the right side panel. Along with these, surface grouping tools are also located under Groups section. All these tools will be discussed later.

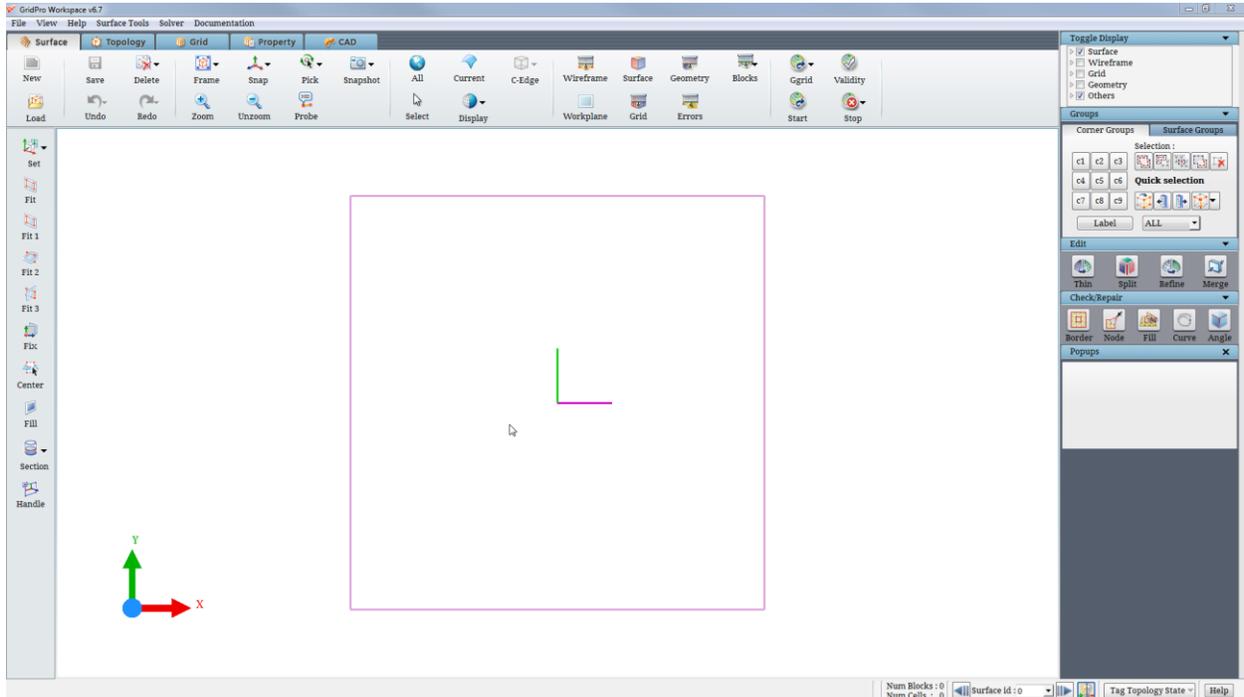


Figure 2.2: Surface tab

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

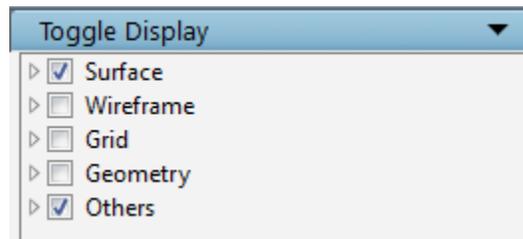


Figure 2.3: Toggle Display section

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

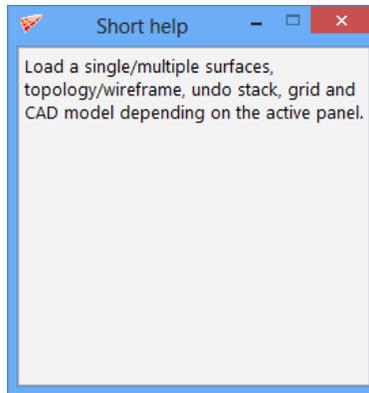


Figure 2.4: Short help window

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

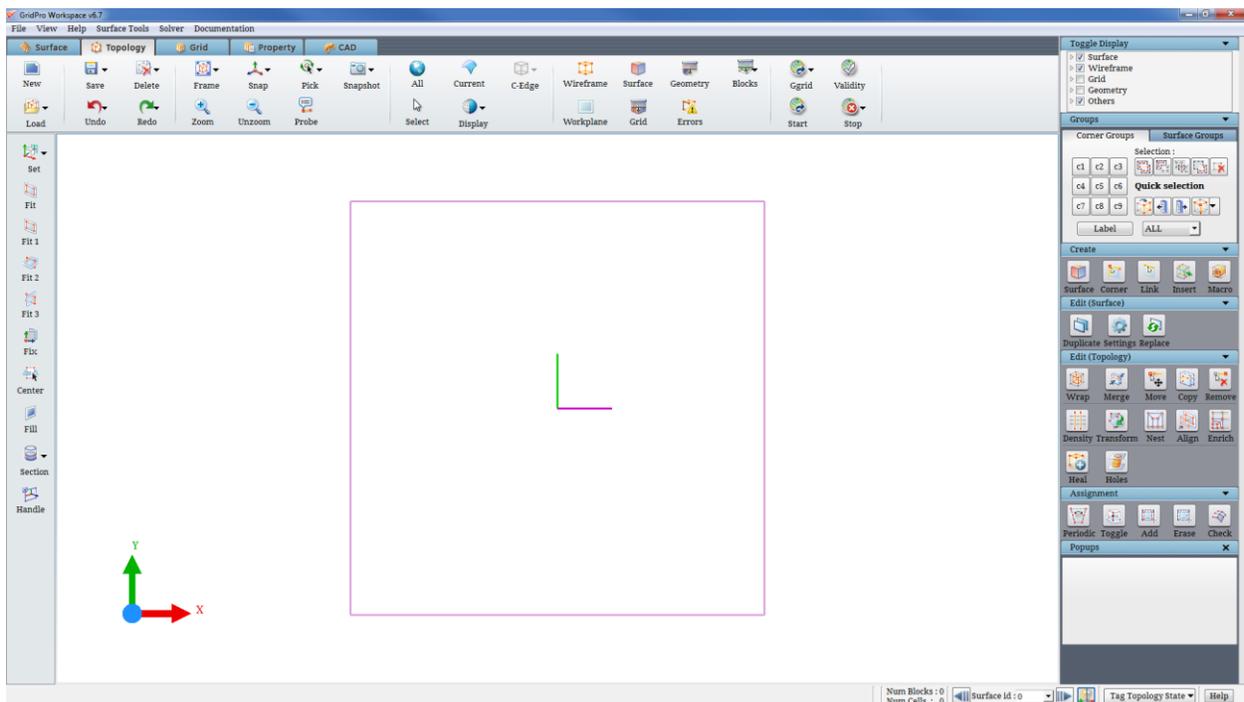


Figure 2.5: Topology tab

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

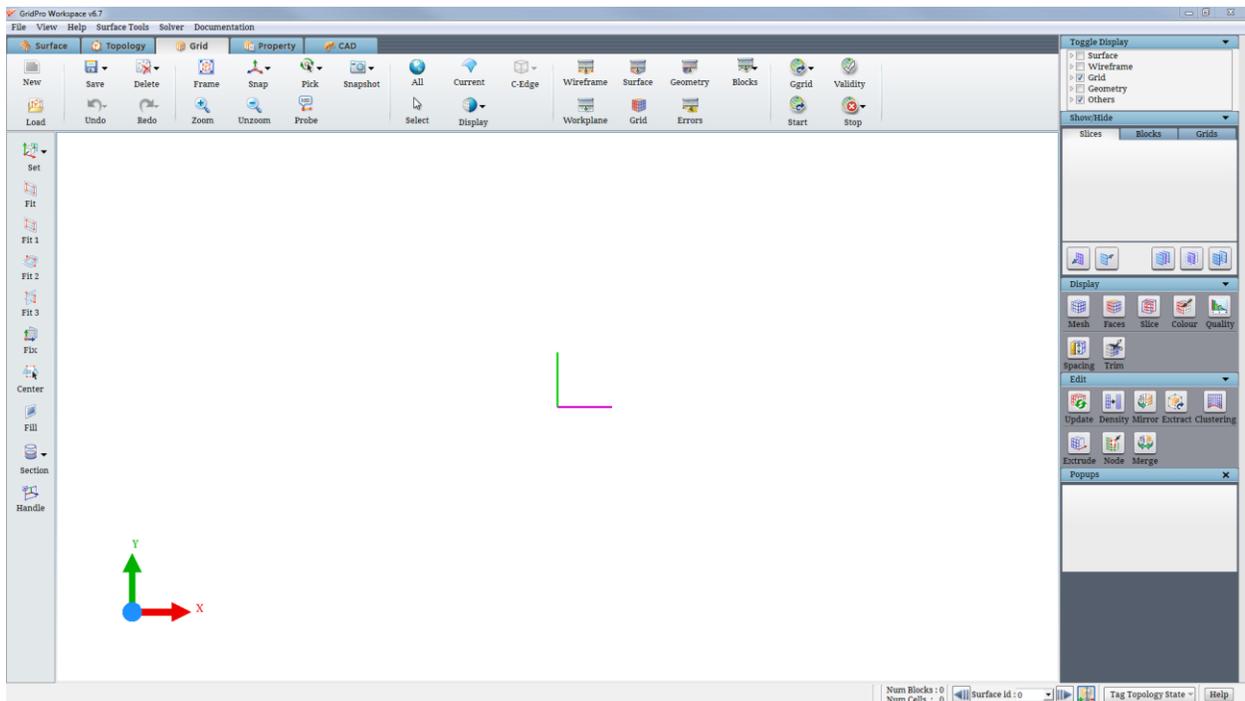


Figure 2.6: Grid tab

The fourth tab is '**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.

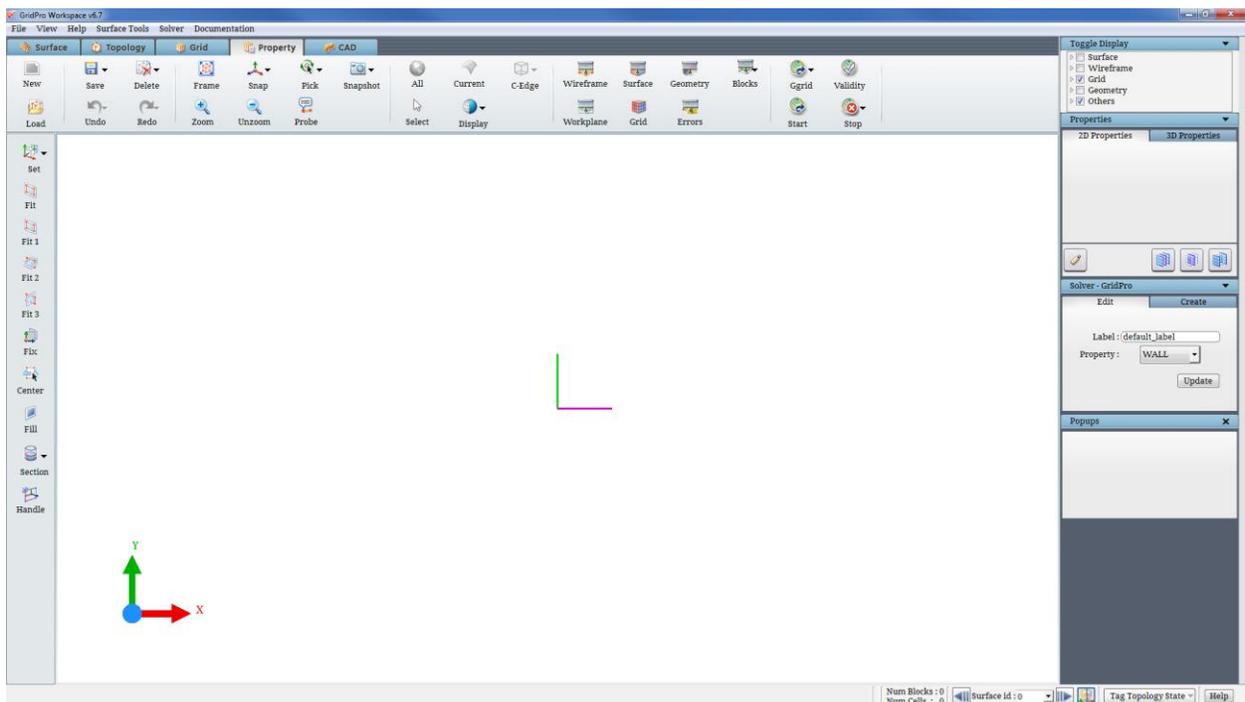


Figure 2.7: Property tab

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

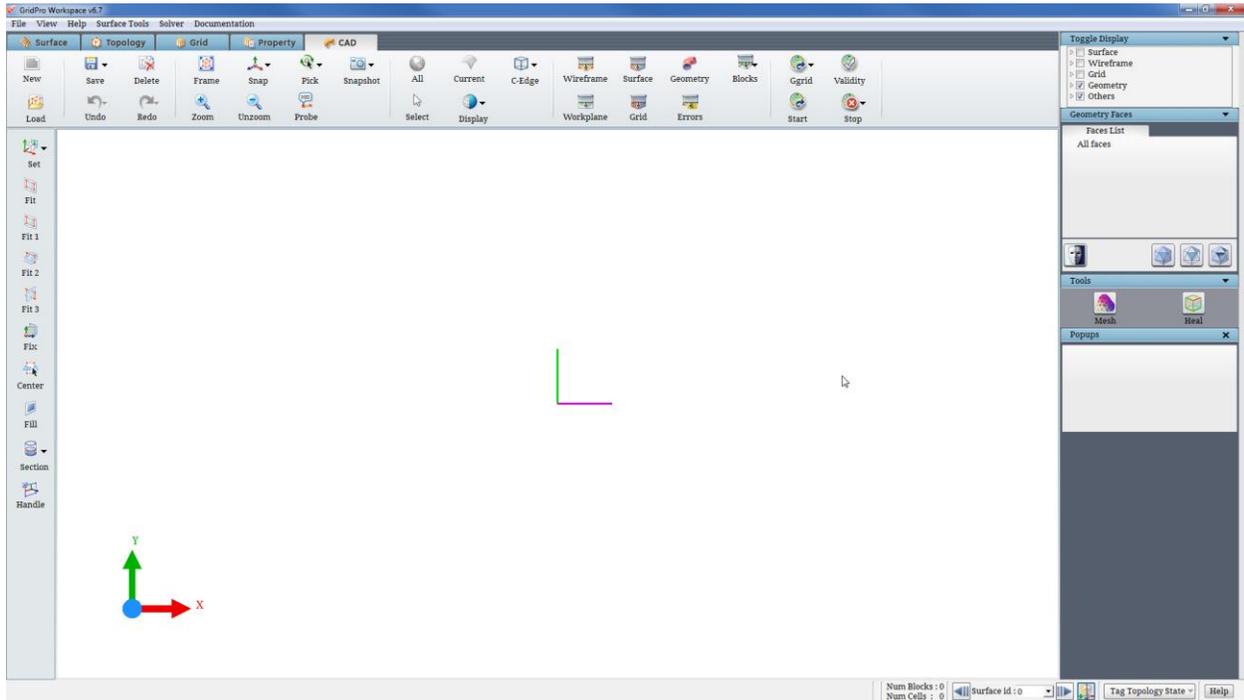


Figure 2.8: CAD tab

## 2.2 Top Menu Bar

The top menu bar of the GUI consists of five menus such as *File* menu, *View* menu, *Help* menu, *Surface Tools* and *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.

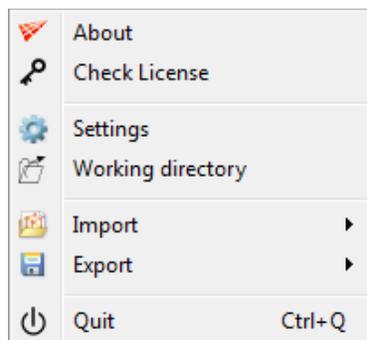


Figure 2.9: File Menu

**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\_mgr directory, a message pops out as shown in Figure 2.11.

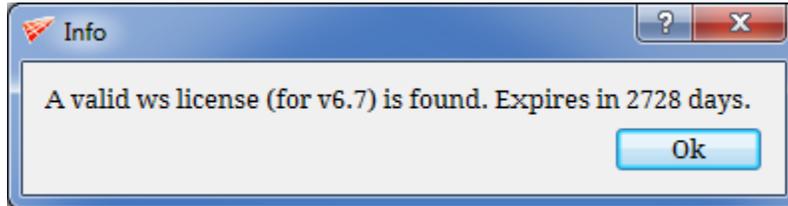


Figure 2.11: License expiry status

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). Same message is also displayed when UI is started and there is no valid license found.

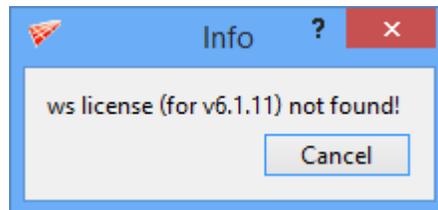


Figure 2.12: License status

From the **Settings** option, you can change the predefined keyboard shortcuts, font size, background color, and autosave frequency. In GUI often-used functions are defined with keyboard shortcuts for easy usage that can be modified. The Figure 2.13 shows the default keyboard shortcut keys. These shortcuts can be modified using the drop-down menu associated with each shortcut. The modifications will work across the sessions except for the background color, to retrieve the default settings click the **Default** button in the popup window as shown in the Figure 2.13.

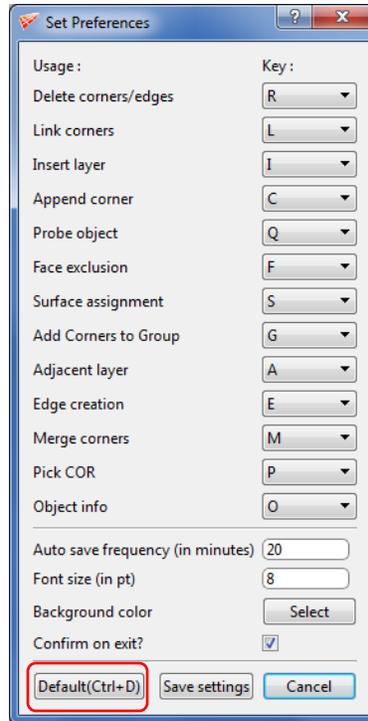


Figure 2.13: Default Keyboard Shortcuts

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

Background color 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 color has to be given in the respective fields. In addition to that, few standard colors can be chosen from the **Sample colors** drop-down list as shown in Figure 2.14.

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

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 is to ensure that even if the application crash 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 *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

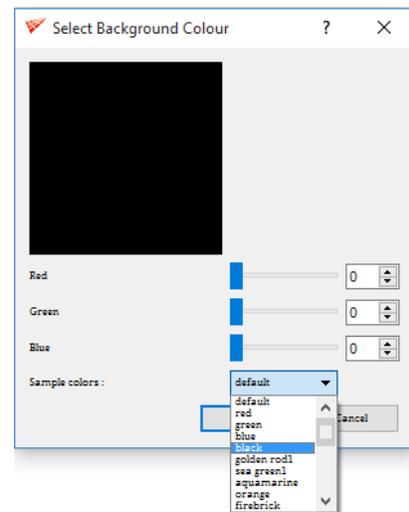


Figure 2.14: Background color dialog box

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. When starting a new session, it is advisable to redefine your working directory as all autosave files (\_auto.fra, \_az.fra, \_undo.stack) and grid files will be saved to that folder which can be accessed easily if 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 2.15. The dialog box allows you to create new folder/directory and select them or select the existing directory.

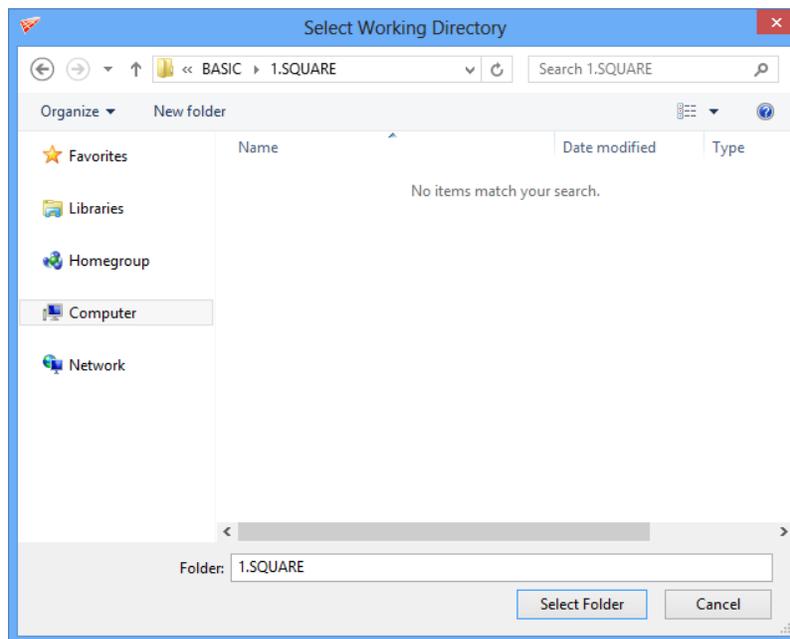


Figure 2.15: File dialog box for the working directory

When the topology or grid is loaded from the different directory other than the current working directory, the working directory will be updated to the recently used directory for loading the topology and it 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 2.16 until a valid directory is chosen.

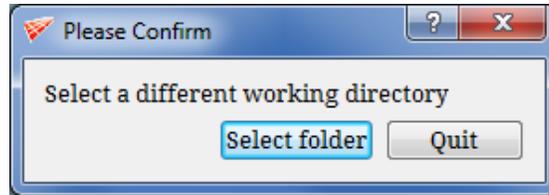


Figure 2.16: Working directory read/write permissions denied

When you quit the UI, 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 from next time. Make sure the settings are saved after you uncheck the box.

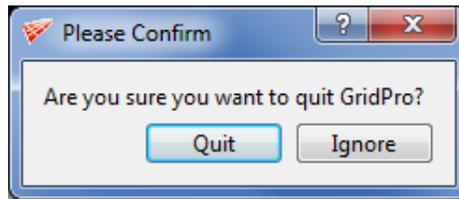


Figure 2.17: Confirmation dialog box

**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 2.18 shows various import geometry and grid formats available in the UI.

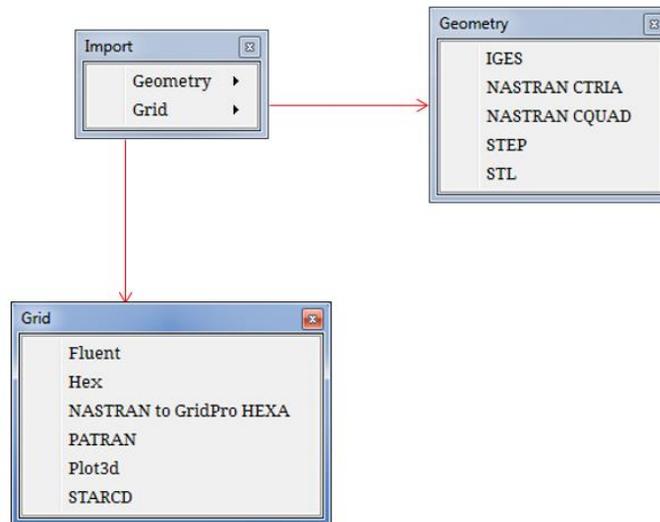


Figure 2.18: Import formats of geometry and grid

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.

**Export** option allows you to export various surface and grid 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 2.19 shows various export surface and grid formats available in the UI.

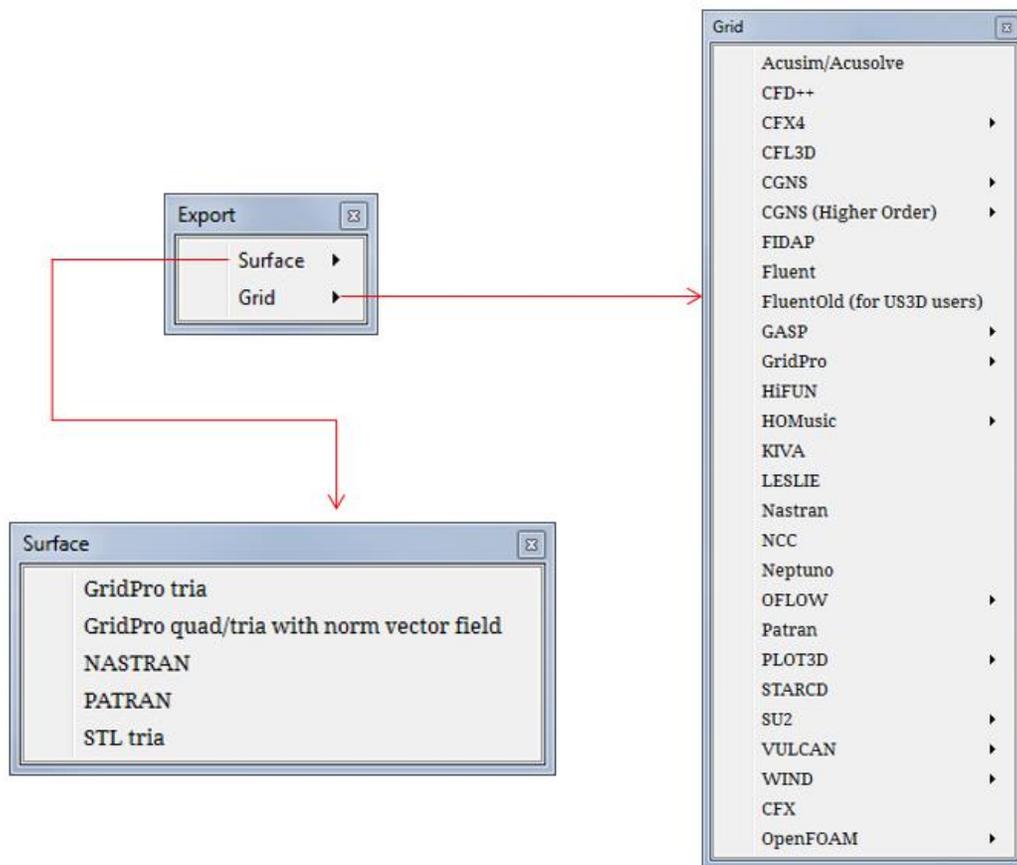


Figure 2.19: Export formats of surface and grid

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

Note that you can also export the grid from superblock format to 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 available only for these solvers.

The **Quit** option defined under File menu can be used to quit the application. A confirmation dialog box as shown in Figure 2.17 is popped up to ensure that it is not closed accidentally. The user has to make sure 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 2.20.

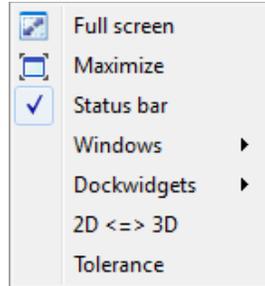


Figure 2.20: View Menu bar

To view the entire screen in Full-screen mode, select **Full-screen** option under the view menu as shown in Figure 2.20. The full-screen mode is shown in Figure 2.21. This hides the desktop panel, system tray, and all other visible applications, but the top dock, left side toolbar and the right side panels are remain unhidden. To bring it back to the default size, click on the '**maximize**' option, which retrieves the default screen size.

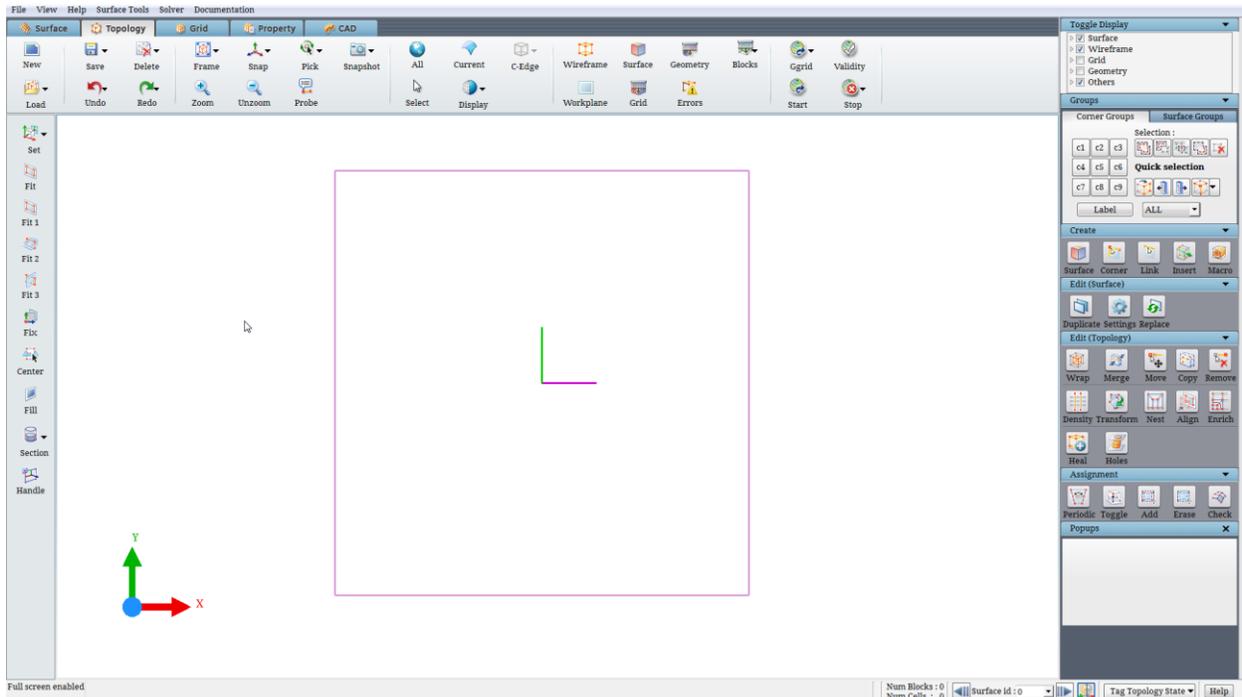


Figure 2.21: GUI in full-screen mode

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 Status bar, right side panel, left side toolbar and top dock can be switched off for display using options defined under **Dockwidgets** in View menu as shown in Figure 2.22. You can switch off all at a time as shown in Figure 2.23 or you can manually switch off one at a time based upon your necessity. This is useful when you want to increase the size of the graphics window. Even though graphics size might have been changed, but the objects in the graphics window may be disoriented and it needs to be updated by simple left-click in the drawing window which would redraw all the graphics objects and reevaluate the view volume appropriately.

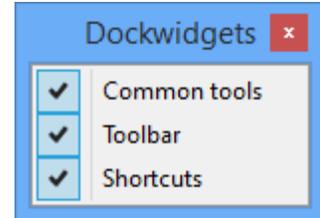


Figure 2.22: Dock widgets drop down list

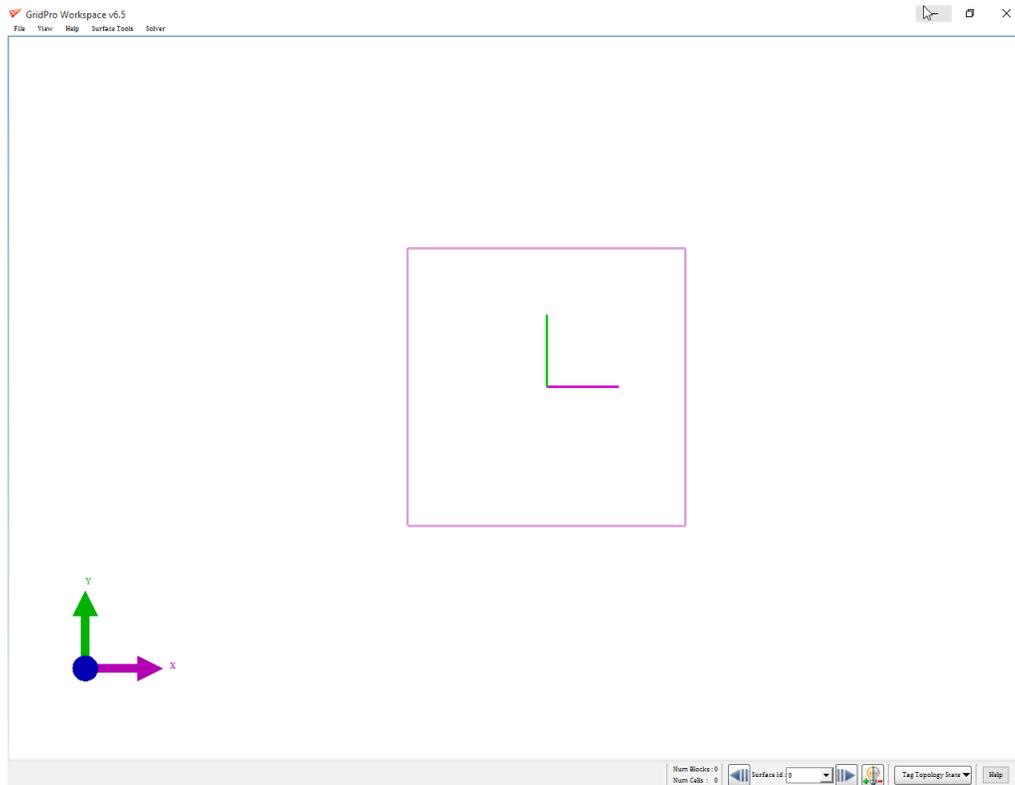


Figure 2.23: UI after switching off all the dock widgets

As you know, the UI can support both 2D and 3D geometries which can be chosen while opening a ‘New’ empty workspace. Suppose if you have chosen a wrong dimension or else if you want to load a 2D surface in a 3D workspace or vice versa, you can use **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 pop out before changing the dimension as shown in Figure 2.24.

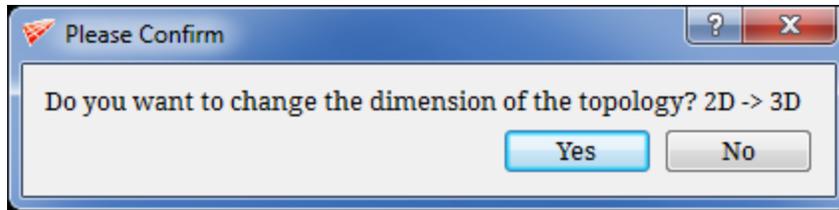


Figure 2.24: 2d to 3d confirmation dialog box

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.

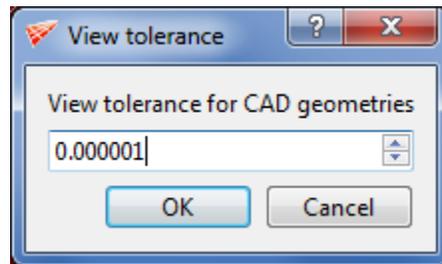


Figure 2.25: Tolerance dialog box

### 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 2.26 shows the options in the drop down of Help menu in top menu bar.

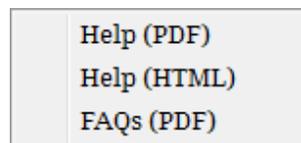


Figure 2.26: Help Menu

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 default pdf reader is opened.

## 2.2.4 Surface Tools

The **Surface Tools** menu has tools used for creating surfaces using wireframe. Wireframe defines the shape of the surface. Figure 2.27 shows the drop-down 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*.

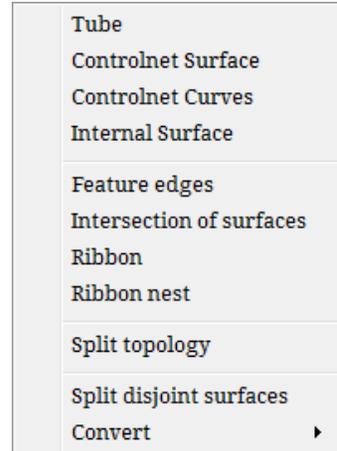


Figure 2.27: Surface Tools drop down list

- **Convert**

Convert is a file format conversion tool similar to Import options whereas this is specific to GridPro formats. Figure 2.28 shows the available conversions.

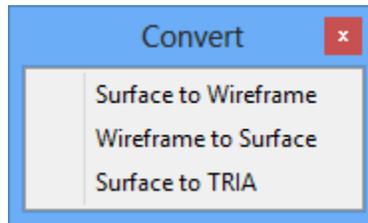


Figure 2.28: Convert drop-down list

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\* contains the surfaces to be converted should be selected in the respective field as shown in Figure 2.29.



Figure 2.29: Surface to Wireframe conversion

Wireframe to Surface: Converts a given corner group as a quad surface and loads into the UI. Once it is selected, Wireframe to Surface dialog box opens as shown in Figure 2.30 where a corner group and an output file name has to be provided.

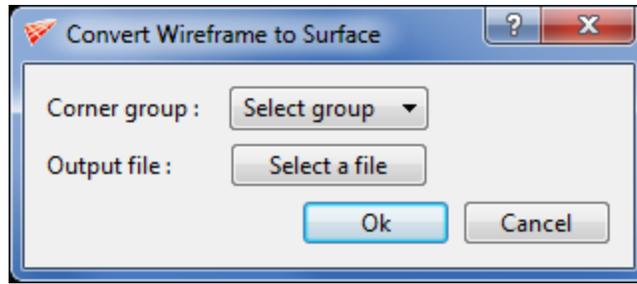


Figure 2.30: Wireframe to Surface conversion

**Surface to TRIA:** Converts a given quad or tube surface to a TRIA surface. Once selected opens 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 Solver menu

**Solver** menu in top menu bar is used for defining the solver format globally. All the surface properties and grid properties will be assigned according to the solver format selected. The various available solver formats can be seen in Figure 2.31.

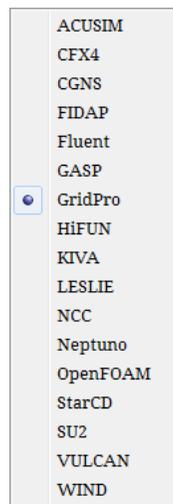


Figure 2.31: List of Solver formats

## 2.2.6 Documentation

Documentation menu in the top menu bar contains all the documents related to the GUI. It opens a drop-down list with two options, tutorials, and manuals. Each one further opens another drop down with the file names as shown in Figure 2.32. Clicking on any file name opens the respective document in PDF format separately.

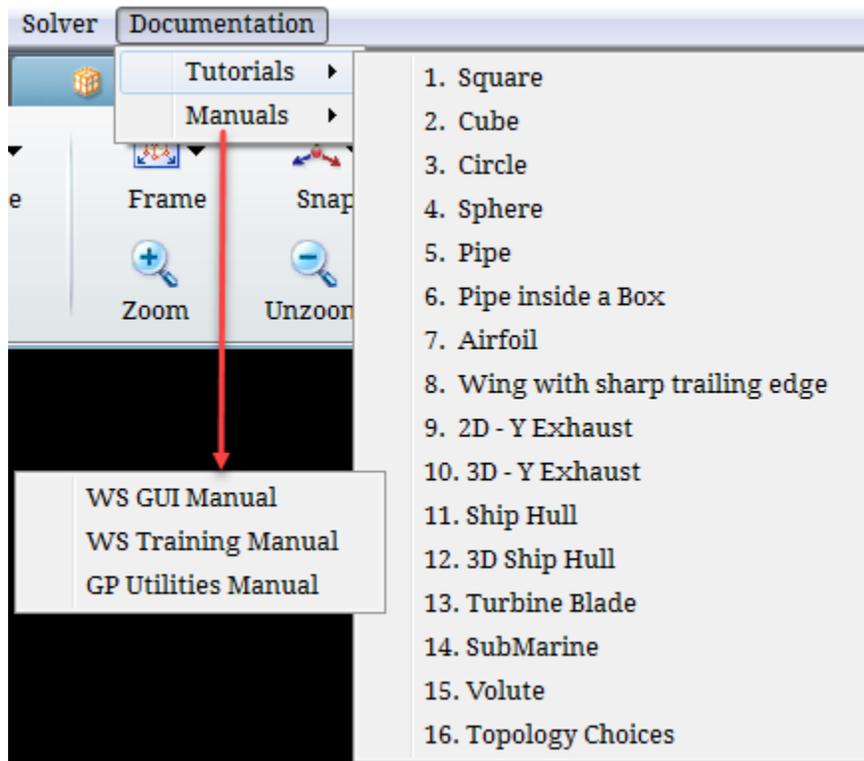


Figure 2.32: Documents Menu

## 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 global dock are discussed below.

### 2.3.1 New

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

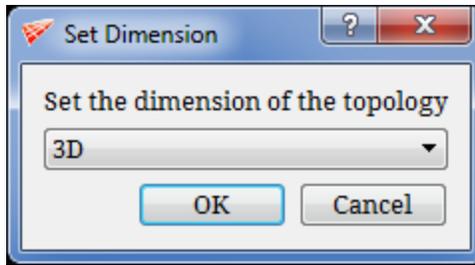


Figure 2.33: Set dimension dialog box

## 2.3.2 Load

The Load button helps to load a topology, surface, grid or a CAD file which you have created or you were working on it previously. To load surfaces into UI, use **Load** button  in the **Surface** tab located below '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) on the basis of order it is loaded into the UI.

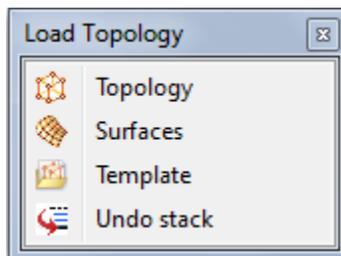


Figure 2.34: Load drop-down list in topology tab

Topologies which 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 2.34 shows the drop-down list of Load button in the global dock of the topology tab.

In addition to predefined templates, a user-defined topology can be loaded into 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 2.35 if **Load → Template** is selected. Select the file you would like to input, by pressing *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

wireframe for every modification becomes a painful and time-consuming process. It helps in loading the previously used wireframe and modifies it according to the surface modification.

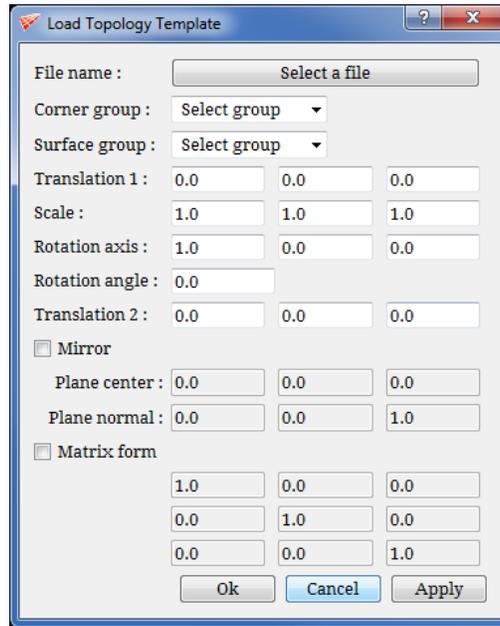


Figure 2.35: Load Topology Template dialog box

While you are working on any topology in GUI, it saves your topology by default every 2 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`. It is the topology file which is having the state before which you did any undo or redo operations. This file can be used to retain a session even after closing a session or if the current session has got 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 **Grid** tab or **Property** tab from **Load** → **grid** option. You can load multiple grids in the GUI.

If you want to repair/fix a geometry using *GridPro* tools, you can load the geometry files from 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 **Topology** tab you can save your topology. When you save a topology you save information related to surface files, its location and its association to wireframe in addition to the wireframe. It allows you to

save with 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 *section 4.4*]; to perform that you can use **Group** under *Save Topology* in **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.

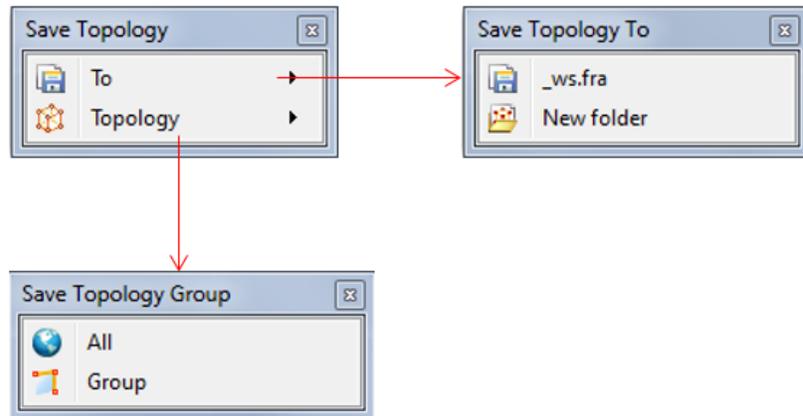


Figure 2.36: Save drop-down list in the topology tab

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 topology to another directory by using *Save to* → *New folder*  in **Topology** tab which allows you to save the topology to another directory, i.e. the wireframe and the associated surface files. It’s helpful for extracting files related to your topology and copying 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’.

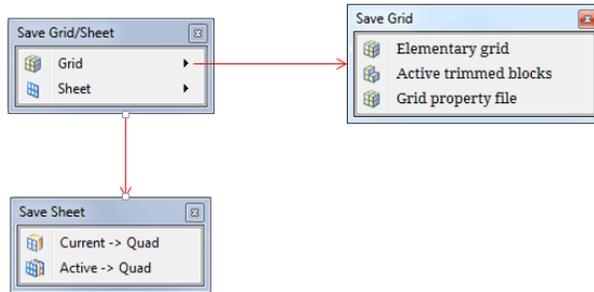


Figure 2.37: Save drop-down list in the grid tab

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 different name by selecting **Save → Grid → Grid property file**. Similarly to save *current* grid sheet, select the grid as *current*, select the sheet you wish to save as *current* sheet, and select **Save → Sheet → Current -> Quad** in the **Grid** tab. This grid sheet would be saved as quad format, which can be loaded into the GUI using **Load → Surfaces** option. You can also save all the visible grid sheets of the *current* grid by selecting the **Save → Sheet → Active -> Quad** option. All the options related to saving grid and saving sheet are also available in Property tab.

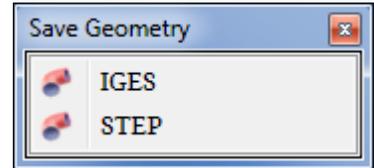


Figure 2.38: Save drop down list

In CAD tab, at any point of time, a CAD geometry can be saved into iges format or step format. **Save → IGES** option in CAD tab can be used for saving the geometry in .iges format. **Save → STEP** option can be used for saving the geometry in .step format. You can save a step format geometry into iges format and vice versa also.



Figure 2.39: Tag Topology State

Apart from these save options, you are provided with one more save option called **Tag Topology State** which can be located at the right side bottom in the status bar as shown in Figure 2.39. 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 back 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 but the tag topology state can be reverted back to a previous tag and

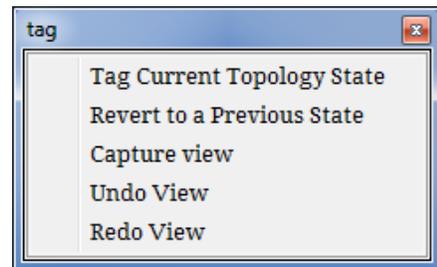


Figure 2.40: Tag topology state drop down list

the topology can be modified. Once it is reverted back to a previous state, all the tags after that will be removed. Clicking on the button will show the drop down as shown in Figure 2.40. To tag a state, select **Tag Current Topology State**, you will be prompted with a pop-up where you need to input a name for tagging the state. In order to revert back to a previous state, select **Revert to a Previous State**, a window will pop-up as shown in Figure 2.41 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.

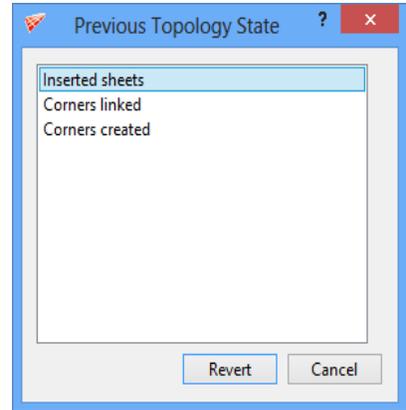


Figure 2.41: Revert to previous state dialog box

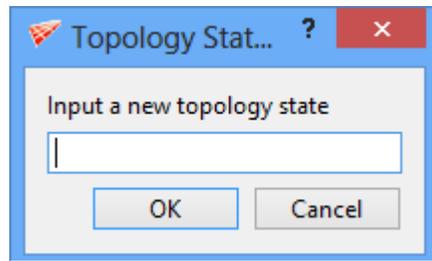


Figure 2.42: Tag current topology state dialog box

Next 3 set 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. More often you tend to change the screen view by mistake or it gets changed 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 point in time. When you want to go back to the previous view, click on Undo view option under the drop down, 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 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 rotation/zooming to get to that desired view.

### 2.3.4 Delete

You can delete any surface, any topological entity or any entity related to grid using **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 **Surface** tab or using **Delete → Surface → Current** in **Topology** tab. You

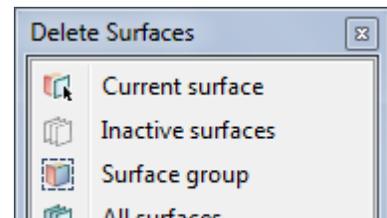


Figure 2.43: Delete drop down list in surface tab

have the facility to display few surfaces in GUI and hide few surfaces. The hidden surfaces in GUI are called *inactive surfaces*. The surfaces that are visible are called *active surfaces*. You can delete all the hidden surfaces at a time by using option **Delete → Inactive surfaces** in the **Surface** tab or using **Delete → Surface → Inactive** in **Topology** tab. You can delete all the surfaces by using **Delete → All surfaces** in **Surface** tab or using **Delete → Surface → All** in **Topology** tab. There is also an option available to group different surfaces. You can access surface groups in different ways which will be explained in *section 4.4*. You can delete a specific surface group by using option **Delete → Surface group** in **Surface** tab or using **Delete → Surface → Group** in **Topology** tab. You can delete all surfaces whether visible or hidden in GUI by using **Delete → All surfaces** option in **Surface** tab or using **Delete → Surface → Group** in **Topology** tab. 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.

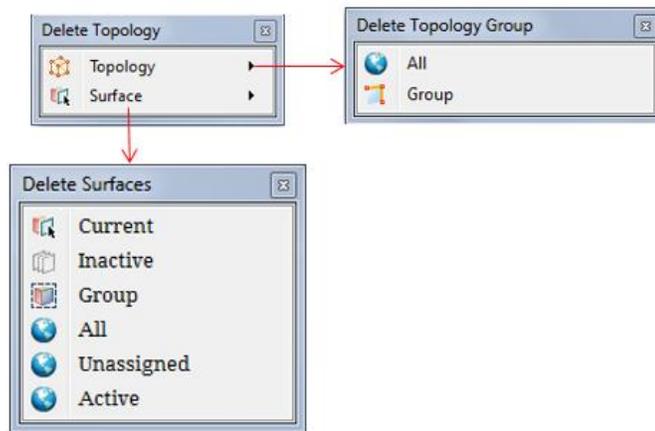


Figure 2.44: Delete drop-down list in the topology tab

As discussed earlier, you can load multiple grids in the GUI. The grid that 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 **Grid** tab and the Current sheet can be deleted using **Delete → Sheet → Current** option. You can hide the unwanted sheets in **Grid** tab to avoid clutter, and hidden sheets can be deleted using **Delete → Sheet → Inactive** in **Grid** tab. All the options related to deleting grid is also available in Property tab.

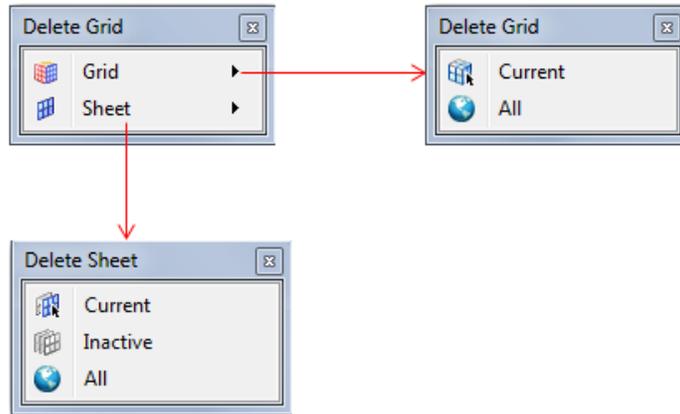


Figure 2.45: Delete drop-down list in Grid tab

Delete button in CAD tab can be used for deleting the geometry loaded into UI. It will pop-up a confirmation dialog box as shown in Figure 2.46 if clicked.

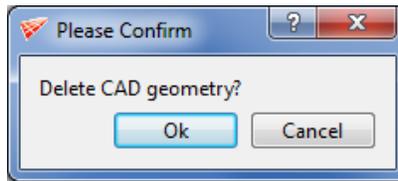


Figure 2.46: Delete confirmation dialog popup

### 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 topology you have created from any point in the

process. Also, you can scroll forward using **Redo** button ; if you realize that you should not change anything. Both the buttons are disabled in all tabs except topology 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.

<b>Simple Insert : 695</b>
<b>Block Insert : 542</b>
<b>Multi Insert : 360</b>
<b>Box Macro : 98</b>

Figure 2.47: Undo/Redo drop down list

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 Now all the major operations are saved with their names under undo/redo drop-down list. So you can directly undo from 10<sup>th</sup> step to 5<sup>th</sup> step with a click of a button by selecting the desired operation from the drop-down. Though you can undo/redo up to 32 times, only 10 operations would be shown in the drop-down at any point in

time. All the minor operations which are not displayed under undo/redo drop down are still exists in the background and it can be undo/redo by simply clicking on the respective button.

### 2.3.6 Frame

When you are maneuvering 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 **Frame** button



located in the global dock. **Frame** → **Current surface**

option will change the view to current surface. The current surface will fill up more than 80% of the screen display. Also, the center of rotation is set to center 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 drop-down list of frame button in surface and topology tabs are shown in Figure 2.48 & Figure 2.49 respectively

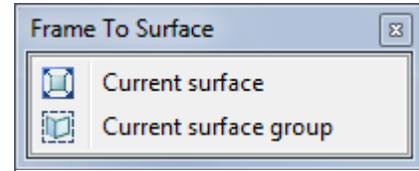


Figure 2.48: Frame drop down list in surface tab

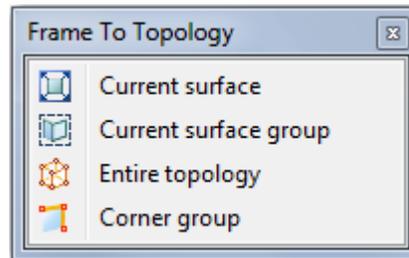


Figure 2.49: Frame drop down list in topology tab

Figure 2.50 shows how the frame option behaves. Figure 2.50 shows the zoom-out view of a topology with active surface group and corner group. Figure 2.50a shows frame to Current surface view. Figure 2.50b shows frame to active Surface Group view. Figure 2.50c shows frame to Topology view. Figure 2.50d shows frame to active corner group view.

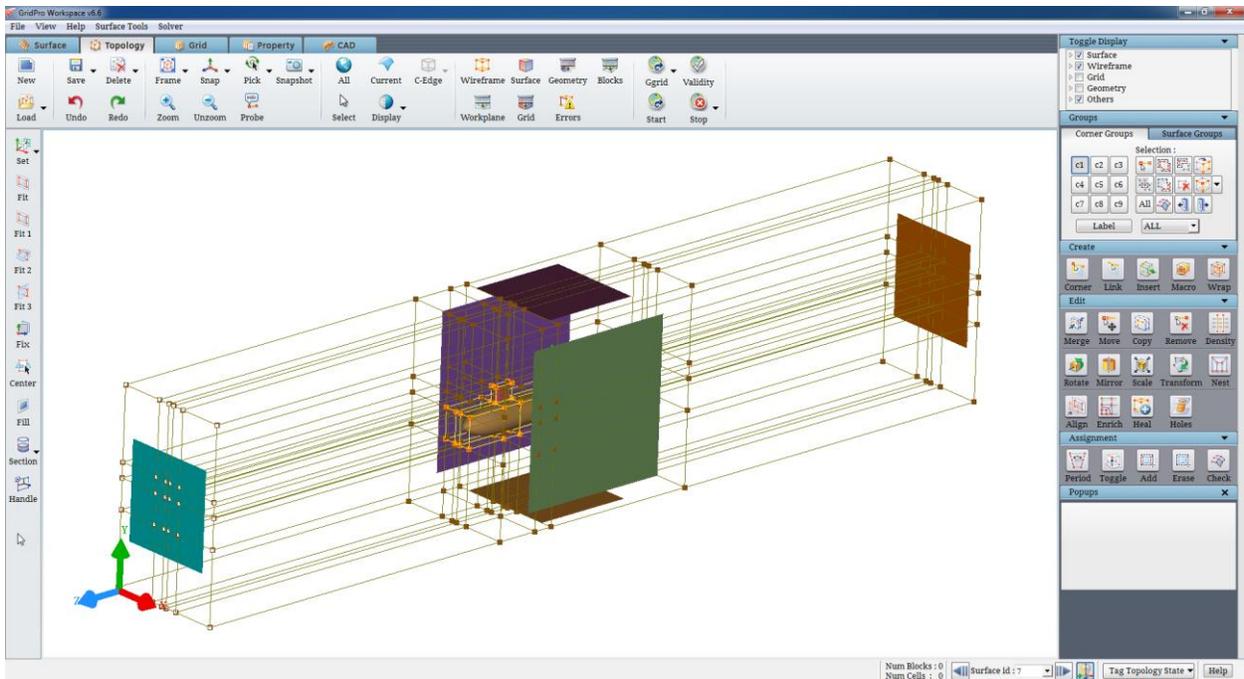


Figure 2.50: Original View Position

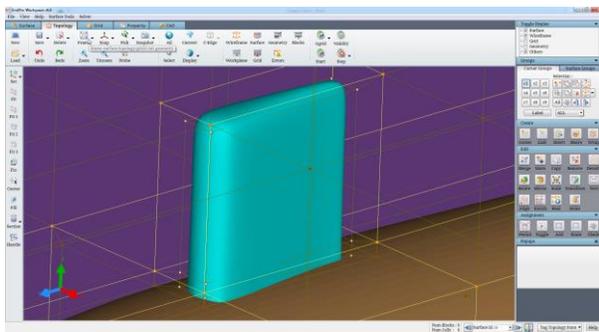


Figure 2.50a: Frame to Current Surface

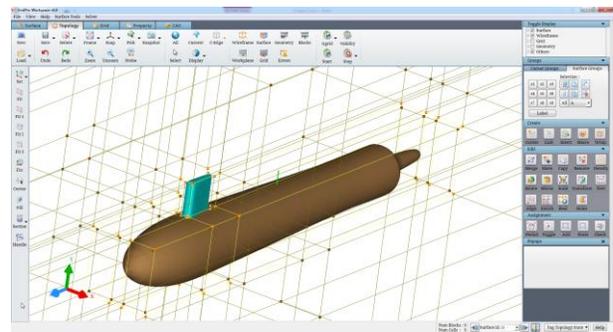


Figure 2.50b: Frame to Current Surface Group

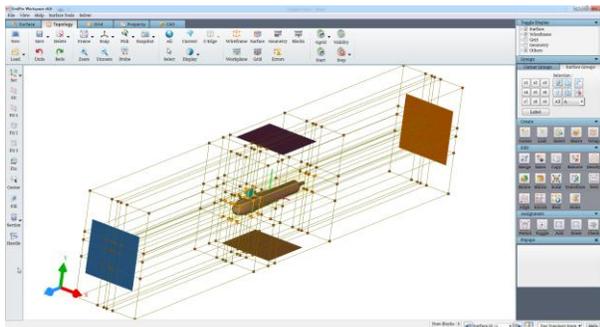


Figure 2.50c: Frame to Topology

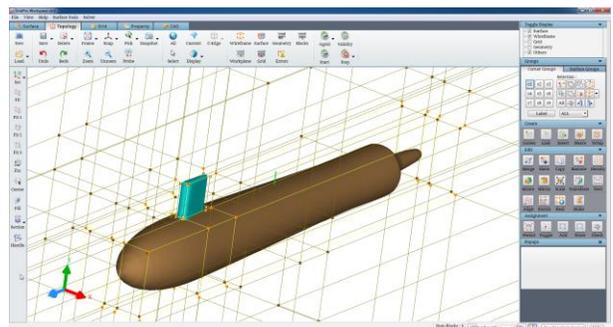


Figure 2.50d: Frame to active corner group

In Grid, Properties and CAD tab, **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 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 as shown in Figure 2.51. *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 *closest* option, here the workplane axis is aligned to the closest screen axis.

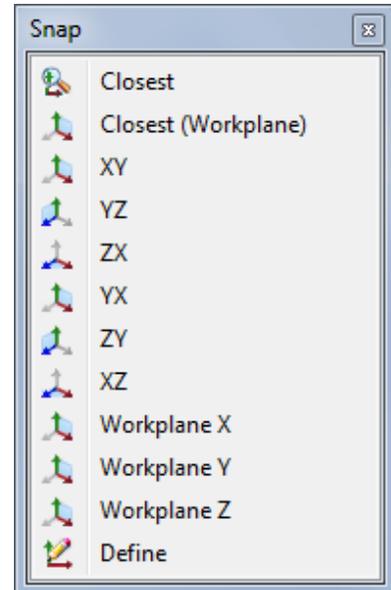


Figure 2.51: Snap drop down list

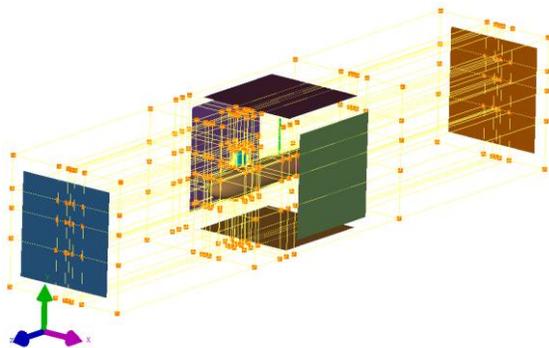


Figure 2.51a: Rotated View

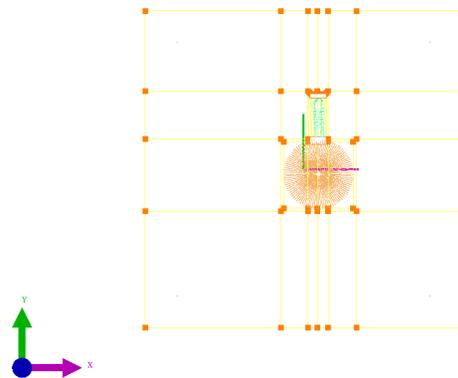


Figure 2.51b: Snap to the closet View i.e. XY plane

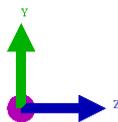
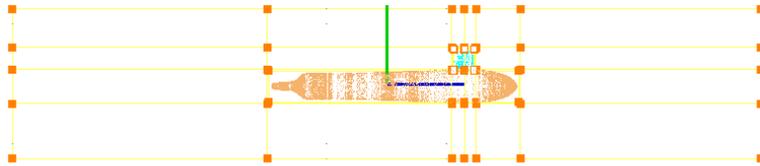


Figure 2.51c: Snap to the ZY plane

You can also snap screen axes to user-defined axes by using **Define** option  under **Snap** menu. If selected, it will pop out a dialog box as shown in Figure 2.52. For that, you need to input the orientation and up direction, where orientation axis is pointing towards the screen and up direction is the axis pointing upwards. The third direction is automatically calculated.

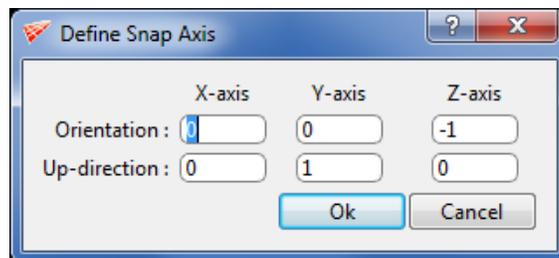


Figure 2.52: User-defined snap axis dialog box

### 2.3.8 Zoom/Unzoom



Zoom



Unzoom

Zoom  and un-zoom  helps in zooming/un-zooming a portion of the screen with respect to the center of the screen. The right mouse button can also be used for zooming into a portion of the screen in order to dilate that portion for a more detailed view. By clicking and dragging a box using the right mouse button, you form a white zooming rectangle. The point where you click 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 zooms in on the specified rectangle automatically. 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 mouse scroll to zoom in and zoom out when you scroll out it zooms out and when you

scroll in zooms in. When the zoom or un-zoom is done using mouse scroll, the center of the screen is assumed as zoom/un-zoom center.

### 2.3.9 Pick

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

of rotation can be picked using **Pick**  option located in the global dock. When the button **Pick** is selected, any entity (corner, edge, surface, edge of the grid, grid sheet or geometry) can be picked using the left-button of the mouse to set that point as the center 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 workplane is not treated as an object and hence cannot be selected using this option.



Figure 2.53: Pick drop-down list

The center of rotation can also be picked using a custom right-click menu of the entities in the graphics window. By holding keyboard shortcut 'P' key and clicking on any point on any entity you can pick the center of rotation. Instead of picking, you can fix center of rotation to the center of view volume or center of the screen by using **Reset**  in **Pick** drop-down list. By selecting **Origin** , you can fix the center of rotation at the origin of the coordinate system. You can also fix the center of rotation to the mass center of the model by selecting **Model** option in **Pick** menu. By selecting **ALL** under **Pick** will set the center of topology as the center of rotation. Else you can define the x, y, and z coordinates for the center of rotation by selecting **Define** in **Pick** menu. Figure 2.54 shows the dialog box of Define COR option.

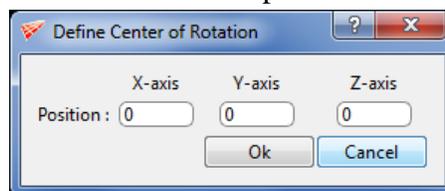


Figure 2.54: Define COR dialog box

### 2.3.10 Probe

You can get the information about topological, surface and grid entities by using *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 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 which is popped up after the selection. When this button is clicked, it is pressed down until the selection of an object from the graphics.

When a corner is selected, the corner data is shown in the tabular form as shown in the Figure 2.55. The corner index, it's 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.

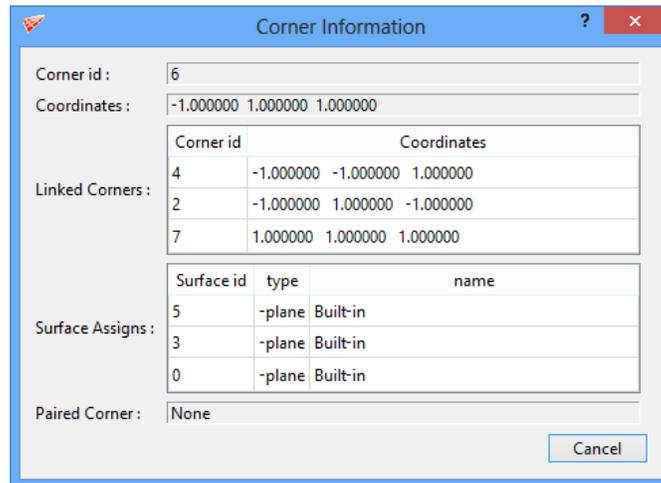


Figure 2.55: Corner information

When an edge is selected, the edge data is displayed in a dialog box as shown in the Figure 2.56. The edge index, its end vertices, its length and the density of the edge 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.

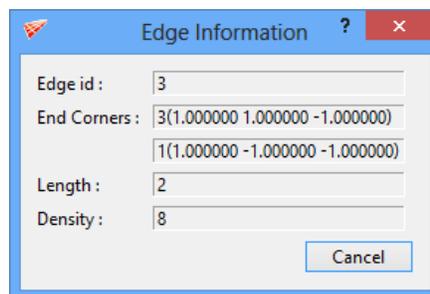


Figure 2.56: Edge information

When the surface is selected, the surface data is displayed in the dialog box as shown in the Figure 2.57 includes it's 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.

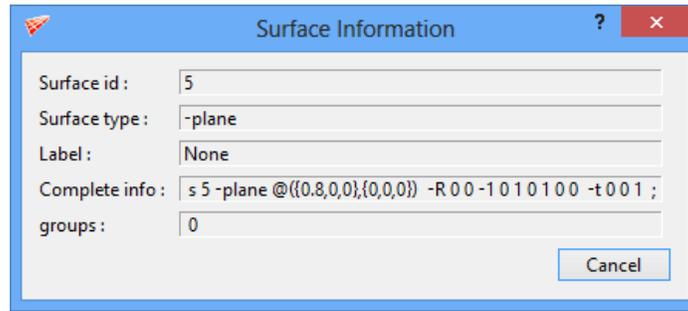


Figure 2.57: Surface information

When a grid block edge is selected, its information is displayed as shown in the Figure 2.58. The block edge information consists of the block index, the edge index within the block and the density of the edge.

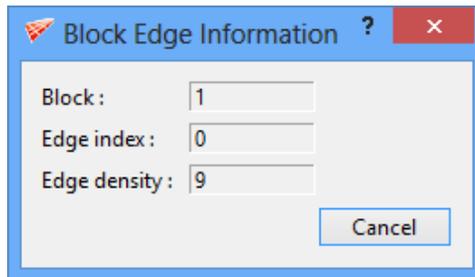


Figure 2.58: Grid block edge information

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 doesn't just contain  $x * y$  number of points on it to display its size. Hence, the only useful information that can be displayed is whether a grid sheet is a surface sheet or not.

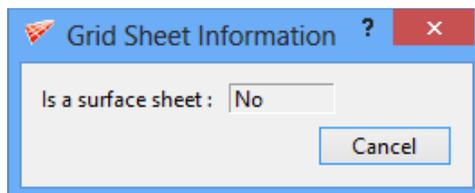


Figure 2.59: Grid sheet information

### 2.3.11 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 2.60. By default, if clicked, it captures the GUI region as shown in figure 2.60a. Entire UI option captures the whole GUI and Graphics window option captures the graphics window of GUI as shown in figure 2.60b. 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 Current working directory. You can change the path of the file. You can save the file in png, and .bmp format.

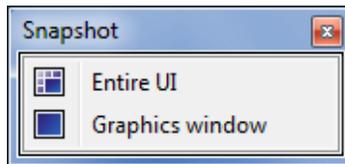


Figure 2.60: Snapshot drop-down list

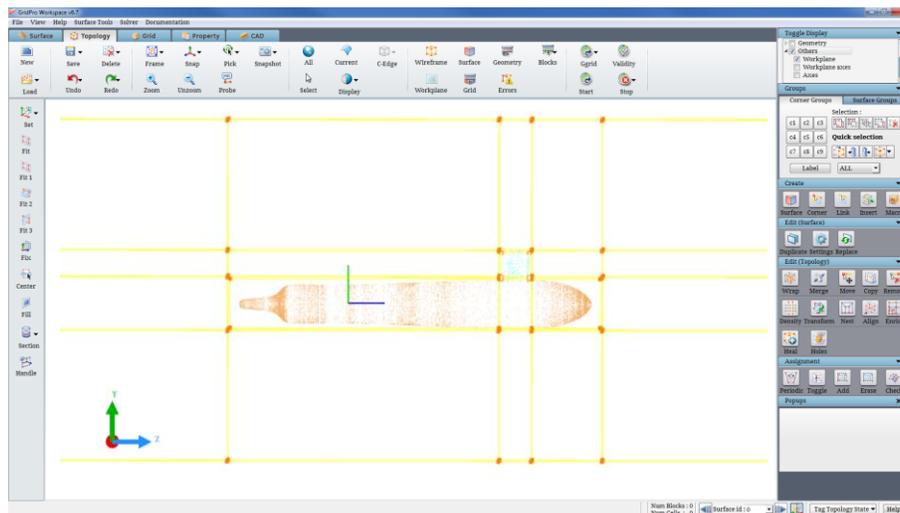


Figure 2.60a: Snapshot of Entire UI

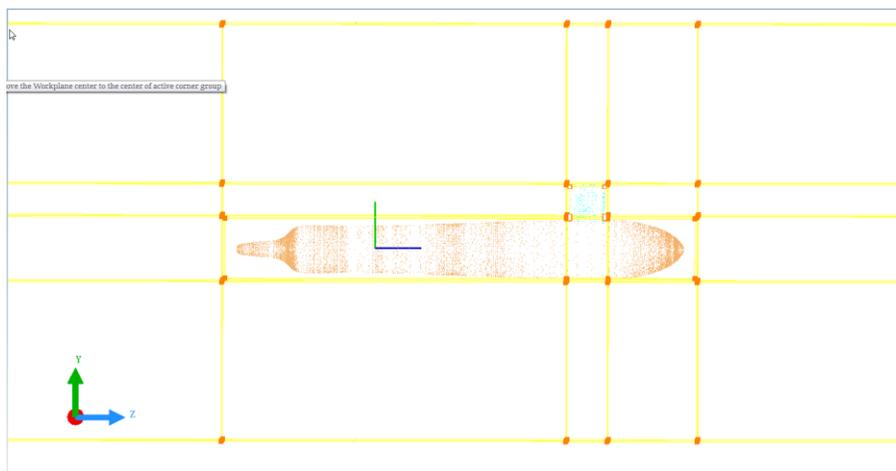


Figure 2.60b: Snapshot of Graphics Window

### 2.3.12 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.

### 2.3.13 Current

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



all surface (excluding the current surface) by using **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 rest all surfaces are hidden including the previous *current* surface.

### 2.3.14 Select

You can also pick a surface to be *Current Surface* by using **Select** button . Depress **Select** button and click on the desired surface you want to select as the *current* surface. Its color would be changed to sea blue color and its id would be updated in the index field.

### 2.3.15 Display

Surfaces, geometries, and grid 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

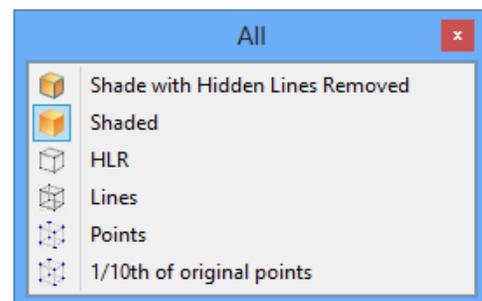


Figure 2.61: Different display modes

When an object is set to **Shade with HLR**  visualization mode, the polygon faces of the object are drawn using the object's own color but the polygon boundaries (edges) are drawn in white. Figure 2.62 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 object's own color. The polygon boundaries (edges) are not drawn at all. The objects on the other side of the object cannot be seen because of opacity.

In **HLR** (Hidden Line Removal)  visualization mode, the polygon faces of the object are drawn using the background color and the polygon boundaries (edges) are drawn using object's own color. 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 object's own color. In this display mode objects on other side are visible.

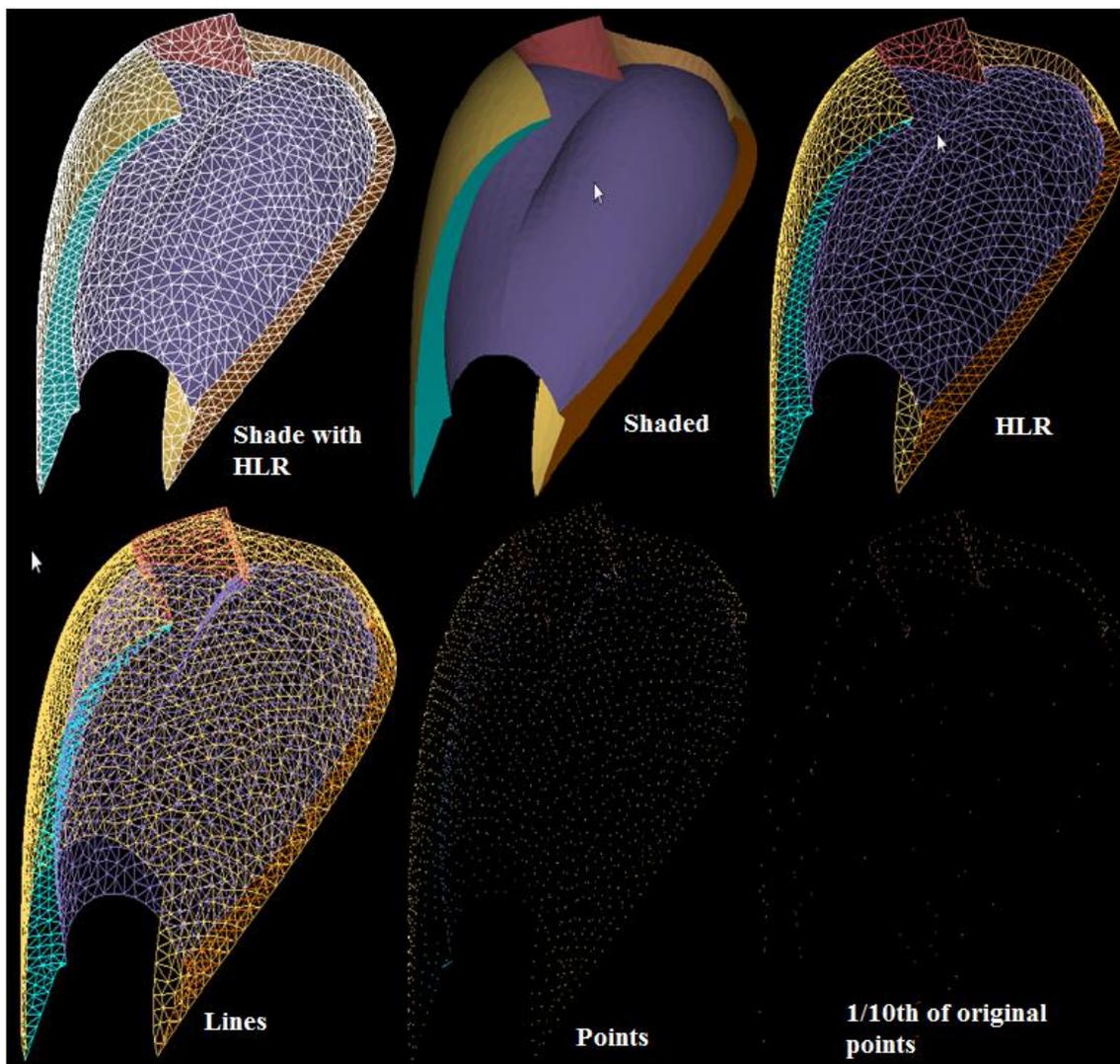


Figure 2.62: Surface with different display modes

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 very preferable for large surfaces.

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

By default, all surfaces are displayed as shaded. The visualization for all surfaces can be changed by specifying the display type under **ALL** drop down 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 in a better way. It can be done by specifying display type for the Current object under the **Current** drop-down menu. The visualization mode of the moving objects (while panning, rotating or zooming) can be set separately by specifying display type **Moving** drop-down, 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 it 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** drop down, one more option is available i.e. **Follow all surfaces display type**, it copies the display type of all surfaces to current surface or while moving surfaces for whichever option is selected.

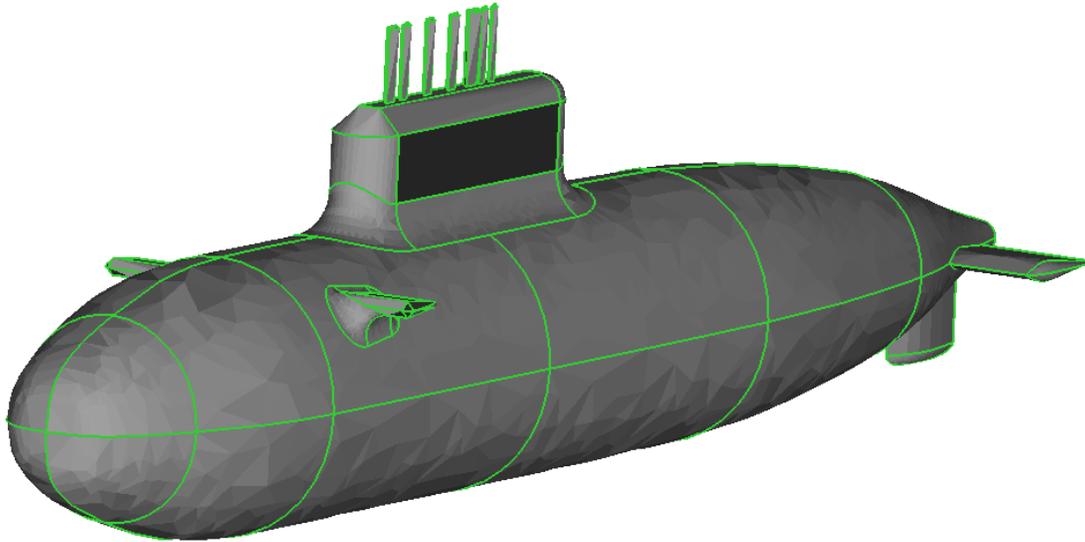
Note that the visualization type can also be changed/set by right-clicking on the object and selecting the appropriate visualization mode (will be explained later). In this case, this mode which is set manually from the right-click custom menu overrides the mode selected through this display menu.

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.

### 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 2.63. If an edge is defined by 3 faces and 4 faces, it is shown in yellow and bright blue colour respectively.



*Figure 2.63: 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 its related objects such as border, path selection, and feature edges.

### **2.3.19 Geometry**

Clicking on this button will toggle the geometry and its 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 UI.

### **2.3.22 Blocks**

Clicking on this button will toggle the shrink blocks in the UI. Please refer section 2.6 for more information on shrink blocks

### 2.3.23 Grid

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

### 2.3.24 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.

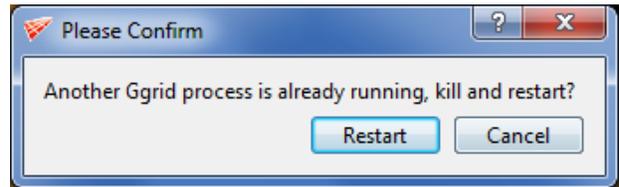


Figure 2.64: Ggrid error message

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 2.64. If there is no topology, an error message is popped-up as shown in the **Error!**

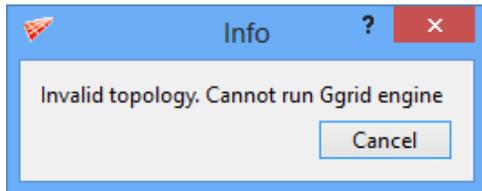


Figure 2.65: Ggrid error message

**Reference source not found.** GridPro will also refuse to begin the gridding process if a 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 highlight which serves to show 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 2.66. You can choose to run either Navier Stokes or Euler grid using the check button. In case if you want to run a previous computation, enable the Restart checkbox and input the corresponding grid file. The output 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. All the edge densities of the topology can be doubled, quadrupled, halved or quartered. 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 dump frequency is the sweep interval after which the output grid is rewritten. This is typically a multiple of hundred.

Then the output file to which the grid is saved can be specified. The grid can be either written in ASCII format or in Binary format. Grid in binary format is first converted into ASCII format and then loaded into UI. GUI automatically detects whether the grid trying to load is ASCII or Binary. Binary grid cannot be reloaded into UI while running Ggrid because the grid loaded into UI is converted from binary to ASCII. If you want to reload the binary grid, delete the existing grid and load it again. The specified file would be written in the current working directory. The log file and all other output files of Ggrid are always written to the current working directory.

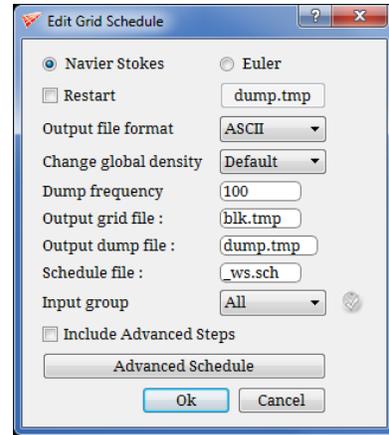


Figure 2.66: Grid schedule dialog box

You can also save the schedule file parameters to user-specified name using the Schedule file input box. Changing the default name ‘\_ws.sch’ will add a checkbox “Update \_ws.sch” file below. Note that the user-specified name will not affect the gridding process run from the UI unless the \_ws.sch file is also updated. The advantage of changing the schedule file name helps in running ggrid from the command terminal.

From v6.6, you can also run Ggrid on part of a 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 gets 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.

You can also input advance schedule options by clicking on the Advance Schedule button. Clicking on it will pop\_up a dialog box as shown in Figure 2.67. Loop count is nothing but the step numbers that are mentioned in the \*.sch files. Rest all are self-explanatory options.

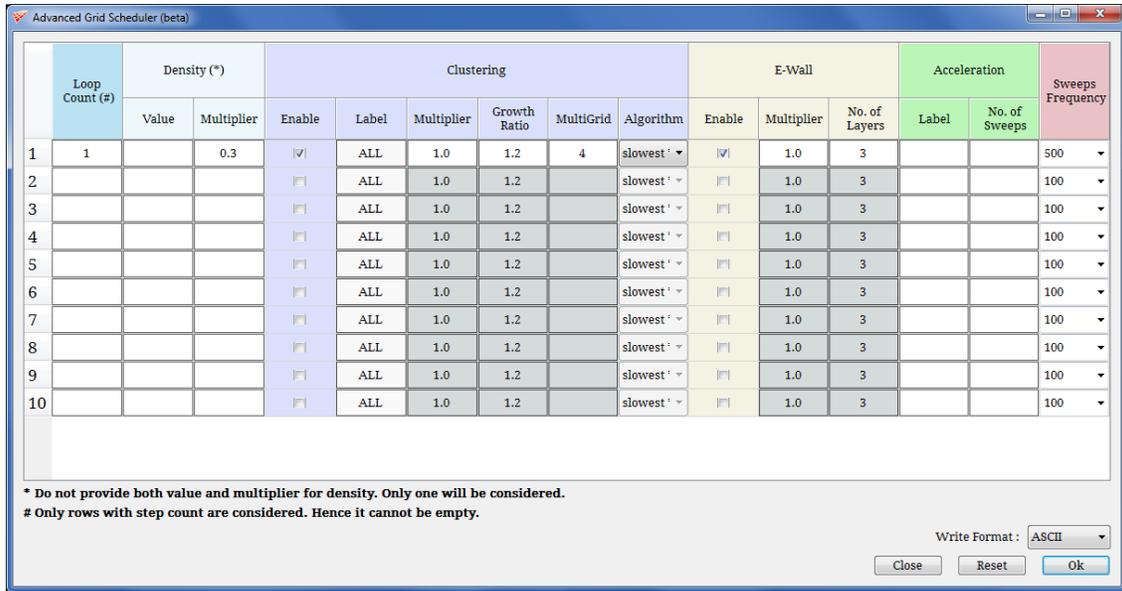


Figure 2.67: Advanced Grid Schedule option

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

The schedule file can be also edited by selecting **Schedule** option under **Ggrid** drop-down list as shown in Figure 2.72.

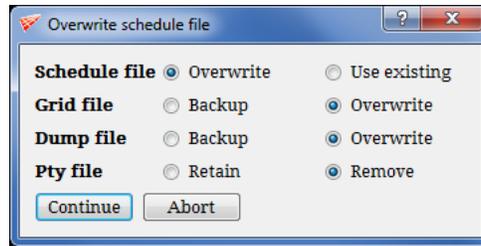


Figure 2.68: Grid file name confirmation dialog box

Note that if the specified schedule, grid, dump, and pty file name already exists in the working directory, a confirmation pop\_up will open as shown in Figure 2.68. For the schedule file, you can either overwrite or use the existing one by selecting the appropriate option. Similarly, you can either back-up the existing grid or dump file or you can overwrite the grid and dump file. It is also applicable to pty file, you can either retain the existing pty file or you can remove the pty file.

If the topology is disjoint, it pops out an error message as shown in Figure 2.69.

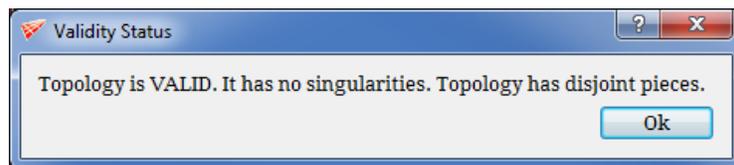


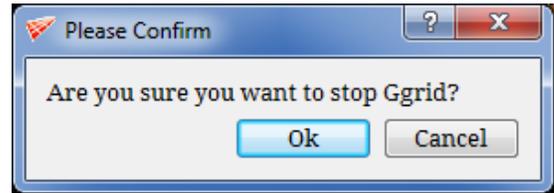
Figure 2.69: Disjoint pieces error message

### 2.3.25 Stop Ggrid

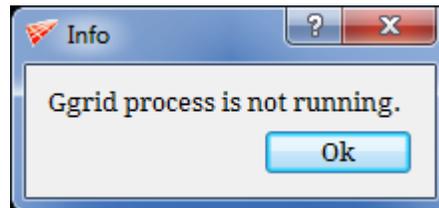
You have to stop the Ggrid engine manually, once started. To stop the Ggrid click on **Stop Ggrid**

icon  located in the global dock. If Ggrid is running, a confirmation dialog is popped-up as shown in the *Figure 2.70*. But if the grid generation engine is

not running, an error message is displayed (see *Figure 2.71*). Sometimes encounter a situation where the UI would have crashed while running Ggrid or you would have stopped the Ggrid but the process will be still running in the background. In such cases, you cannot run a new Ggrid process; click on **Kill Process** under **Stop** button to resolve the issue on such scenarios. If Ggrid is tried to stop while it is writing the grid file, it will finish the writing first and then it will stop the process. In a case, if you don't want it the write the grid file, *Kill process* option can be used.



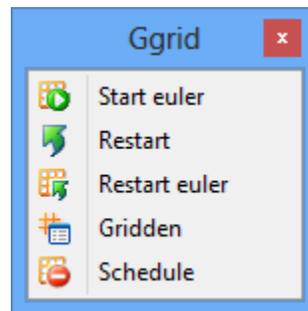
*Figure 2.70: Confirmation dialog box*



*Figure 2.71: 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 also stopping the grid writing. It does not pop-up for confirmation box also.

### 2.3.26 Ggrid



*Figure 2.72: Ggrid drop-down list*

- **Restart**

Once GGrid is stopped, and after looking at the grid you realized that it would have been better to run it for few more sweeps to get a good quality grid, you can use **Restart** option  under **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 is popped-up as shown in the **Error! Reference source not found.** and if the topology is invalid, an error message is displayed as shown in Figure 2.64. If the topology is valid, a file dialog box as shown in the Figure 2.73 is displayed to select the appropriate dump file (having *.tmp* or *.grid* extension). By default, the dump.tmp file in the current working directory is selected.

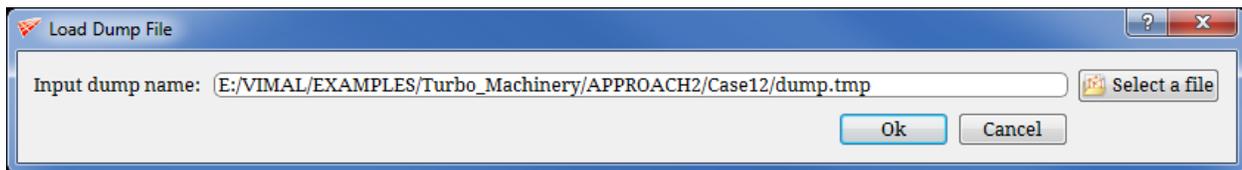


Figure 2.73: Restart Ggrid dialog box

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 was 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 this, it behaves in the same way as the **Start GGrid**.

- **Restart Euler**

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

- **Gridden**

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

- **Schedule**

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

### 2.3.27 Validity

Before starting Ggrid, at any stage validity of



topology can be checked using **Validity** button . If

the topology is valid, it will pop-out a message as shown in Figure 2.74. If the topology is invalid, it will pop-out an error message as shown in Figure 2.75.

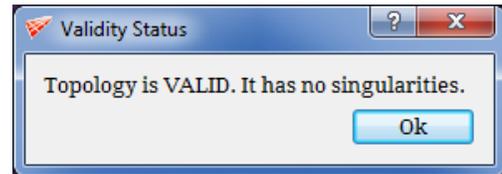


Figure 2.74: Valid topology status

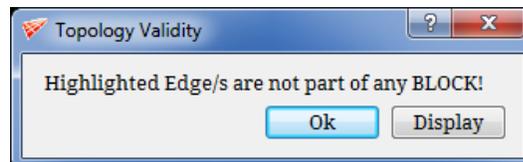


Figure 2.75: Invalid topology status

## 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 color as shown in Figure 2.76. This rectangle is called the workplane rectangle. It can be shrunk or expanded in order to work on a global or local scale.

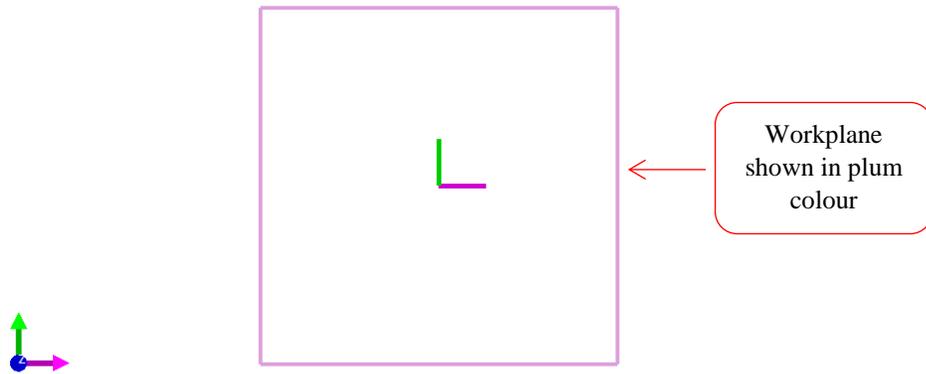


Figure 2.76: Workplane is shown in plum colour

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 workplane are located in the left side toolbar as shown in Figure 2.77 and can be accessed from all four tabs.

- **Workplane Axes**

Similar to global axes, there is a set of workplane axes that is represented in pink, green and dark blue color. Along with axes, there are four violet squares, each of which is placed on the vertex of the workplane rectangle as shown in Figure 2.78. 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 to 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.

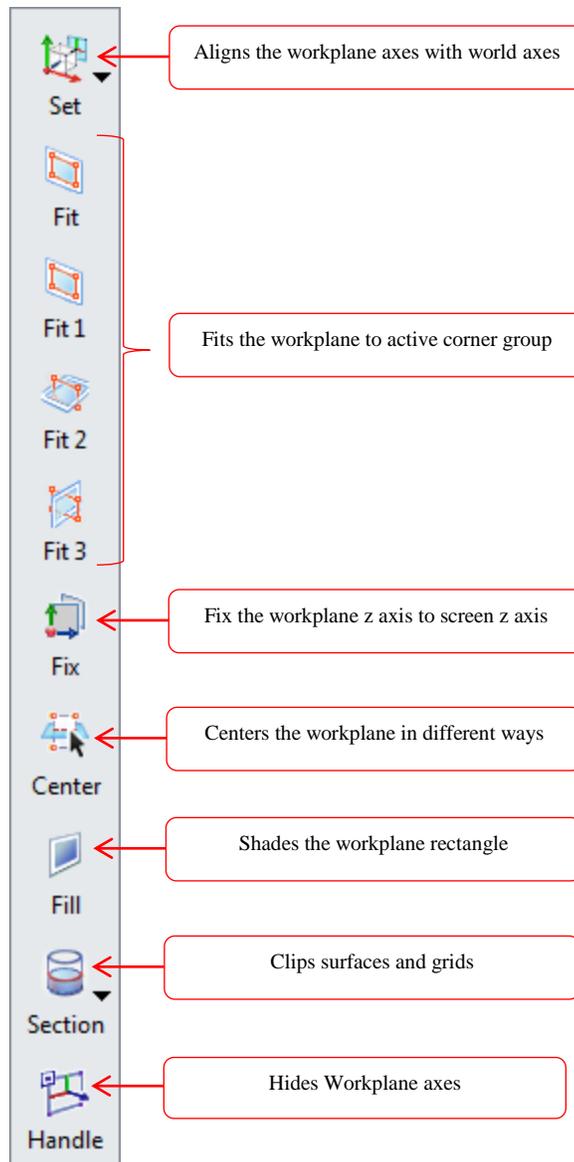


Figure 2.77: Workplane toolbar

Usually, workplane axes are referred as handle as it helps you to 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) in order 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 center dot.

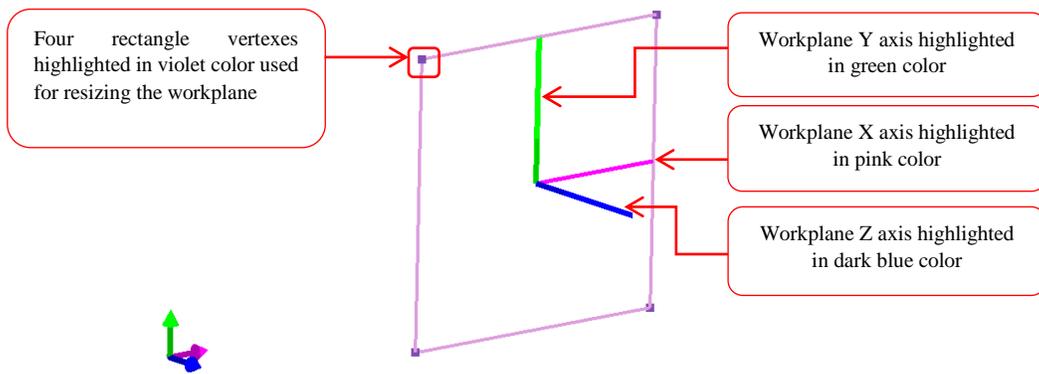


Figure 2.78: Workplane with Handles switched ON

## 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 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 2.79. 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 center of the workplane axis with the center of the respective screen axis.

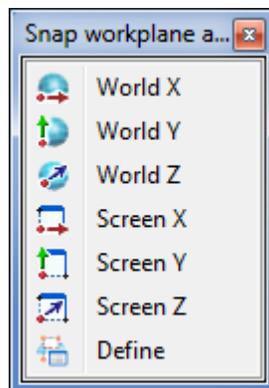


Figure 2.79: Set drop-down list

It can also be positioned anywhere you want if you have the center and normal of the desired location using the **Define**  option under **Set** drop down. For define option, the *normal* and *center* 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 center 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 *center* 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** in order to implement your input as shown in Figure 2.80. The workplane should immediately conform to its new properties. The workplane center 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 center of the workplane and the user can define normal and size of the workplane.

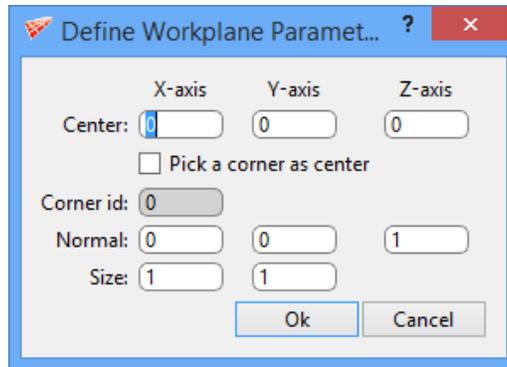


Figure 2.80: Define Workplane Parameters dialog box

## 2.4.2 Workplane Group fit

In three dimensions, you will almost surely use both corner groups and the workplane that are useful in creating a topology with ease. Therefore, *GridPro* provides you with the operations that integrate group and workplane features. Mastering these operations will help you maneuver 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 center of workplane rectangle with the center of mass of the active group. The other fit buttons (**fit1** , **fit2** , **fit3** ) centers the workplane in all three axes with respect to different reference frames. The **Fit 1** operation aligns the center of the workplane with the center of mass of active group and also 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 center of workplane with the center of mass of 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 2.81 displays how the workplane changes position for different fit options for the same group of active corners.

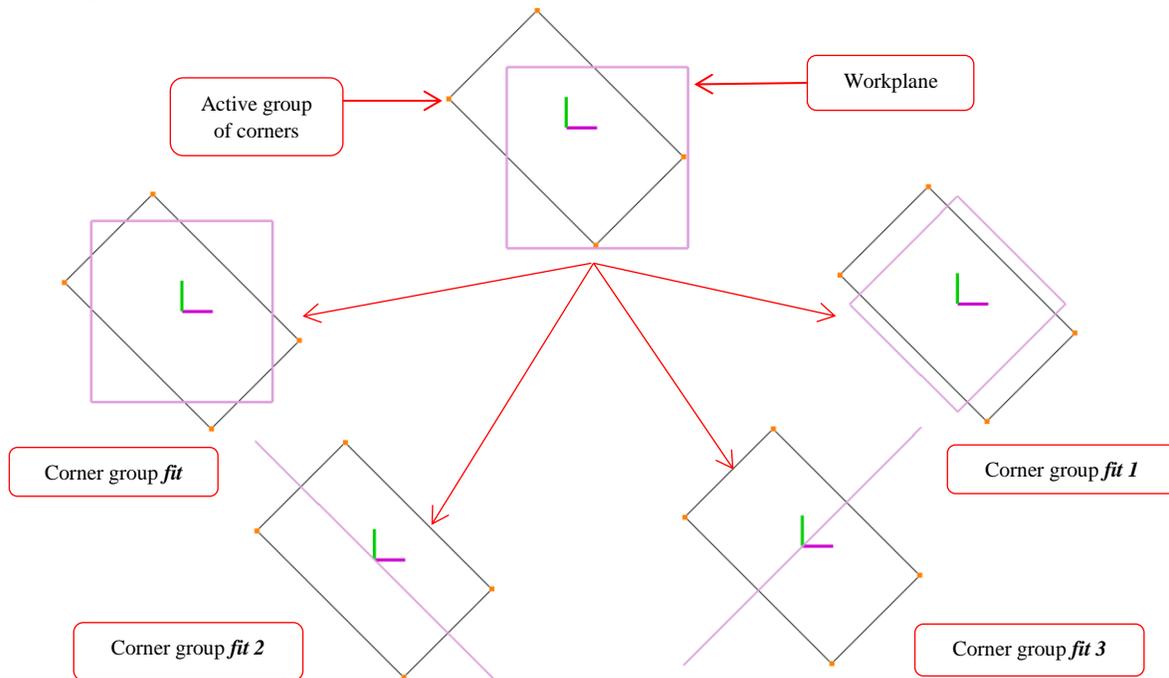


Figure 2.81: Group fit position for active corner groups

### 2.4.3 Fix

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

### 2.4.4 Center

You can position the workplane center on any surface or any corner/edge on the wireframe or the grid sheet with just a click on it using the **Center**  button in the workplane toolbar. Click on the **Center** button and click on any surface or corner/edge to transform the center of workplane to the clicked position.

## 2.4.5 Workplane Fill

**Fill** operation of 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 2.82.

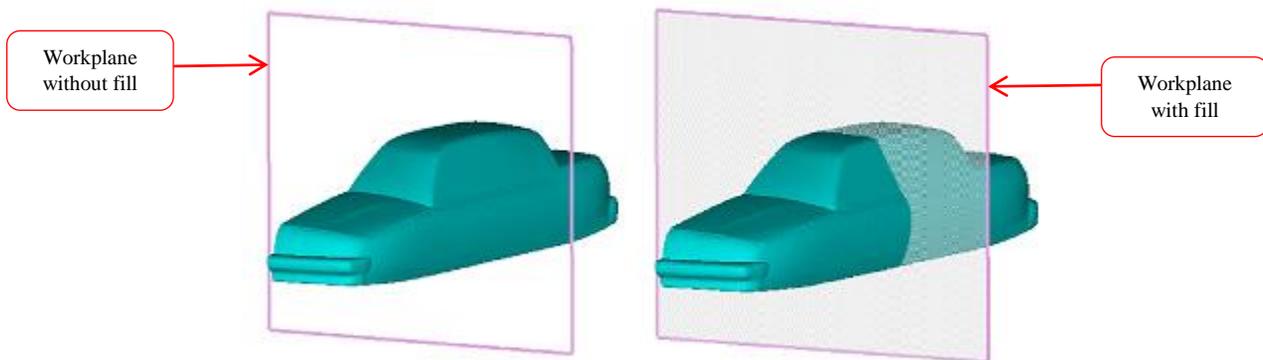


Figure 2.82: With and without fill for workplane

## 2.4.6 Section

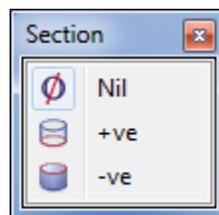


Figure 2.83: Section Drop-down list

Workplane is useful for viewing surface and grid cross-section. The options under **Section** drop-down (as shown in Figure 2.83) can be used to look inside different parts of surfaces and grid. By positioning the workplane so that it cuts through a surface/grid, you can actually clip away part of that surface/grid by selecting the **+ve** option from the dropdown list. Only the portion of the surface/grid on one side of the workplane will remain. The other side of workplane can be seen by selecting **-ve** option from drop-down menu. The previously showing portion of the surface/grid will disappear while the other side will appear. Since the clipped graphics display pierces objects, you can clearly observe 2-dimensional slices of a surface/grid. A surface with section views is shown in Figure 2.84.

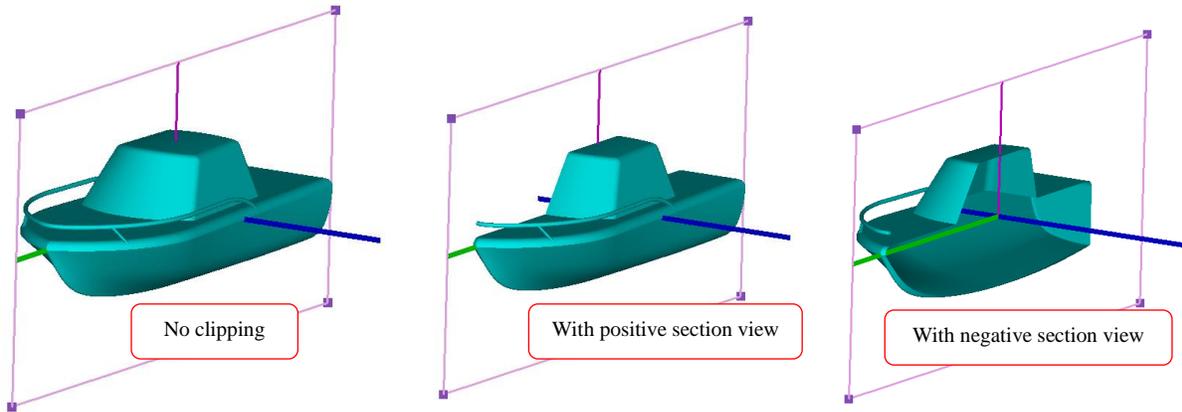


Figure 2.84: Different sectional views

## 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 times, 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 for 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 **toggle surface** section as shown in Figure 2.85 provides 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 center.

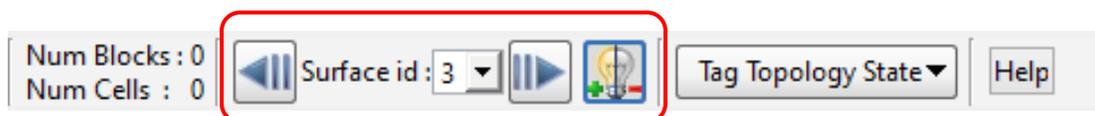


Figure 2.85: Toggle surface section

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 *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 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 *inactive 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 right side top. From the toggle display section, you can hide/show the surface, topology, grid, and workplane related objects in the graphics window. Figure 2.86 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.

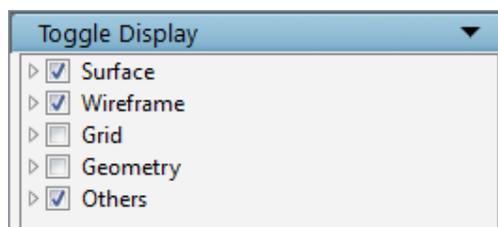


Figure 2.86: Toggle display section

- **Surface**

In the toggle display section, the first branch is **Surface**. By default, surfaces are visible only in **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 it with grid. GUI provides two ways of toggling the surfaces, either locally or globally. The toggle display section helps in toggling globally while the **toggle surface** section will help in toggling locally which is discussed in *section 2.5*. In this toggle display section, under the surface branch, there are three sub-branches such as **Feature edges**, **Border**, and the **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 color. The border is the path around the border of the surface that is highlighted in olive color 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 which will hide/show the respective option. The surfaces can be toggled by unchecking the surface checkbox as shown in Figure 2.87. To toggle a particular surface option, expand the surface main branch and uncheck the checkbox of the corresponding option.

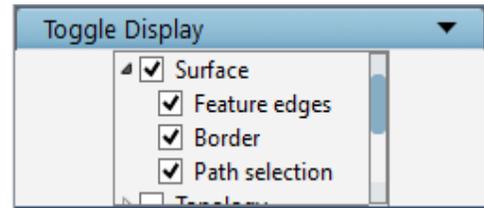


Figure 2.87: Surface sub-menu

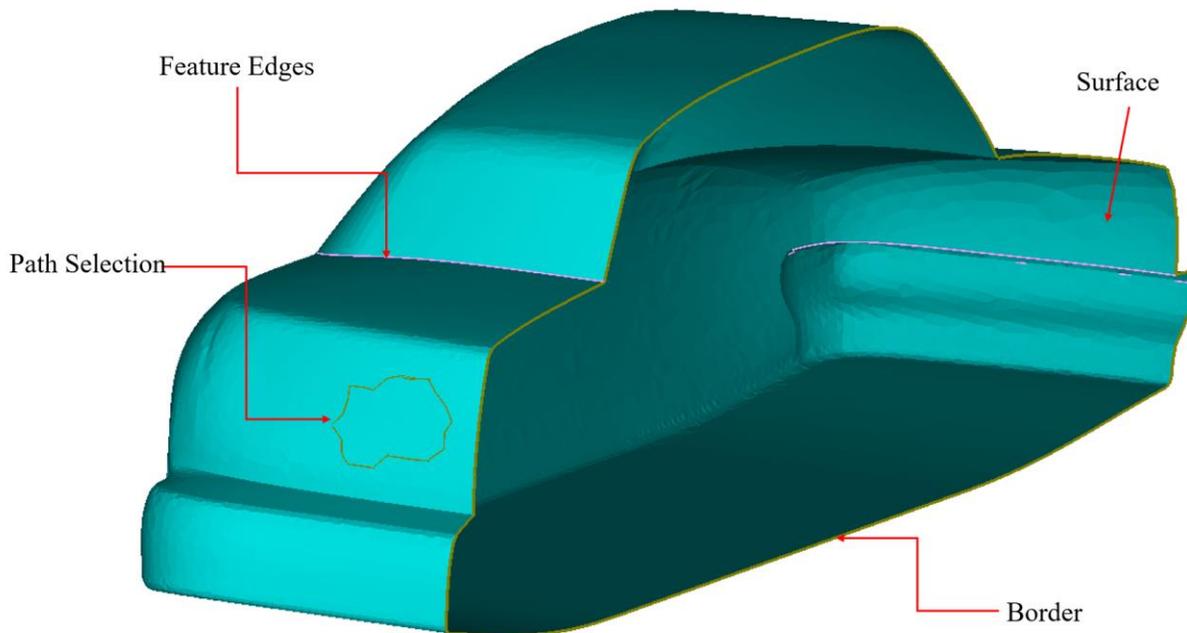


Figure 2.88: Surface entities displayed

- **Topology**

There are various options to hide/show the topology-related objects such as **Wireframe**, **Corner id's** and **Error** as shown in Figure 2.89. By default topology is only visible in **Topology** tab, in rest all tabs it is hidden. 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 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 color when a face is excluded from wrap. The face exclusion lines can be toggled by unchecking the Exclusion. **Shrunk blocks** which 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 section 2.3.22 for more information.

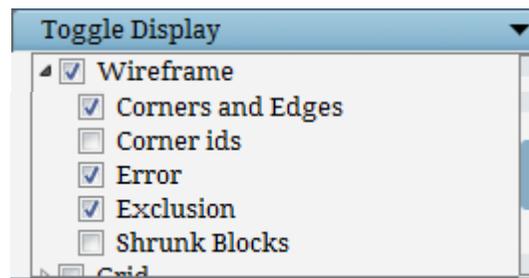


Figure 2.89: Topology sub-menu by default is shown

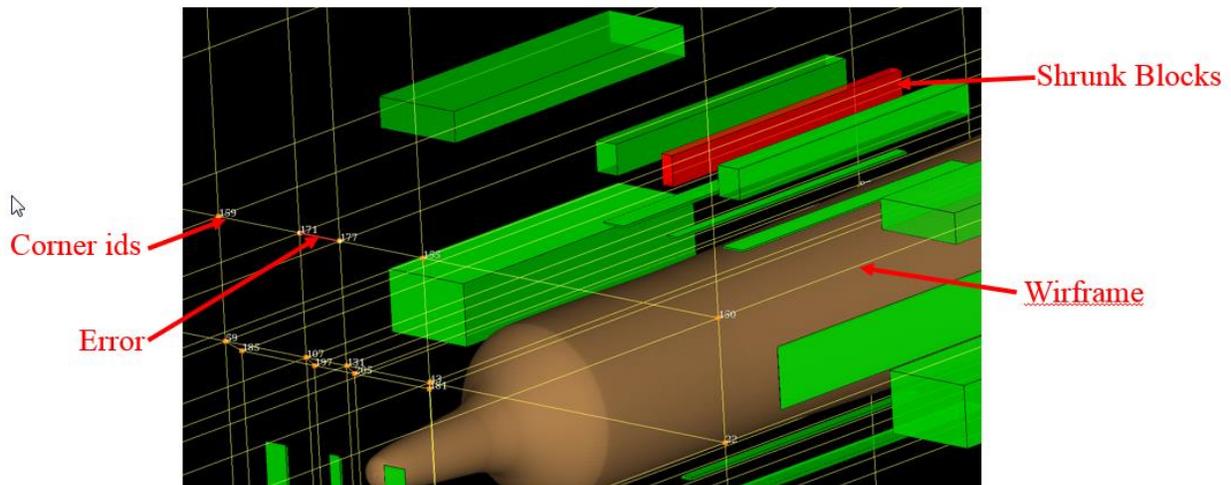


Figure 2.90: Topology entities

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 **Topology** section

in Toggle display section. Errors are displayed only when you attempt to start the gridding process of an incorrect topology.

- **Grid**

By default, grid is only displayed in **Grid** and **Property** tab where it is viewed and analyzed. Sometimes you may want to hide/show 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 box next to **Blocks** under **Grid** branch. Similar to surfaces, each block is given a unique number called block id and each block has unique IJK positioning, in order to see the IJK axes for each activated block, check the **block orientation** under grid branch in toggle display panel. The IJK axes will appear near the center of each block. When the grid blocks are hidden, their id's and their orientation are also hidden. To show/hide the block indices, use the checkbox of **block ids**. Checking **block orientation** box can toggle the block orientations. Grid sheets are useful in viewing grid cross-section, 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 grid branch can toggle this line.

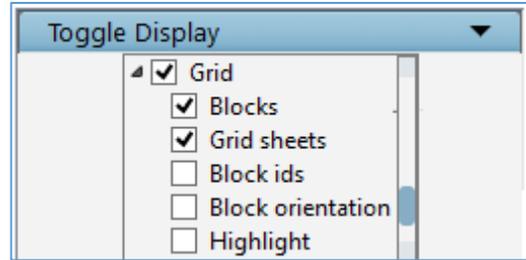


Figure 2.91: Grid sub-menu

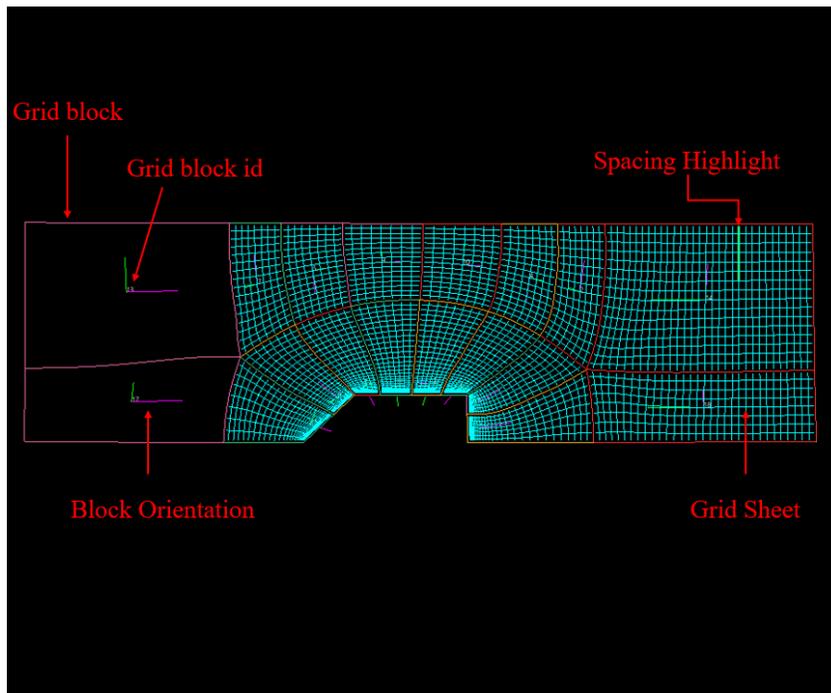


Figure 2.92: Grid entities

- **Geometry**

All the geometry related entities are grouped under ‘Geometry’ branch as shown in Figure 2.93. **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 are represented in green, yellow and red color depends 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 color.

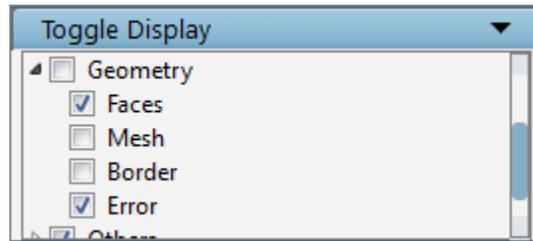


Figure 2.93: Geometry sub-menu

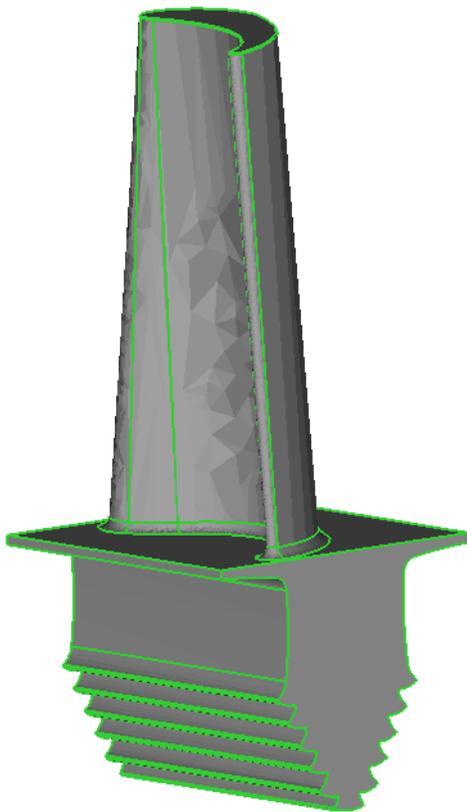


Figure 2.92a: Geometry with Face

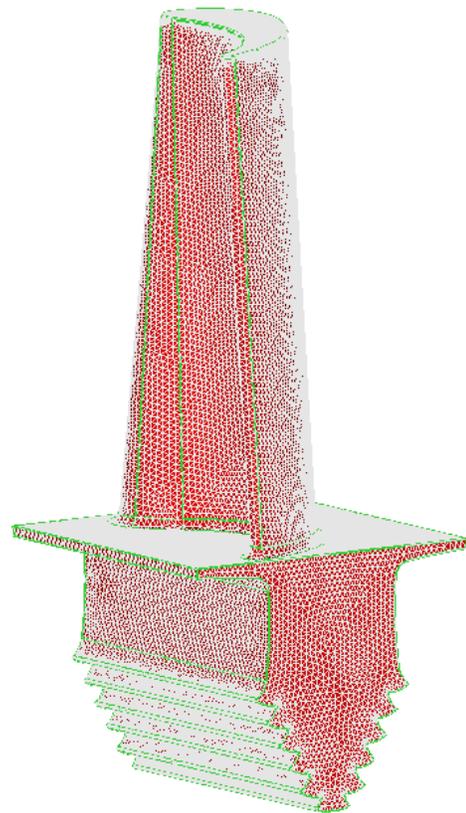


Figure 2.92b: Geometry with mesh

- **Others**

Other than surface, topology and grid entities that are visible in graphics window are **Global axes**, **center of rotation axes**, **Workplane** and **workplane axes** that are classified under ‘Others’ branch in Toggle display section. By default, these are visible in both **Surface** and **topology** tabs except for the **global axes** and **center of rotation axes** that is 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 center of rotation axes can be toggled by checking the **Axes** check box.

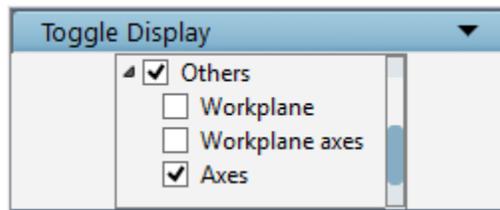


Figure 2.94: Others sub-menu

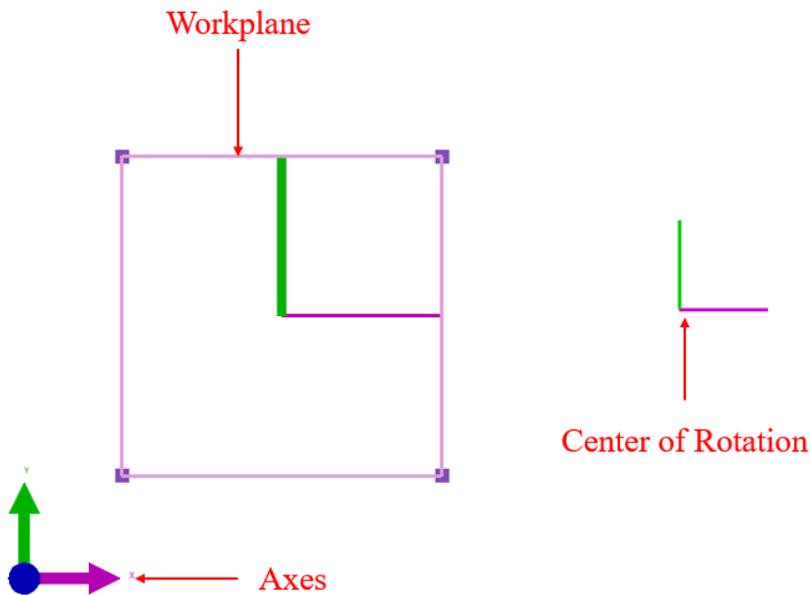


Figure 2.95: Other entities

**NOTE:**

1. If the workplane is used to clip the surfaces/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 surface tab, it would be still visible in Topology tab or any other tab.

## *CHAPTER 3*

# **SURFACE**

The foremost desired objects to generate a grid are surfaces. Before gridding a region, user must often read in several surfaces. *GridPro* can read surfaces from roughly eight different formats. Surfaces loaded into GUI must be analyzed and repaired if required, so that it can be used for grid generation. All operations related to surface restructure and repair are done in **Surface** tab. So 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 you surface repairing tools, surface probing tools and surface reconstruction tools. Each one is explained in different sections in this chapter.

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

In **Surface** tab, tools are divided into two sections such as Edit and Check/Repair. Edit section include those tools which are used for modifying the already loaded surfaces. Check/Repair section includes tools that help to check whether the surface is sufficiently smooth enough to be used for 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 plane surfaces. *Cylinder* refers to cylindrical surface. *Ellipse* and *sphere* refer to as ellipsoid surfaces. *Periodic* surfaces are rotational symmetric and translational 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 Create section of **Topology** tab as shown in Figure 3.1. The surfaces can be either 2D or 3D. Depends 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. List of surfaces can be created for 2D are

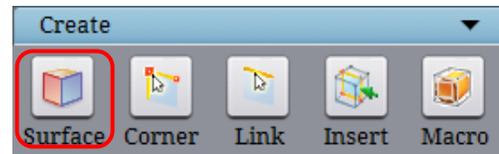


Figure 3.1: Create Section

- Line,
- circle,
- ellipse,

and for 3D surfaces are :

- plane,
- cylinder,
- ellipsoid,

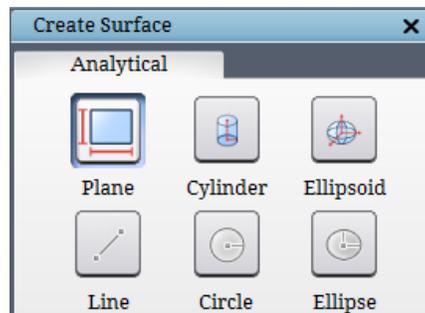


Figure 3.2: Surface creation dialog box

Even though 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 center 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 center and normal direction are required in the UI. Like the workplane, surface planes are represented by rectangles. The center of this rectangle is the point that helps define the plane. The normal of a plane helps to define the plane's orientation. For visualization purposes, the infinite plane is displayed as a square of finite size. Apart from the center and normal, the plane size is also required to create the plane surface. It's just a visualization parameter. The larger the size, the larger the rectangle will appear. In order 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.

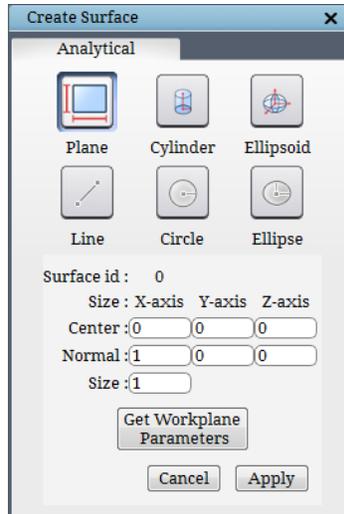


Figure 3.3: Plane dialog box

The surface id is an automatically generated number which starts from 0. It cannot be modified manually. The fields corresponding to center, normal and size of the plane can be manually edited. The plane normal and the center field require three real numbers separated by a space which represents the value on all three coordinates in the order x, y, and z and the size field requires a positive real number which is 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 color. And the surface id is displayed at the top of the dialog box as 'Index'.

Figure 3.4 shows two planes created with center 0,0,0 and 10,0,0. The normal for both the plane is 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, the size parameter is purely used for visualization purpose. Please note that the surface id of the first plane is 0 and for the second plane is 1.

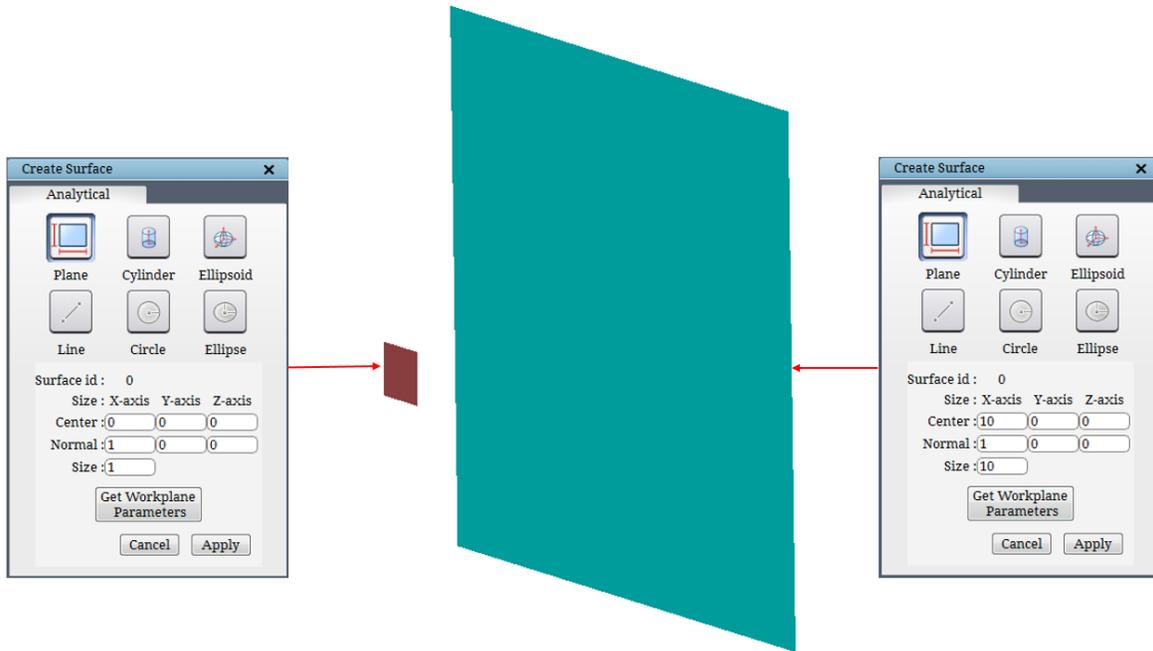


Figure 3.4: Plane with different size

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 center, normal direction and size of it (length \* width). This data is directly copied to the fields corresponding to center, normal and the plane size. Note that 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 one of its axes. When clicked on *Cylinder* button, Create Cylinder parameters appears on the right side panel as shown in Figure 3.5. The center, 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 for visualization purposes, its length (or view scale) is also required. The radius, power and length field require a single real positive number

which represents 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 center, axis, and the length direction field require three real number separated by a space which represents the coordinates of x, y, and z-axis for the corresponding input.

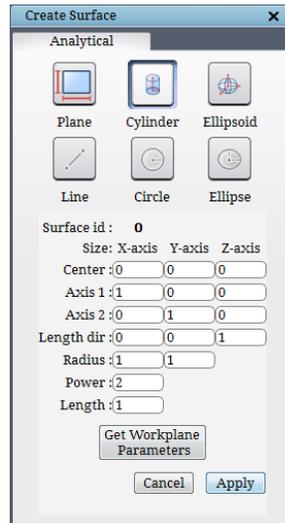


Figure 3.5: Cylinder dialog box

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

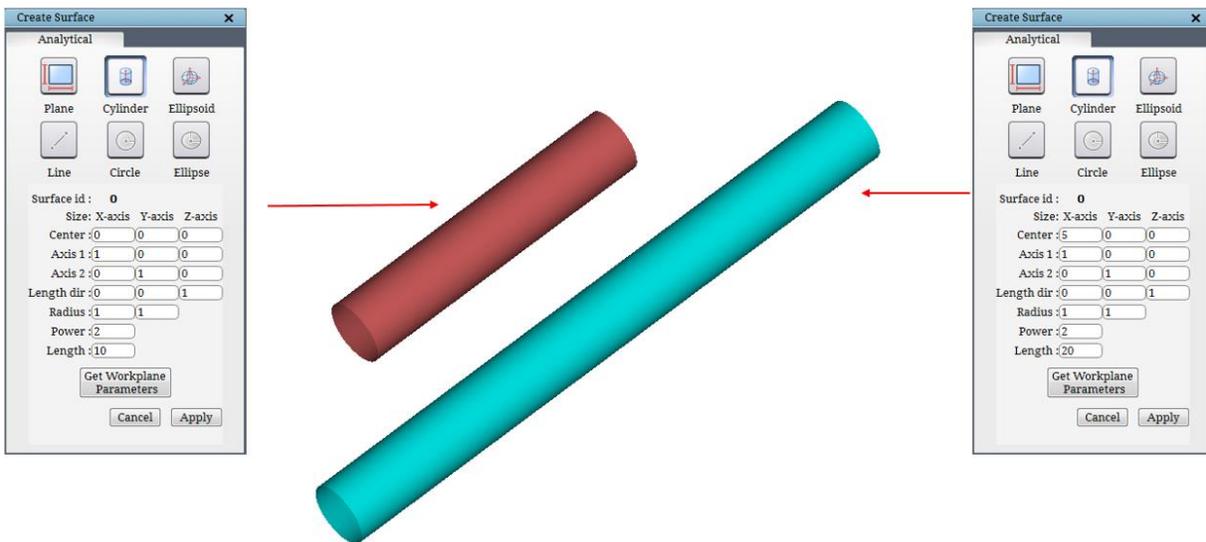


Figure 3.6: Cylinder with different lengths

The workplane data can also be used to create the cylinder surface by clicking on the *Get workplane parameters*. The center of the workplane is copied to the 'center' 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 remain unaltered when the workplane data is used.

**NOTE:** The “power” option defines the relation that provides the shape of the built-in surface. An ellipsoid (includes cylinder, circle, ellipse) is defined by a second order function which 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.

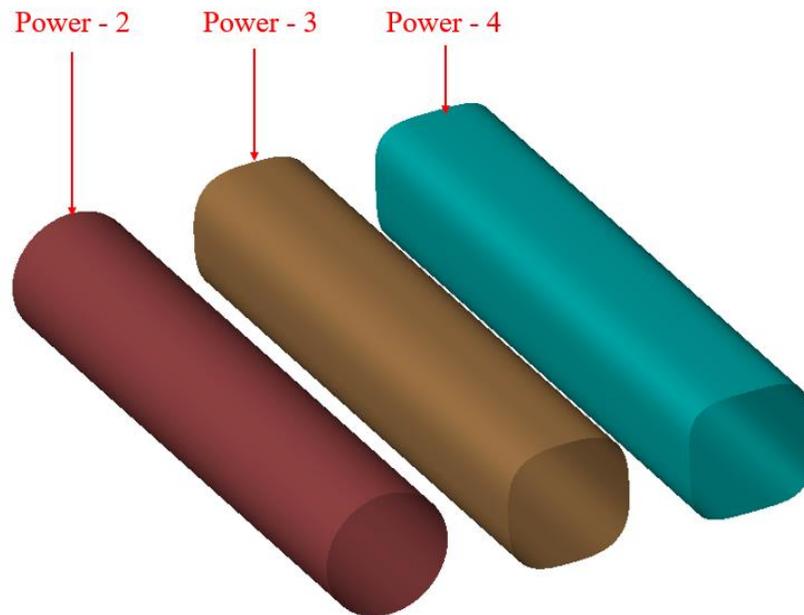


Figure 3.7: Cylinder with different power

- **Ellipsoid**

An ellipsoid is defined by the center, three axes, their radii and the power of the ellipsoid. In



order 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 which appears on the right side panel. The radius and power field require a single real positive number which represents the radius on all three axes and power determines the shape of the ellipsoid. The center and axis fields require three real number separated by a space which represents the coordinates of x, y, and z-axis for the corresponding input.

Surface id : 0

Size :	X-axis	Y-axis	Z-axis
Center :	0	0	0
Axis 1 :	1	0	0
Axis 2 :	0	1	0
Axis 3 :	0	0	1
Radius :	1	1	1
Power :	2		

Get Workplane Parameters

Ok Reset Cancel Apply

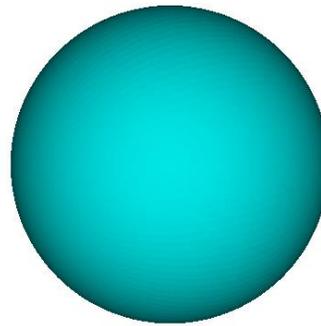


Figure 3.8: Ellipsoid with equal radius in all directions

Surface id : 0

Size :	X-axis	Y-axis	Z-axis
Center :	10	0	0
Axis 1 :	1	0	0
Axis 2 :	0	1	0
Axis 3 :	0	0	1
Radius :	5	3	5
Power :	2		

Get Workplane Parameters

Ok Reset Cancel Apply

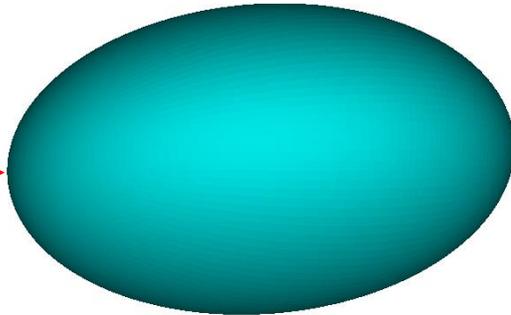


Figure 3.9: Ellipsoid with equal radius in two directions

Surface id : 0

Size :	X-axis	Y-axis	Z-axis
Center :	0	0	0
Axis 1 :	1	0	0
Axis 2 :	0	1	0
Axis 3 :	0	0	1
Radius :	5	3	1
Power :	2		

Get Workplane Parameters

Ok Reset Cancel Apply

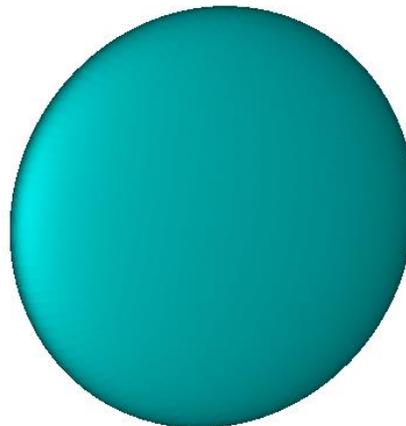


Figure 3.10: Ellipsoid with radius different in all directions

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

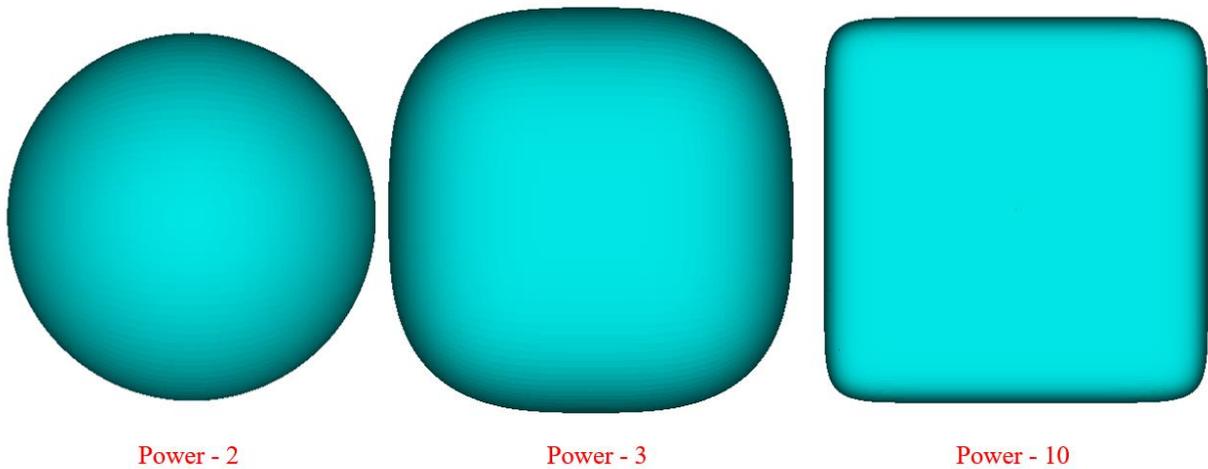


Figure 3.11: Effect of power for ellipsoid on the same parameter

The dialog box of axis shows the three axes i.e. axis 1, axis 2 and axis 3. The first column in 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 in axis 1 or in axis 2 or axis 3. It will take the values based upon the axis vector you input. For example, the Figure 3.12 shows 3 dialog box, for which output will be same i.e. an ellipsoid with radius 10 in the X direction (1, 0, 0) will be created and radius 5 in other two directions i.e. Y and Z directions. The same ideology is applicable to cylinder surfaces also.

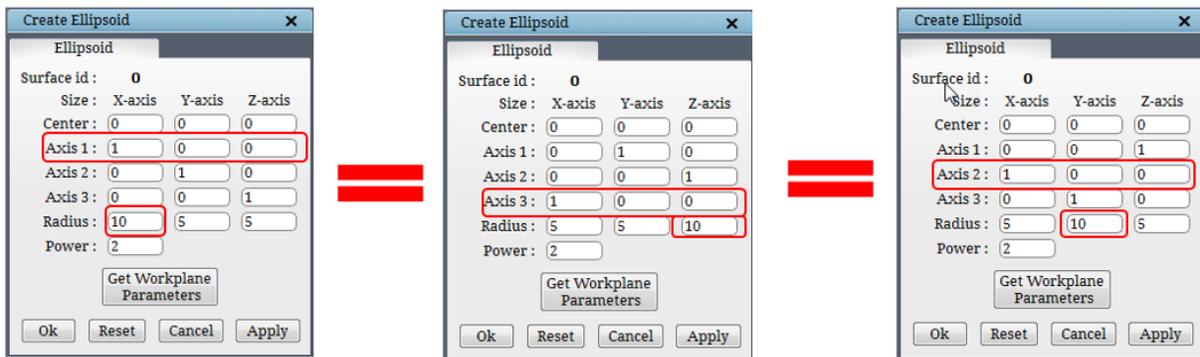


Figure 3.12: Same output for different dialog box

Similar to cylinder, the workplane data can also be used to create the ellipsoid surface by clicking on the **Get workplane parameters**. The center of the workplane is copied to the center field, 'center' 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 the 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, then duplicate that section to construct easily the grid about the whole object. This process saves time and effort and allows to construct 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 section 4.10.4) in the right side panel of topology tab to create and 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.

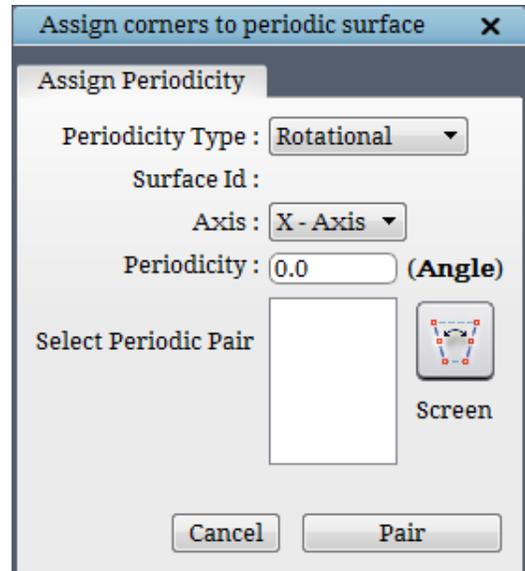


Figure 3.13: Periodic surface dialog box

- **Rotational**

A rotational periodic surface is defined by its periodicity, orientation, and center. When clicked on the **Periodic** button, the dialog box opens as shown in Figure 3.13. Origin is taken as the default center of the surface. The orientation of the axis can be changed from the Axis drop-down list. The center 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 the section 3.4.9). Hence, two parameters are required for the generation of a default *rotational periodic* surface (which passes through the origin of the global coordinate system) are 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 center 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.

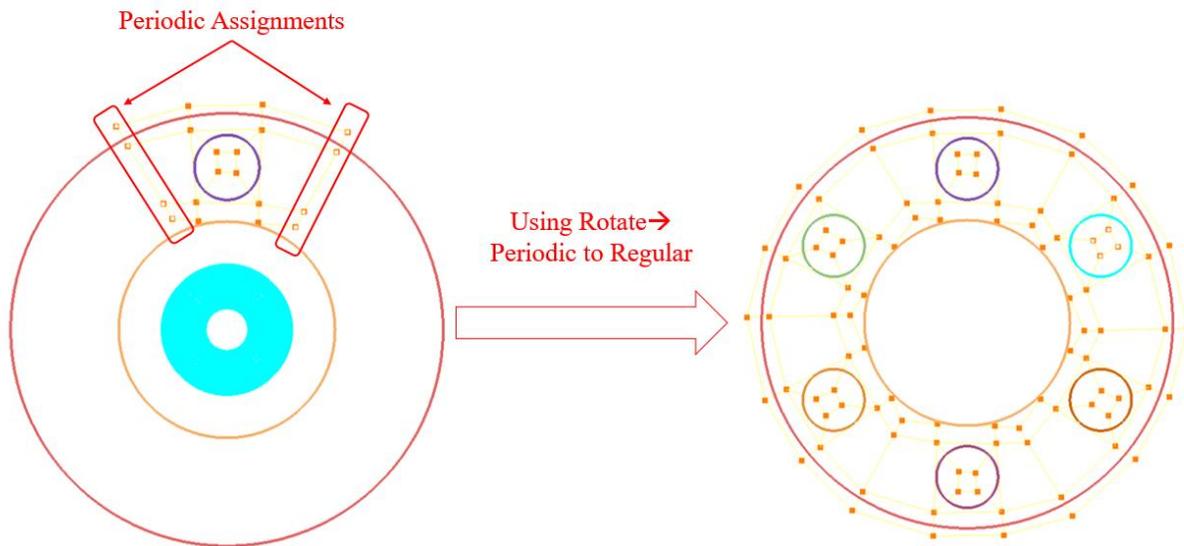


Figure 3.14: Using Rotate option to rotate the Periodic Topology

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

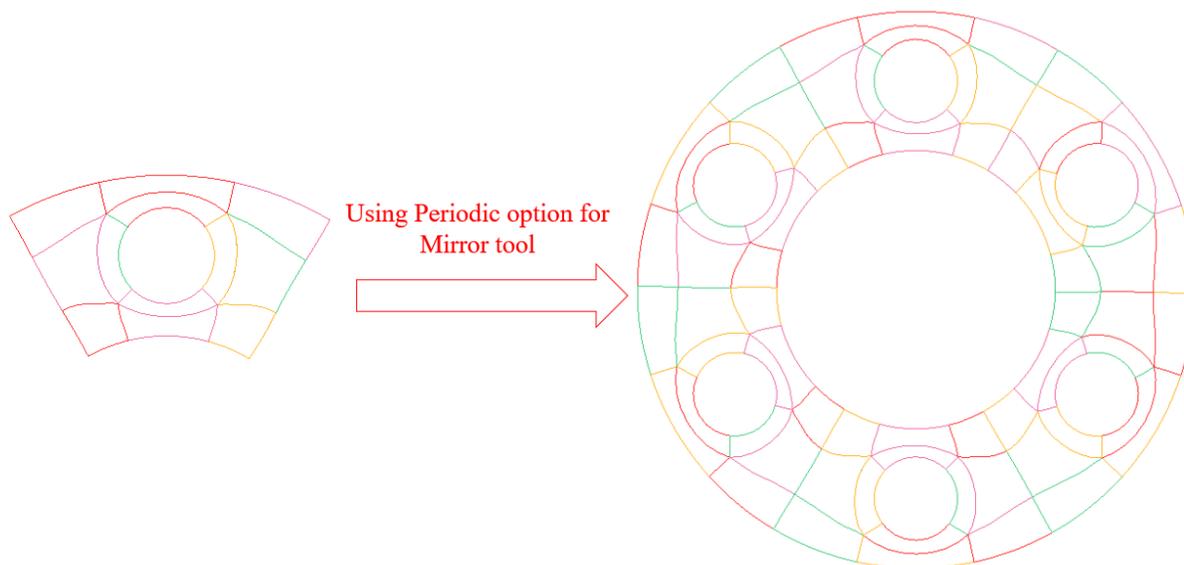


Figure 3.15: Using Mirror option on level to generate a full grid

A full grid can be generated using the periodic grid as shown in Figure 3.15 by using Periodic option specified under **Mirror** tool in 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 center is the origin of global system coordinates which can be transformed using transformation option (discussed in a later section). Figure 3.16 shows the dialog box of translational periodic surface. Similar to Rotational, the Translational object also doesn't have a representation, but 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.

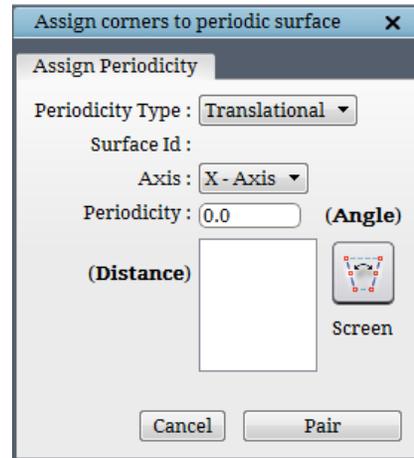


Figure 3.16: Periodic surface XYZ tab

**NOTE:** Both the periodic surfaces are defined in the same way irrespective of its 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, a variety of 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 center 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 **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 center and normal fields require only two coordinates (x and y) to represent the line. The size field requires a single real positive number which defines the length of the line.

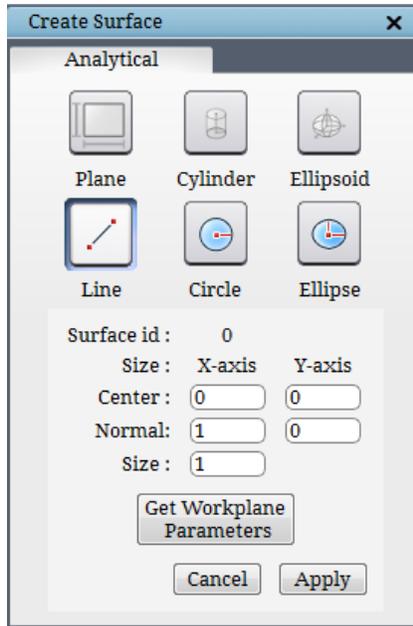


Figure 3.17: Line dialog box

The workplane data can also be used to create the line surface by clicking on the **Get workplane parameters**. The center of the workplane is copied to the 'center' field of the line. The workplane X-axis is used as the normal of the line and the workplane diagonal length is copied to the size field of the line.

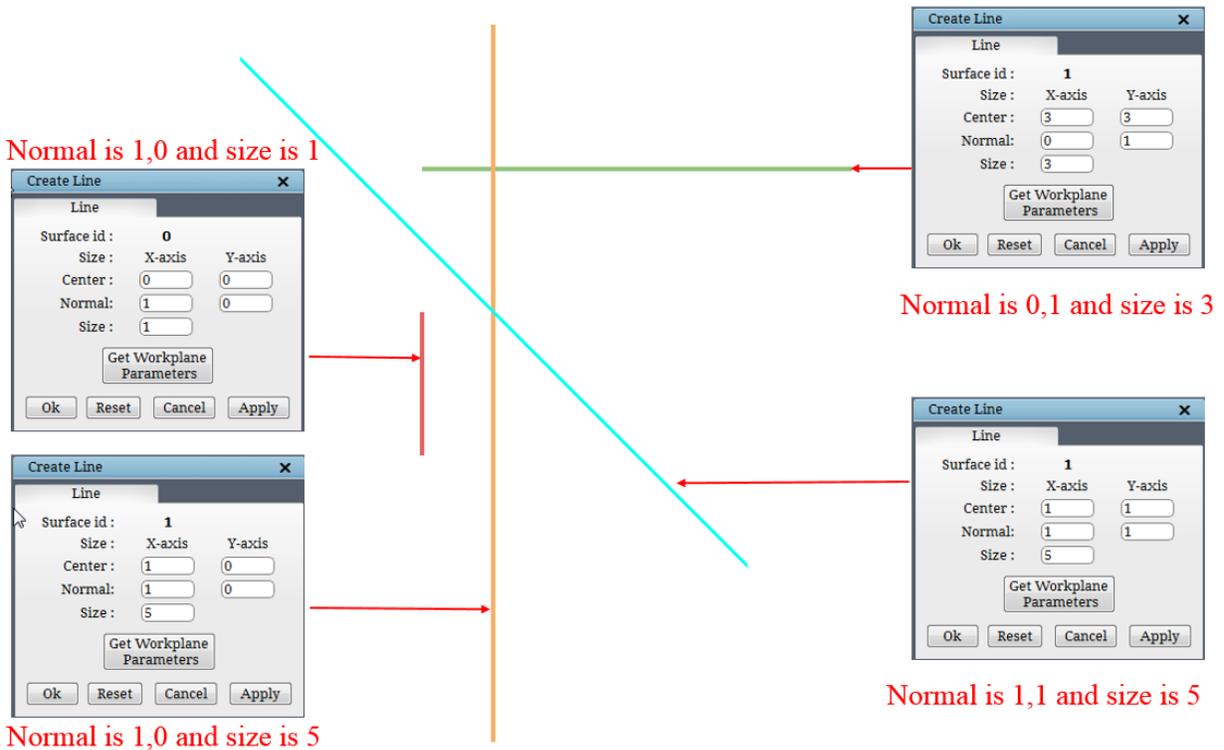


Figure 3.18: Shows the effect of normal and size

- **Circle**

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



the *Circle* button *Circle* in the **Create Surface** dialog box. Figure 3.19 shows the Create Circle dialog box which appears on the right side panel. The center field requires two real numbers separated by a white space representing the center of the circle and the radius and power field require 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.

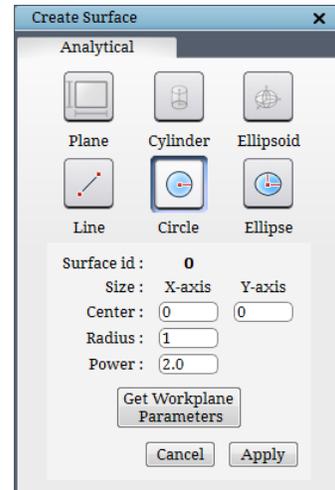


Figure 3.19: Circle dialog box

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

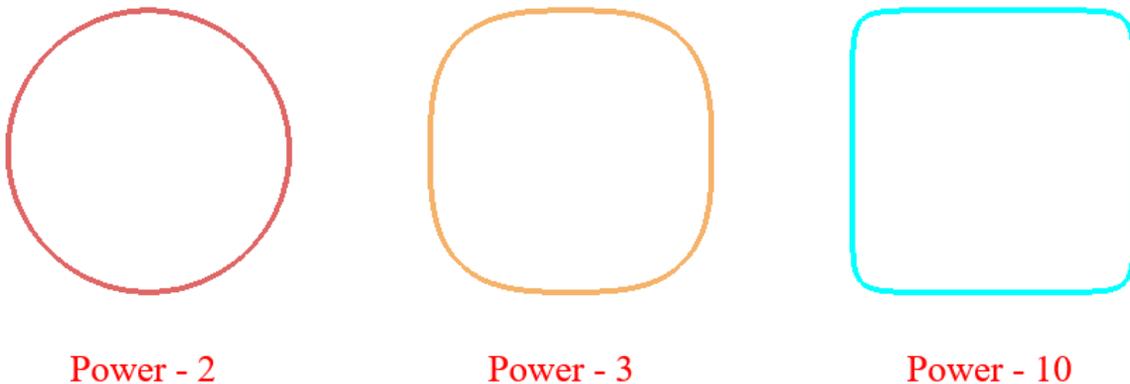


Figure 3.20: Effect of power value in cylinder

The workplane data can also be used to create the circle surface by clicking on the *Get workplane parameters*. The center of the workplane is copied to the ‘center’ 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 center, major and minor axis and radius along there respective axes and the power of the elliptic surface. Clicking on *Ellipse* button



*Ellipse* in **Create Surface** dialog box opens ellipse parameters as shown in Figure 3.21. As discussed earlier, the center, 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 center of the workplane is copied to the center field of the ellipse. The workplane length (along workplane x-axis) is copied to major axis radius field. The workplane width (along 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.

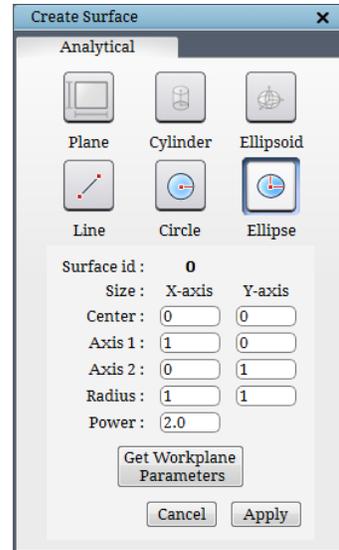


Figure 3.21: Ellipse dialog box

### 3.1.3 Explicit surfaces

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

- **Tube**

A *tube* can be defined by the revolution around a center curve. The revolution for each point on the center 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 center 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*.

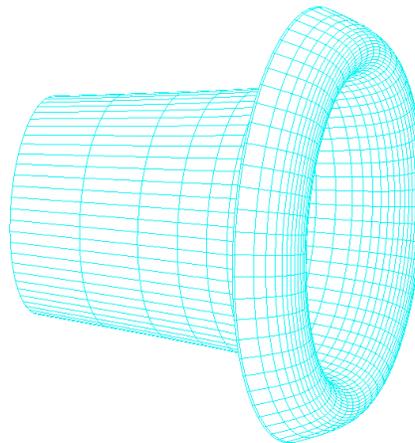


Figure 3.22: Tube Surface

- **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 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.

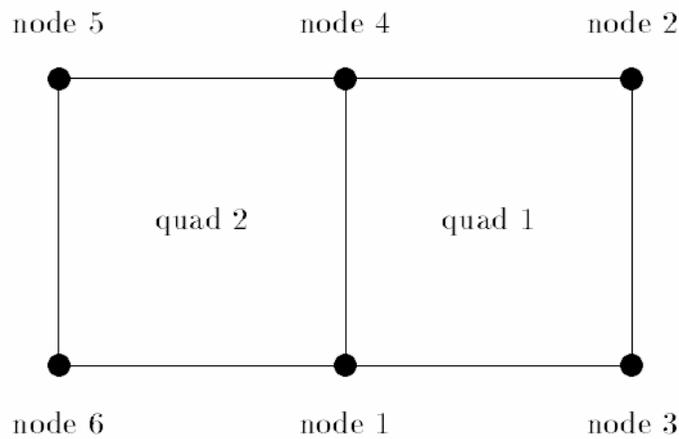


Figure 3.23: Surface with two quads

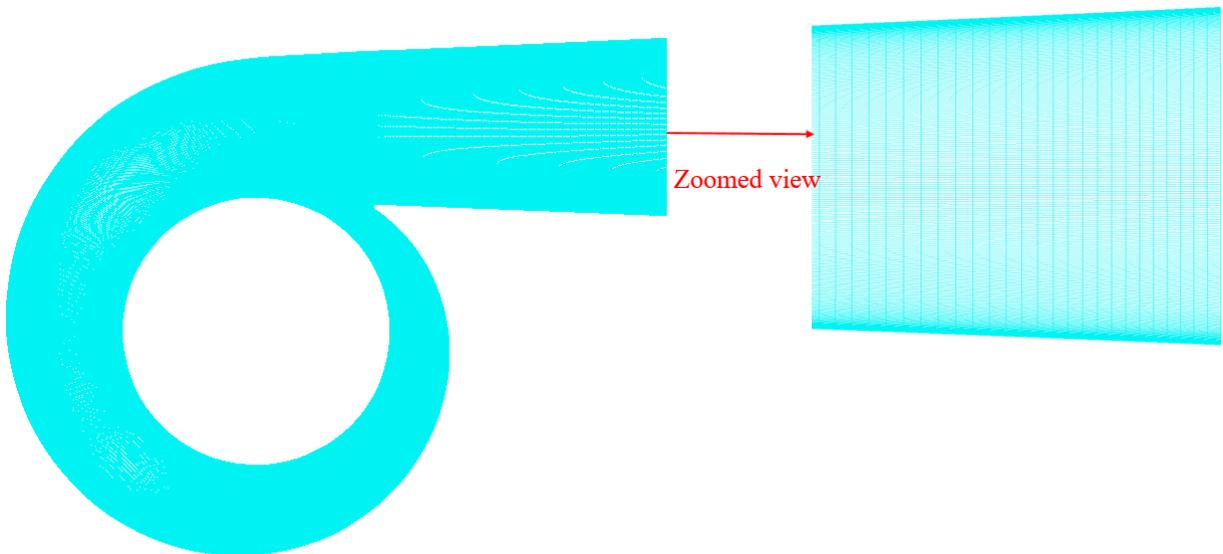


Figure 3.24: Quad surface

- **Lin**

The only explicit 2D surfaces that can be created in UI is in *.lin* format. *.lin* are linear surfaces defined by an I\*J array of points for some I and J in the file. The bilinear interpolation is used to determine 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 surfaces 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 total number of points, coordinates of each point, number of triangular elements in the 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 *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.

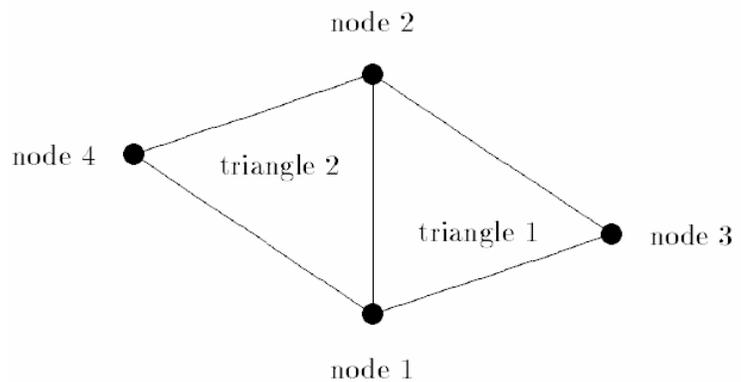


Figure 3.26: Tria surface with two triangular elements

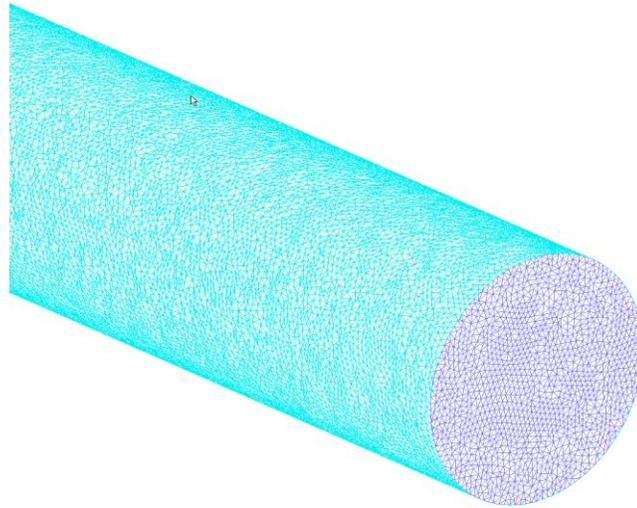


Figure 3.27: Tria surface

## 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; that is, 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 grid only on one side either inside the surface (internal flow geometry) or outside the surface (External flow geometry). The external surfaces are surfaces which acts 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 the sides. i.e. Surfaces which are defined inside the grid to represent the fluid/solid regions or to enhance the grid lines would 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 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.

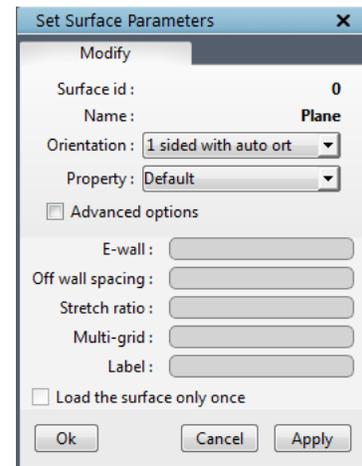


Figure 3.28: Surface Settings dialog box

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 point points to the “+side” of the surface. If a surface is oriented in the “1 sided -ve” direction, the normal vectors at each surface point points to the “-side” of the surface.

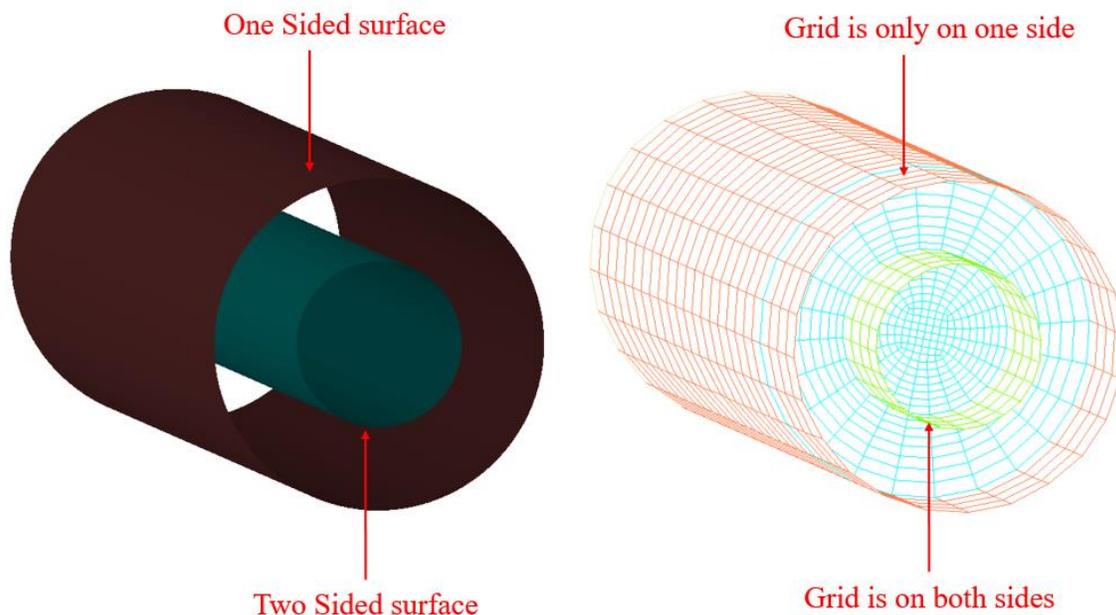


Figure 3.30: Surface orientation with the grid

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. In order to impart 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 it may be having some leaks or patches. Sometimes geometry is perfectly fine, just require some modifications so that it can be readily used by grid engine. 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* color 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.

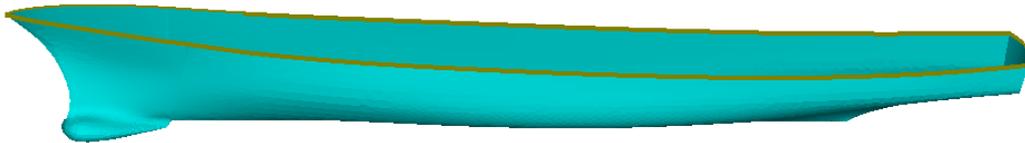


Figure 3.31: Surface boundary (border)

#### 3.3.2 Angle

The ***Angle*** operation illustrates 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. ***Angle*** tool is available with two option such as incremental and above the specified. By the default, ***Angle*** tab is displayed as shown in Figure 3.32 when the ***Angle***



button is clicked.

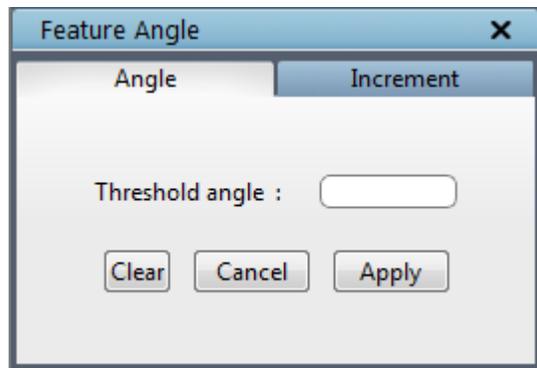


Figure 3.32: Angle tab dialog box

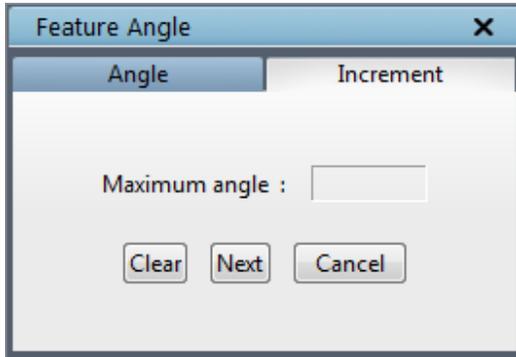


Figure 3.33: Increment tab dialog box

Once the threshold angle is given, it calculates the angle in all the surface nodes and creates a path in baby pink color on the current surface, wherever the cells are having angle more than the specified value. To generate a smooth grid, feature angle of more than 45 is not desirable. Another option is the **Incremental**, available in 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 angle value is shown in 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.

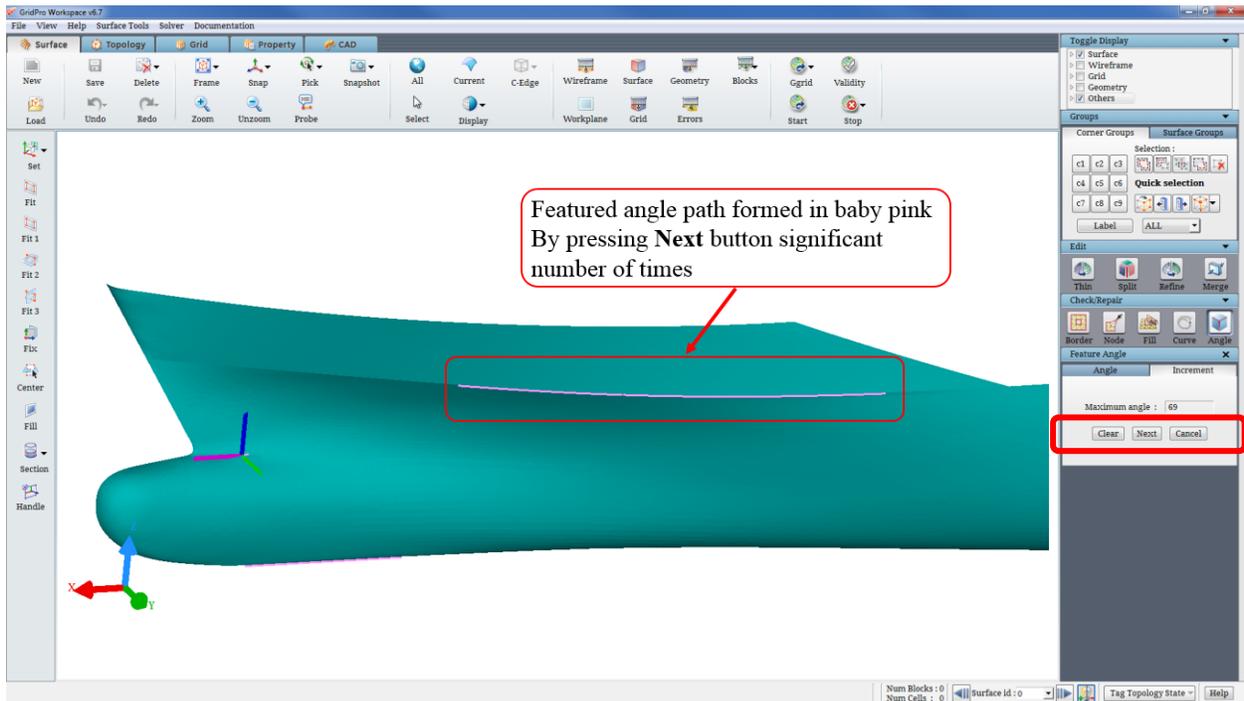


Figure 3.34: Feature angle path formed on the Current surface

### 3.4 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 that are 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 **Edit (surface)** section. Since these 3 buttons are often used, they are positioned in Topology tab itself to provide a better workflow.

### 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 of triangulations on



the basis of angle or ratio. **Thin** button is located in the right side toolbar under **Edit** section in **Surface** tab. If clicked, a dialog box opens as shown in Figure 3.35.

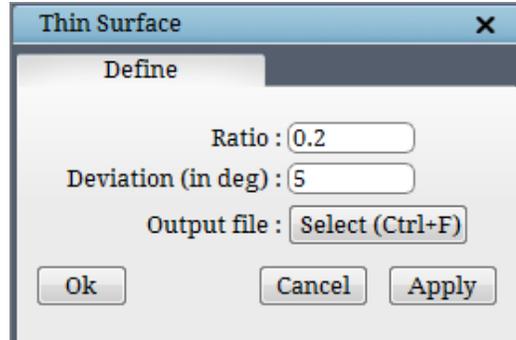


Figure 3.35: Thin dialog box

In the *Deviation* field, specify the maximum allowable surface cell angle. In the *Ratio* field, specify the ratio by which the number of cells has to be reduced. Any, one of the options can be specified; if both specified it will try to reduce the number of cells in an iterative manner to satisfy both the conditions. Figure 3.36 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.

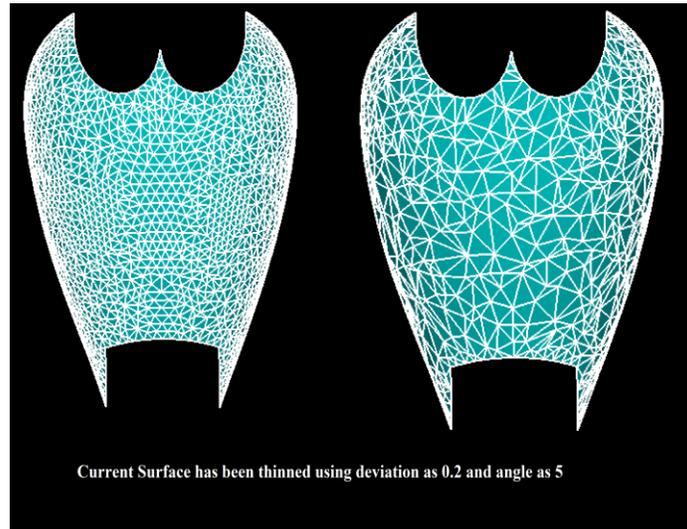


Figure 3.36: Before and after applying thinning tool

### 3.4.2 Split

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

- **Split disjoint surfaces**

If in case the loaded surface has disjoint surfaces and yet 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 3.37 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 which has to be split should be grouped under a surface group and must be switched on while executing the command. Refer section 4.4 for more information on groups.

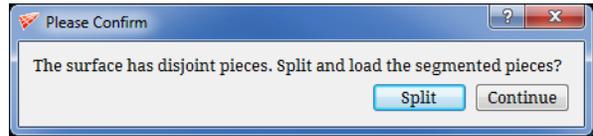


Figure 3.37: Dialog box for split disjoint surface

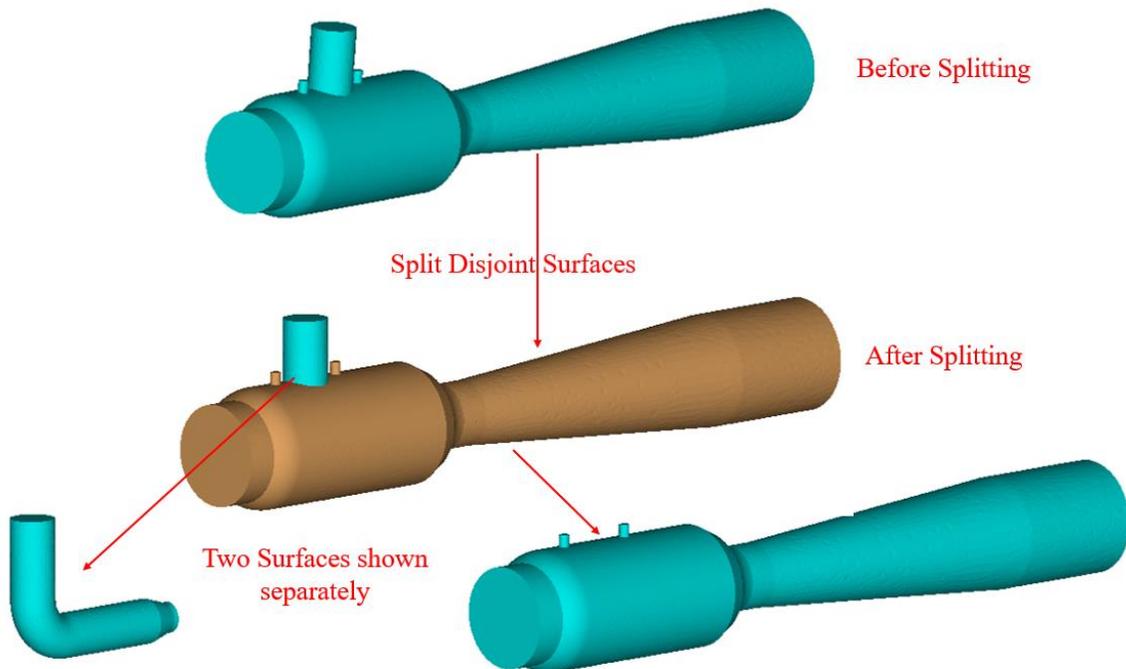


Figure 3.38: Before and after splitting disjoint surfaces

- **Angle**

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

In **Angle** tab under **Split**  button as shown in Figure 3.39, a surface is split based upon feature angle. If *calculate automatically* is selected, it will split the current surface if it is forming a closed path with the feature angle of more than 45. Another option in **Angle** tab is to specify **threshold angle** which would split the current surface above the specified angle if it forms a closed path. Press **Apply** to split the current surface.

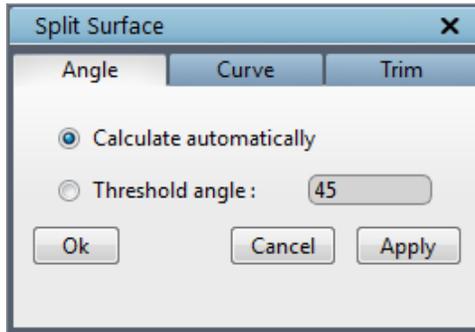


Figure 3.39: Angle tab for the Split button

- **Curve**

In order to split up the surface, form a closed curve around it using options available in **Curve** tab under the **Split** button as shown in Figure 3.40. 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. Curve appears as olive coloured lines. The endpoint of a path appears as a white square. Figure 3.41 illustrates examples of paths.

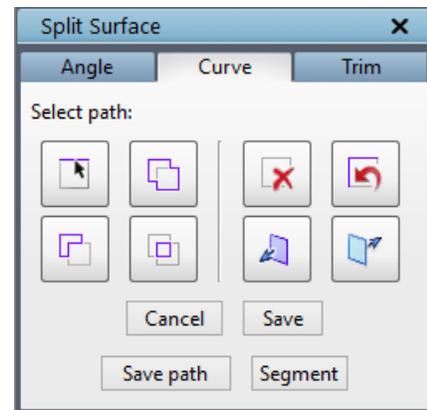


Figure 3.40: Curve tab for Split button

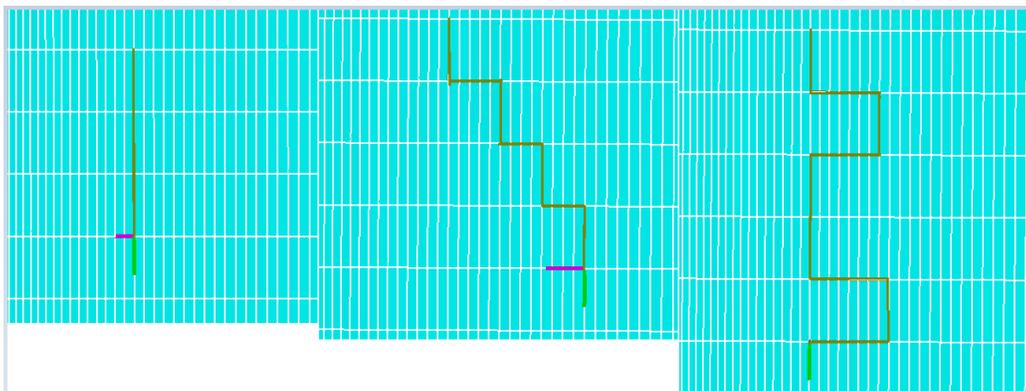


Figure 3.41: Various ways a path can be created

In order to create a curve, the user can either use the **toggle path**  or **add path**  button. Toggle path  button helps to pick a path between two nodes one at a time. After depressing the  button, click (with the left mouse button) on between the two nodes that will add to the curve. Keep clicking between the nodes to add it to curve. Similar to toggle path , add path  button allows to create 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 shortest distance as a path between the two nodes. If the line connecting the two points is not along a surface cell, GridPro will automatically create a “zig-zag” curve among the intervening cells. Notice that once used, the toggle path  and add 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 add path  button is depressed, a long and precise curve can be created as per the requirement. The three operations to the right of the add path 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. To select the specific part of the path, depress the retain path  button and drag a box on that part of the path using left click. Notice that the remove path  and retain path  buttons remain depressed until it is re-clicked. The delete path  button, when clicked, deletes all the paths. The undo path  operation acts as a partial one-step undo. The operation will delete the path that was just created but does not re-create the path or part of the path that was deleted.

Once a closed curve is formed or a part of a surface cell is selected, it can be segmented using the **Segment** button. If the curve is not a closed one, the splitting operation will fail. In other words, the endpoints of the curve must intersect or lie on the border of the surface (which can be

checked by depressing border  button). Also, if two disconnected curves are created, the **Split** operation would fail. An error message as shown in Figure 3.42 pops out when the split operation fails.

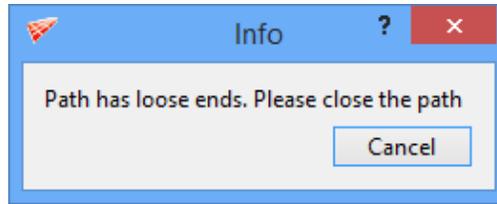


Figure 3.42: Split error message

Figure 3.43 illustrates examples of segmented surfaces. If Split succeeds, a message in the status bar will read “surf has 2 pieces,” and one such piece will become a different color (while one remains sea blue). 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 the previous  and next  buttons. To save the current piece, click on the *Save* button. 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.

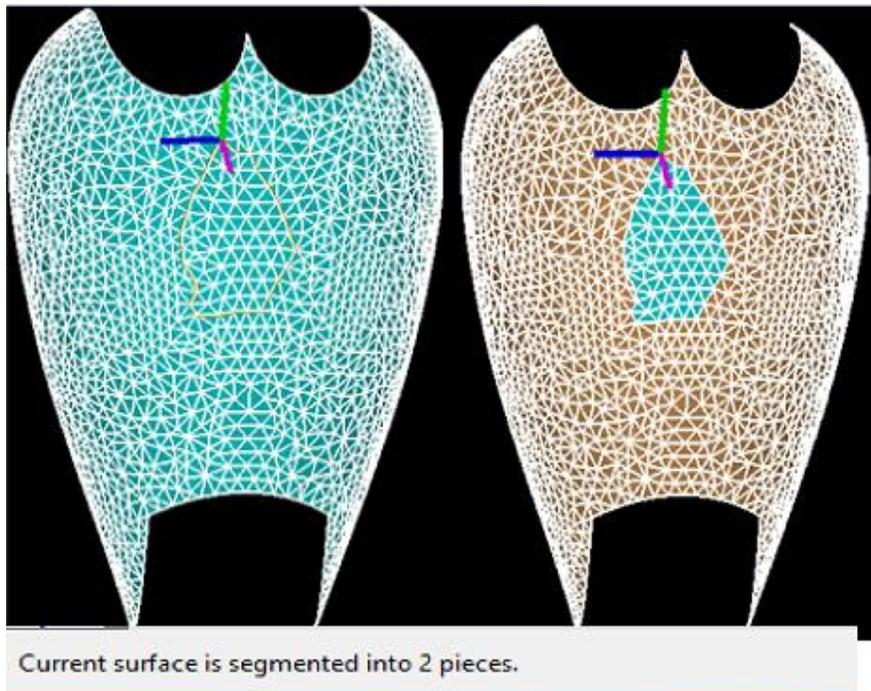


Figure 3.43: Surface split into two pieces

- **Trim**

The third tab under the **Split** button is **Trim** as shown in Figure 3.44. It is similar to **Curve**, instead of selecting a path, surface elements are removed to split the current surface. Surface elements can be added or removed one by one from the current surface by using  button. After switching on  button, clicking on visible surface elements will hide the selected elements. This

button remains depressed. It needs to be clicked to switch it off.  button is used for removing elements from the current surface. By clicking on  button, and dragging a box using right click will hide all the elements inside the box. The button will switch off, once the operation is done. To add elements,  button can be used. A box must be dragged, after clicking on  button to add elements. Its behavior is similar to . 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 Save button.

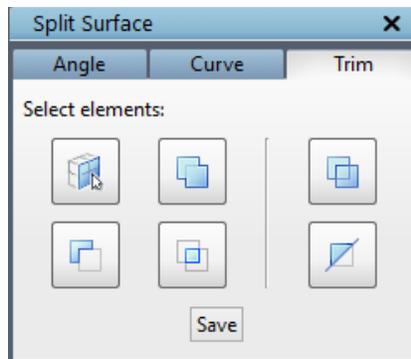


Figure 3.44: Curve tab for the Split button

### 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 3.45 shows the Refine Surface dialog box. Only a current surface can be refined in the GUI, so whenever you want to refine a surface, select that surface as *Current* surface first.

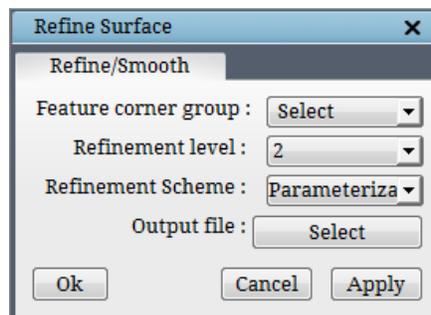


Figure 3.45: Refine dialog box

The three options under **Refine**  button give the user a 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 the 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 among 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.

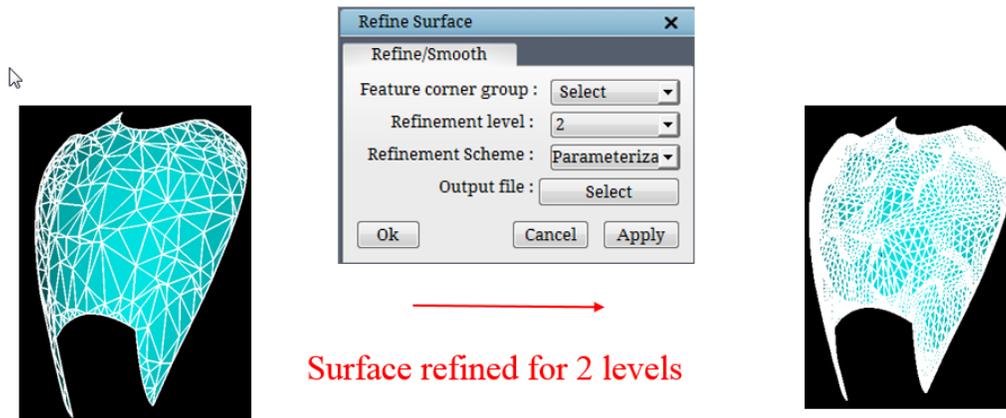


Figure 3.46: Surface before and after refining

### 3.4.4 Merge

If one surface can be split into two surfaces using the *Split* tool, two surfaces having 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 3.47. Select the two surface numbers that have to be merged and 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.

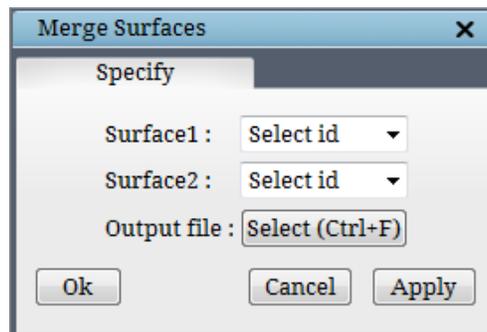


Figure 3.47: Merge Surfaces dialog box

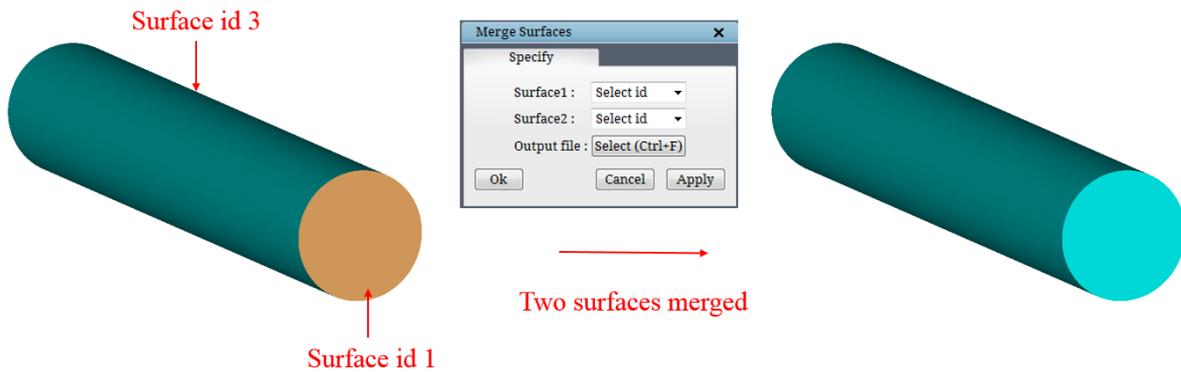


Figure 3.48: Surface before and after the merge

### 3.4.6 Duplicate



Duplicate tool **Duplicate** is located under *Edit (surface)* section of the Topology tab and it is used to duplicate *current* surface to create a new surface and append it to the current topology. This is very useful if a new surface, with the 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, so that the surface position can be modified and append the surface. For instance, to create a plane parallel to the current planar surface, it can be duplicated and the center 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 3.50. The duplicate operation can be undone and it is also stored in the undo stack.

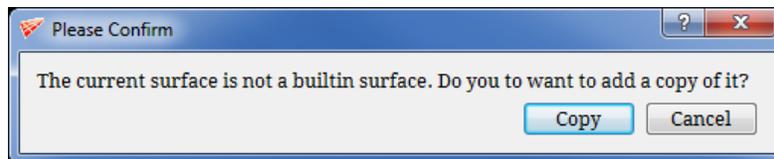


Figure 3.49: Copy confirmation dialog box

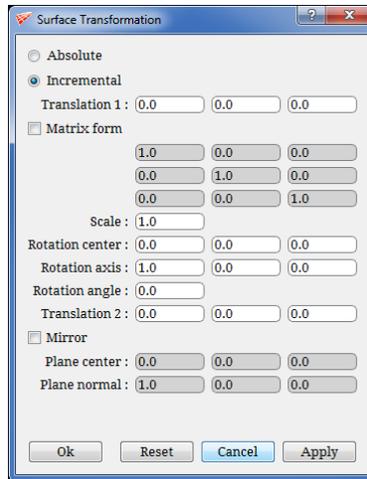


Figure 3.50: Surface Transformation dialog box

### 3.4.7 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 **labeling the surfaces**. If the



**Settings** button under *Edit (Surface)* section of the Topology tab is clicked, Set Surface Parameters dialog box opens as shown in Figure 3.51. 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'.

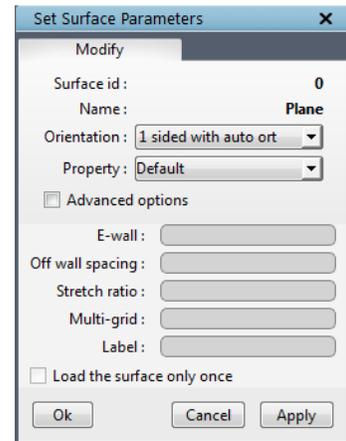


Figure 3.51: Surfaces Settings dialog box

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 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 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 state of the topology. 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 which are 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.

Clustering parameters for a *current* surface can be set by checking advanced options in properties dialog box. Checking advanced options enables the clustering parameters as shown in Figure 3.52. 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.

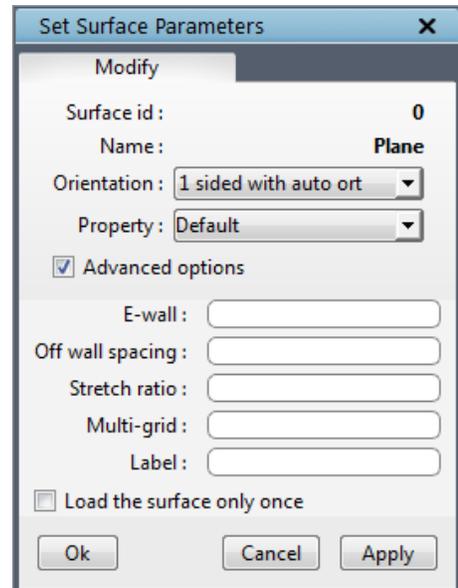


Figure 3.52: Surface Settings parameters with Advanced options enabled

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 3.53.

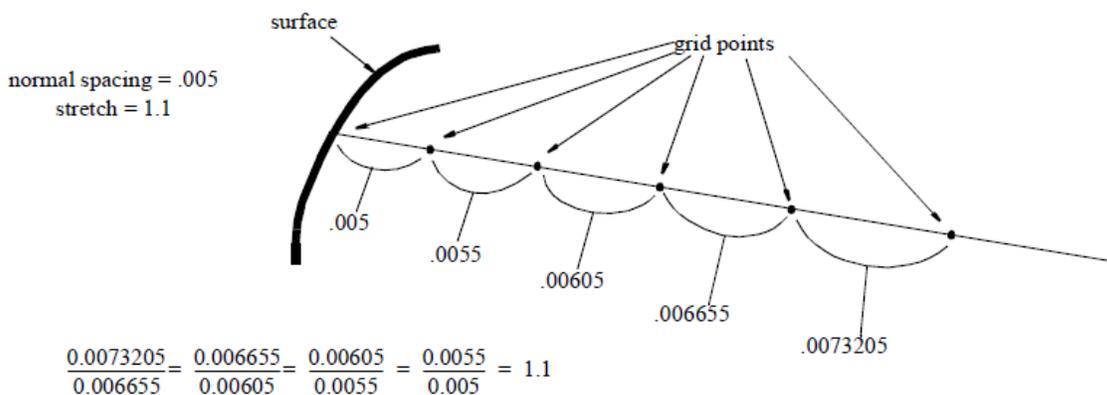


Figure 3.53: Stretching value for grid points

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 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 same label name. These labels attached to the surface can be loaded in association with property in Property tab by using **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 surfaces are identical in type, location, and transformation.

### 3.4.8 Replace



Replace  current is used to reposition or modify the surface parameter inputs for the CURRENT surface. It is one of the 3 buttons which are positioned under the **Edit (surface)** section of the Topology tab. Once clicked a dialog box opens as shown in Figure 3.54, 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 replace button can be undone from the **Topology** tab.

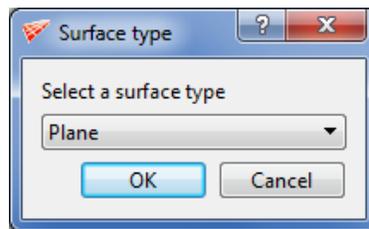


Figure 3.54: Surface type dialog box

### 3.4.9 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 **Transform**  button under the Topology tab. Please refer to section 4.6 for more information.

### 3.4.10 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 3.55. 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 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.

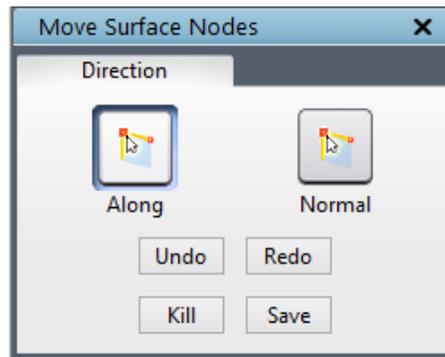


Figure 3.55: Move surface dialog box

Similar to topology undo and redo operations, node displacement operations can also be revert back using **Undo** and **Redo** buttons in 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.

### 3.4.11 Fill

GridPro allows the user to merge segmented surfaces (pieces), as well as 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 3.56 shows the dialog box of Fill button. Create a path around the region that needs to be filled using the path creation options available, then click on **Fill** button in the dialog box. -Tria surfaces will automatically fill the hole. Filled holes are not part of the surface until the modifications are saved using **Save Modifications** button.

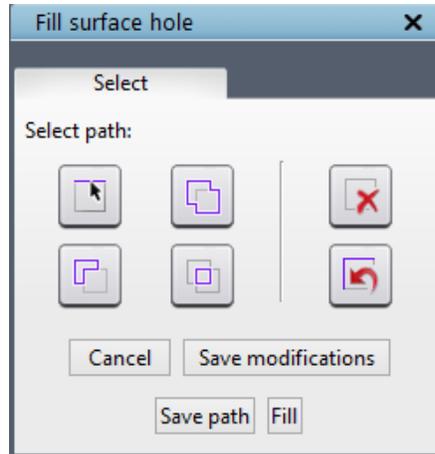


Figure 3.56: Fill Surface hole dialog box

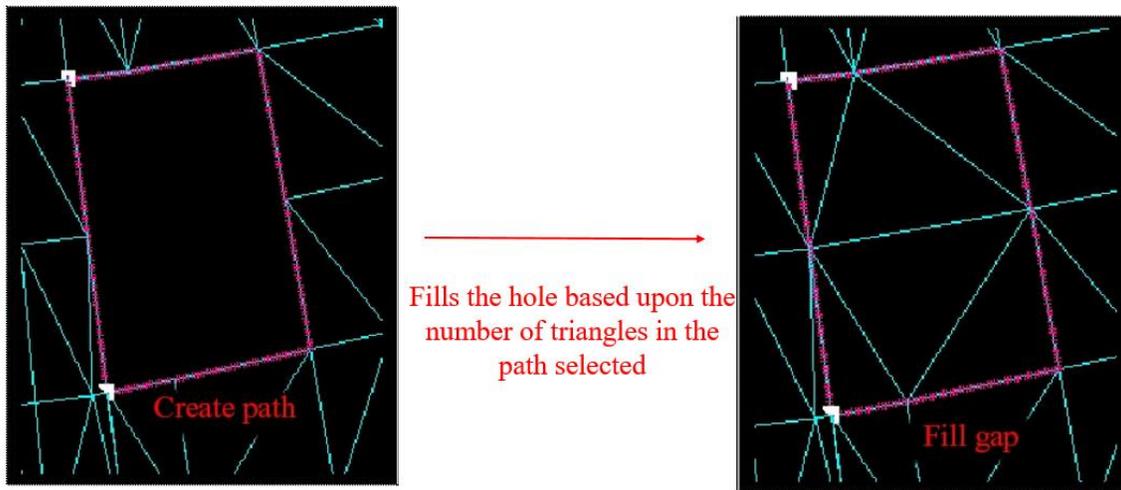


Figure 3.57: Fill Surface hole dialog box

### 3.4.12 Curve

Sometimes when you load a \*.dat or \*.lin file, notice that there will be two nodes (coordinates) located at the same position either at the origin or end-point. *GridPro* cannot generate grids on where two surface nodes are overlapping. It will report an error message in the terminal as shown in Figure 3.58 while running *Ggrid*. To resolve such cases, you need to find and remove the overlapping nodes from the surface file. Using the **Curve** tool, you can resolve this issue. Clicking on the **Curve** button opens a dialog box as shown in Figure 3.59. 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 3.58: Error message

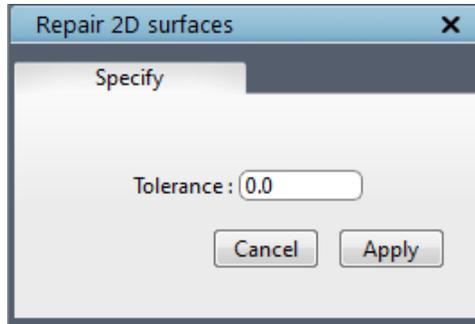


Figure 3.59: Curve button dialog box

- **Right Click Menu**

Apart from the properties options, the often used options are kept under surface right-click menu as shown in Figure 3.60.

- **Center of rotation** picks the right-clicked position on the surface as the center of rotation of the screen.
- **Make** the current option used to select any surface *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.

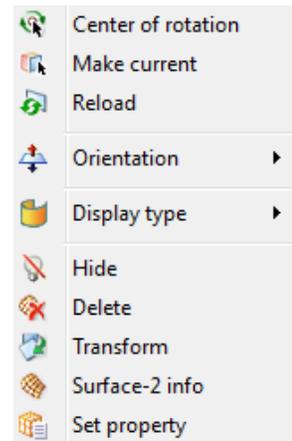


Figure 3.60: Surface Right click menu

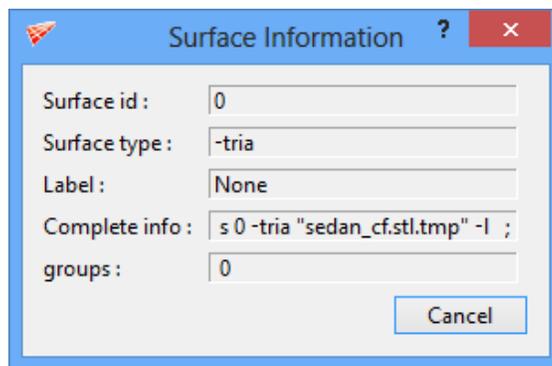


Figure 3.61: Surface information

- **Orientation** 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 *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 topology panel.
- **Transform** button can be used to translate, rotate, scale and mirror the surfaces which are explained in *section 3.4.9*.
- **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 3.61.
- **Set property** helps in setting 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 a wireframe and surface assignments.

*GridPro* is a unique multiblock grid generation program because it revolutionizes how volumes are gridded. Usually, multiblock 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 exactly fit the boundaries for the region to be gridded. 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 roughly follow the geometry of its bounding surfaces.*

*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 Projection 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 **Edit (Surface)** section, 3 buttons which are often used are located. They are Duplicate, Settings and Replace. Under **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 section such as **Groups**. Topology consists of wireframe and surface assignments. A separate section called '**Assignment**' consists of the tools that help in assigning the corners to its 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 **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 first to 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 to define the region which 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.

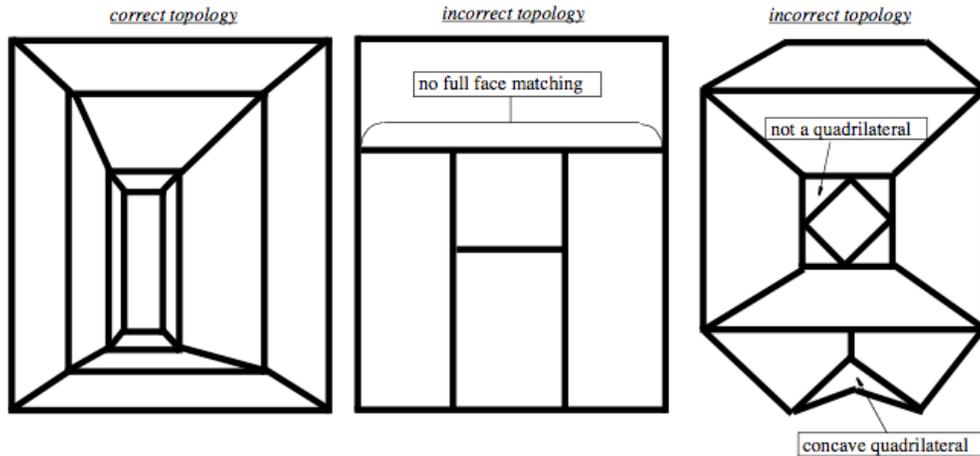


Figure 4.1: Topology examples

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. There are some rules, however, which 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.

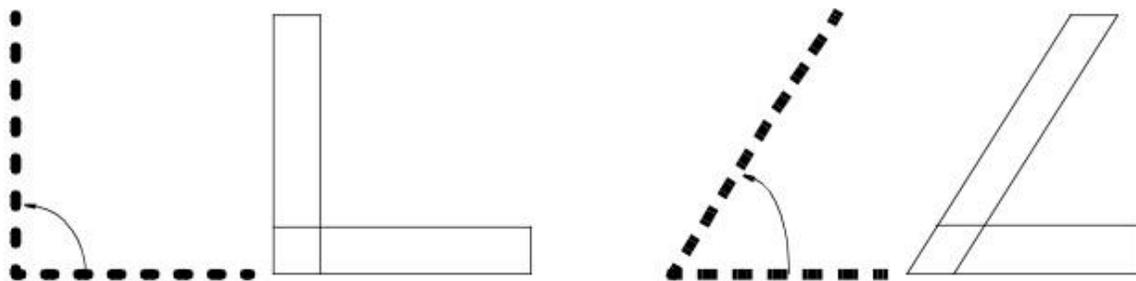


Figure 4.2: Topology for angles between 0 and 90 degrees

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:

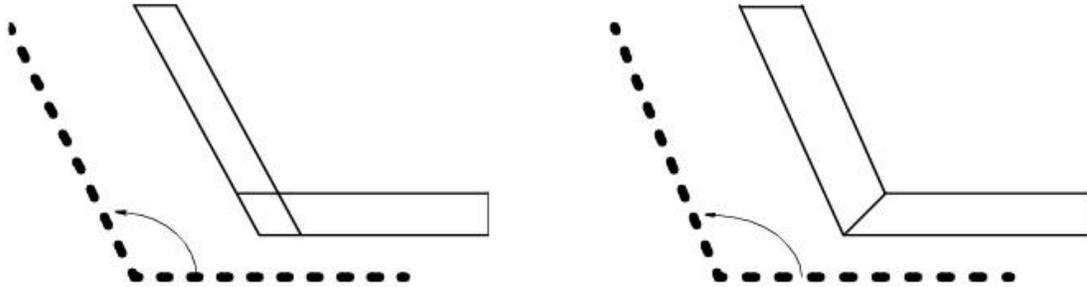


Figure 4.3: Topology for angles between 90 and 180 degrees

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

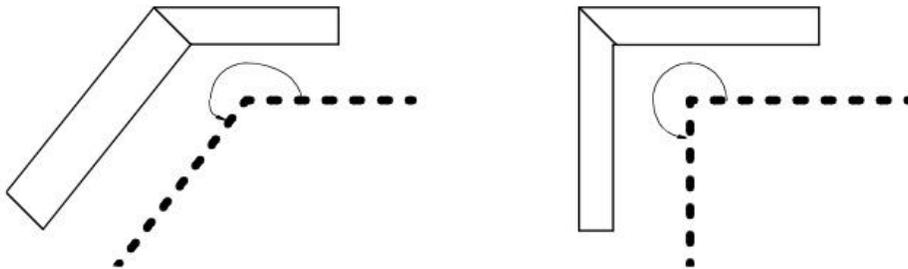


Figure 4.4: Topology for angles between 180 and 270 degrees

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

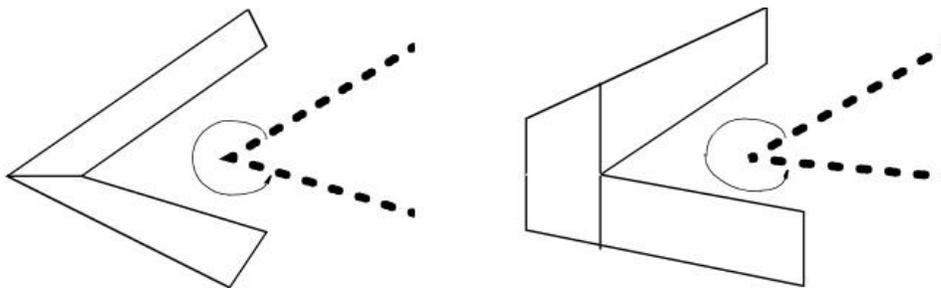


Figure 4.5: Topology for angles between 270 and 360 degrees

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:

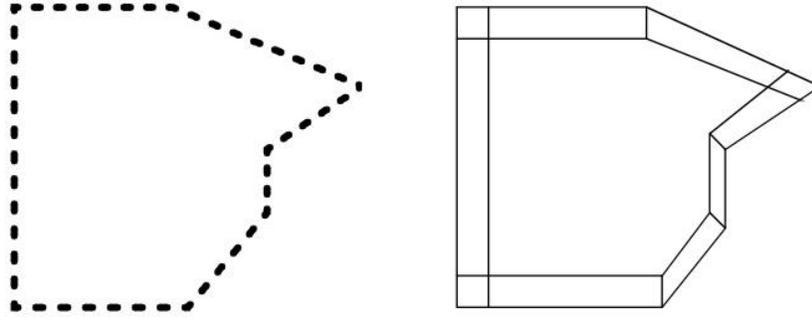


Figure 4.6: Example showing the above ideas incorporated

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

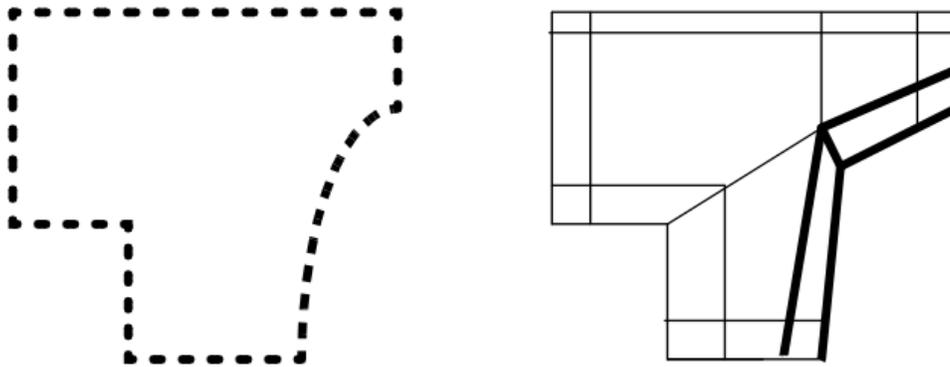


Figure 4.7: Topology for a smooth surface

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.

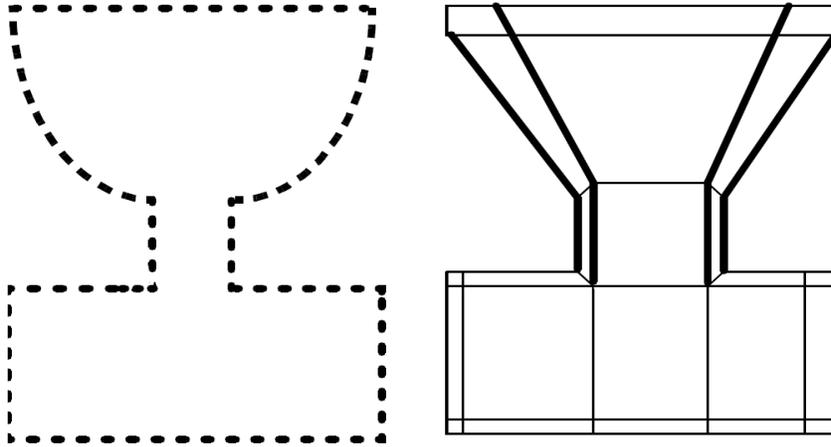


Figure 4.8: Topology for continuous curves

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

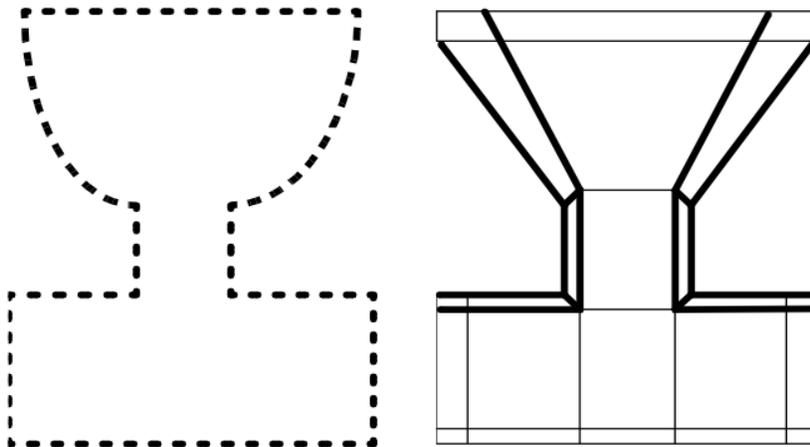


Figure 4.9: Topology with wrap layer highlighted

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

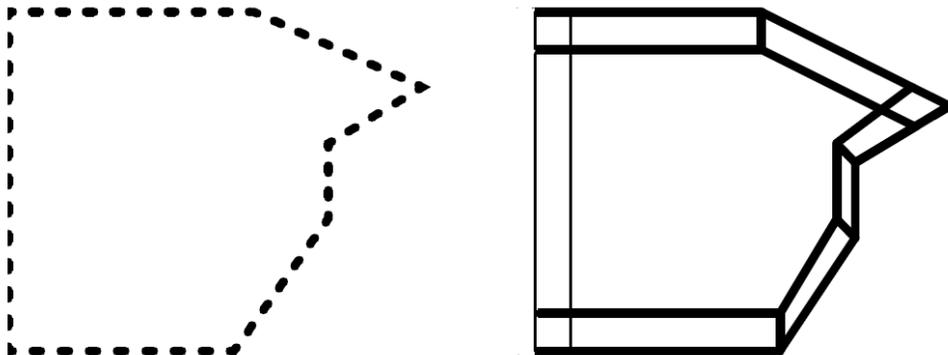


Figure 4.10: Topology with wrap layer highlighted

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

All real-life configurations have three dimensions. Therefore, in order 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 which is grouped into different category. Of which the first category is the *topology creation utilities* grouped under **Create** section of **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 objects to generate a grid are surfaces. Before gridding a region, user must often read in several surfaces. *GridPro* can read surfaces from roughly eight different formats. Of these eight, analytical implicit surfaces are one of the formats which include plane, cylinder & ellipsoid in 3D and line, circle and ellipse in 2D. All these surfaces can be created using the *Surface* button under **Create** section. Please refer Chapter 3 for more information on each of the options under the *Surface* button.

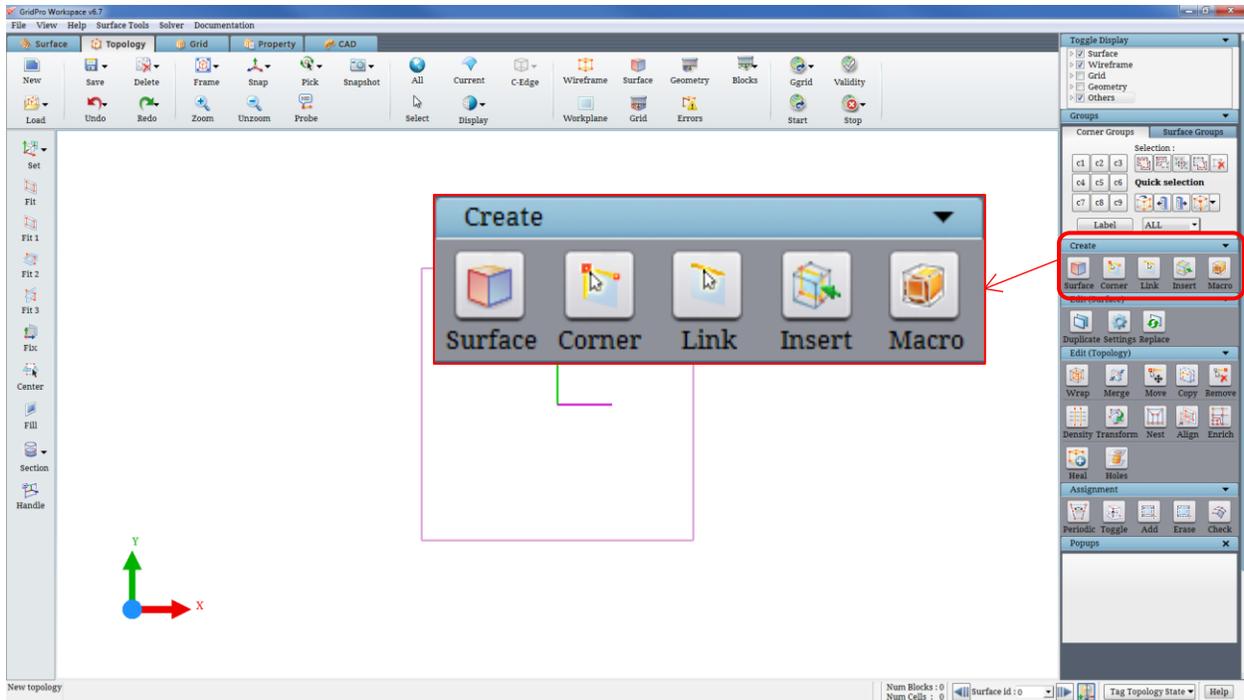


Figure 4.11: Create section

## 4.2.2 Corners

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.

In order to input a corner, switch on the *Corner* button  in the **Create** section of topology tab. Figure 4.13 shows the Create Corners dialog box. Select the mode of creation i.e. either using mouse click or by specifying the coordinates. By default the mode of selection is set to mouse that is specified as *Create On* and using the mouse click, the corner can be either created on the **workplane** or on **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 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 click the left mouse button on the workspace. Edges can then be created after corners are created; the workplane has no effect on edges.

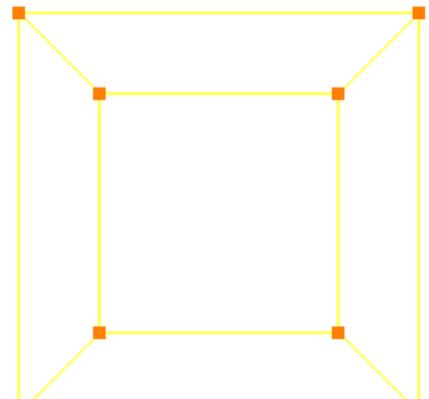


Figure 4.12: Wireframe

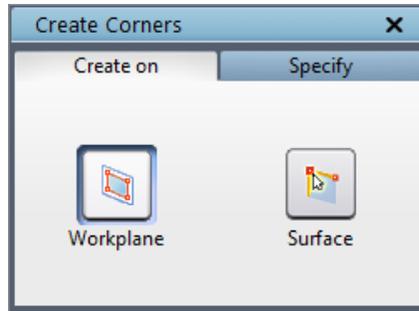


Figure 4.13: Corner dialog box

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

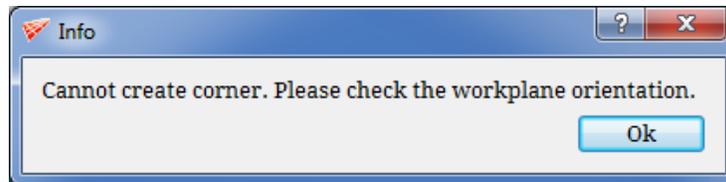


Figure 4.14: Error message when workplane is normal to screen

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

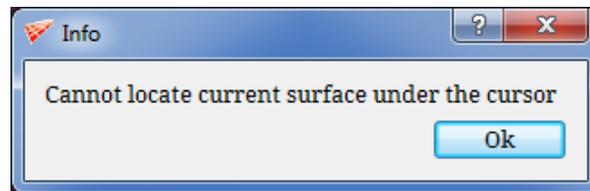


Figure 4.15: Error message when cursor is not on surface

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 and enter the coordinates in the provided fields and click **apply** to create the corner.

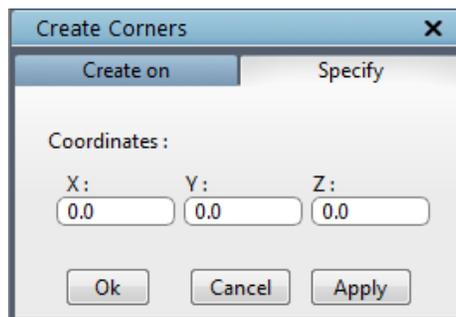


Figure 4.16: Specify tab dialog box

There are three shortcut options located under 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 mouse which will show the options as in Figure 4.17. The **center of rotation** option fixes the center of the rotation of the screen to the corner position. An axis similar to the global axis whose center appears on the corner which represents the center of rotation is fixed in that place.

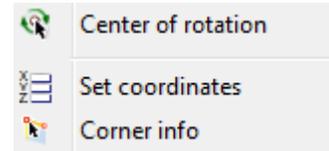


Figure 4.17: Right click menu of corner

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

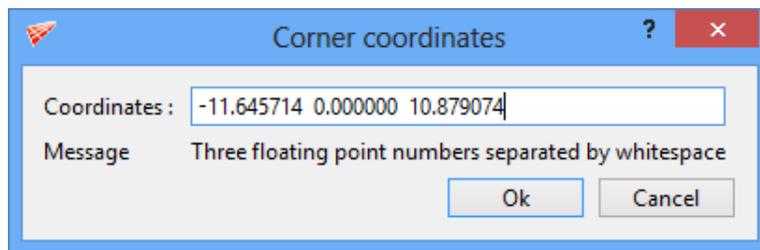


Figure 4.18: Set coordinate dialog box

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 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*.

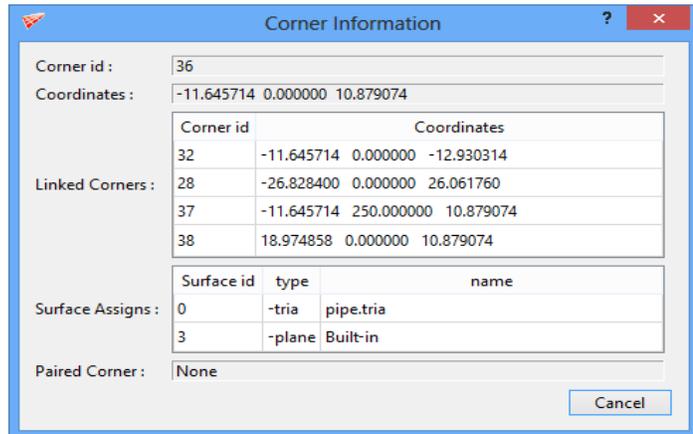


Figure 4.19: Corner info dialog box

### 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 **Topology** tab. Move the cursor to one corner, and click on it using 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.

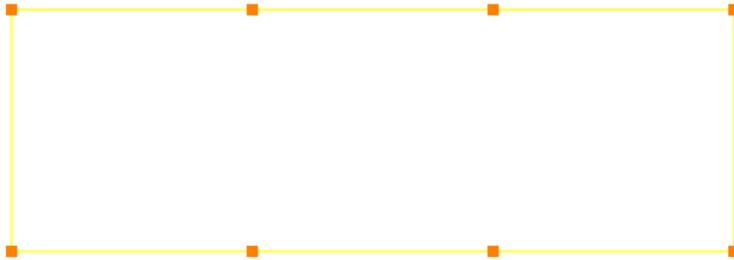


Figure 4.21: Strings of corners linked

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.

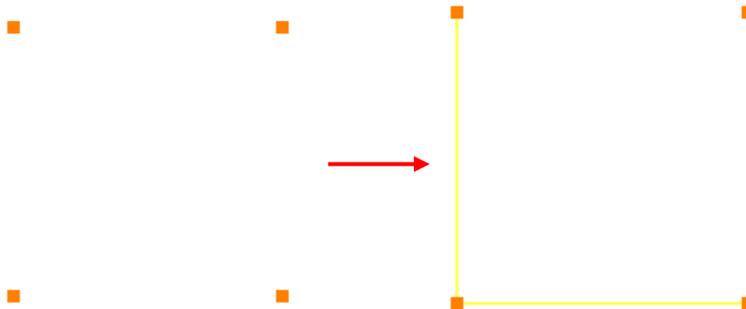


Figure 4.22: Corners linked after dragging a box around them

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 group corners to be linked.

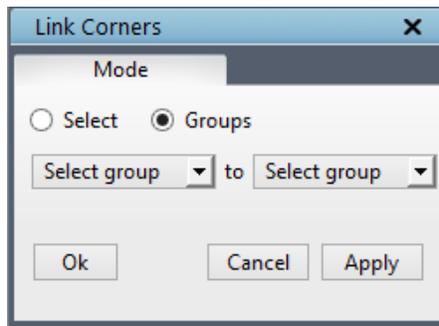


Figure 4.23: Link dialog box with Groups mode selected

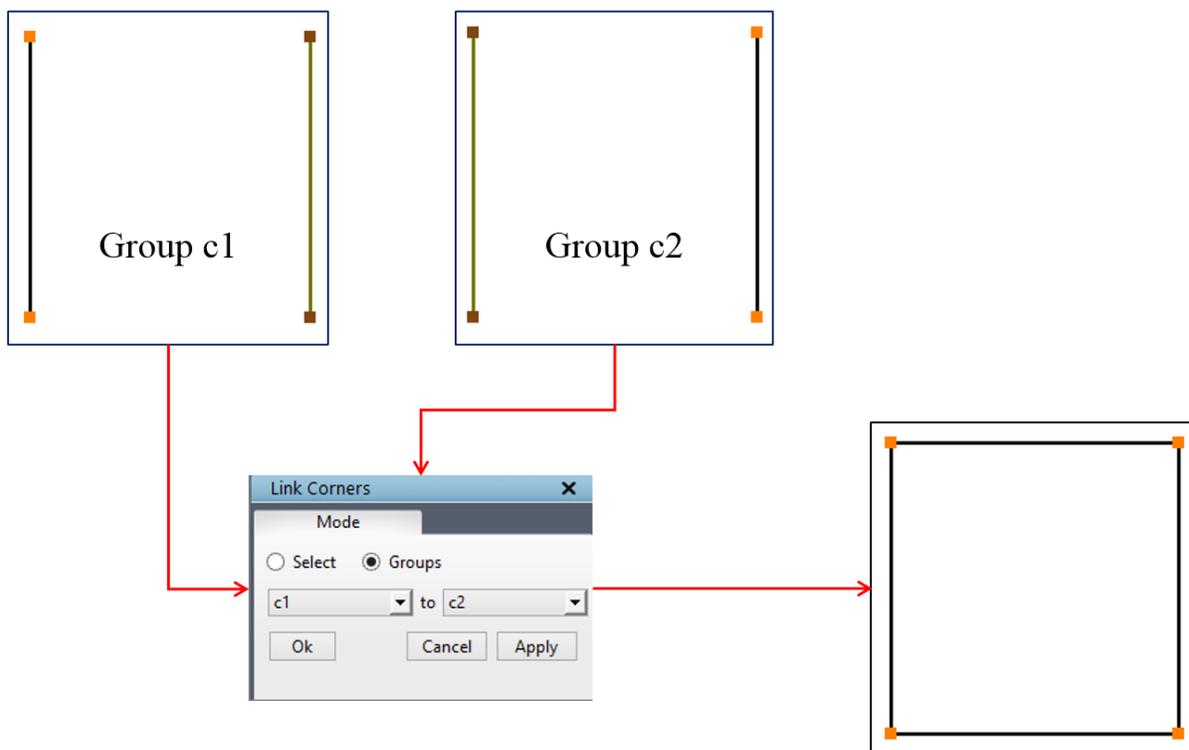


Figure 4.24: Linking corners using Groups

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 which enables you to create quickly these (often multiple) corners/sheet, within the boundary of an already existing wireframe.

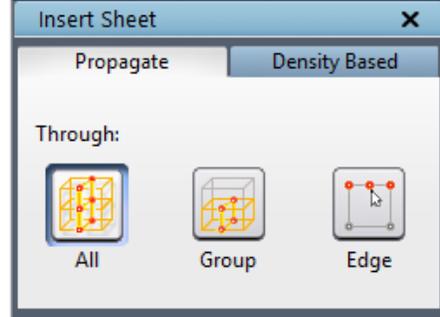


Figure 4.25: Insert dialog box

In order to insert a sheet on an already existing



wireframe, click on the **Insert** button in the *Create* section and a dialog box opens in pop\_up section as shown in Figure 4.25, which asks whether single inserts are to be done or multiple automatic inserts. By default **Propagate** method is selected which allows you to do manual inserts based upon the propagation mode selected.

Corner/Sheet inserts are created using a feature of the insert mode. The three insert modes are All, Group and Edge. All these three modes are shown when the Insert button is depressed. 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 '**T**' key on the keyboard and click on the edge where the insert has to be done.

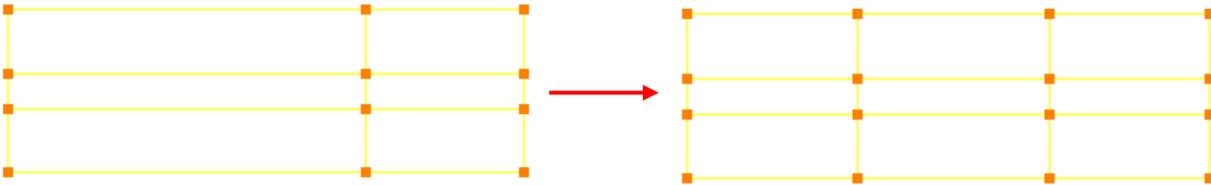


Figure 4.26: Insert propagation in All mode

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.

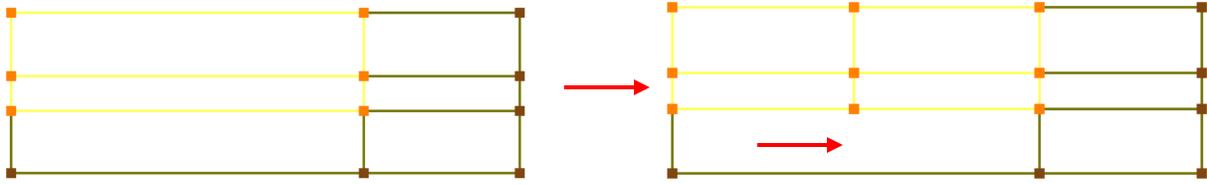


Figure 4.27: Insert propagation in group mode

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.

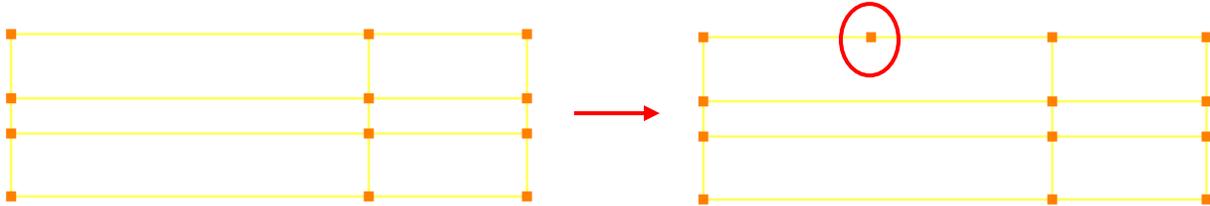


Figure 4.28: Insert propagation in edge mode

If mode **All** or **Group** is ON, and click on an edge, not only are corners formed proportionally along every edge parallel to the one you clicked; 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. Switch the **Insert** tab from **Propagation** to **Density based** 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 input the corner group in the fields provided. If no corner group is selected, it will consider the entire topology.

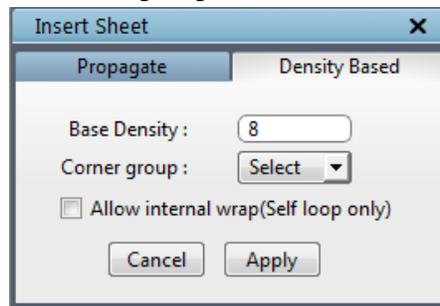


Figure 4.29: Density-based dialog box

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

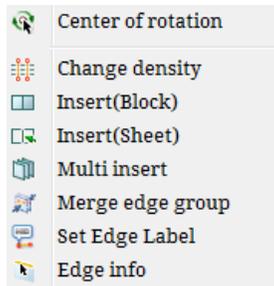


Figure 4.30: Right-click menu of edge

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 propagates 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 separate block hence it added an edge connecting those two edges but the Insert(Block) option evaluates such cases and avoids creating such links.

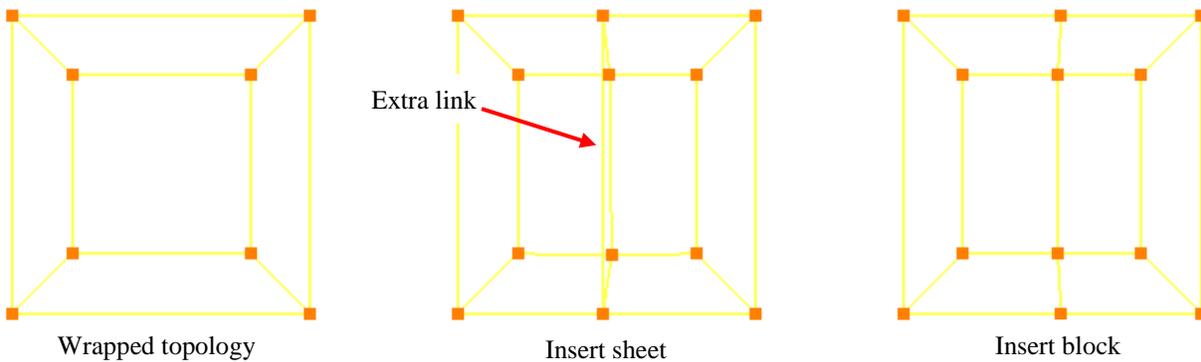


Figure 4.31: Difference between Insert sheet and Insert block

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

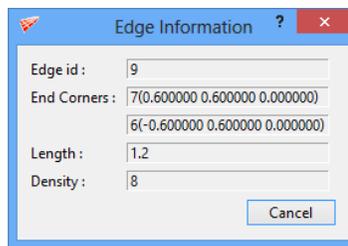


Figure 4.32: Edge info

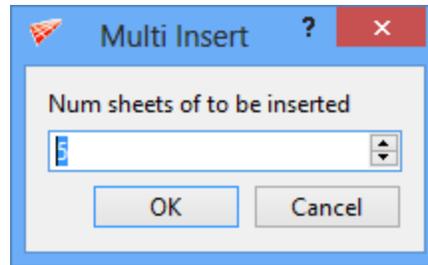


Figure 4.33: Multi insert dialog box

Similar to COR in corner right click menu, the COR in 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.

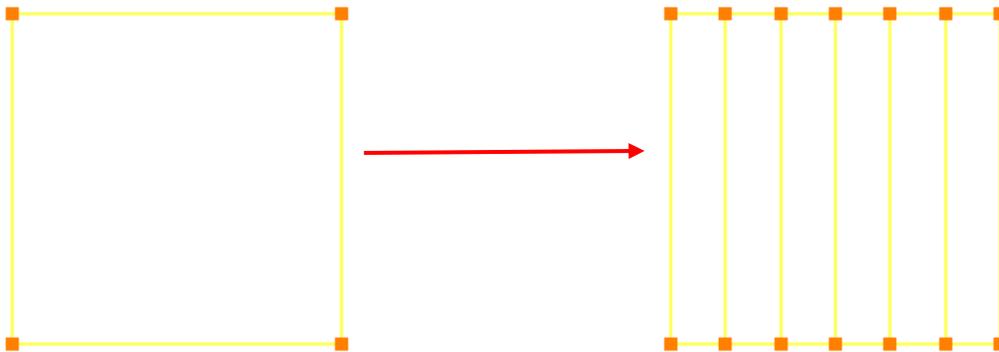


Figure 4.34: After Multi insert

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 otherwise it will pop\_up an error message. Similar to block and surface labels, **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 which can be often used for simple geometry creations and to create the outer domain. Therefore, the software offers some topological templates from which you can input objects such as surfaces and/or wireframe. **Macro** button is located in **Create**

section of the right side panel in **Topology** tab. Similar to built-in surfaces; macro has different options for 2D and 3D. In case of 3D - box, ellipsoid and cylinder macro are available as shown in Figure 4.35 and in 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 own macro dialog box which is located in right-hand side panel.

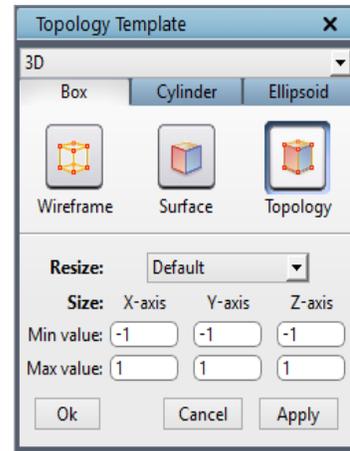


Figure 4.35: 3D Macro options

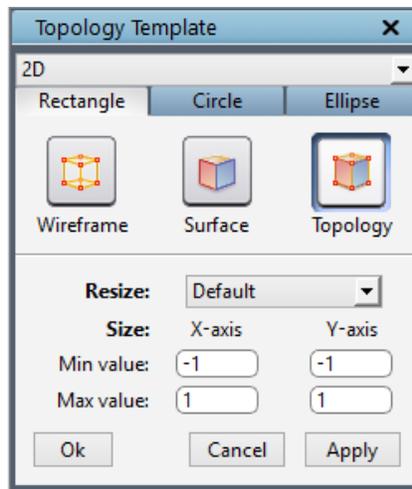


Figure 4.36: 2D Macro options

- **Box**

When **Macro** button is clicked, by default **Box** macro dialog box opens as shown in Figure 4.35, if it's 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** drop-down 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 drop down. This resize option automatically fills up the dimensions fields in the dialog box of respective macros. The default option allows you to

- 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 drop down list

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 axes and updates it in the below value fields, whereas the *fit current surface* option uses only the current surface boundaries. Similarly, *fit current surface group* option uses the surfaces in active surface group to calculate the boundaries and *fit workplane* uses the workplane position and its dimensions. *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 increases or decreases the existing values in the fields below the *resize*, depending upon 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.

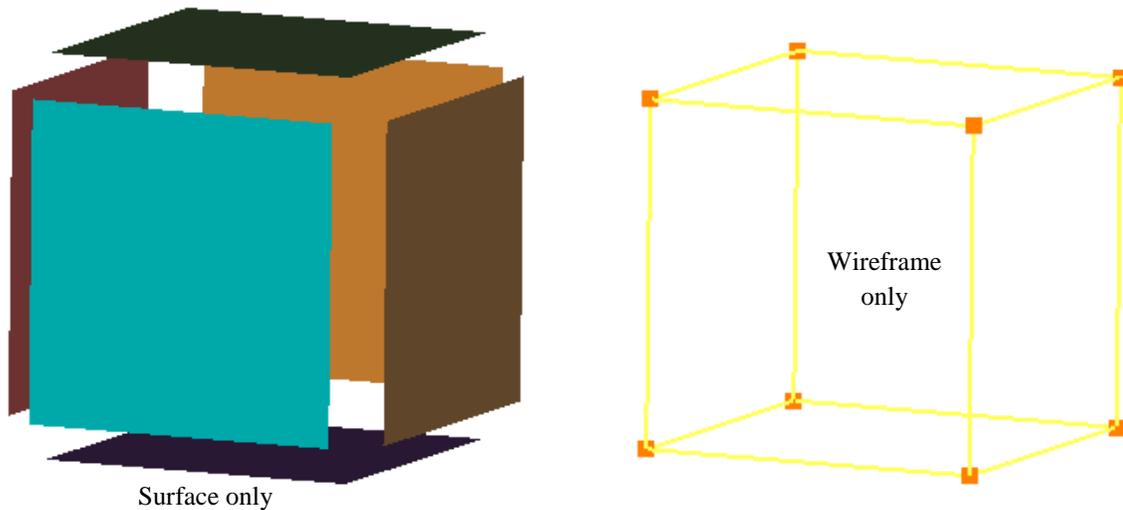


Figure 4.38: Box macro

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

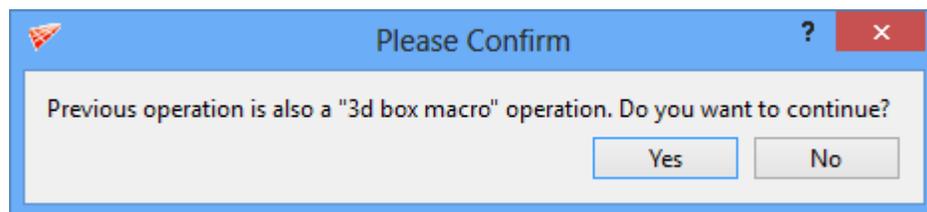


Figure 4.39: Warning message

- **Cylinder**

Next option in 3D macro is Cylinder. The *Cylinder* macro allows you to automatically create a cylinder with a center, radius, length, and rotation axis that you provide. Two planes will automatically be positioned so as 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 same for all the macro options. Rest all fields in each macro box require you to input dimension values belonging to each macro.

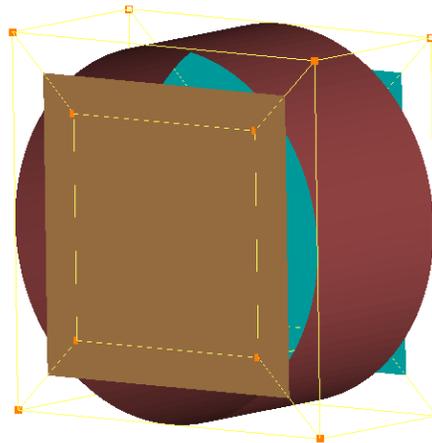
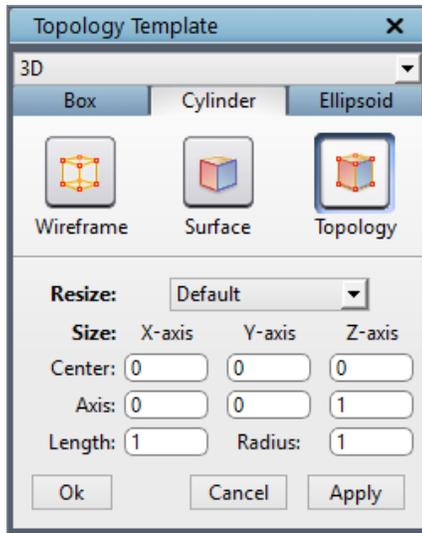


Figure 4.40: Cylinder topology macro

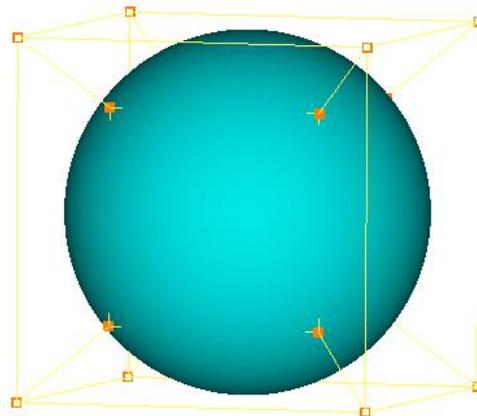
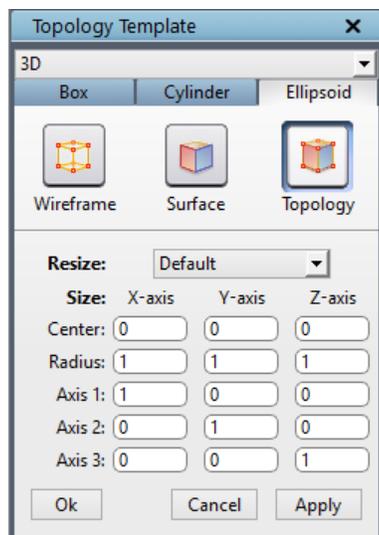


Figure 4.41: Ellipsoid topology macro

- **Ellipsoid**

The *Ellipsoid* macro enables you to create an ellipsoidal shape region with a center, axis and radius. Sphere is a special case of ellipsoid; when all the three axes have the same radius, 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, dimension field requires a center, axis and radius value in all three directions. These fields can be either filled manually or automatically using the *resize* drop-down list by choosing one among the list which best suits your need.

- **Rectangle**

Only if a 2D workspace is opened, 2D macro options are enabled. You cannot create a 2D macro in a 3D workspace. Figure 4.42 shows the drop-down list of the same.

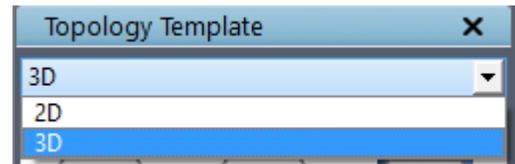


Figure 4.42: Dimension drop down

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's a 2D, you only required 4 values x min-max and y min-max. Figure 4.36 shows the dialog box of rectangle macro.

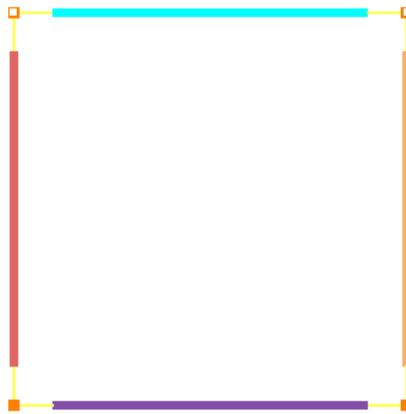


Figure 4.43: Rectangle topology macro

- **Circle**

The Circle macro enables you to create a circular surface and its corresponding topology. It can be created with a center and the radius. Circle center in x and y co-ordinates and radius need to be defined. Figure 4.44 shows the dialog box of the circle macro.

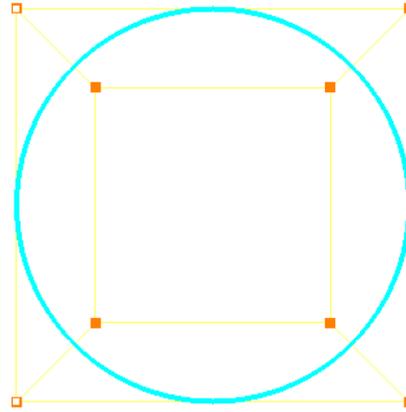
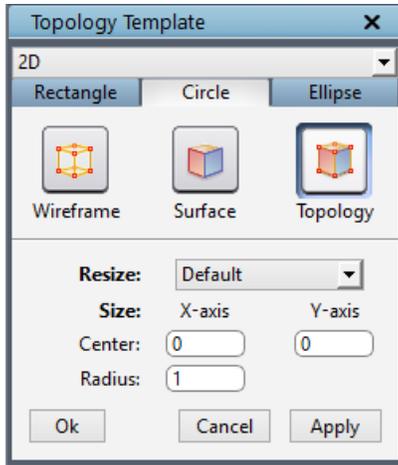


Figure 4.44: Circle topology macro

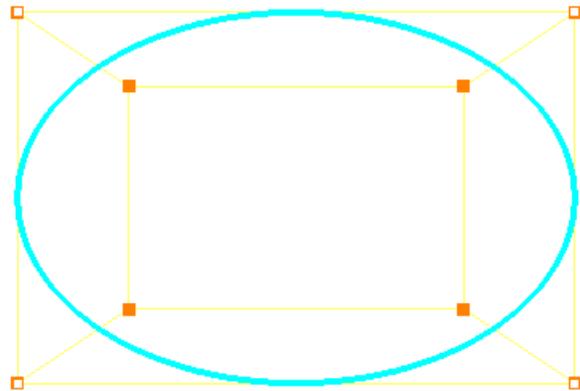
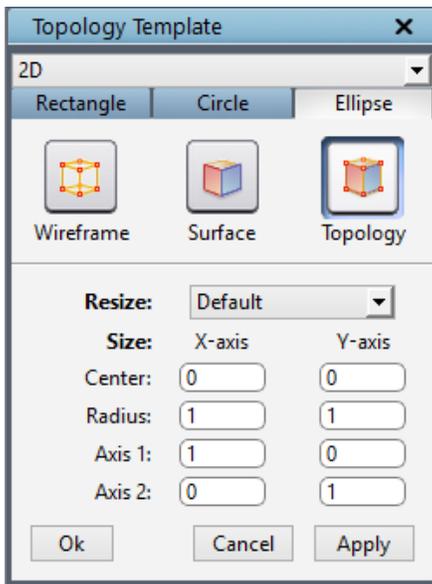


Figure 4.45: Ellipse topology macro

- **Ellipse**

The *ellipse* macro enables you to create an elliptical region of any shape with a center, axis and radius. To change the shape of the ellipsoid, change the radius value of axis 1 and axis 2. By giving same radius value on both the fields, you will create a circle. Instead you can use the circle macro. Since it's a 2D surface, only two coordinate axes is required. Figure 4.45 shows the Ellipse dialog box.

## 4.3 Automatic Topology Creation

Automatic topology creation tools help in creating topology automatically using the surface definition. **Holes** is the only tool which is currently added to the UI, more tools will be added in the upcoming releases.

### 4.3.1 Holes

**Holes** or **Hole topology** helps in creating topology and internal surfaces automatically. For. E.g. Consider an injector which has many pipes connecting two surfaces as shown in Figure 4.52. In such cases, topology has to be built individually for each pipe using insert and wrap. In order to ease this process, you can automatically create an individual topology for each pipes using Holes command. Clicking on **Holes** will open the dialog box as shown in Figure 4.46.

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 which 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.

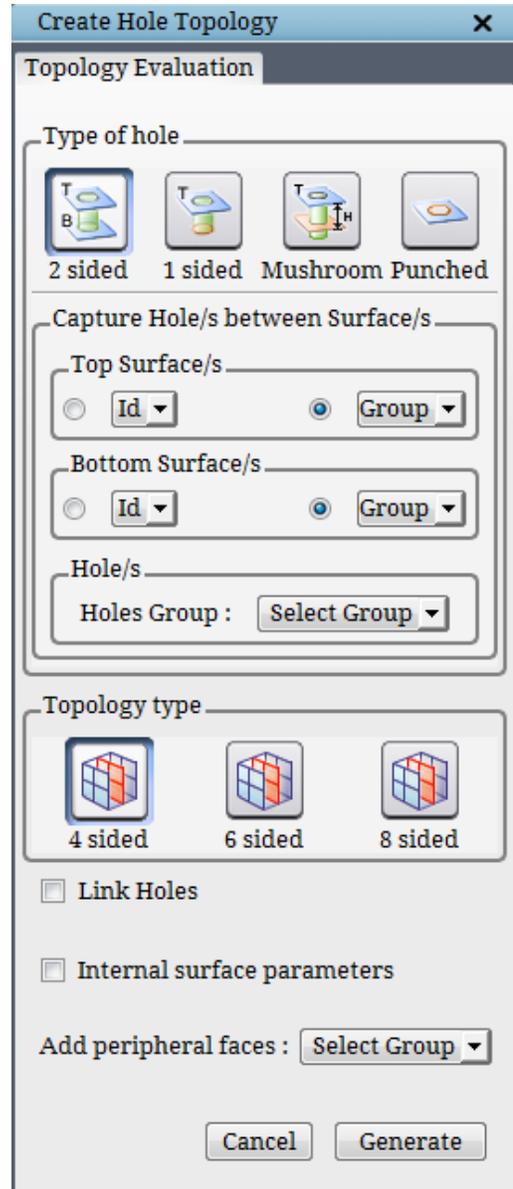


Figure 4.46: Hole topology dialog box

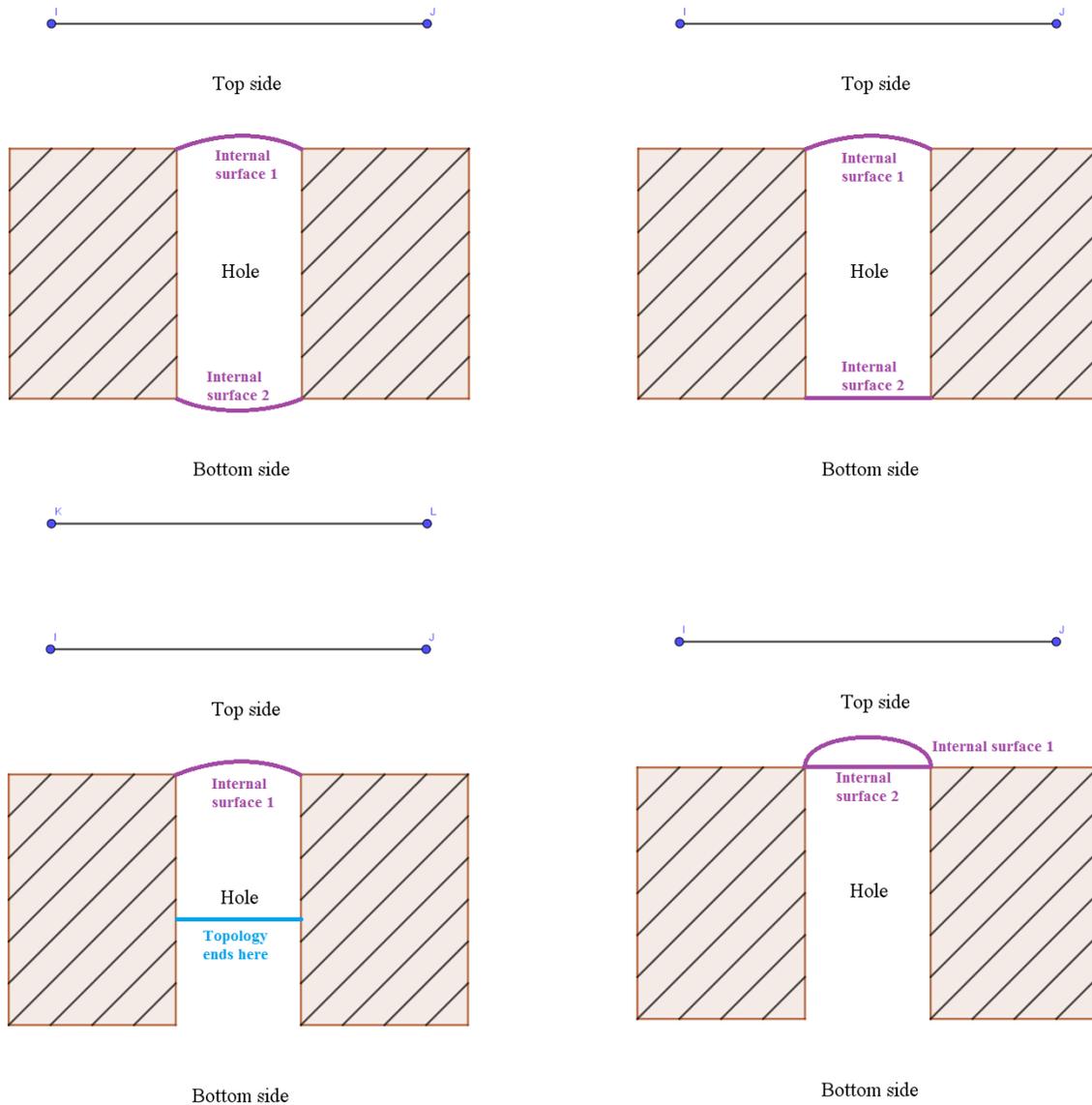


Figure 4.47: Type of Holes

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 type. For mushroom and punched hole type, you only need top surface id/group and hole/s surface id/group.

Next section is the **Topology Type**, which can either be 4 sided, 6 sided or 8 sided. i.e. at any given plane/axis, the output topology has 4 faces like in a cube, 6 faces as in hexagon and 8 faces as in octagon. In v6.7, only 4 sided topology has implemented the rest two are yet to be implemented in the upcoming release.

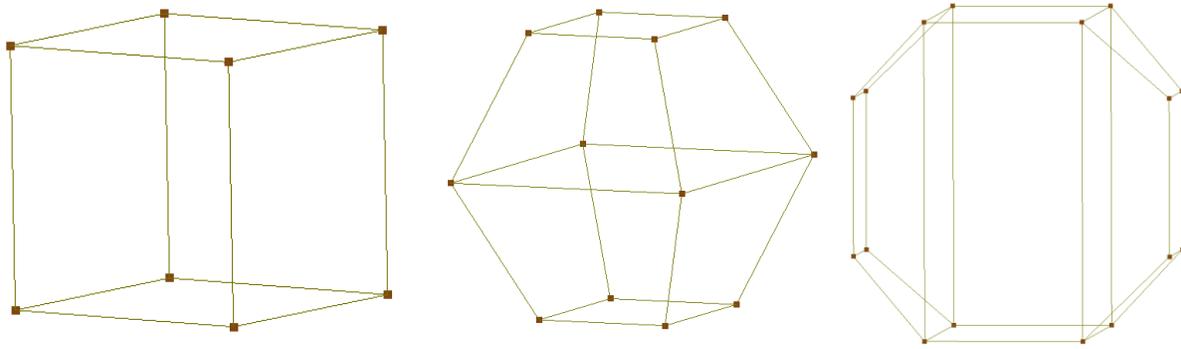


Figure 4.48: Type of Topologies

These are the mandatory input parameters which are necessary for generating the hole topology. For advanced controls, more options are grouped inside Link Holes and Internal Surfaces checkbox. Enabling the **Link Holes** checkbox opens the required parameters as shown in Figure 4.50. Link Holes helps 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 edit them manually after getting the respective hole centers. There are 5 different pre-defined patterns which are

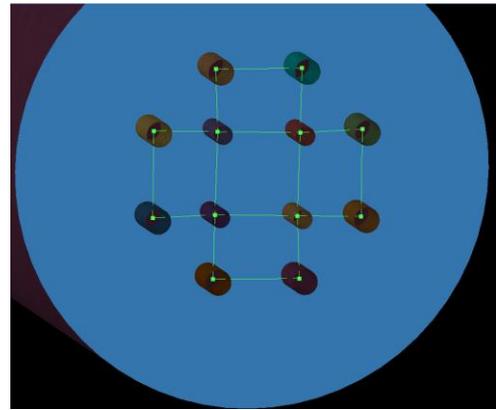


Figure 4.49: Pattern calculated using Rectangular option

1. **Hole centers** – which creates a corner at every hole center for linking them manually if the user does not get the desired pattern from the pre-defined options.
2. **Linear** – which will be able to capture the linearly arranged holes.
3. **Circular** – which will be able to capture the holes which are arranged in circular manner. For e.g. Holes in Injectors.
4. **Rectangular** – which will be able to capture the holes which are arranged in rectangular pattern.
5. **General Tria** – This uses the triangulation of the surfaces and determines the holes pattern

All these options, capture the hole centers 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 of the hole center and linked together as per the selected option. If none of the options have given the desired linking pattern, users can select hole centers which will create the green dots at each hole without linking them. Then enable the **Edit pattern** checkbox to convert the

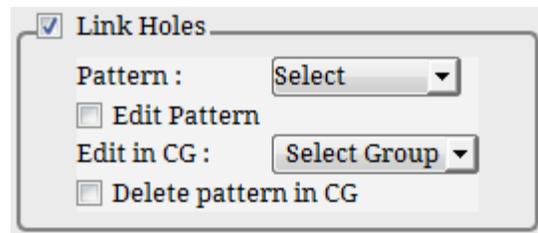


Figure 4.50: Link Holes dialog box

green dots to corners and group them in the desired corner group using the **Edit in CG** drop down. 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.

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 4.51. It is same as the parameters given in the Internal Surface creation. Please refer section 4.11.8 for more information on those parameters.

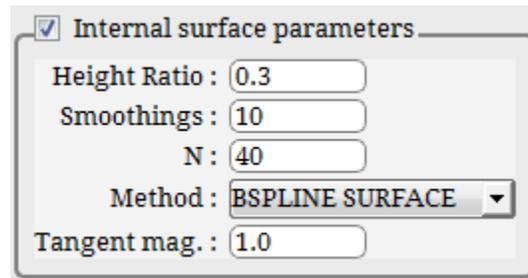


Figure 4.51: Internal Surface dialog box

The last option is to group all the outer corners of the hole topology in order to proceed with the topology building for the rest of the regions. If a corner group is selected from the drop-down, 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 command line.

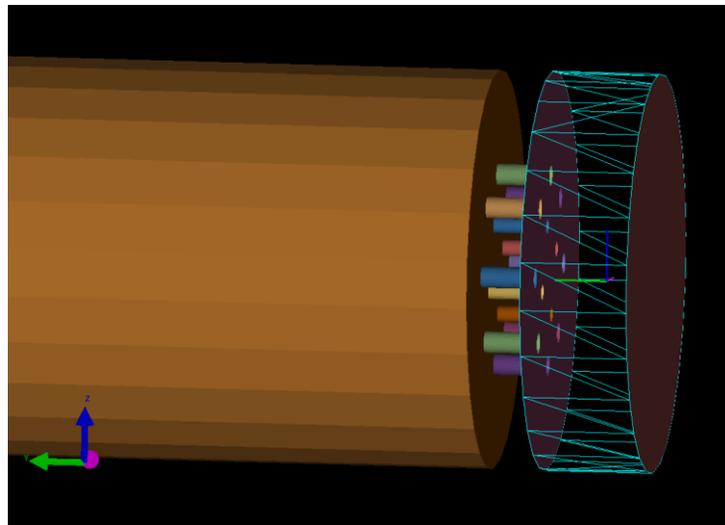


Figure 4.52: Injector Surfaces

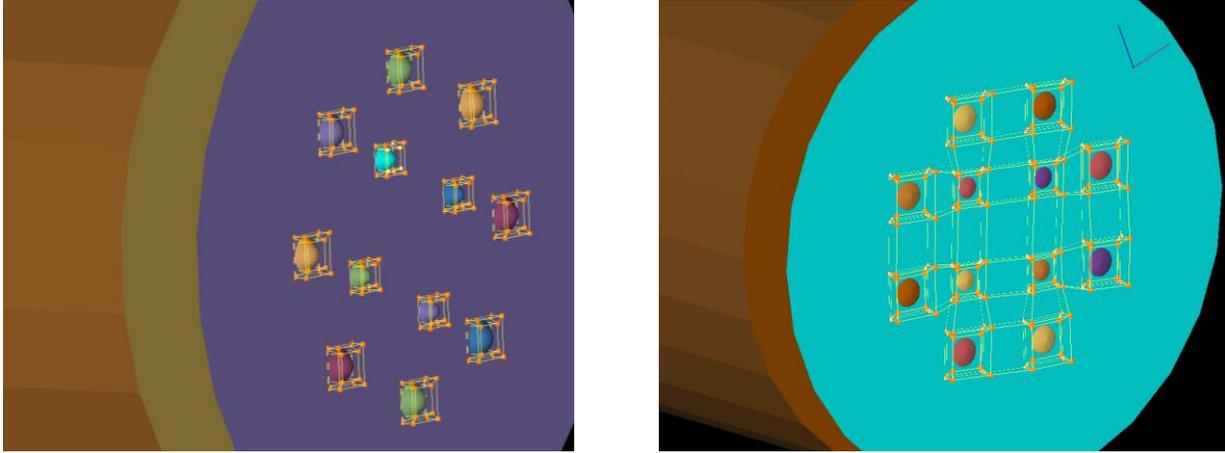


Figure 4.53: After executing hole topology command with and without links

## 4.4 Topology Grouping Utilities

One tool that helps you create three-dimensional topologies is the workplane; another tool is 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 also. 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 prefix for corner groups and 's' as prefix for surface groups. Groups are located below toggle display section in right side panel. Groups are divided into two tabs such as **Corner Groups** tab and **Surface Groups** tab. Corner groups tab can be seen in Figure 4.54. 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.

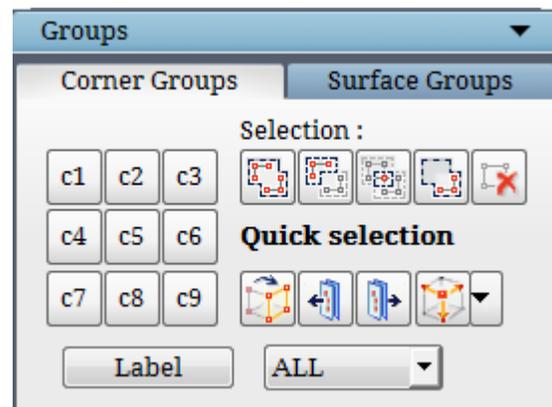


Figure 4.54: Corner Groups tab

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 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 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 Selection section are persistent, i.e it acts like 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 will add to the active corner group. Similarly switch ON the  button and click on 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.

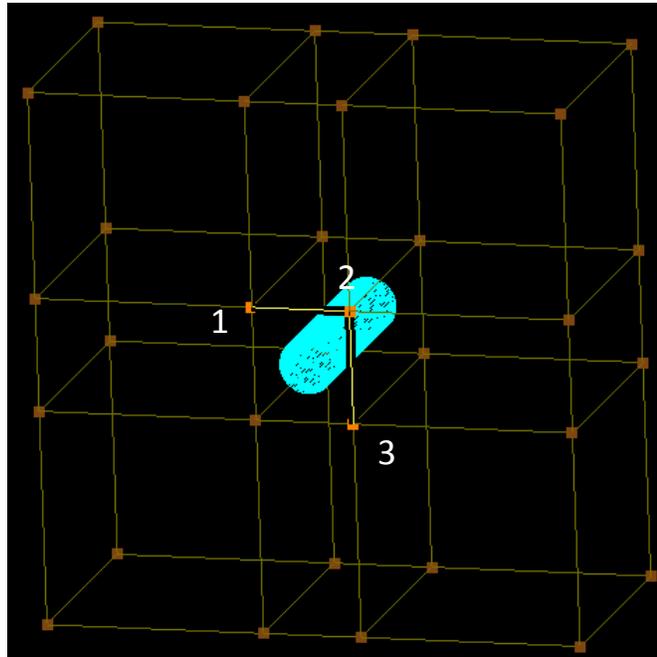


Figure 4.55: Grouping corners by picking individually

The next two, intersection and equal operations help you more efficiently add corners to, 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.

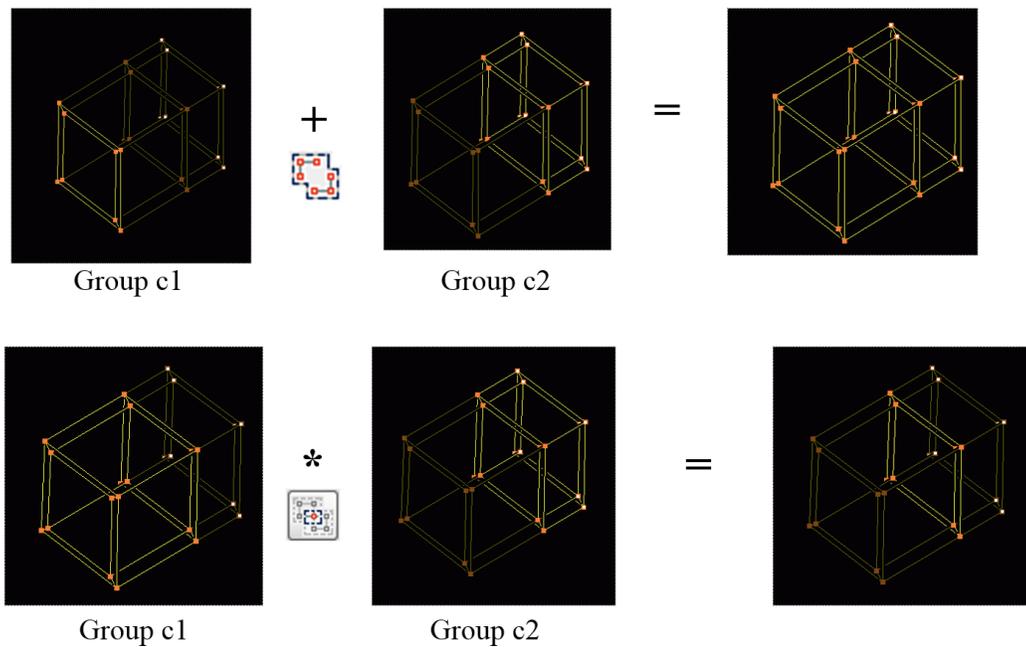


Figure 4.56: Grouping using two different corner groups

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 also add/subtract two groups, i.e. you can 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 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 4.57. These buttons are more like a separate groups which help in quick selection of the desired corners.



Figure 4.57: Quick selection widget

Depends 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, will add all the corners to the active group.  represents the surface assignment group as explained in *section 4.4.3*,  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] drop-down list in **Groups** section. There are 13 group display modes available as shown in Figure 4.58. 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 drop-down list. By default, the grouping display mode is All. When **ALL** 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 dimmed. **GROUP** mode refers to Active Group. When group display

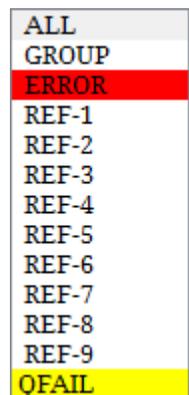


Figure 4.58: Grouping modes



Figure 4.59: Validity check

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. By setting the group mode to reference group (Ref-1 to Ref-9), you localize your work, eliminate screen clutter, and effectively manipulate 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 4.59. **QFAIL** group refers to the corners which will output a bad quality. These are highlighted by the Validity button as singular topology. For e.g. the topology may have reported mildly, mediumly or very severe singularity and the corresponding edges would be highlighted by the validity button. Similar to Error group, QFAIL group also automatically groups those singular corners into it and 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*. If the output grid has folds, the faces which are associated with these folds are automatically grouped into QFAIL group while running the quality check.

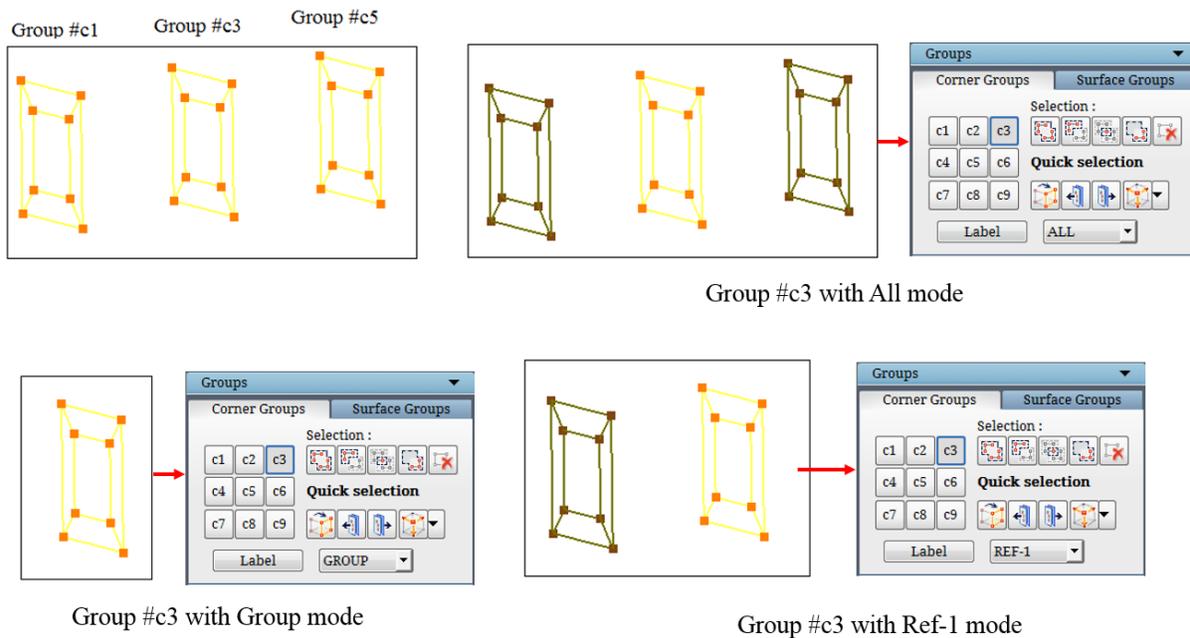


Figure 4.60: Different grouping modes

### 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.

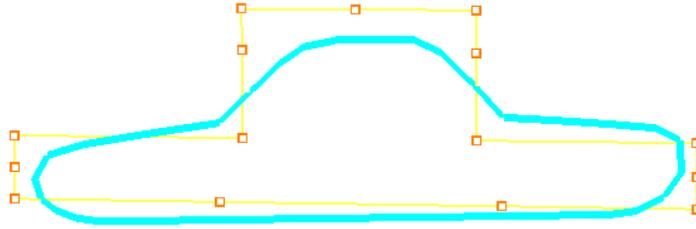


Figure 4.61: Surface assignment group

#### 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 on the parts of a group that was the results of your actions (e.g. wrap, copy, add, subtract, etc).  shows previous selections while  shows forward selection.

#### 4.4.5 Span out

You can directly add or remove sheets of corners from active group by using span out or peel off button. The drop list under this button is shown in Figure 4.62. It has 9 options, 5 options belonging to span out means adding corners to active group and 4 options belonging to peel off mean subtracting corners from an active group. Using **span out** layer 1, you can add one layer of corners in all direction 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 upon your selection of group mode display (A or R)]. **Span out all**, adds all the corners in reference topology to the active group.

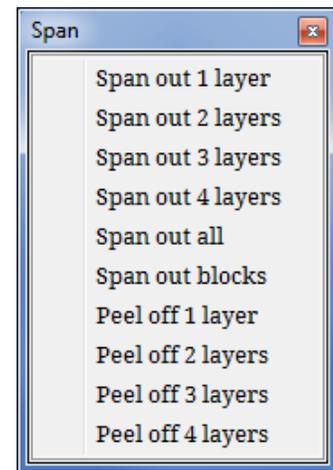


Figure 4.62: Span drop down list

This option is really useful for checking the connectivity between the topology. If you have two distinct pieces topology, you can separate that out 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 4.57, 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.

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

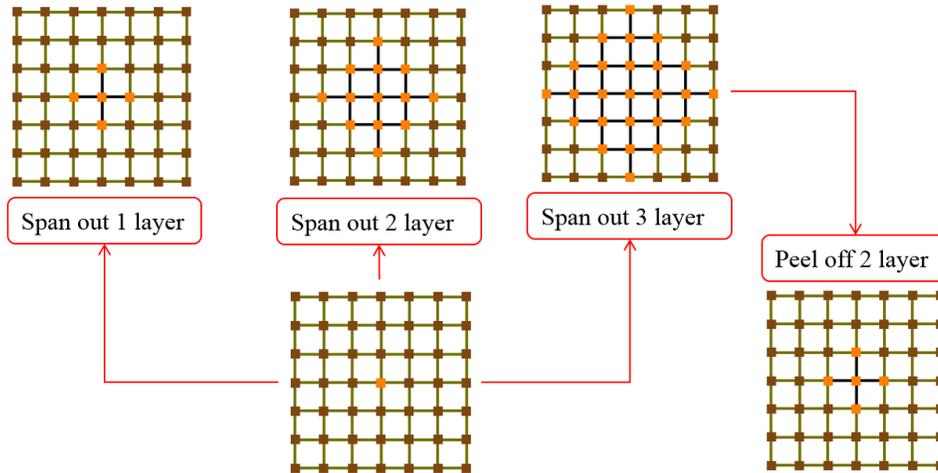


Figure 4.63: Examples for Span out and Peel off options

#### 4.4.6 A KEY

In addition all the buttons for grouping operations, keyboard shortcut such as **A** key available to select a sheet from 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 active group of corners by holding **A** key and clicking on 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 active corner group. Figure 4.64 shows the sheet selected after using **A** key .

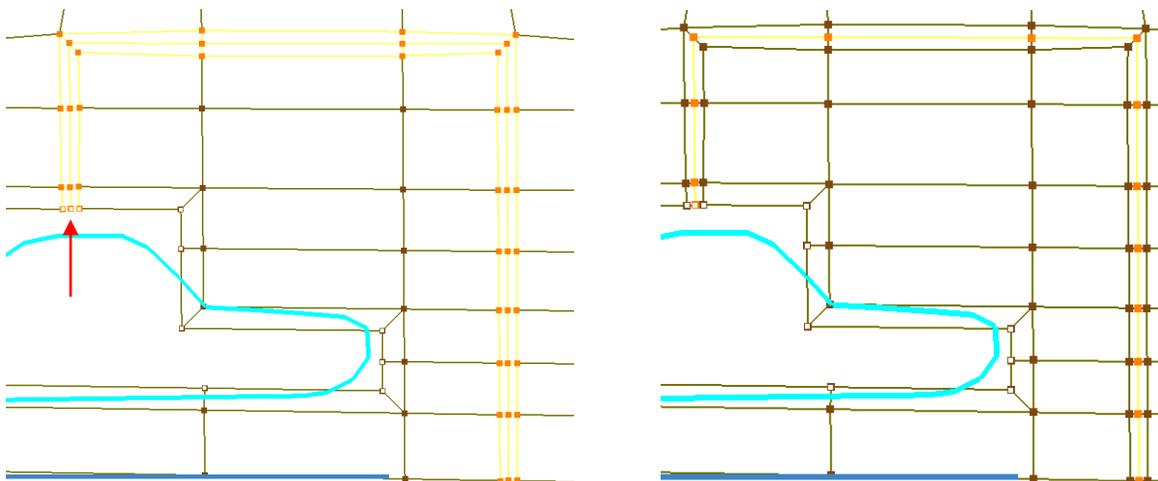


Figure 4.64: Sheet selection

## 4.4.7 Surface Grouping Tab

You can access surface grouping options by depressing *Surface Groups* tab in *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 exactly to similar to corner group options. Surfaces belonging to active groups are displayed in different colors whereas inactive surfaces are displayed as white grey color 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 you can use this ALL group to add all surfaces to active group.

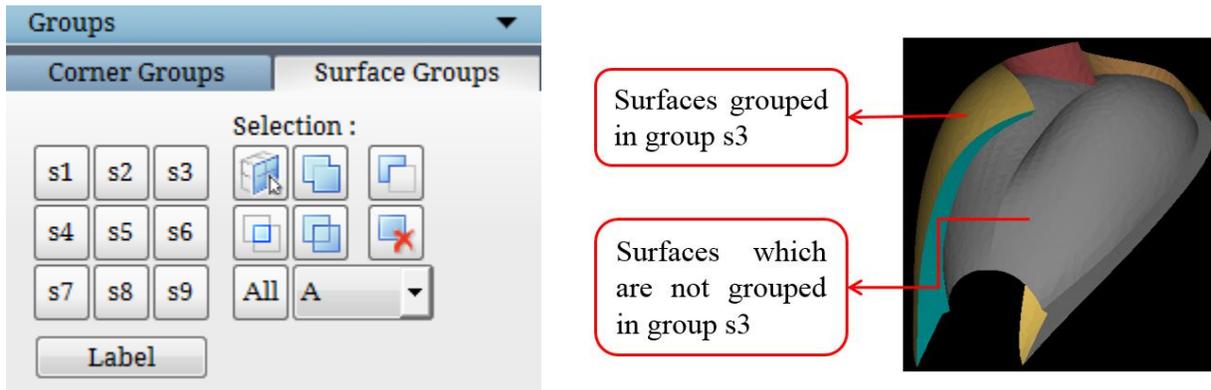


Figure 4.65: Surface Groups

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

## 4.4.8 Labels

Labels refer to labeling a given group of corners or surfaces for easier selection and visualization. Label button is accessible from the Groups section in the right side panel. Clicking on the Label button will pop\_up a dialog box as shown in Figure 4.66. You can choose to label either blocks, surfaces or both. By default Block label is checked, select the corner group and enter a name to label the blocks and click 'Apply'. Once it is labeled, the same can be seen from the Group modes drop-down list. It can also be seen in the \*.fra file once the topology is saved.

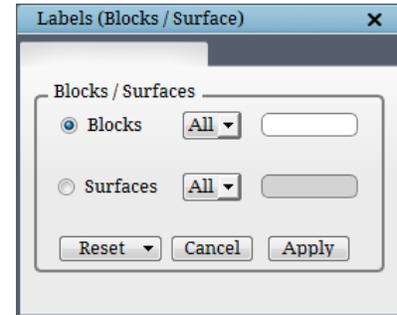


Figure 4.66: Labels dialog box

Similarly, the surface groups can also be labeled. 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*.

## 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 which helps in modifying the topology is also explained. It is located under **Surface Tools** menu in Menu bar.

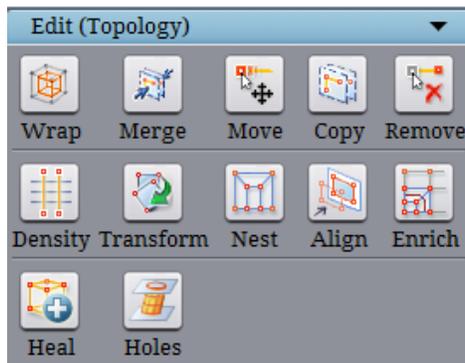


Figure 4.67: Edit (Topology) section

### 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 similar pattern with same number of corners into one. Merge can be done in two ways by dragging 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 4.68. First is **To Center** option that will position the merged corners to the center 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'. 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.

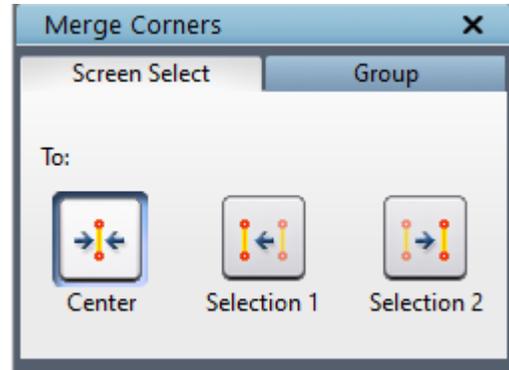


Figure 4.68: Screen select tab dialog box

Figure 4.69 shows the corners merged at different positions.

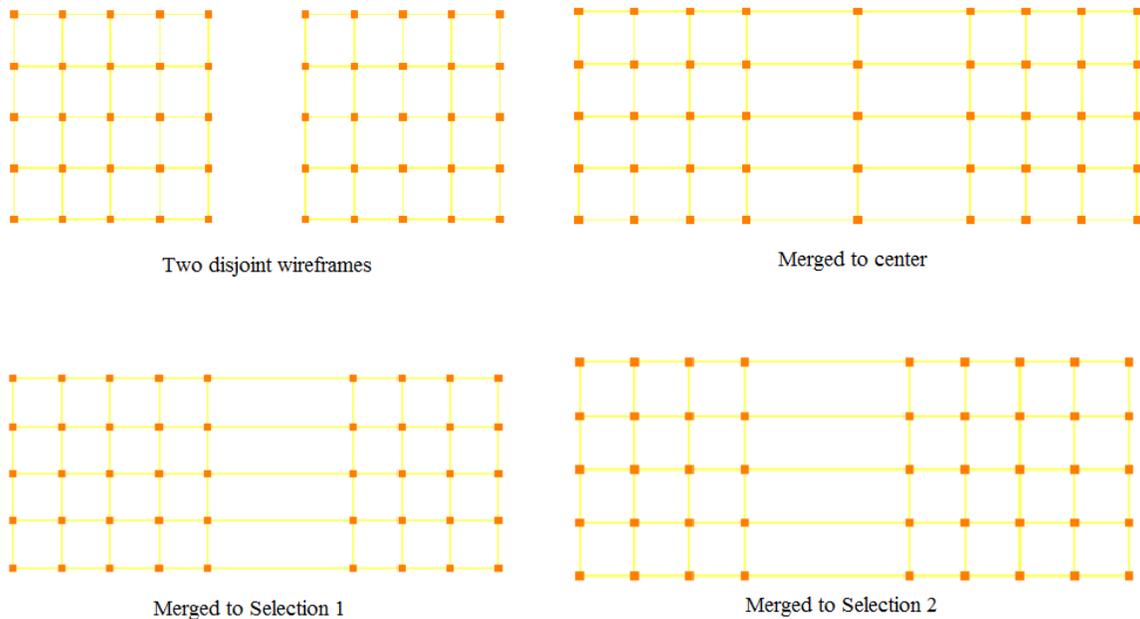


Figure 4.69: Corners merged into different positions

The other way to merge is by using **Groups**. Switch to **Group** tab under Merge dialog box as shown in Figure 4.70. Firstly 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. If you want to merge it to center of the two selected corner sheets, enable **To center**. If **To center** is not enabled, it will position the merged corners on the second selected group.

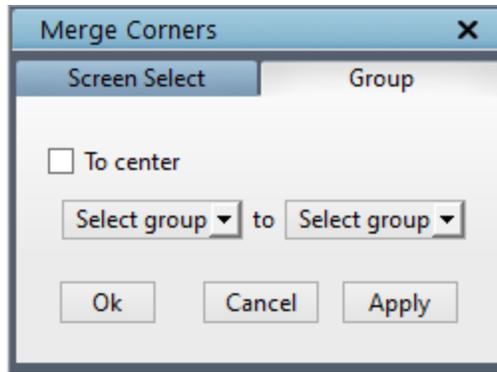


Figure 4.70: Group tab dialog box

The third way to merge is by using **Merge edge group** option under 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 4.71 shows the 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. Corner move on screen 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. If the flags surface and parallel is selected, and if you move the corners outside the surface, 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.

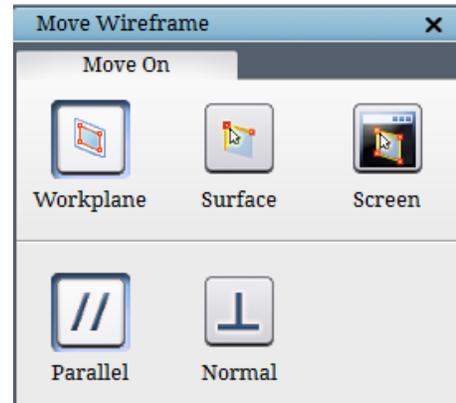


Figure 4.71: Move on dialog box

You can also move a corner group along with the workplane. In order to move a corner group, first, fit the workplane center to the center 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 in 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 get shrunk) 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.

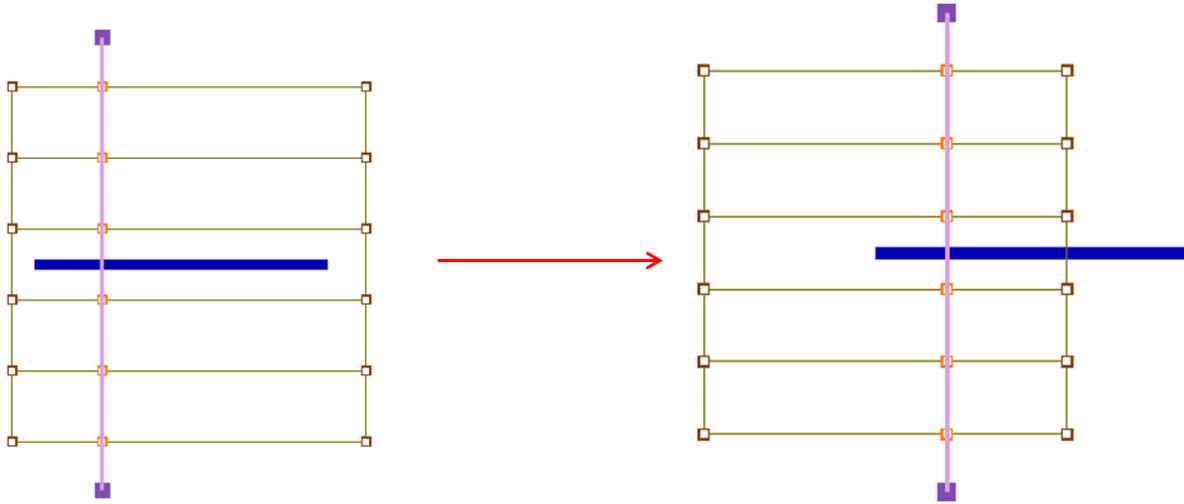


Figure 4.72: Corner group moved using workplane

### 4.5.3 Copy

For geometries with same cross section or symmetric sections, it is easy 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 right side panel in **topology** tab. Figure 4.73 shows the Copy Wireframe dialog box.

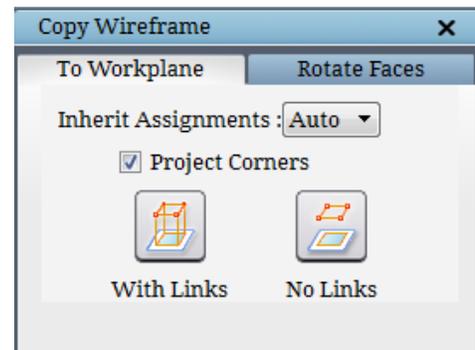


Figure 4.73: Copy dialog box

- **To Workplane**

The **“To Workplane”** tab 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 first operation **With Links** represents perhaps the often used tool. This operation in addition to translation of the grouped corners links the duplicated set of corners to original set of corners. Essentially, the **With Links** operation saves you time because the operation creates a duplicate of a wireframe in a different plane, then creates edges between the wireframe and it’s duplicate automatically.

The other operation **No Links**, simply duplicates the group structure on the workplane such that the center of mass of the copy of the group and the center of the workplane rectangle coincide.

NOTE: After this action, a new group is formed, one comprised of the original group and the other, it's 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 **'Project Corners'** along with **'With Links'** option. The checkbox next to *Project corners* should be enabled to use this feature in addition to *With Links*. This operation duplicates the active group by first moving its center of mass to the workplane center 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 two operations in the Copy dialog box, this operation adds duplicated corners to the original group.

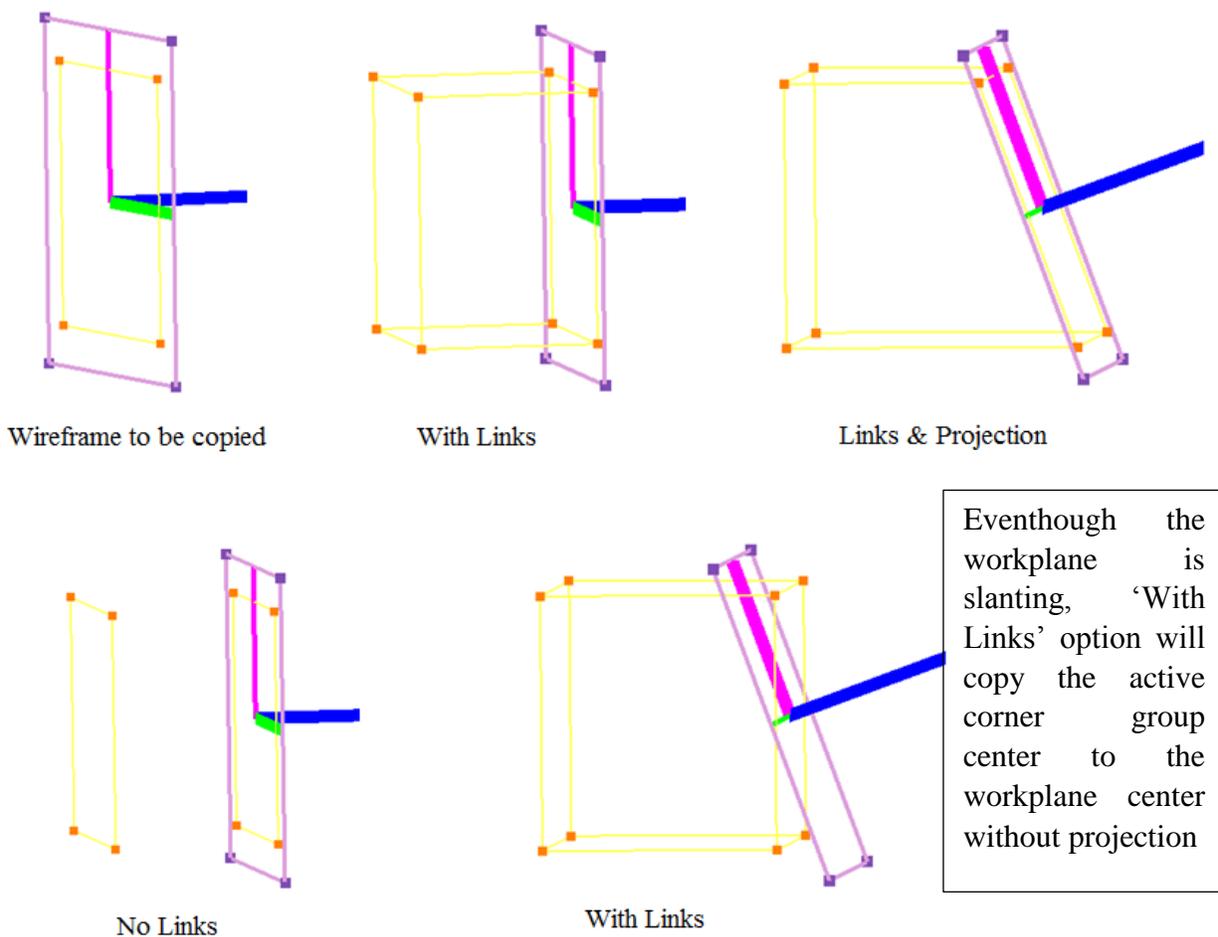


Figure 4.74: Various copy options

All the above 3 operations can choose various ways of carrying forwarding the surface assignments if there are any in the given corner group. It is same as the **'Inherit Assignments'** option in the Wrap button. **Auto** smartly decides whether to copy or transfer the assignments to

the copied group based on the neighboring surface assignments. **All** copies all the surface assignments to the newly created corners. **None** retains the surface assignments with the input group and does not copy/transfer to the newly copied group.

- **Rotate Faces**

**Rotate Faces** tab is used for copy and rotating the given wireframe using the angle, number of instance and the pitch. This tool can be useful where geometries have a 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 Sheet rotation can be used to copy the wireframe. Clicking on the **Faces** button under **Edit (Topology)** section opens a dialog box as shown in Figure 4.75.

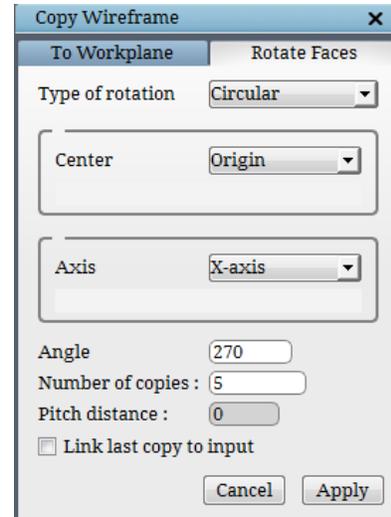


Figure 4.75: Faces tab dialog box

The type of rotation defines whether it is a Circular or Helical path. If helical is chosen, the pitch distance option gets enabled otherwise it is disabled. The **center 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 instance is 2 then you will have the wireframe at the starting point and then at the ending point; if it's 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. And finally, 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 at the end.

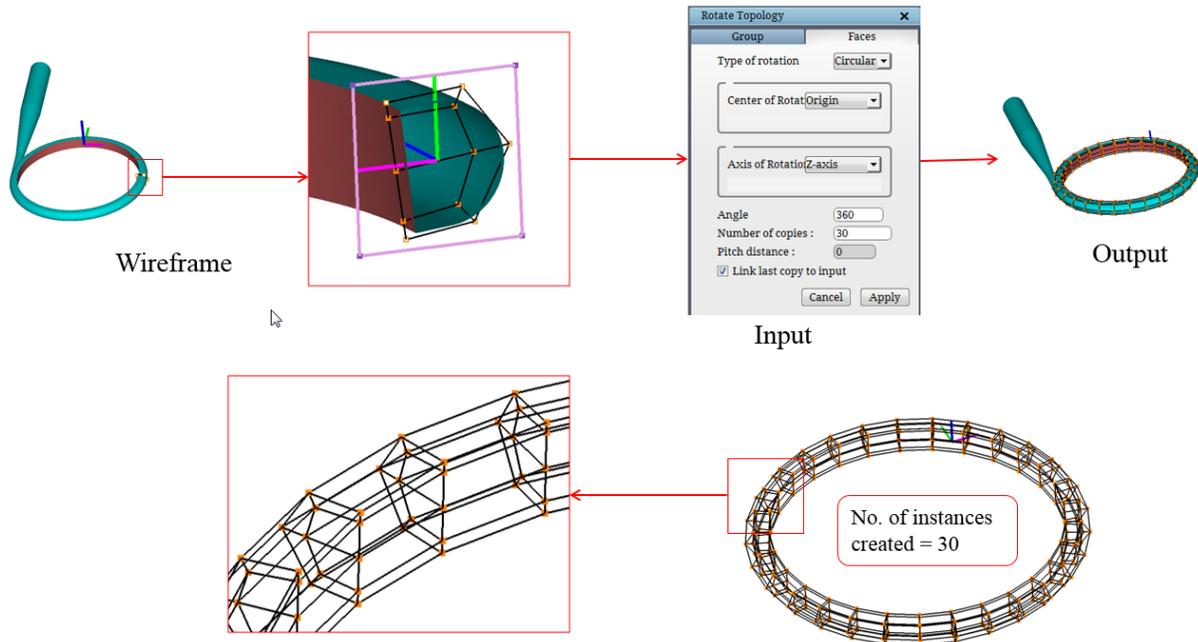


Figure 4.76: Rotate → Faces

#### 4.5.4 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 right side panel in 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 key 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.

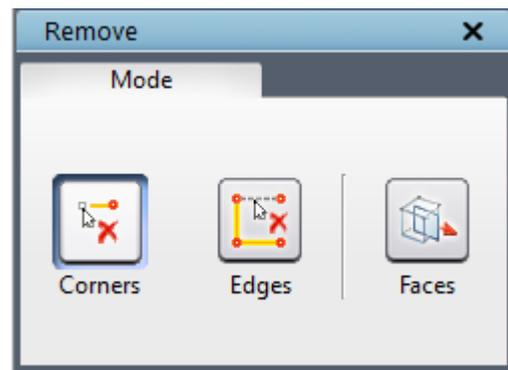


Figure 4.77: Remove dialog box

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

There is another operation which is also embedded with remove button i.e, **Faces**. **Faces** removes a sheet of corners if its pattern replicates the adjacent sheets. It is the reverse of insert, where it inserts a sheet similar to adjacent sheets. But **Faces** remove the sheets if it is similar to the adjacent sheets. It is helpful in removing the extra blocks in order 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 4.78 if remove 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 of human effort.

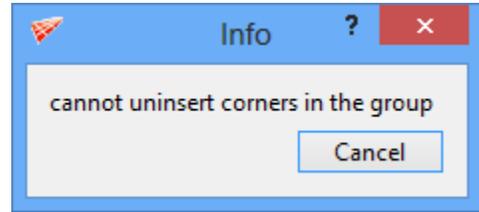


Figure 4.78: Error message

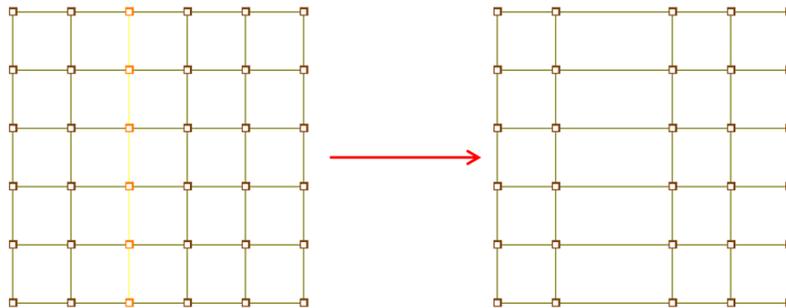


Figure 4.79: Remove → Faces

### 4.5.5 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 **surface tools** menu. This is helpful when you want to run 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 drop-down opens a dialog box as shown in Figure 4.80.

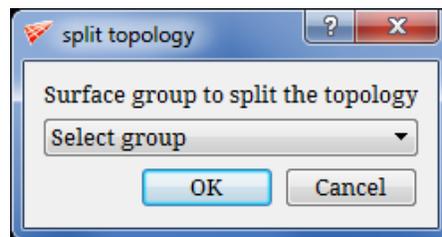


Figure 4.80: Split topology dialog box

For 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 4.81.

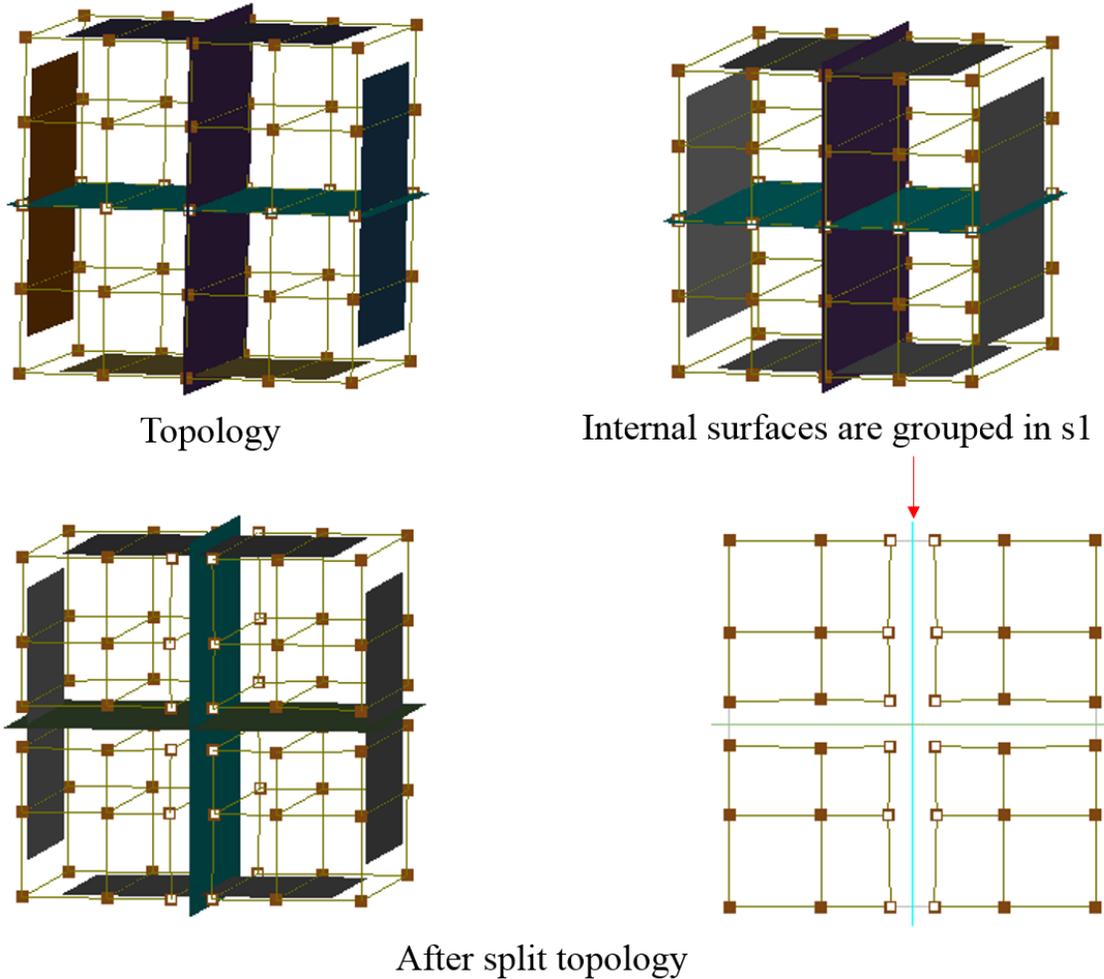


Figure 4.81: Split topology

## 4.6 Topology Transformation Utilities

Operations such as Translate, Rotate, Mirror, Scale and General transformation under **Transform** 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 4.82. The first section contains all the common input parameters such as corner and surface group, options to enable transform current surface 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.

## 4.6.1 Translate



When **Translate** button is clicked, it opens a dialog box as shown in Figure 4.82 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 drop-down list will be translated. So make sure the right group is selected before executing the operation.

## 4.6.2 Rotate



When **Rotate** button is clicked, Rotate Topology dialog box opens as shown in Figure 4.83. It deals with rotating the given corner group and surface group to the desired angle with the given rotation center 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 periodic topology is selected, all of them would be disabled as it does not 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 center, axis and angle of rotation in the respective field and click on **apply** to rotate. The center 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 center 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 center of rotation is evaluated from the workplane center and the axis of

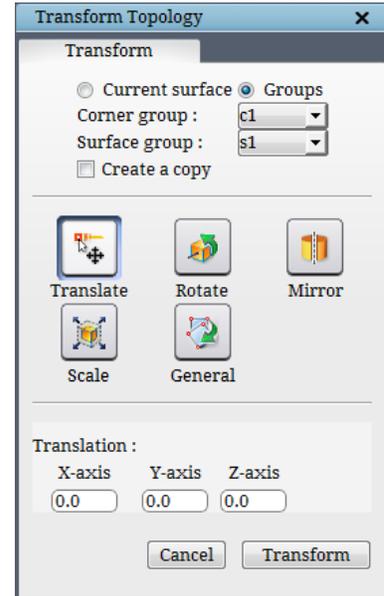


Figure 4.82: Transform dialog box

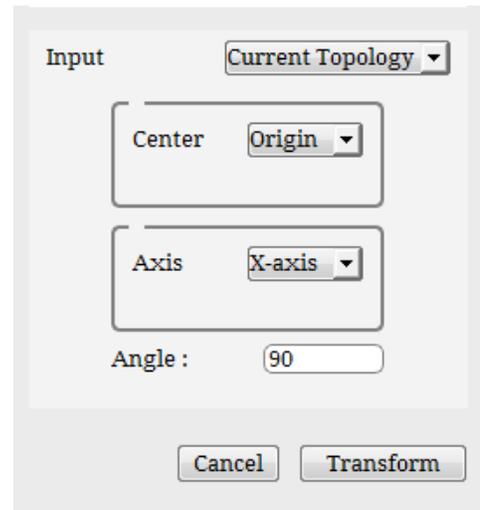


Figure 4.83: Rotate dialog box

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 like 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 even. 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 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. 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 as shown in the Figure 4.84.

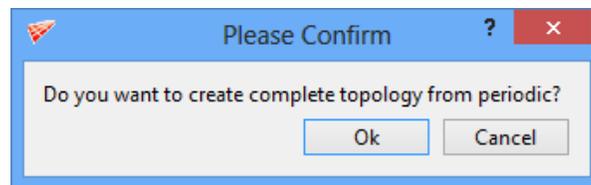


Figure 4.84: Confirmation dialog box

### 4.6.3 Mirror



Mirror tool **Mirror** is helpful when you work on a symmetrical geometry. It can mirror and/or copy & 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 of symmetry is required. Figure 4.85 shows the *Mirror Topology* dialog box. Similar to rotate, mirror is also having some predefined options apart from the user defined. 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 center field which will calculate the plane of symmetry while the normal of symmetry needs to be given in the normal

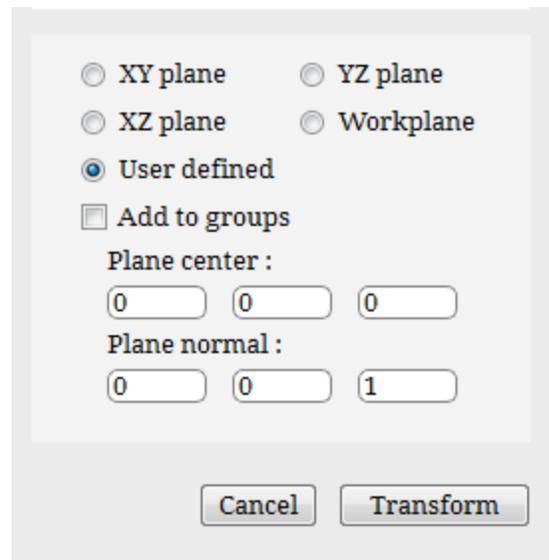


Figure 4.85: Mirror dialog box

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 are assumed implicitly using the options itself, i.e. XY plane defines that the normal is on XY plane and the plane center for these planes would be the origin. Checking the ‘**Create a copy**’ checkbox, will mirror and copy the topology; otherwise, the original topology will be mirrored. **Add to groups** option adds the corresponding mirrored corners to its respective group. For e.g. the corresponding mirrored corners of corner group c2 will get added to c2 if the checkbox is enabled.

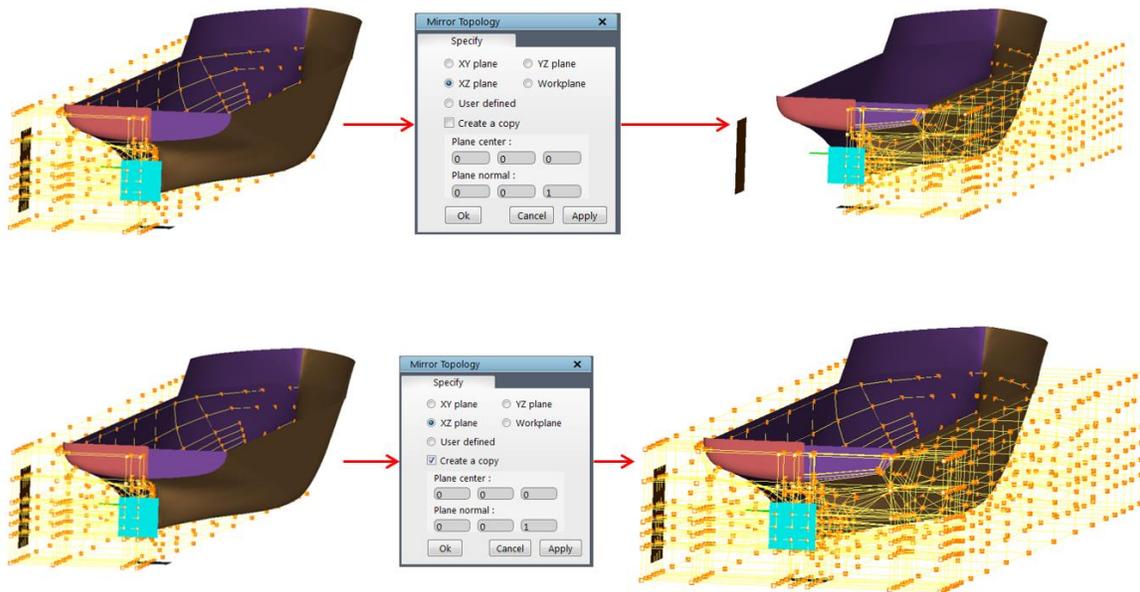


Figure 4.86: Mirror and Copy & Mirror corner and surface groups

#### 4.6.4 Scale



Scaling **Scale** can be done on any active corner and/or surface groups by giving the scaling center and the ratio of scaling in all three directions. Figure 4.87 shows the dialog box of the **Scale** button. You need to provide the **center** of scaling in the center field and the **ratio** of scaling in the ratio field. Each axis is provided with a separate field. Except workplane, there are no pre-defined options like Rotate and Mirror. Clicking on ‘**Get workplane Center**’ will automatically find the center of workplane position and input it in the Center field.

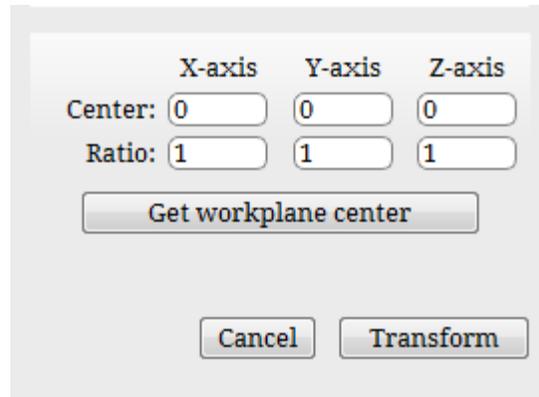


Figure 4.87: Scale dialog box

### 4.6.5 General Transform

All the above-discussed transformations are put together in this transformation tool. Figure 4.88 shows 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 *scale* where the **scaling ratio** is given. By default, the origin is used as scaling center and the corresponding ratio is used for the 3 directions. Then *rotation*, which can be defined similar to the rotation button, by using rotation center, rotation axis and the angle of rotation. **Translation 2** field 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 center other than origin. You can translate the group to the origin using translation 1 and apply scaling and then revert the topology back to the same position using the translation 2. All these transformation will be executed one after the other in the order as listed in the dialog box.

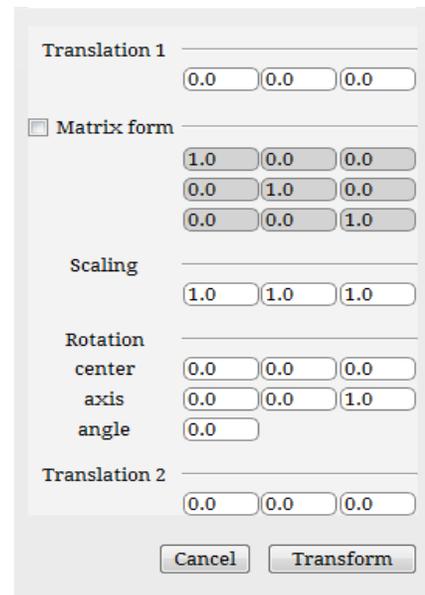


Figure 4.88: General Transform dialog box

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.

## 4.7 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 enhances 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

A 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. 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 4.89 illustrates wrappable one-dimensional objects, and Figure 4.90 illustrates unwrappable one-dimensional objects.

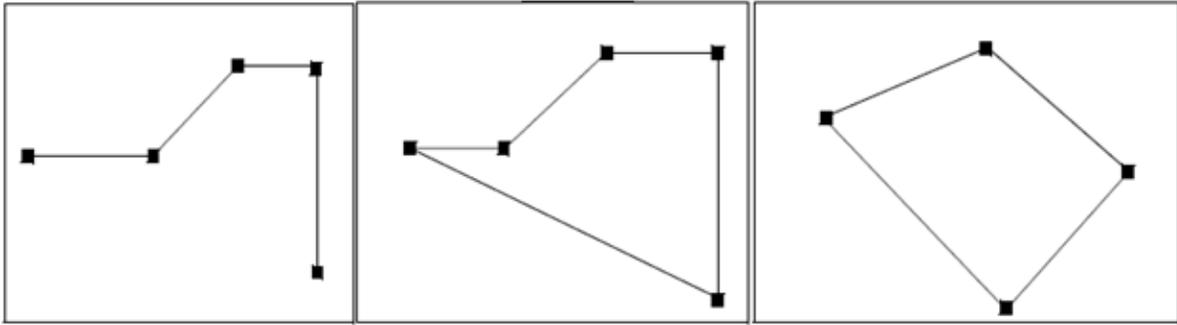


Figure 4.89: Wrappable 1D wireframes

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 must be inside a group. If these conditions are met, you have a wrap-able group.

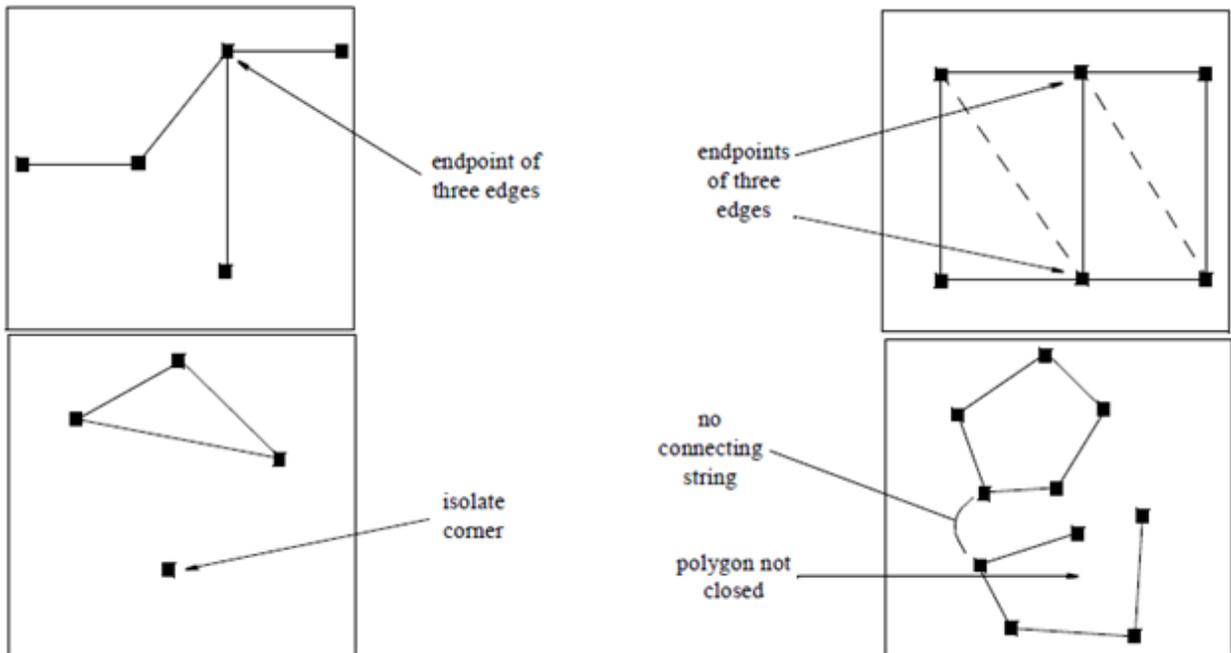


Figure 4.90: Unwrappable 1D wireframes

In order to wrap a group, switch on a corner group (Refer *section 4.4*), and then press the **Wrap** button located in **Edit (Topology)** section. The button opens a dialog box shown in Figure 4.91. By selecting one of the values in the dialog box, you can dilate or shrink your wrap by the percentage shown. In status bar, you will be informed if a group is wrap-able or unwrappable.

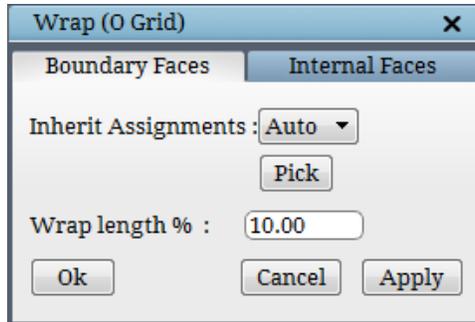


Figure 4.91: External wrap dialog box

There are two kinds of wrap; one is external and the other is internal. **External wrap/Boundary Faces** is a wrap which extrudes from the existing topology either on positive normal side or on negative normal side. **Internal wrap/Internal Faces** is a wrap which is inserted in between the existing topology and topology connections are adjusted such that the topology remains valid. This is helpful when there is a need for local refinement without affecting the far-field boundaries.

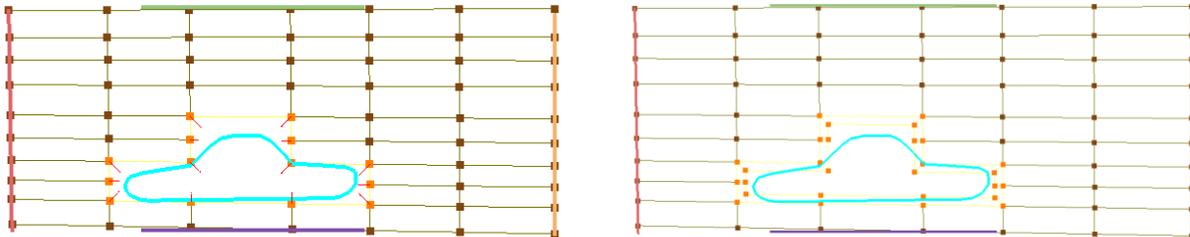


Figure 4.92: Before and after External wrap

The percentage of the wrap layer should be mentioned in the percentage field. By default, it takes 10%, this value is the measure of the average length of all the edges in the corner group used for wrap. It can also be adjusted using the **Pick** button provided before the input field. The **Pick** button primarily helps in picking the direction of the wrap as shown in Figure 4.92, either inwards or outwards. Depends on the length of the arrows, the wrap length input field will also get filled automatically, which can still be edited manually if required. The Inherit assignments option determines how the surface assignments have to be carried forward. It consists of 3 options provided in a drop down. The **Auto** (Intelligent warp) option removes the surface assignments from the corner group used for wrap and assigns it to the wrapped corners. The **All** option retains the assignments with the corner group used for wrap and also carried forward to the wrapped corners. And the **None** option retains the assignment with the corner group used for wrap and will not be carried forward to the wrapped corners.

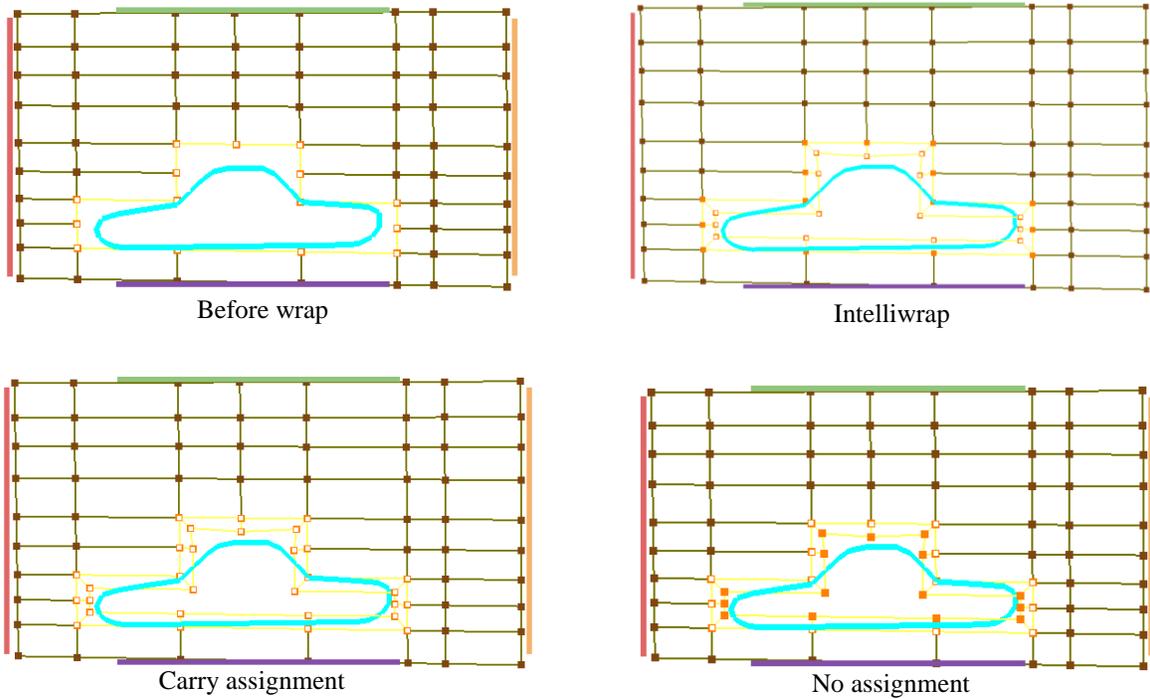


Figure 4.93: Different surface assignment propagation type

Switch to Internal Faces tab in Wrap dialog box. Figure 4.94 shows the dialog box of the internal wrap. Since it is a wrap which is inserted inside an existing topology, it takes care of the assignments and hence outputs a valid topology. Enabling **Add insert** will add an insert in the middle of corners used for wrap and the wrapped corners.

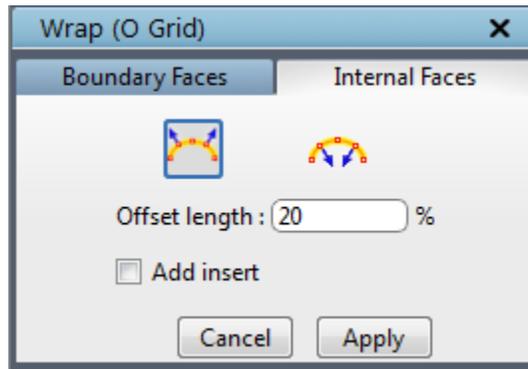


Figure 4.94: Internal wrap dialog box

Internal wrap will work similar to All (Intelligent Wrap) option under wrap if the input corner groups have assignments. It 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.

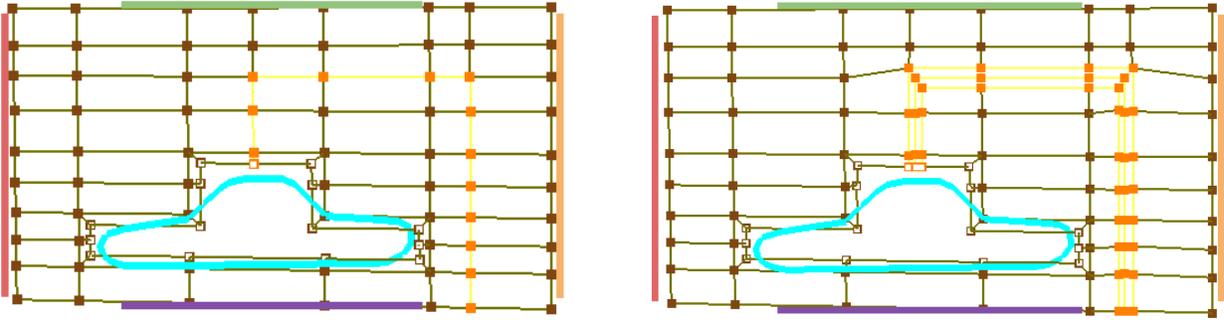


Figure 4.95: Before and after Internal wrap

- **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 read correctly 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.

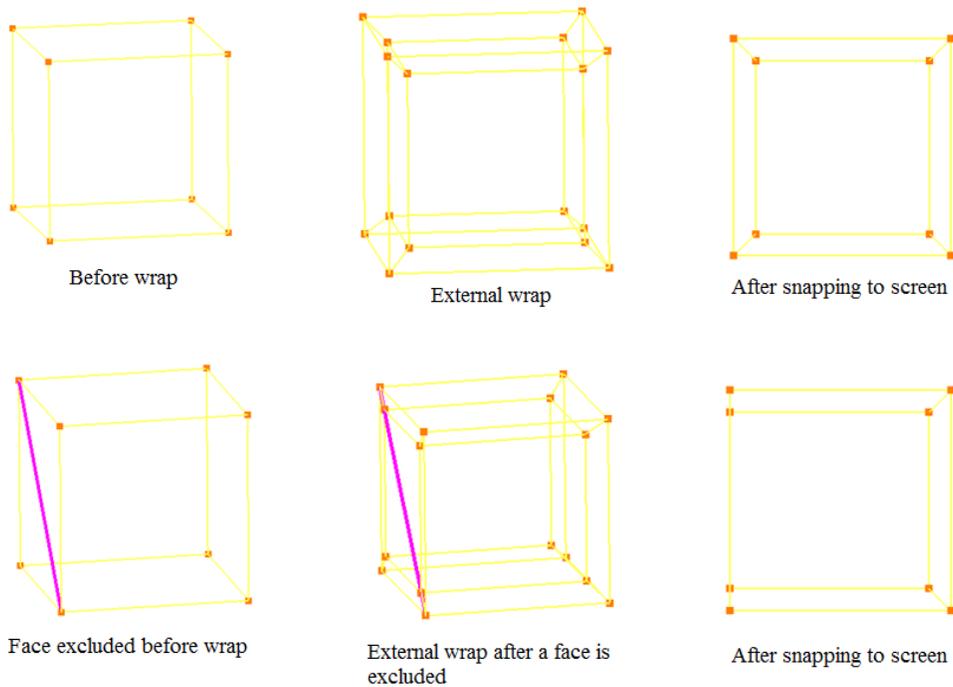


Figure 4.96: Wrap on face excluded wireframe

Second, you must exclude faces in order to exclude a portion of the wrap. Figure 4.96 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 4.97.

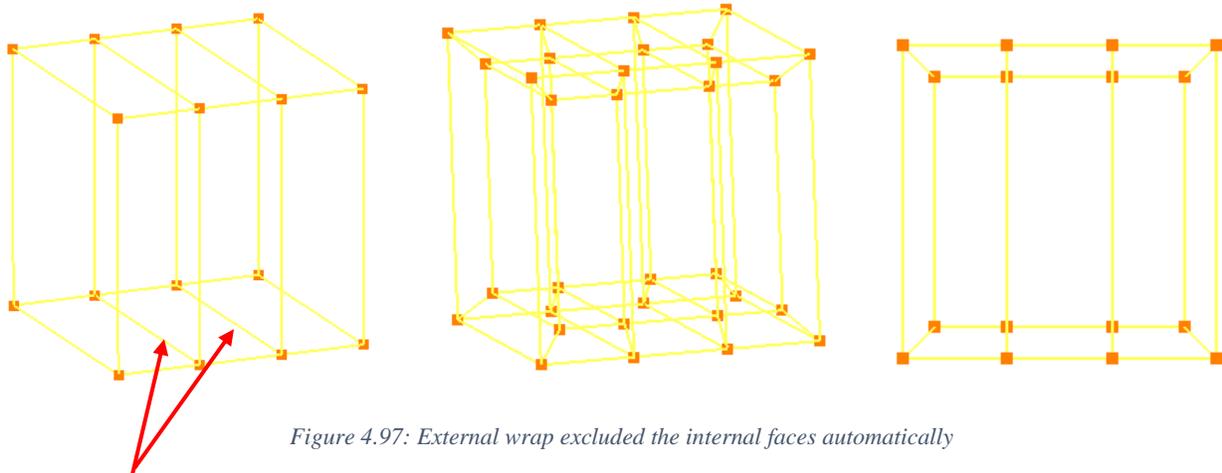


Figure 4.97: External wrap excluded the internal faces automatically

**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, before the grid generation process is started, all the face exclusion (pink line) in the topology should be 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 problem, you can hide the exclusion markers by unchecking exclusion in topology tree of the toggle display section which is explained in *section 2.6*.

## 4.7.2 Heal

- Singularity

In 3-d, a regular grid point in a structured grid has 6 neighboring 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 on a *singular edge*.

For 3-d 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

neighboring blocks. For internal edges (i.e. edges not on surfaces) in 3-d, a regular one has 4 neighboring blocks. An internal edge having three neighboring blocks is called a singularity of degree 3. Similarly, an internal edge having five neighboring 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 his 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 2-d case, Figure 4.98 is a local view showing how a singularity of degree 6 can be split into two singularities of degree 5.

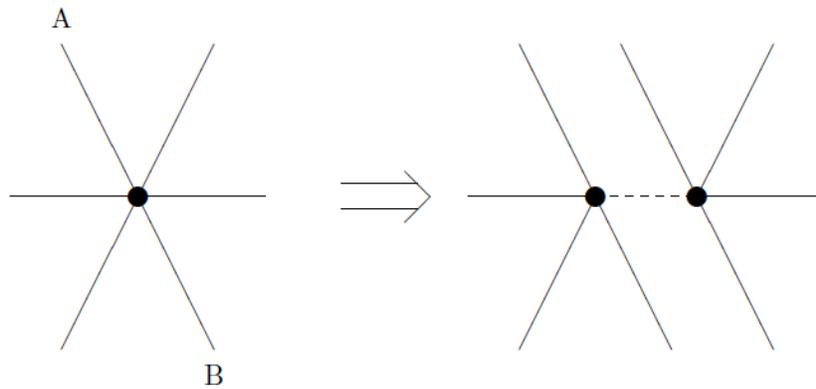


Figure 4.98: 6 degree singularity split into two 5 degree

Singularities of degree 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 in 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 (3-d): an edge on a single surface having two neighboring blocks; and, an edge on the intersection of two surfaces having only one neighboring block. Any other surface edge is a singular surface edge and should be avoided in the design. For 2-d cases, things said about edges can be said about corners. Thus, the topology design in Figure 4.99 has 4 singularities on the surfaces. They are the 4 corners of the outer box. A better design should avoid them and it is shown in Figure 4.100. In some situations, insisting on using regular corners or edges on surfaces may result in bad topology. However, one

does not have to use singularities on surfaces to solve the problem. An alternative is to redo the surface division.

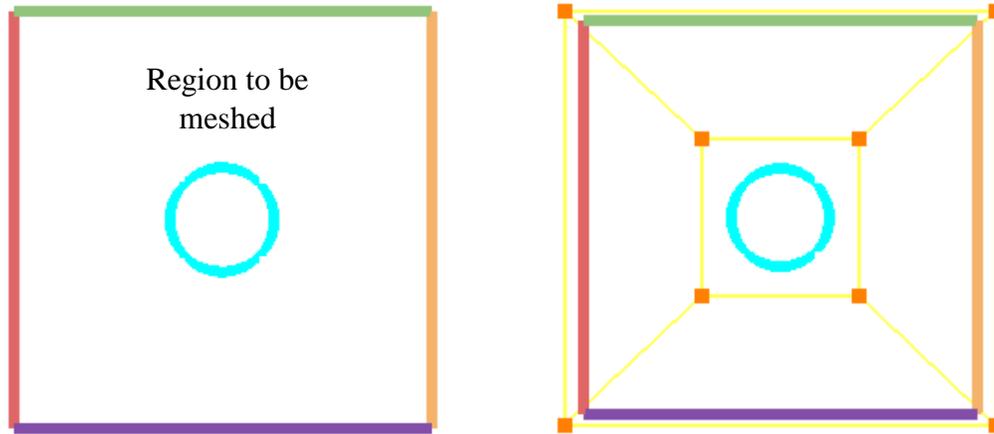


Figure 4.99: Topology with singularities on surface

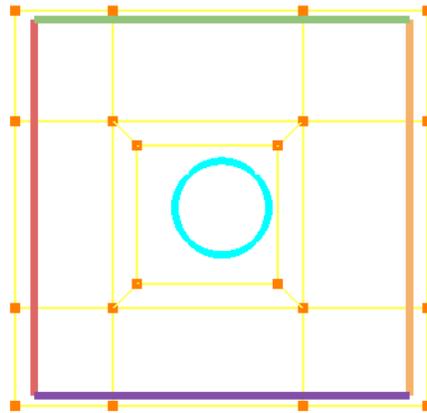


Figure 4.100: Modified topology without singularity

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 neighboring 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 emerge out a fixed corner in case of external surface intersections. If there are 3 or more edges emerge out of a fixed corner, then it is called mildly severe corner and the singularity is mildly severe singularity. In 3D, if there are 3 or more faces emerge out of an edge, then it is called *mildly severe edge*.

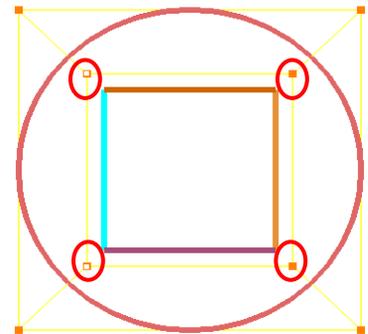


Figure 4.101: Mildly severe topology

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 edge emerging out of a corner, it would be a face emerging out of an edge.

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

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

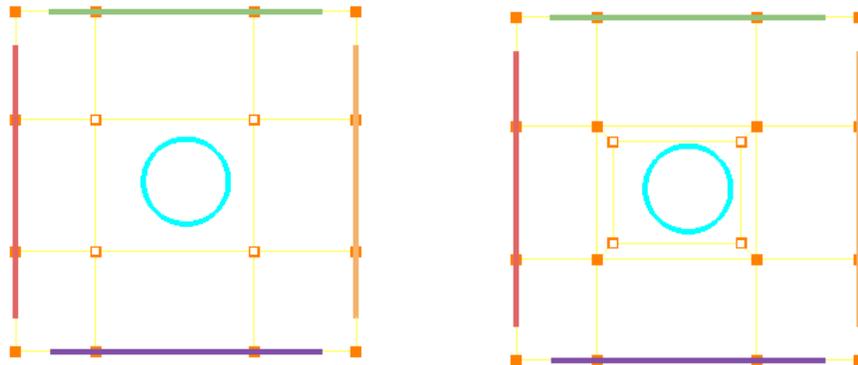


Figure 4.102: Mediumly severe topology and its solution

- **Very severe singularity:**

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

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

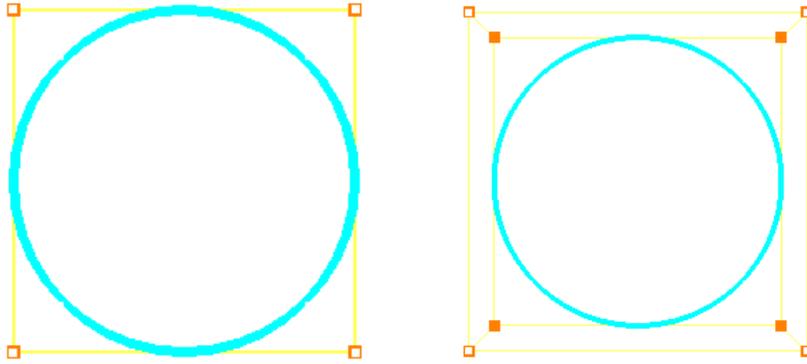


Figure 4.103: Very severe topology and its solution

All these singularities can be either solved manually or automatically using **Heal** button located under **Edit (Topology)** section. Figure 4.104 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.

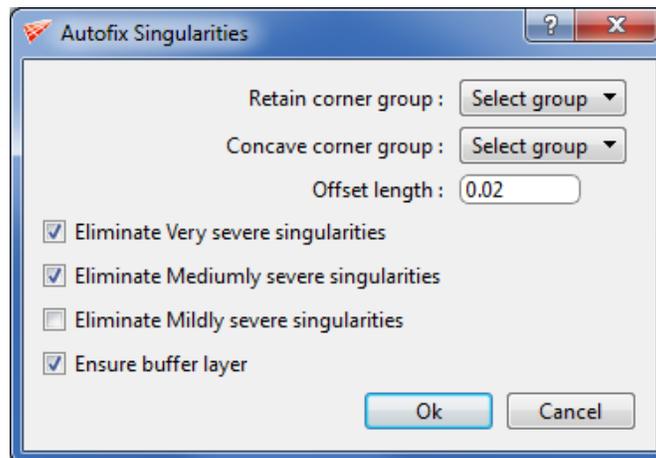


Figure 4.104: Heal dialog box

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 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 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 Projection 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 *Align*  button. When Align button is clicked, Align Wireframe dialog box opens as shown in Figure 4.105. There are five options divided into two tabs i.e. **Align To** and **Align On**. ‘Align To’ provides option to project the group of corners to the Workplane and Grid.

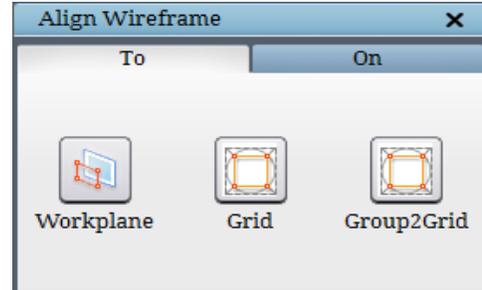


Figure 4.105: Align To dialog box

### 4.8.1 Align To Workplane

The *align to workplane* operation projects the active corner group to the workplane such that the center of mass of the active corner group and the workplane are relative. Figure 4.106 shows the projection of corners to workplane. Note that projection on workplane does not place the corners at workplane position.

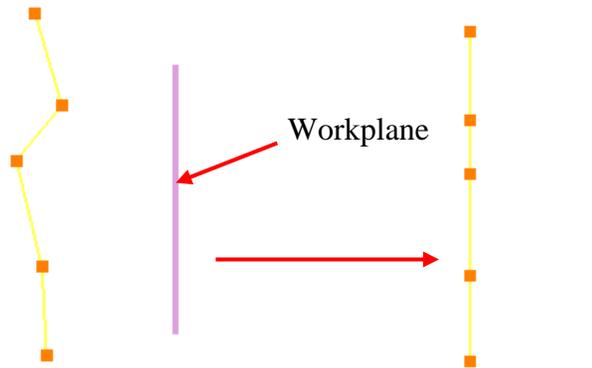


Figure 4.106: Before and after aligning to workplane

### 4.8.2 Align To Grid

*Align To Grid* is a kind of projection that synchronizes the position of the topology corners and the edges 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 4.107 shows the topology before and after sync.

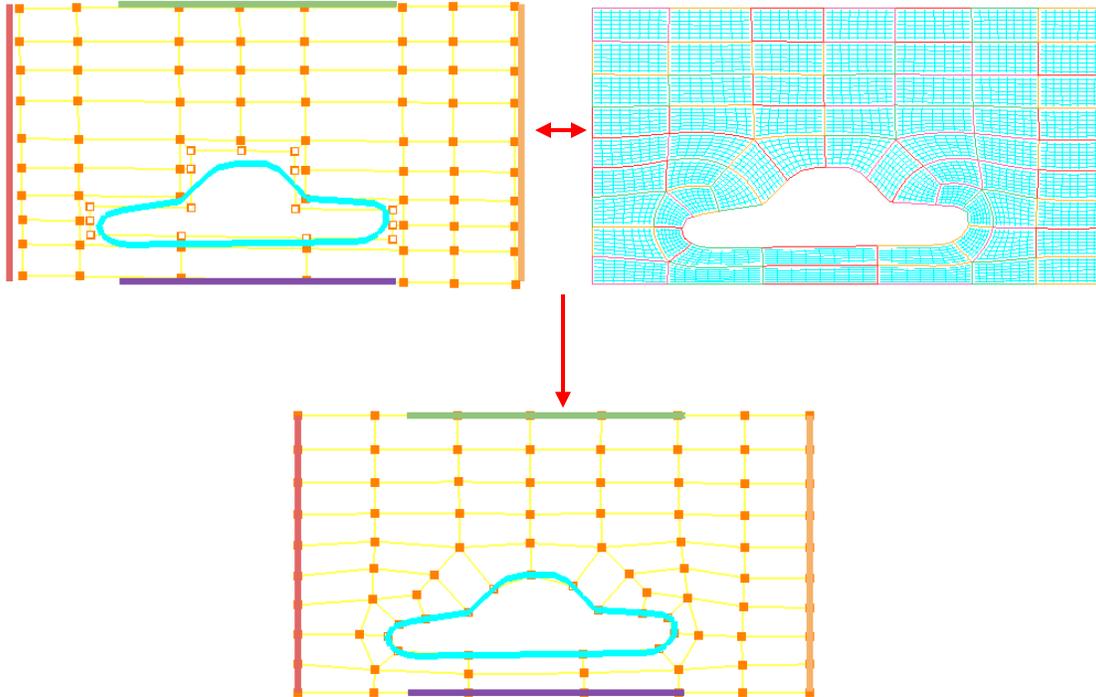


Figure 4.107: Before and after aligning to grid

If the grid blocking pattern is not matching with the topology, an error message pops out as shown in Figure 4.108.

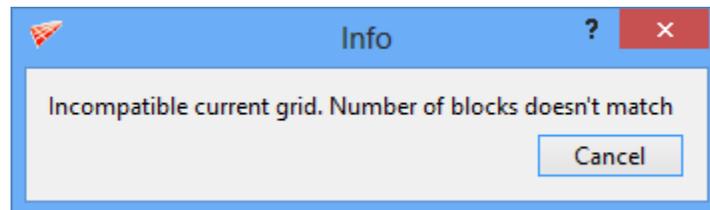


Figure 4.108: Error message

### 4.8.3 Align Group2Grid

Align Group2Grid is similar to Align To Grid operation. This operation is used along with Ggrid groups. You may have run Ggrid on part of a topology in order to position them better. Once you get the grid for the same, use Align → Group2Grid to achieve that. Upon clicking on the button, it shows a confirmation pop\_up as shown in Figure 4.109. 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.

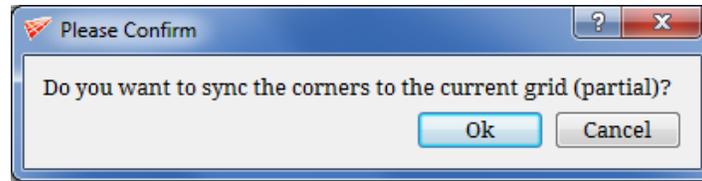


Figure 4.109: Confirmation message for Group2Grid

#### 4.8.4 Align On Surface

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. A corner group that is empty cannot be used for projection, which will pop up an error message. Note that the corners can only be projected to the *current* surface.

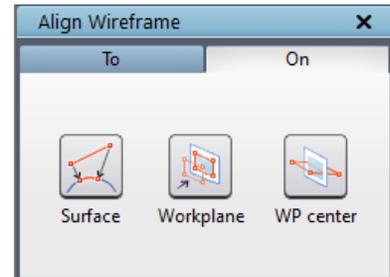


Figure 4.110: Align On dialog box

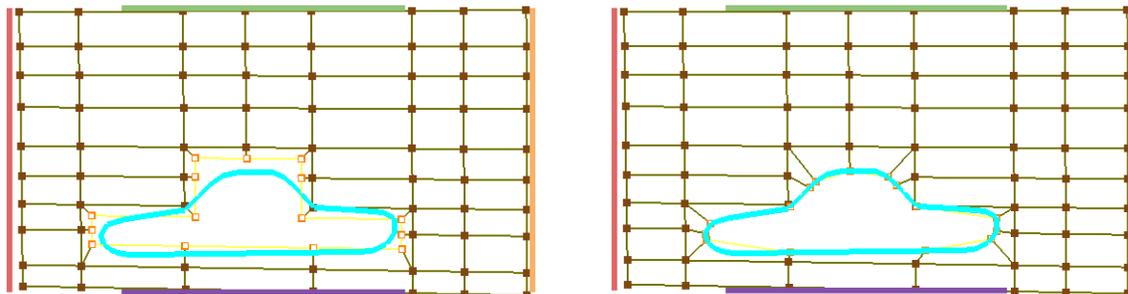


Figure 4.111: Before and after aligning on current surface

When project on surface is clicked, it pops out a dialog box asking for number of smoothing as shown in Figure 4.112. A smoothing algorithm will run on the projected corners based on number of smoothings given.

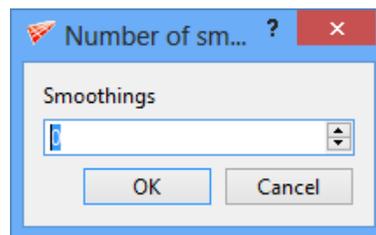


Figure 4.112: Number of smoothing dialog box

#### 4.8.5 Align On Workplane

The **Align on Workplane** 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.

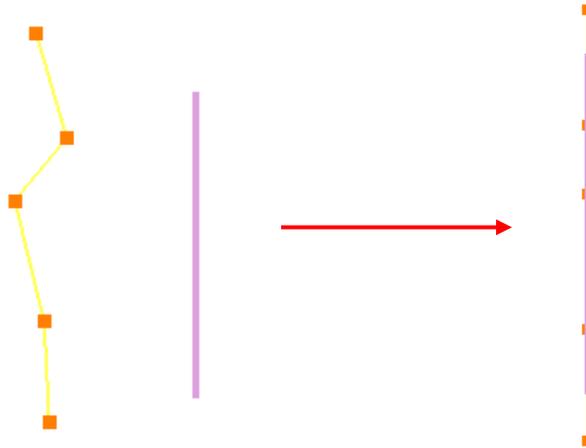


Figure 4.113: Before and after aligning on workplane

#### 4.8.6 Align On Workplane Center

The **Align on Workplane** center operation translates the center of mass of active corner group to the center of the workplane.

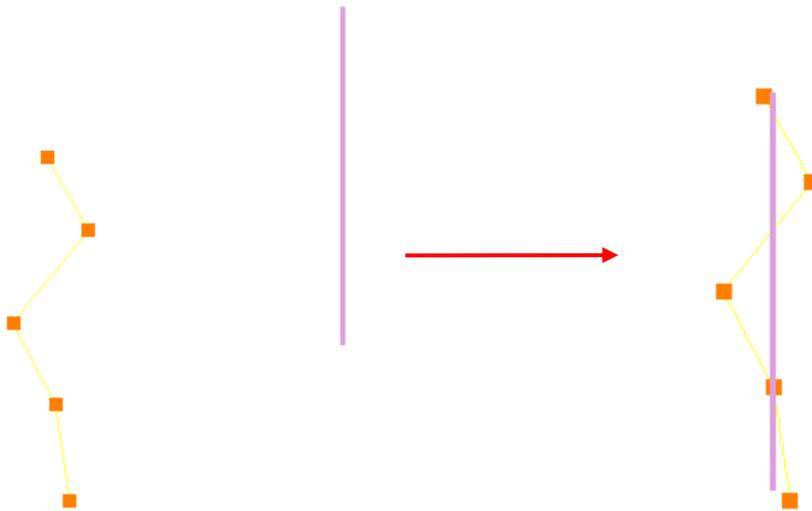


Figure 4.114: Before and after aligning with workplane center

## 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; a grid can be used for analysis only if the gridded region is sufficiently 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. Including both endpoints of an edge, the number of mesh lines present is, one more than the number of cells inputted. You must set the density of the grid in the **Topology** tab; you will not see the results of any density alterations

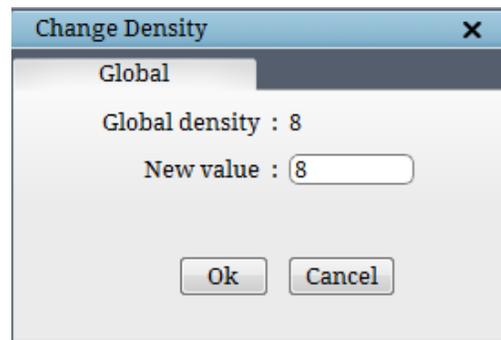


Figure 4.115: Density dialog box

until you actually see the grid (in the **Grid** tab). By accessing the **density**  button under **Edit** section in **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. Clicking on an edge automatically highlights all the parallel edges whose densities will be changed.

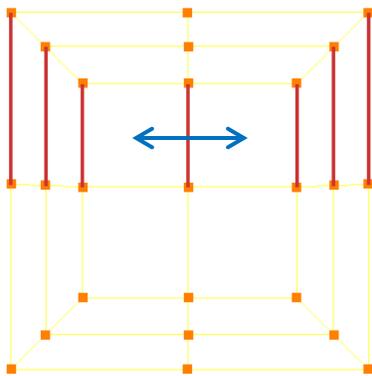


Figure 4.116: Parallel edges highlighted

The **Global density** field displays the current density of all edges and the **New value** field displays the new density of all the edges. By default both the tab shows 8 always that will change once it is modified.

If you wish to alter the density of a set of topologically parallel edges, first click on **density** button, then click on one of the parallel edges. All parallel edges will then highlighted with red lines. Once the density button is clicked and the edge whose density has to be changed is selected, Change Edge Density dialog box opens as shown in Figure 4.117 which

contains three fields: the *edge*, *current density* and *new density* fields. The edge field reads the selected edge id and the id's of the corners to which the edge is connected with. The current density field displays the current density of the parallel edges. The default setting is 8 per set of parallel edges. The *edge* and *current density* fields contain fixed information; you cannot type new information in these boxes. The *new density* field illustrates the new density of the edges you wish to alter. Therefore, you must type the desired density value of the edges in this field. The dialog box has 3 confirmation buttons such as apply, ok and cancel. Clicking on *apply* applies the change in density and retains the dialog box so that you can assign density for other edges without clicking on *density* button again. Clicking on *ok* will apply the change in the density and close the dialog box. Clicking on *cancel* will close the dialog box without any modification.

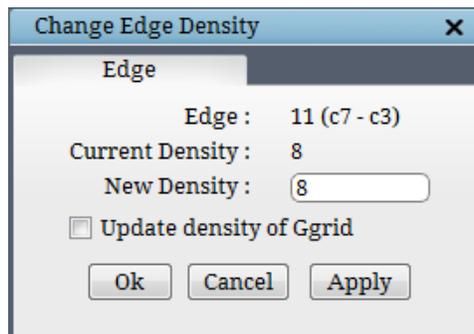


Figure 4.117: Edge density dialog box

The greater the density of the topology, the longer the Ggrid process will take to complete. For two-dimensional grids, every time you double the density of the topology edges, you increase the Ggrid processing time by a factor of 4. For three-dimensional grids, every time double the density of the topology edges, you increase the Grid processing time by a factor of 8.

If you have initiated the Ggrid process, but notice that you wish to change the density of your grid, you can do so without interrupting Ggrid. As Ggrid is running, change the density to the desired value using the *Density* button, then enable the checkbox in the pop\_up before clicking 'Apply' to update the modification in the gridding process also. Ggrid will automatically update the grid and the modification can be viewed when the grid is written next time.

## 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 number of grid points

near the geometry of concentration locally, with one to one connectivity, without increasing the points elsewhere. The nested structure loops back blocks near the geometry and in the defined region.

This nest utility is further divided into 4 different tools having a dedicated tab, each works

in a different way. The *Nest* button  is located in the *Edit* section of the right side panel in **Topology** tab. Clicking on the button opens a dialog box as shown in Figure 4.118. The first three deals with the nesting as explained above and the reverse nest works opposite to the nesting, which is helpful in coarsening the grid as it moves towards the far field.

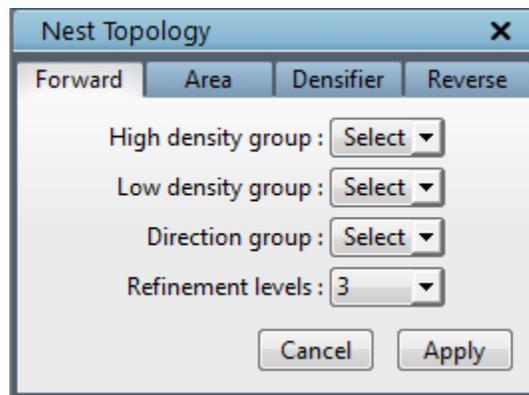


Figure 4.118: Nest dialog box

- **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, mostly it's along the wrap direction and finally the **number of refinements** which decides the number of layers of nesting. Figure 4.119 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.

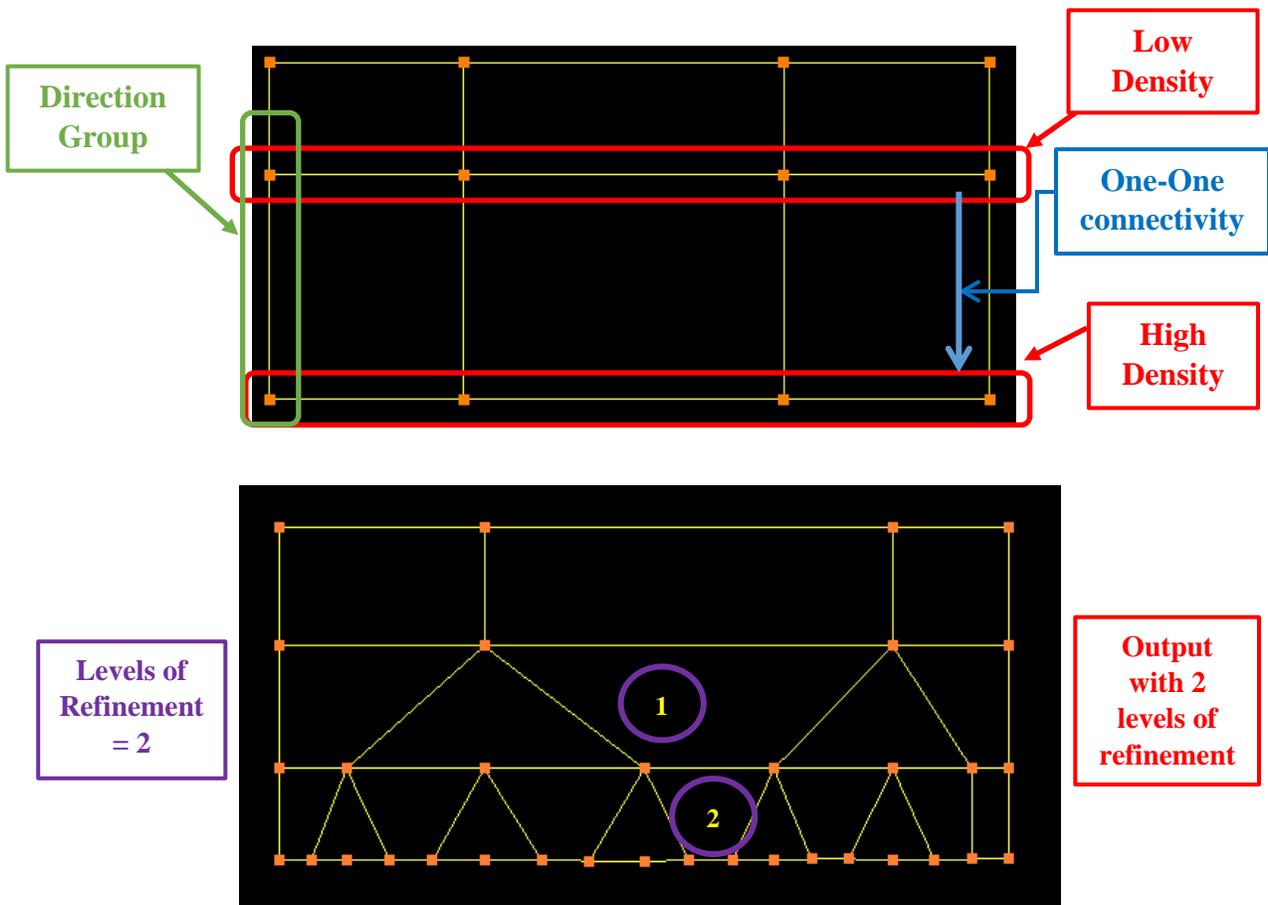
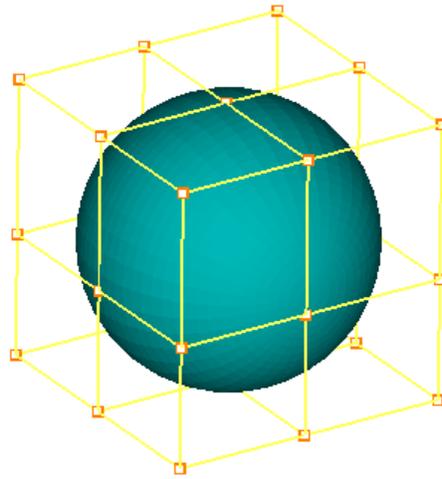


Figure 4.119: Nesting structure – Input & Output

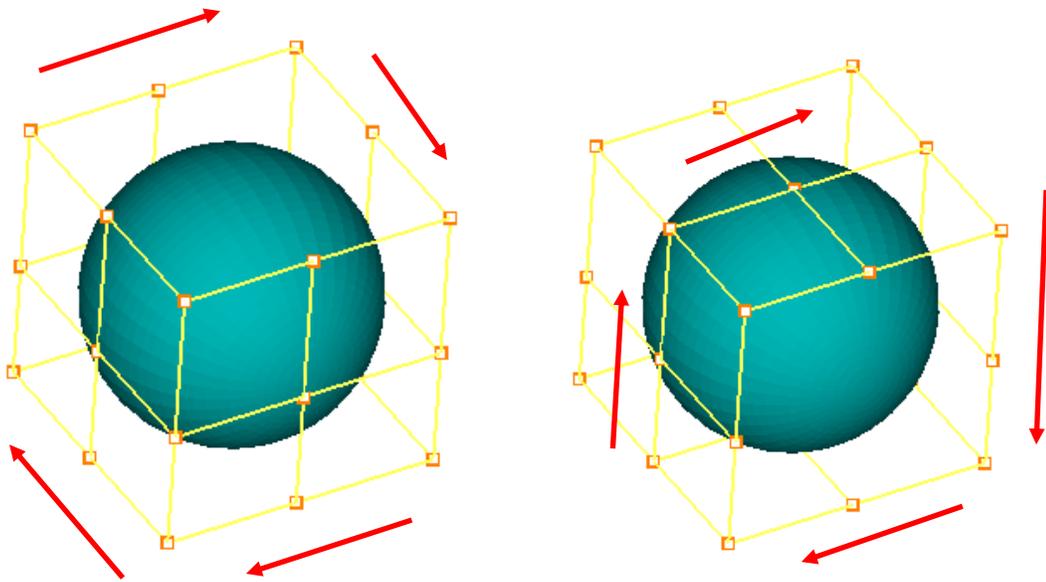
There are few requirements needs to be taken care in order to do the nesting efficiently.

They are,

1. There should be one-to-one connectivity between Low Density Group and High Density Group as shown in Figure 4.119.
2. Nesting cannot be applied for 3-D as it may encounter a singular point. As shown in Figure 4.120, it cannot be applied along 3D sheet.
3. Nesting can be performed along 2-D sheet only and it is applied in 1D as seen in figure Figure 4.121.
4. Topology structure of the high and low density group should be same at every cross-sectional sheet as explained in Figure 4.122.



*Figure 4.120: Topology where nesting cannot be applied*



*Figure 4.121: Topologies where nesting can be applied*

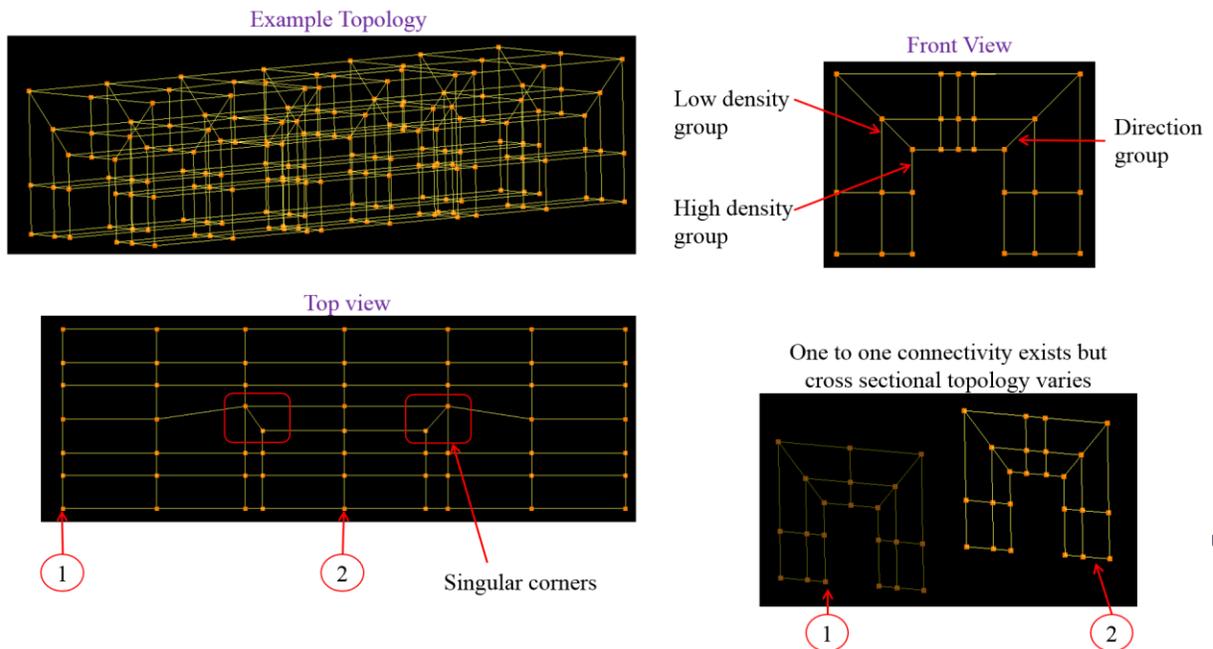


Figure 4.122: Topology structure of high and low density group

- **Area Nest**

Area nest is similar to the forward nest tool where nest is applied along one direction but in the area nest, it is applied on 2 directions simultaneously. Figure 4.123 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 also except for the last one since area nest is applied in 2 directions. Figure 4.124 shows an example of how area nest is being done.

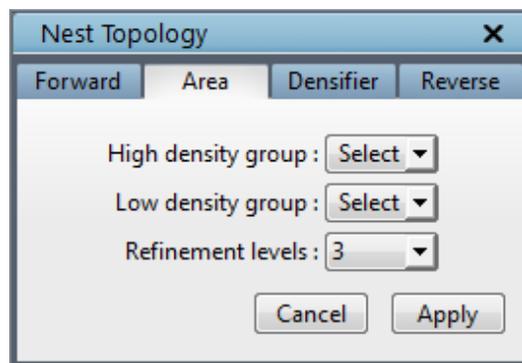


Figure 4.123: Area nest dialog box

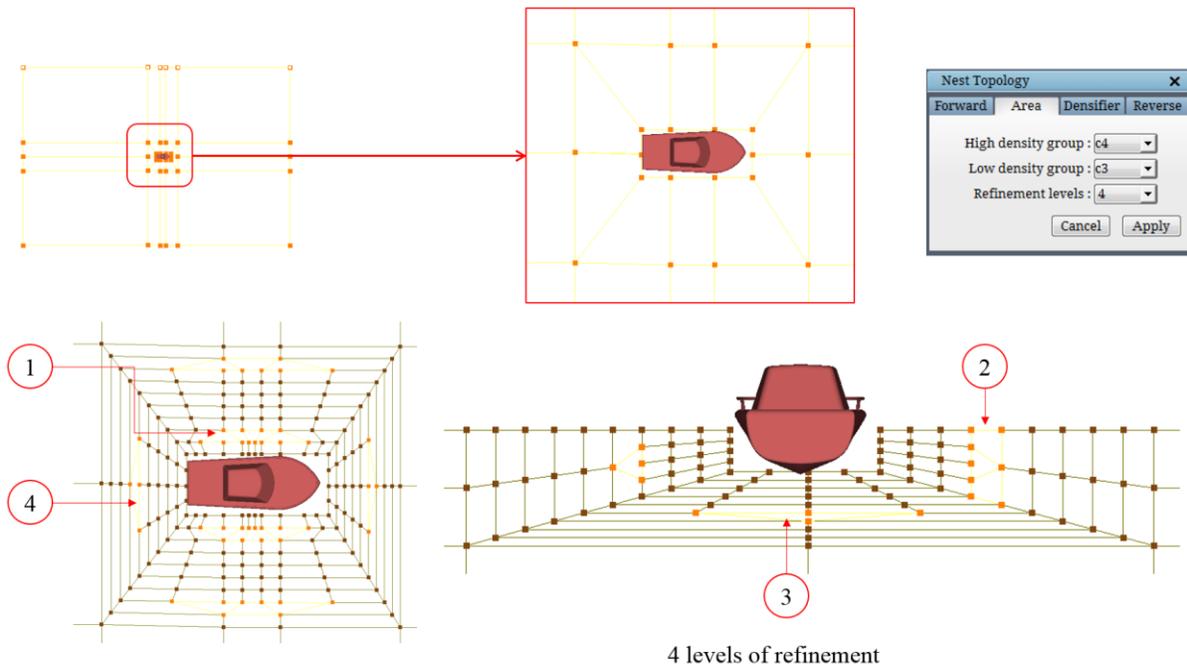


Figure 4.124: Area nest topology

- **Topology Densifier**

Topology Densifier is a special case of *Area nest*. Once clicked, it opens a dialog box as shown in Figure 4.125. The number of refinements and the percentage of wrap needs to be provided in their respective fields. Once the input is provided, it takes the assignments of current surface (Make sure the current surface is the surface where the nesting has to be applied), wraps the corners externally or internally. The wrap type is determined from the surface orientation. i.e. if it is a 1 sided surface, an external wrap is done and if it is 2 sided surface, an internal wrap is done. Then an insert is made in between the wrap and the area nest is applied as 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 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.

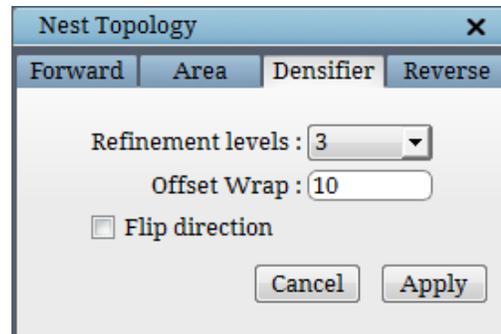


Figure 4.125: Densifier dialog box

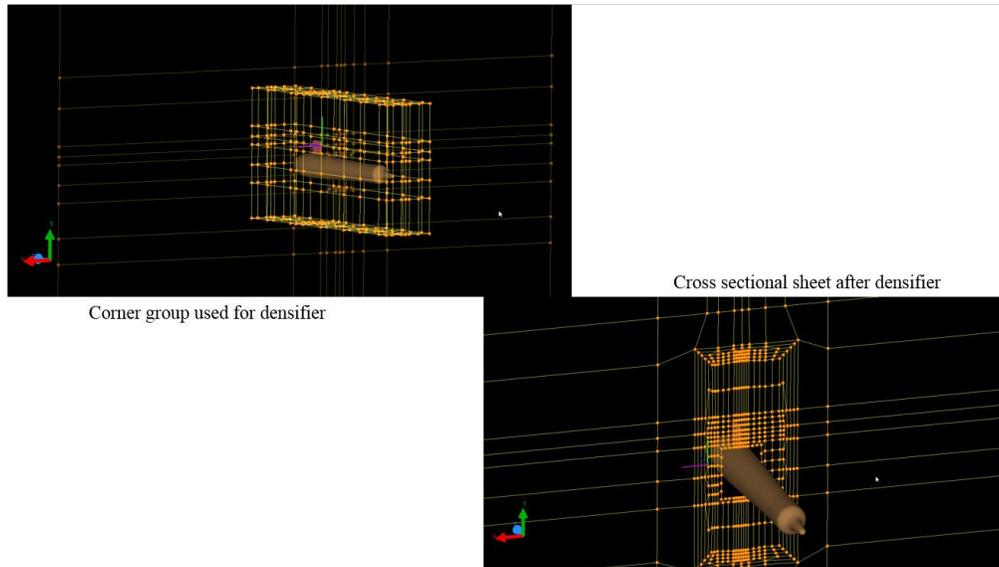


Figure 4.126: Topology Densifier

- **Reverse nest**

The *forward nest* creates a high density topology from a low density topology whereas *reverse nest* works in the opposite 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 4.127 shows the dialog box of the reverse nest.

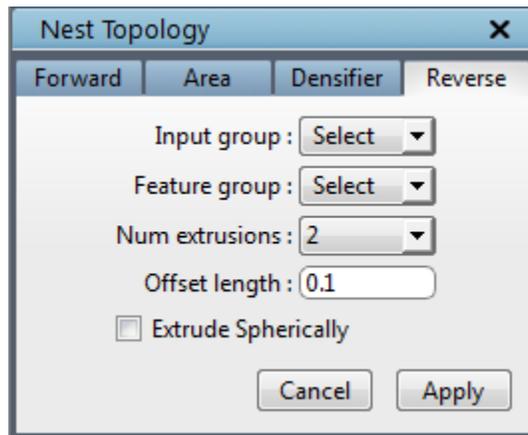


Figure 4.127: Reverse nest dialog box

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's 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 4.119 shows how the forward nest works, whereas reverse nest works in the opposite direction and in 2 directions similar to area nest.

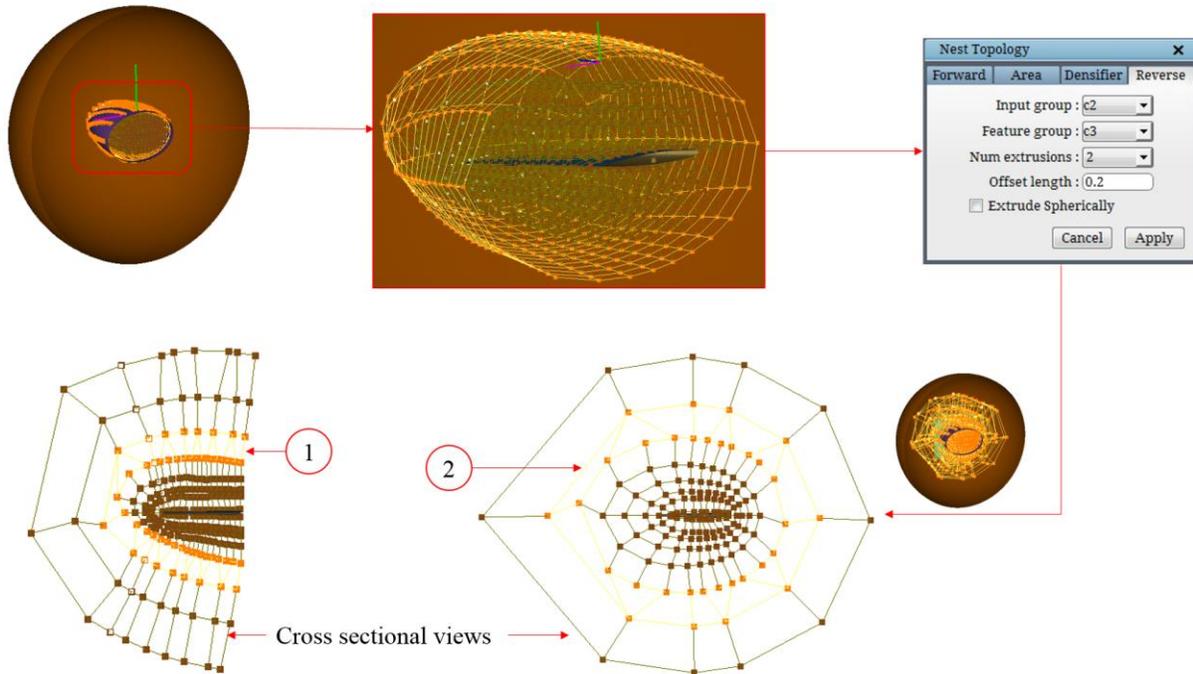


Figure 4.128: Reverse Nest topology

### 4.9.3 Compact enrichment

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 internal wrap. Figure 4.129 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.

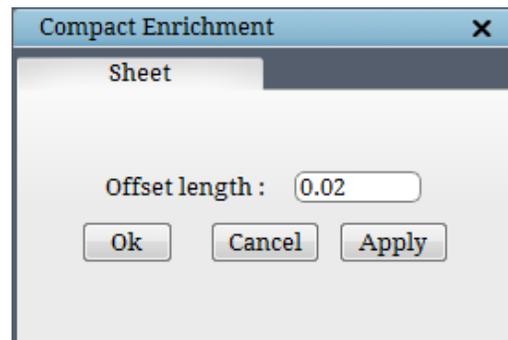


Figure 4.129: Enrich dialog box

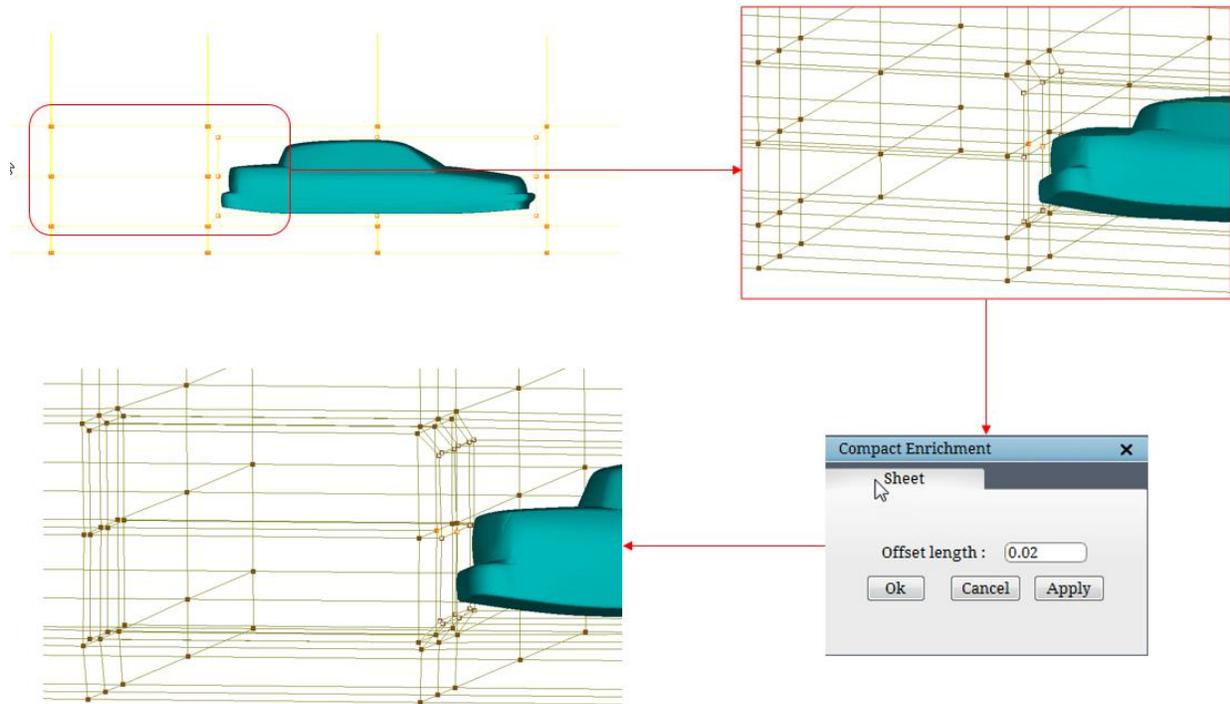


Figure 4.130: Enrich Topology

## 4.10 Surface Assignment Utilities

Unlike other grid generation programs, *GridPro* automatically generates a multiblock 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 are 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.

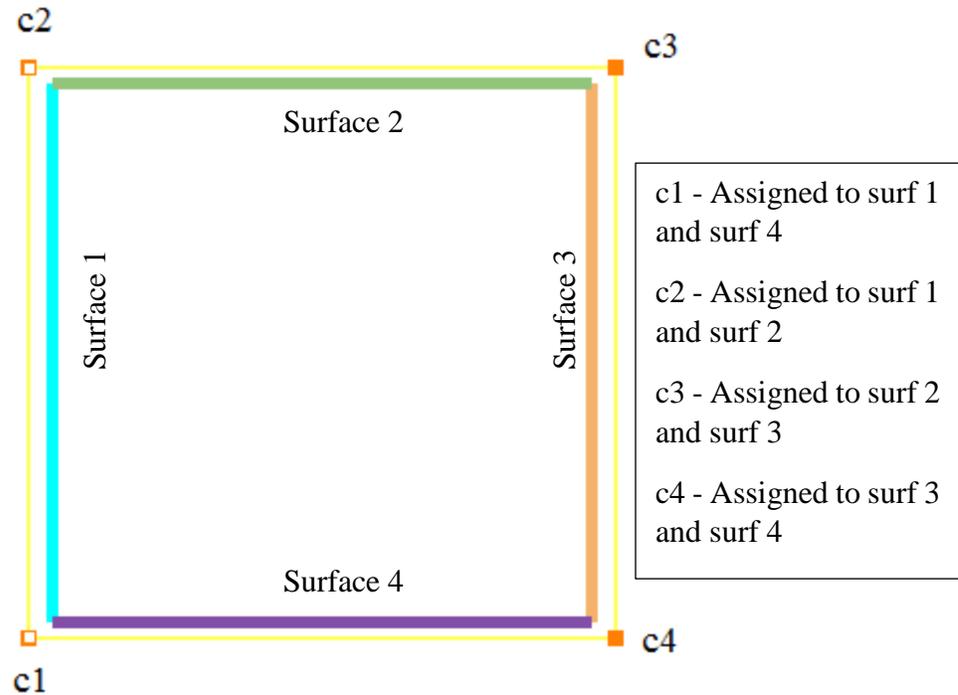


Figure 4.131: Rectangle topology with double assignment

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 the *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 4.132 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*.

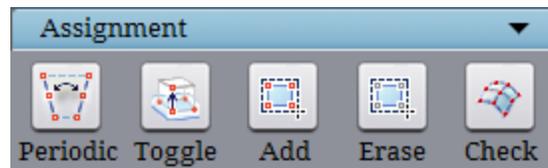


Figure 4.132: Assignment section

### 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 are 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 center. 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 are 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. Similar to **Toggle** button, **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. A dialog box as shown in Figure 4.133 once the **Erase** button is clicked. **Screen Select** lets you un-assign multiple corners that are selected by dragging the mouse. While **Clear** removes all assignments for *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 mouse right-click 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 are 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.

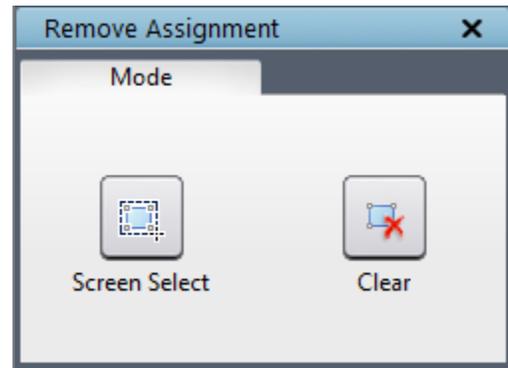


Figure 4.133: Erase dialog box

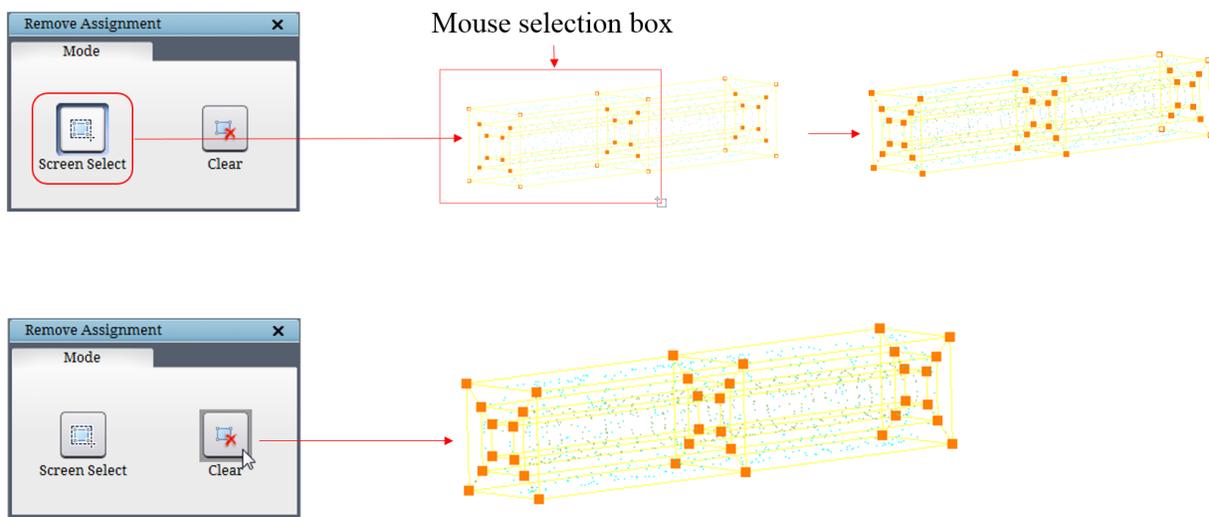


Figure 4.134: Various Erase modes output

- **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 are 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.

## 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 4.135. Both the surface creation and assignments are done at the same time for 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 drop down. It is preferable to use any one of these axes. Hence rotate

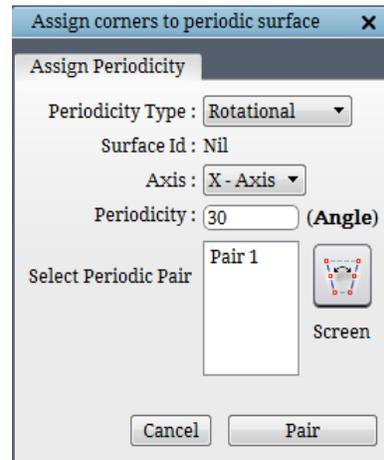


Figure 4.135: Period dialog box

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. Depends 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 under periodic pair column. Clicking on the pair displays the faces in red colour as shown in Figure 4.136. Then click on pair to create a periodic surface and assign the selected faces to it. This automated pair detection eases the assigning the corners to periodic surface operation a lot. In some cases, if it's not able to detect the pairs then there might be some topological errors.

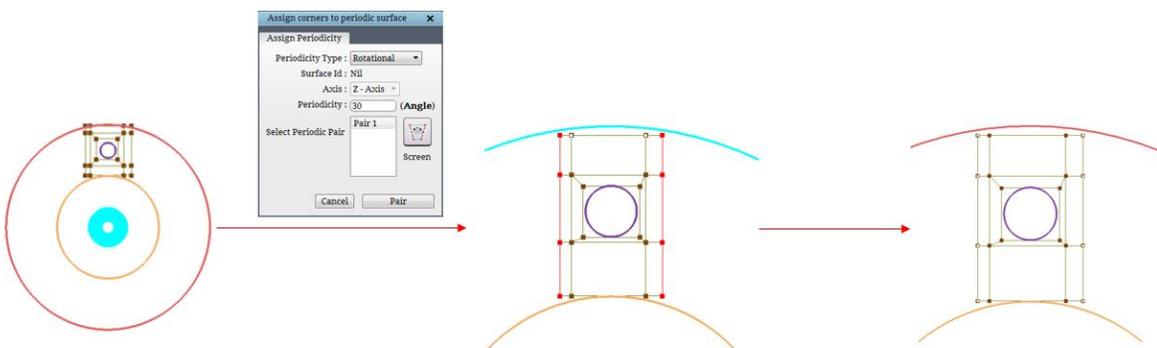
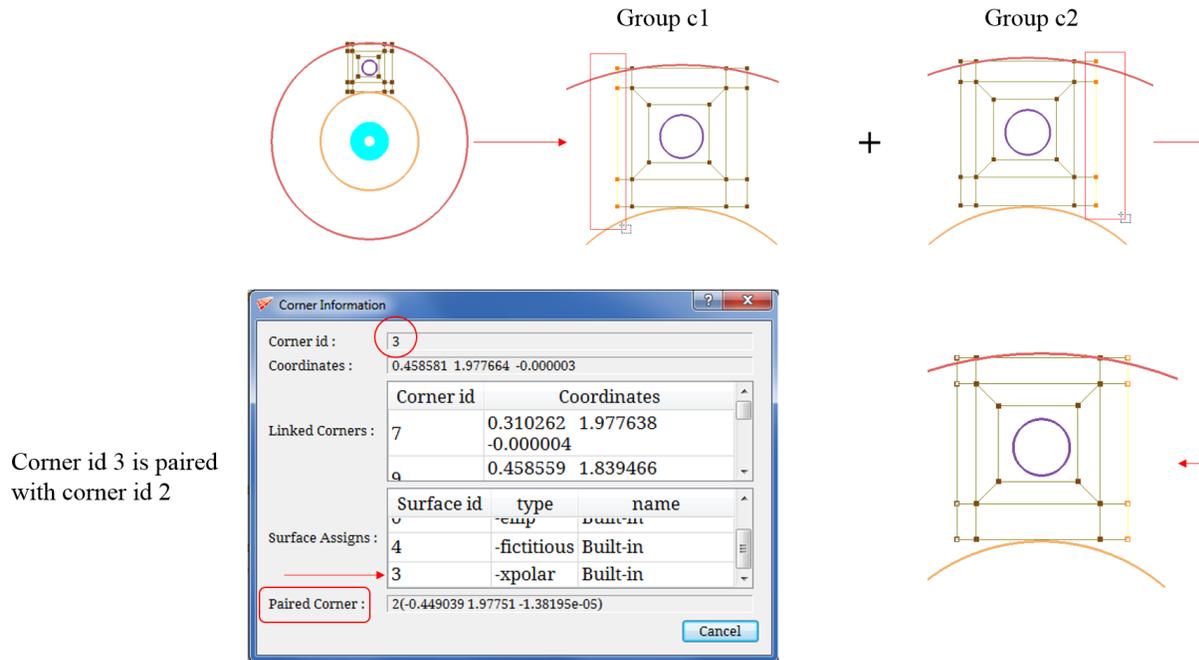


Figure 4.136: Automated periodic pairs detection

You are also provided with the manual assigning button through which the periodic pairs can be assigned manually as explained below. Before start assigning the corners, you have to group the corners on the periodic boundary in two different group or in 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 4.137: 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 and select the first corner group corners and then switch on the other group and select all the corners in that group. 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 *current* surface. It displays only the corner assigned to the *current* surface. All the corners except the corners assigned to current surface are hidden. Assigning corners to a surface is a process where you are very prone to make mistakes by assigning corners that are supposed to not. 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.

## 4.11 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 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** is selected from the drop-down list, a dialog box opens as shown in Figure 4.138. 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.

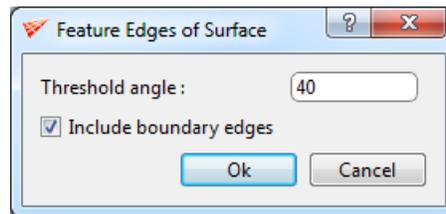


Figure 4.138: Feature edges dialog box

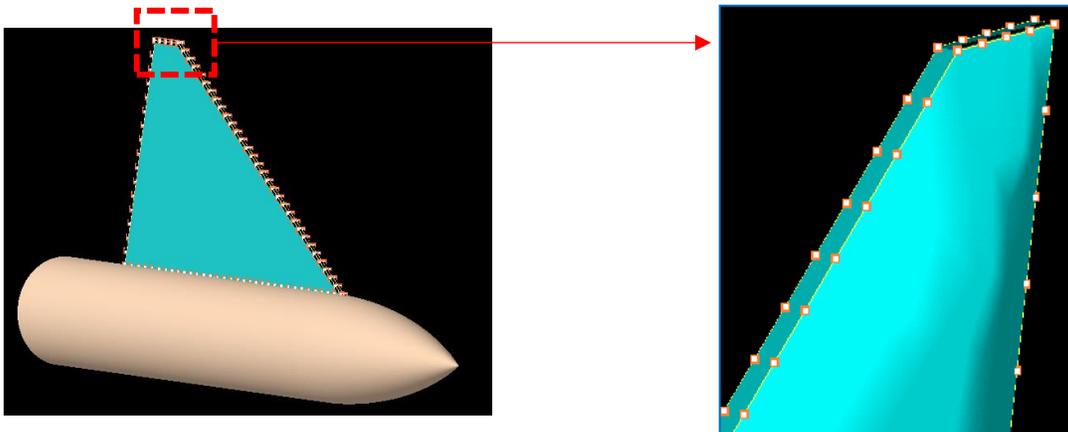


Figure 4.139: Feature edges created on a blunt edge wing

## 4.11.2 Intersection of Surfaces

*Intersection of surfaces* option under *Surface tools* drop down 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.

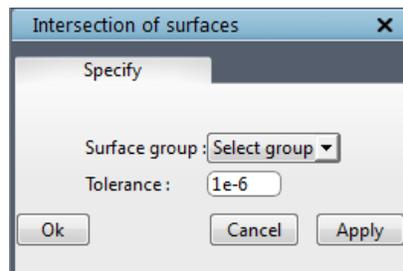


Figure 4.140: Intersection of surfaces dialog box

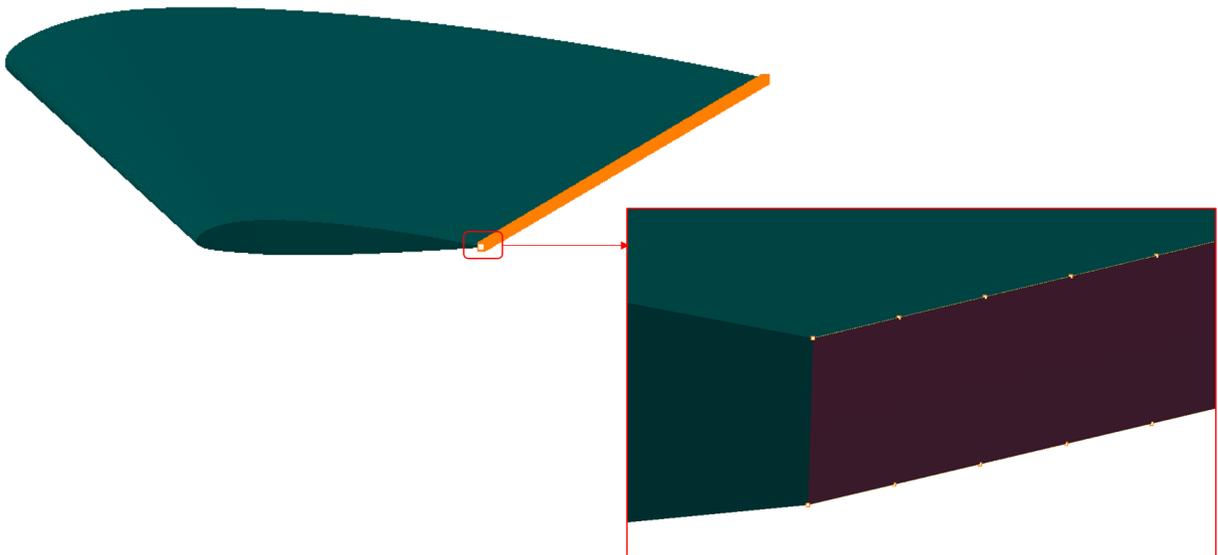


Figure 4.141: Corners created at the intersection of two surfaces

## 4.11.3 Ribbon

Once the corners are created on the slope discontinuity of the surface or on 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 opens the Generate Ribbon dialog box as shown in Figure 4.142. The *path group* is the corner group where the corners created by the feature edge/intersection is grouped. 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, path group is 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.

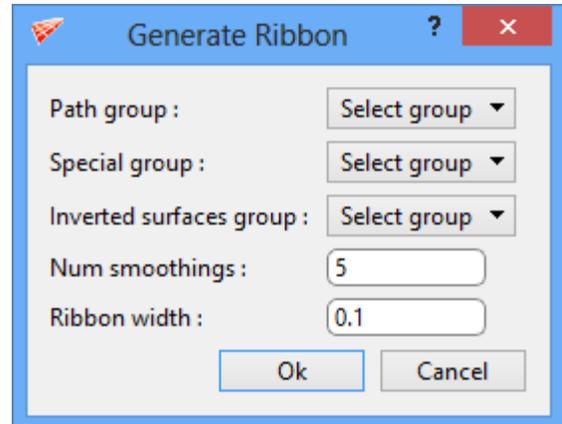


Figure 4.142: Ribbon dialog box

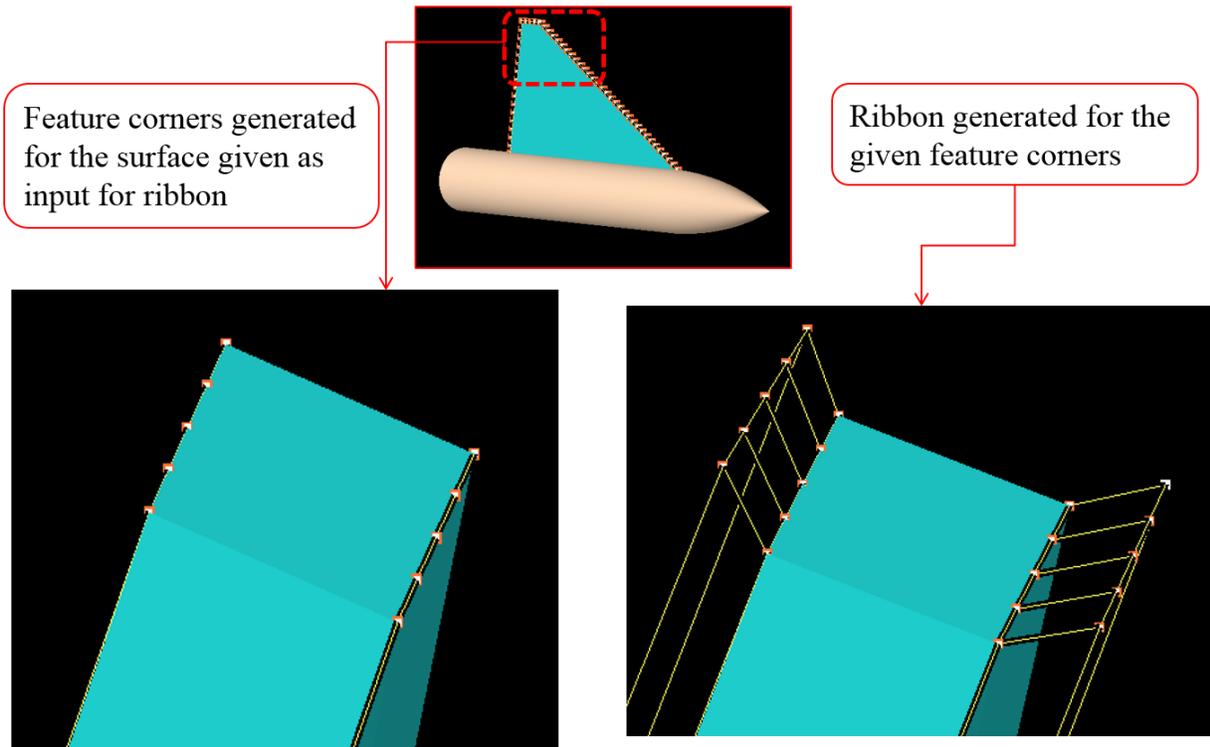


Figure 4.143: Ribbon generated from the feature edge corners

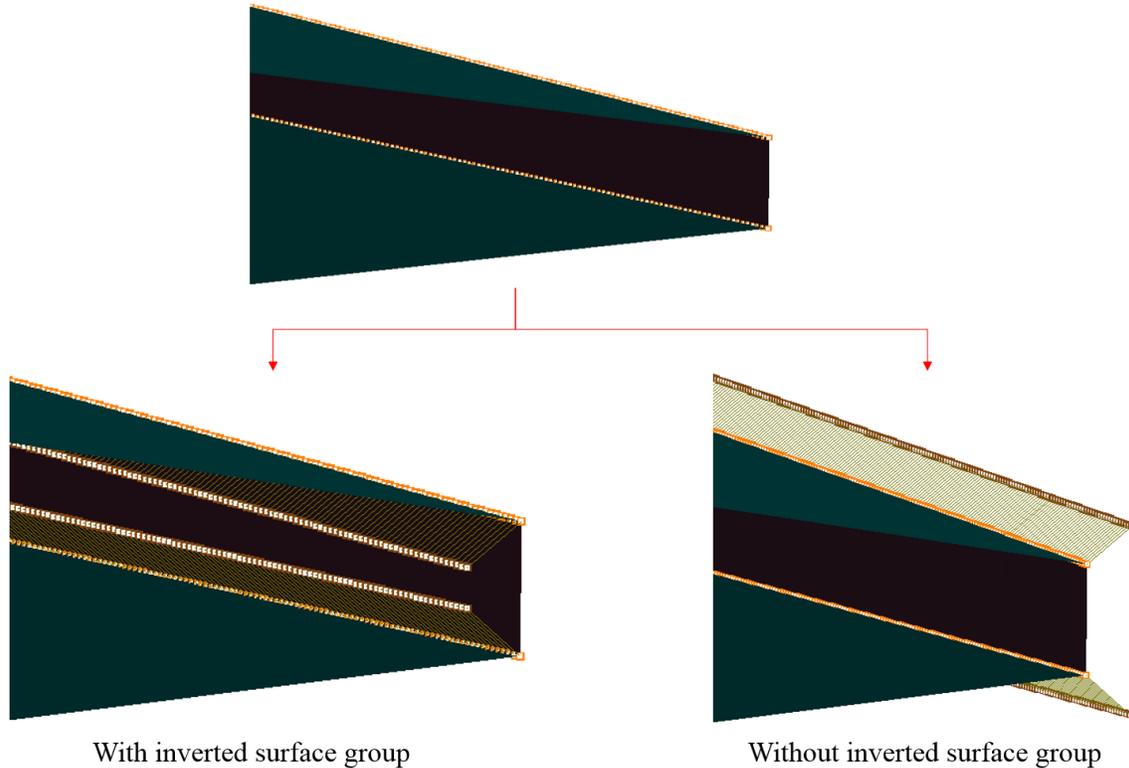


Figure 4.144: Ribbon – Inverted surface groups

#### 4.11.4 Ribbon nest

Many times, the nodes on the surface are more in order 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 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 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 as **reverse nest**. Reverse nest is used in case of 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 in order to retain the normal. Once selected, Surface Nest dialog box opens as shown in Figure 4.145.

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* drop-down list. Ensure the group that you provide in the outer group is an empty group.

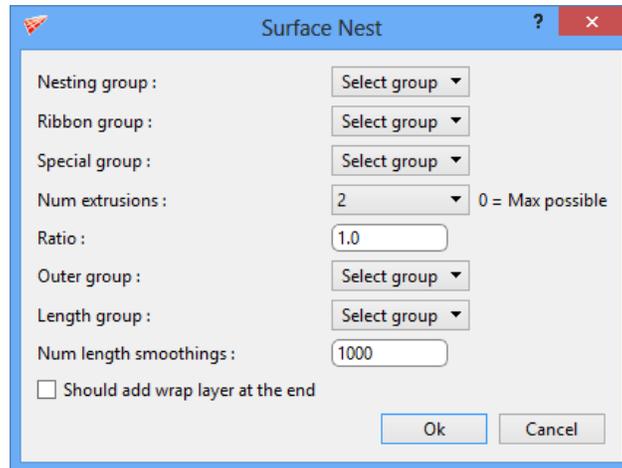


Figure 4.145: Ribbon nest dialog box

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's better to add a wrap layer after creating the ribbon nest.

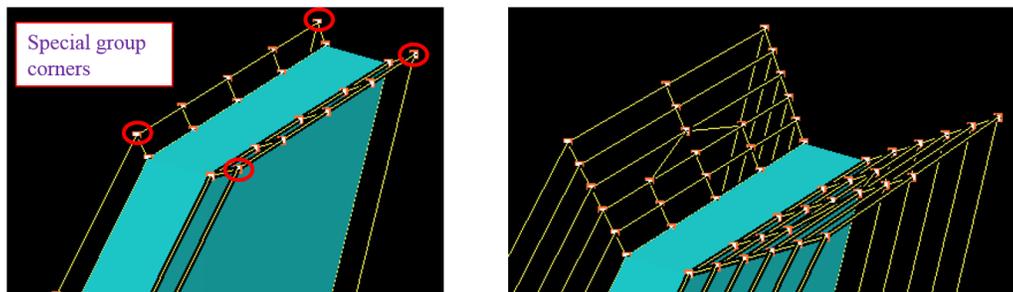


Figure 4.146: Ribbon nest generated from the ribbon corners with one level of refinement

### 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 drop-down list of the **Surface Tools** menu in the menu bar. Figure 4.147 shows the controlnet surface dialog box. Controlnet surfaces are created based upon 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 given feature corner group. **Smooth corner group** smooths the *surface* in that given smooth corner group region using the smoothing scheme which is inputted in the '**Smooth Scheme**' field. You can select the smoothing scheme from the drop-down button next

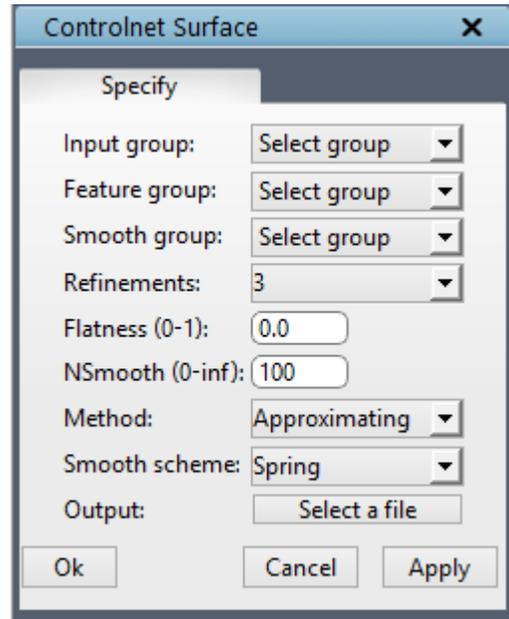


Figure 4.147: Controlnet Surface dialog box

to smooth scheme. In addition to this, smoothing or refining the whole surface can be done by inputting a refinement level in the **Refinements** drop down. 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 smooth corner group. You can define the **flatness** and **number of smoothing** required for your surface. These parameters will apply on whole input corner group. Similar to 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 passes 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 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 ways are explained below with an example image.

- **Default values**

Consider a topology which looks similar to an Elbow as shown in Figure 4.148. The surface is created using default values such as input group with 3 level of refinements using the approximating method and spring smooth scheme.

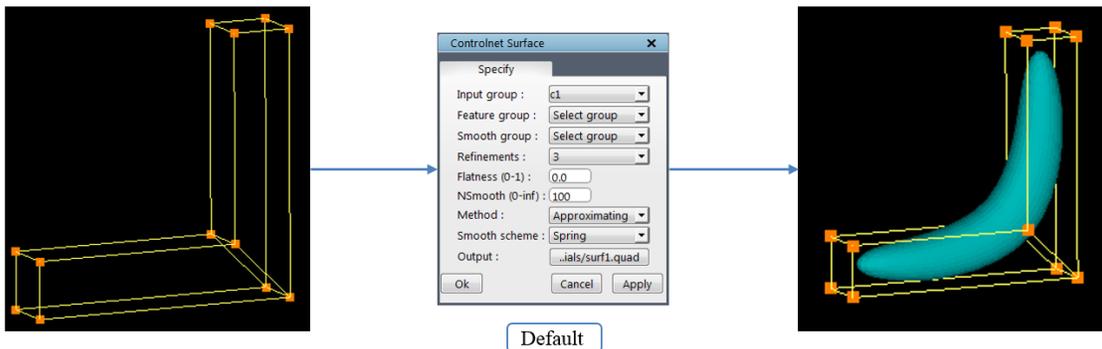


Figure 4.148: Controlnet surface using default values

- **Face exclusion**

Notice that the output in Figure 4.148 is closed on both the 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 steps that are followed for wrapping a quad as in 'Medium Tutorial 4 – Y-Exhaust'

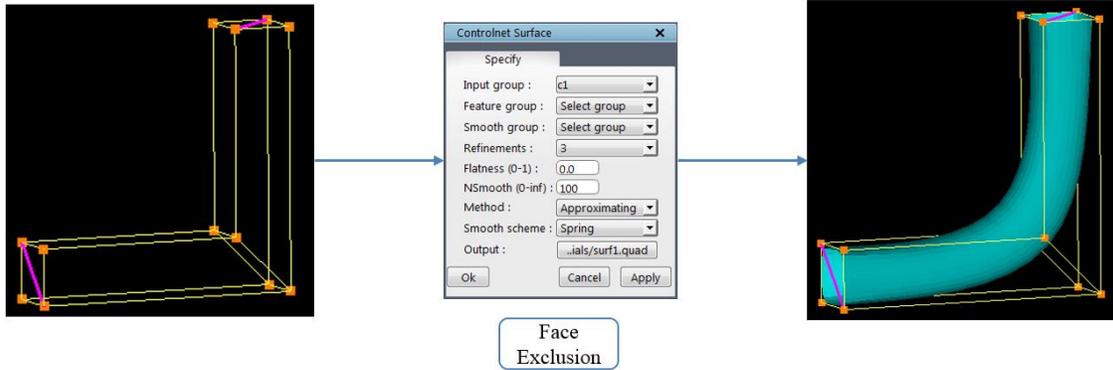


Figure 4.149: Controlnet surface using face exclusion

- **Inserts**

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

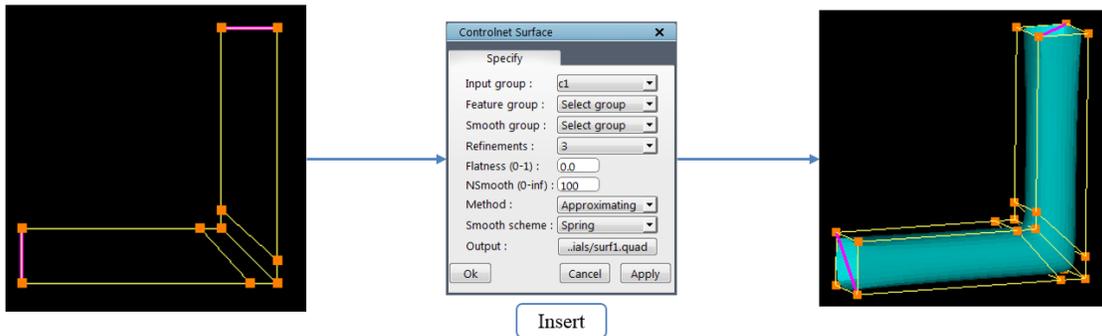


Figure 4.150: Controlnet surface using Inserts

- **Feature group**

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

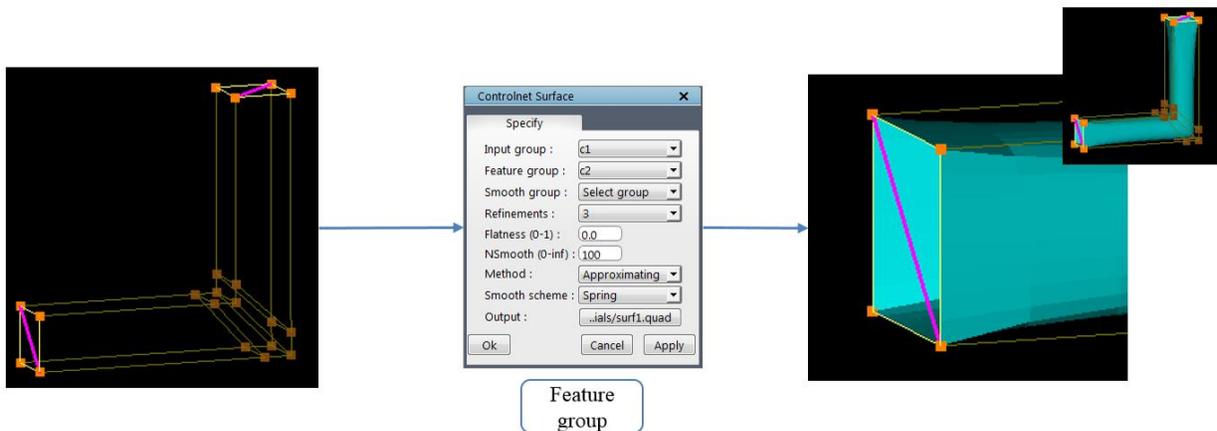


Figure 4.151: Controlnet surface using feature group

- **Smooth groups**

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

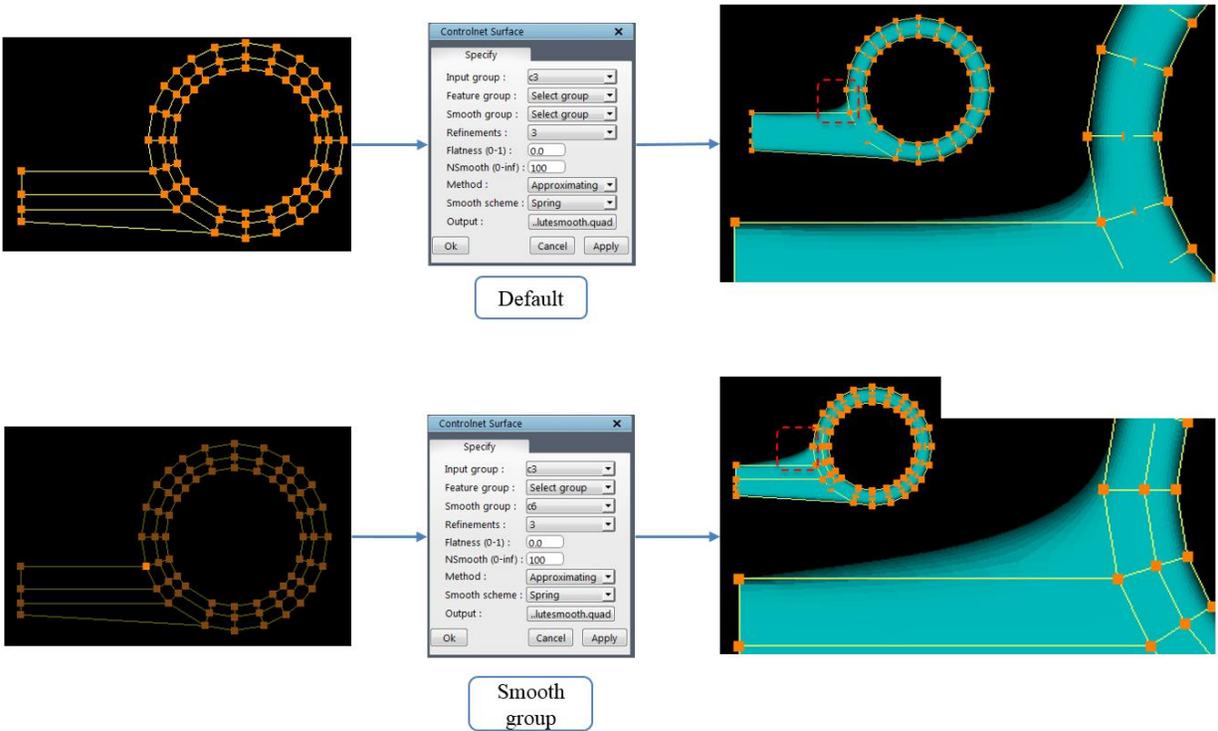


Figure 4.152: Controlnet surface using smooth group

- **Different Algorithms**

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

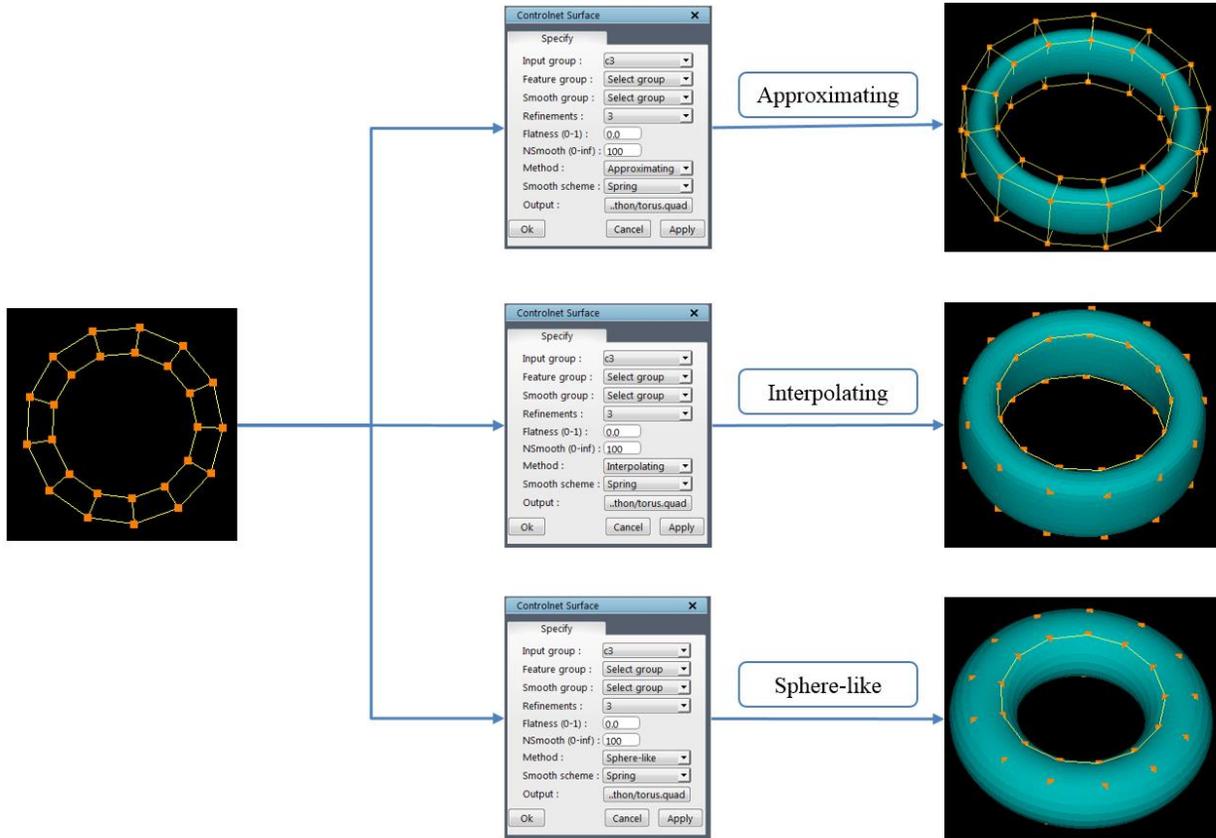


Figure 4.153: Controlnet surface using different algorithms

- **Refinement**

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

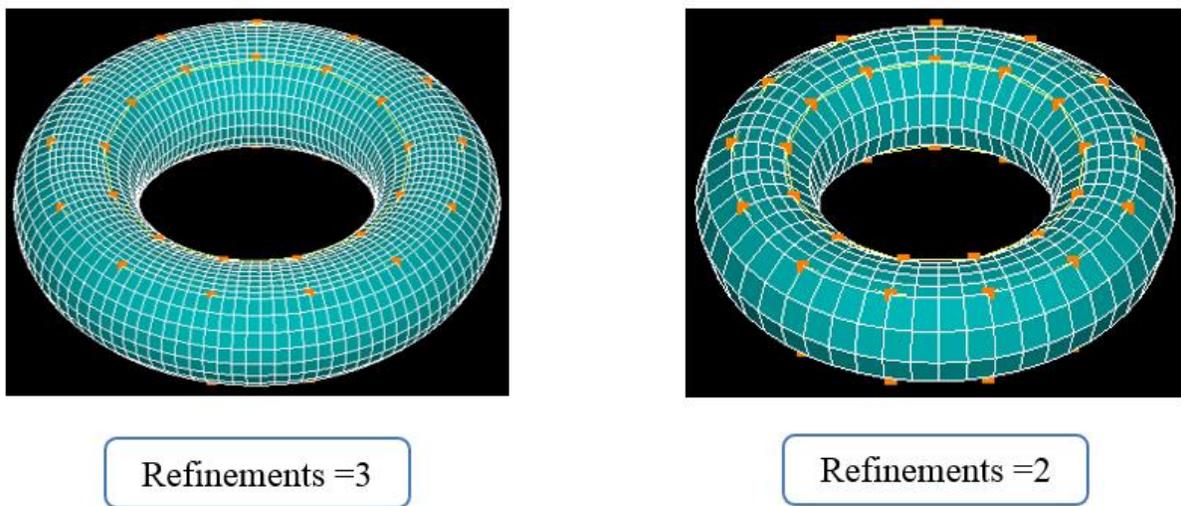


Figure 4.154: Controlnet surface using different refinement levels

- **Flatness**

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

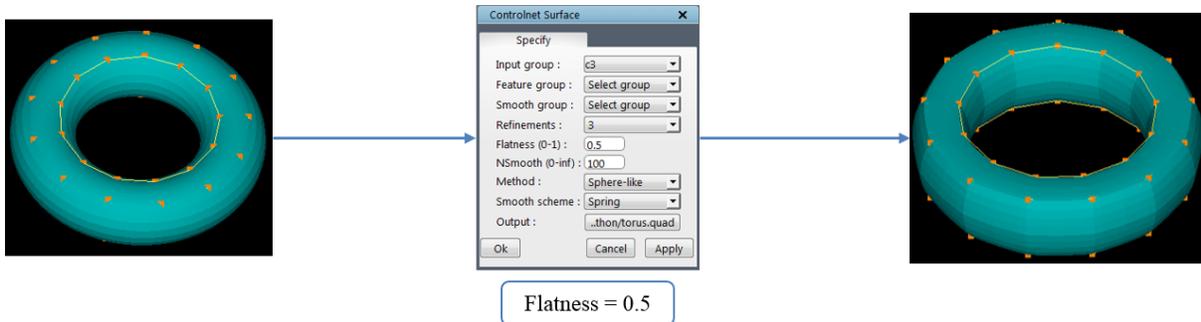


Figure 4.155: Controlnet surface using flatness

### 4.11.6 Controlnet Curves

To create a curve, click on the *Controlnet Curves* option in drop down list of **Surface Tools** menu in the menu bar. Figure 4.156 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 number of refinements is zero, the output curve is just a copy of the corners provided. Two different schemes are provided for generation of these curves which can be selected from the drop-down 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 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.

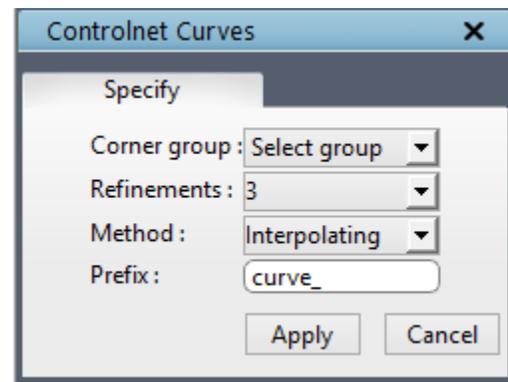


Figure 4.156: Controlnet Curves dialog box

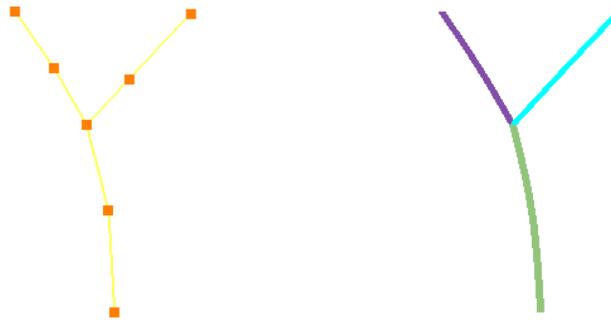


Figure 4.157: Branching wireframe and its corresponding controlnet curves

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 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 4.157 shows the branching input controlnet and output curve.

#### 4.11.7 Tube

A *tube* surface can be defined by the revolution around a center curve. The revolution for each point on the center 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 center 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 drop-down list of the **Surface Tools** menu in the menu bar. Figure 4.158 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 are having an equal number of corners.

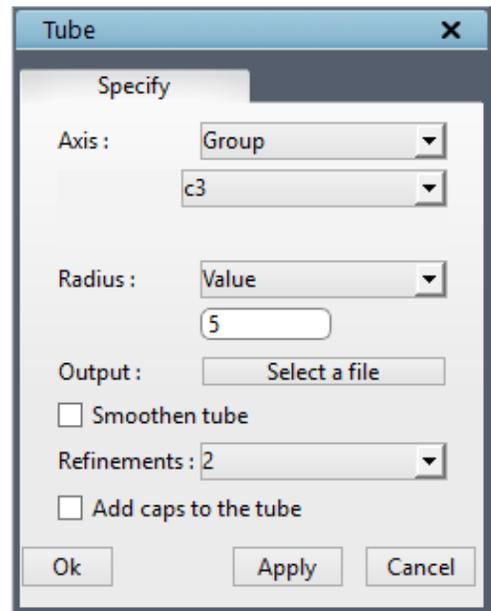


Figure 4.158: Tube dialog box

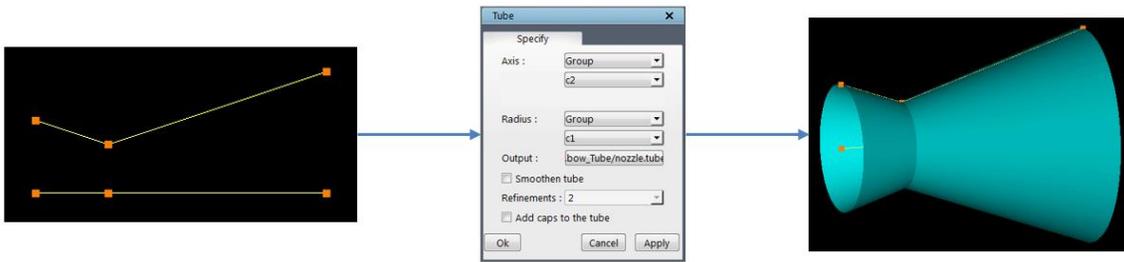


Figure 4.159: Tube surface with group-group combination

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

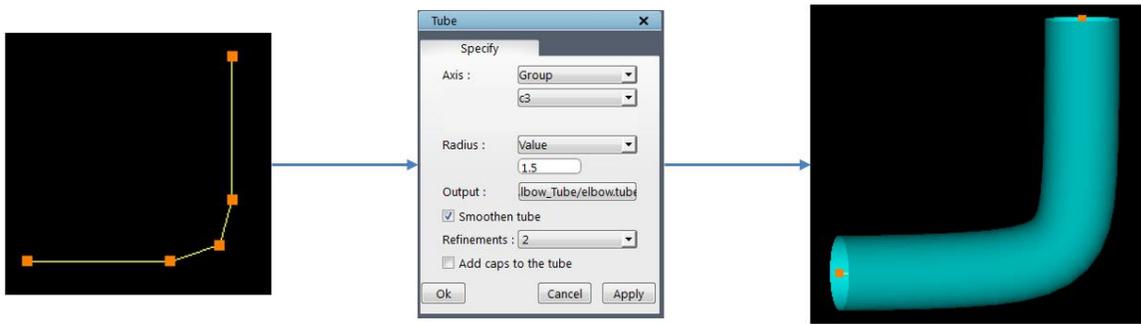


Figure 4.160: Tube surface with group-value combination

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

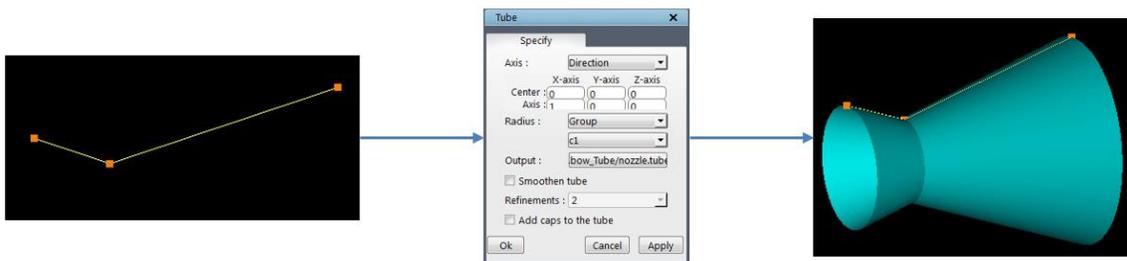


Figure 4.161: Tube surface with orientation-group combination

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

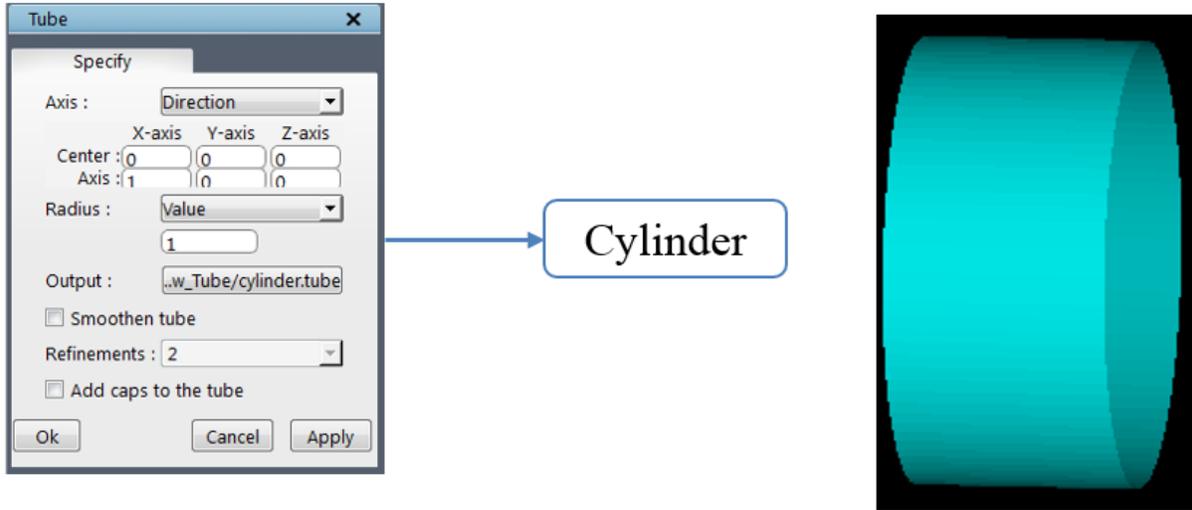


Figure 4.162: Tube surface with orientation-value combination

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 popped-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 drop-down list provided next to the **refinements**. This drop down 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.

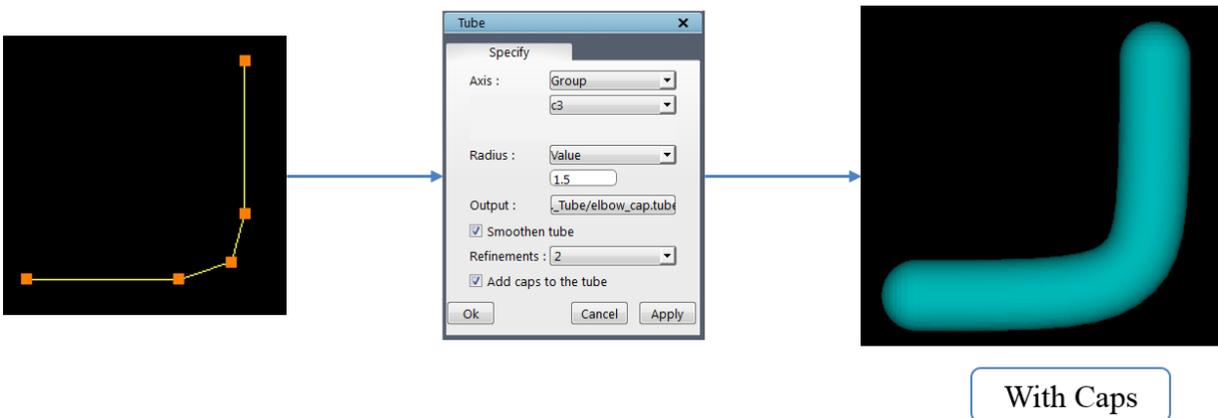


Figure 4.163: Tube surface with caps

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.

### 4.11.8 Internal surface

*Internal surface* is another tool which creates a quad surface at the intersection of surfaces. Figure 4.164 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 drop-down, 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

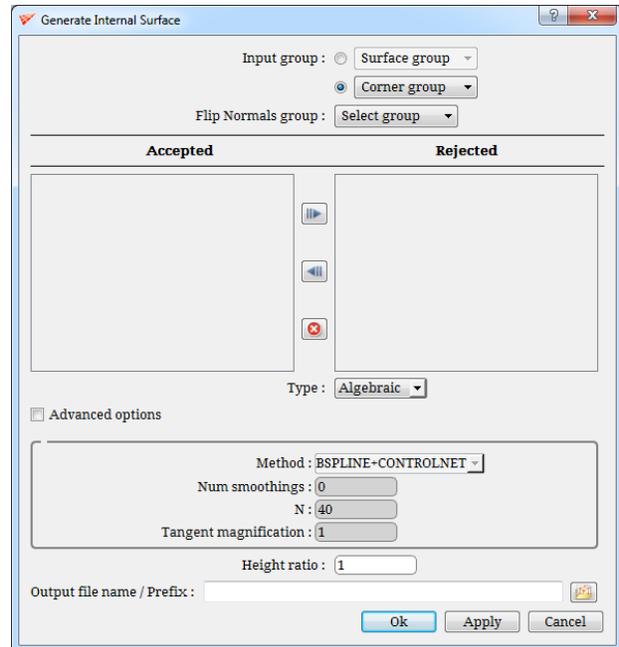


Figure 4.164: Internal surface dialog box

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.

If corner group is enabled, the desired corner group should be selected. To create the internal surface using corner group option, then 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 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* drop-down list. These three 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. First option is the method which is a separate algorithm for creating the surface which will be applied after the algebraic/variational type. A list of 6 different methods available under the *method* drop-down list. Next is the *number of smoothings* to be applied on the surface created. Next is the *N*, which is 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 advanced option is the *height ratio* which controls the maximum height of the surface.

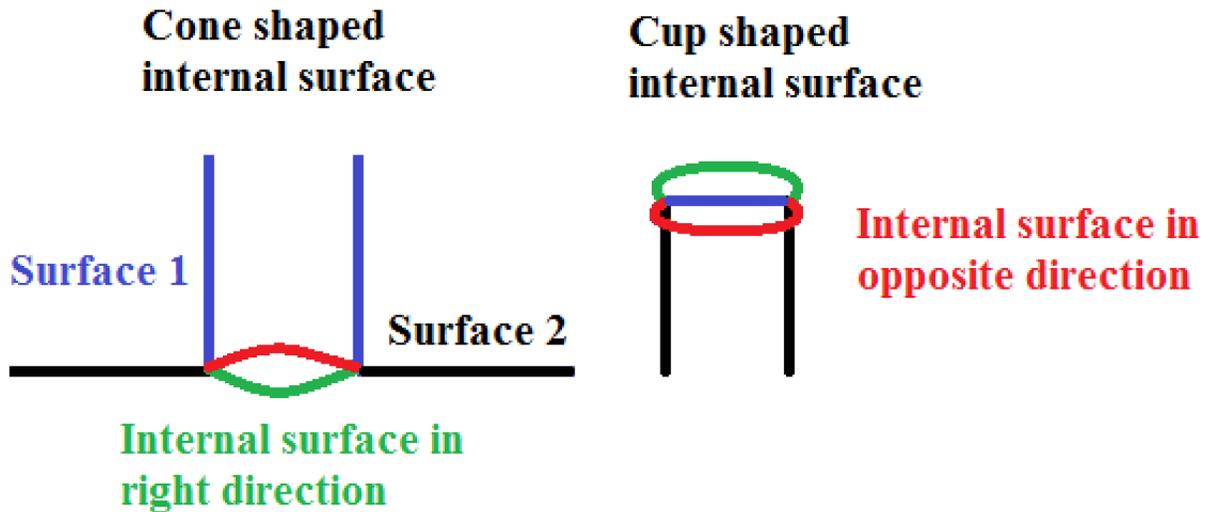


Figure 4.165: Four different ways on internal surface creation

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 4.165.

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

In such cases, input the surface group in ‘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 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 colors 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 out. 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’* files, *GridPro* automatically creates one with the default value. A sample schedule file is shown in Figure 4.167. 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 4.166: 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 4.167: Sample schedule file

### 4.12.1 Ggrid Groups

From version 6.6, ggrid can be run on part of a topology. i.e. on any given input group. The given group should have its boundary faces assigned to any surface either internal or external. 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 & grid sheets. All these utilities are placed under 2 sections namely *Grid toggling*, and *Tools* section at the right side panel of **Grid** tab.

Once the grid is generated, you can load a *blk.tmp* (default name saved by schedule file) file from the directory, you are working in it. Any number of grids can be loaded in the **Grid** tab at a time. You can work independently on each grid. UI allows you to use the tools on different grids without affecting other grid 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, centers of mass, and skeletons (links from the center of mass to the center of each face). These various features can be hidden or activated in various ways, in order to display different parts of a grid, are located in the **Grid** tab.

## 5.1 Grid Viewing Utilities

Because meshes are used extensively for analysis testing, *GridPro* allows you to visualize them in different ways. First and foremost, in order 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) in order 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).

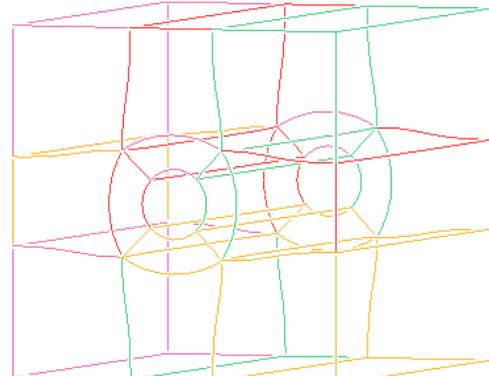


Figure 5.1: Grid blocks

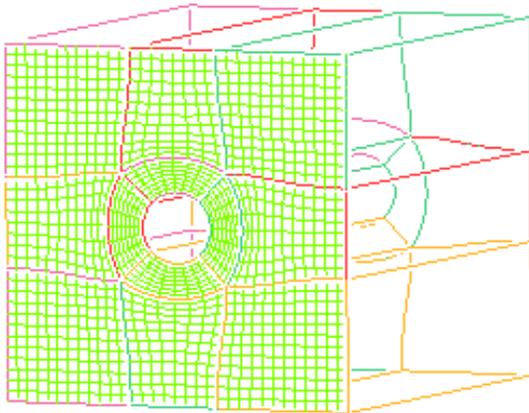


Figure 5.2: Grid blocks with one face highlighted

A *face* is a grid about a boundary of a block. More technically, a face is a network of links along 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 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.

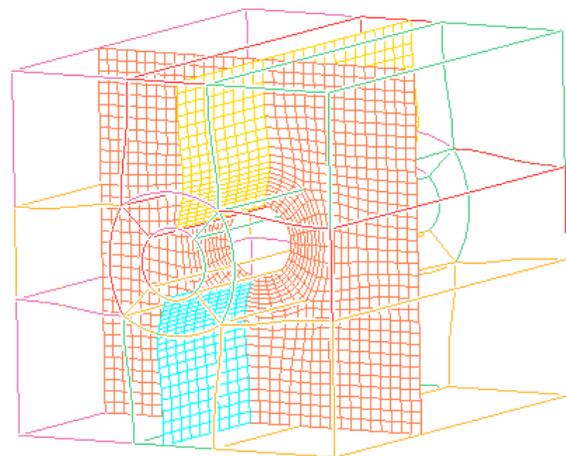


Figure 5.3: Grid blocks with a sliced sheet

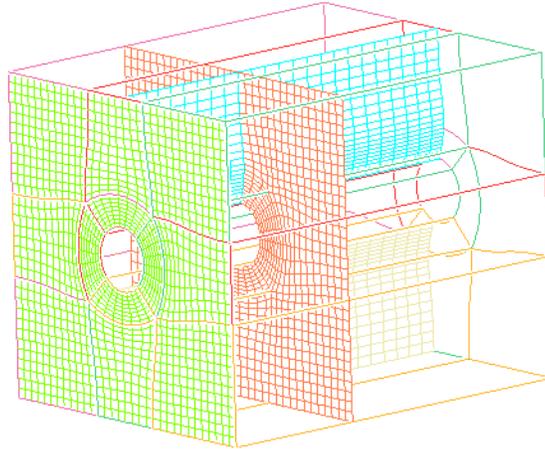


Figure 5.4: Grid block with face sheet and slice sheets

Let us summarize: Distinct blocks have distinct faces and distinct slices. Many linked faces or many 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 to 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.

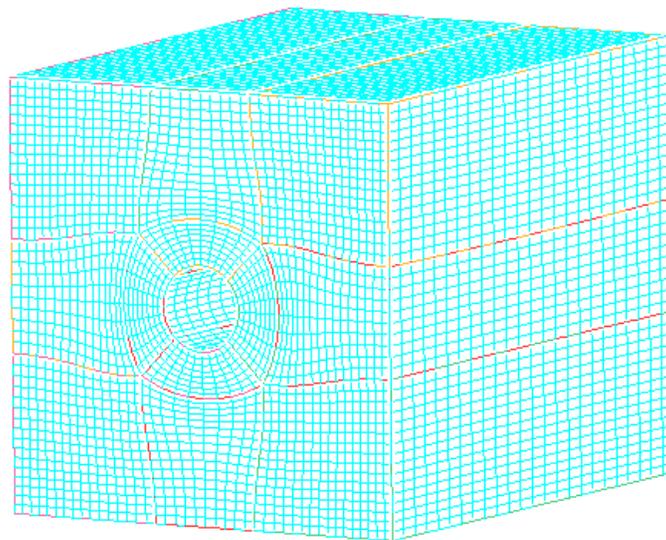


Figure 5.5: Grid blocks with mesh

## 5.1.2 Faces

**Faces**  button is used to create sheet consist of all block faces on all the external grid surfaces. It creates grid sheets for all surfaces, displayed in different colors. 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 **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.

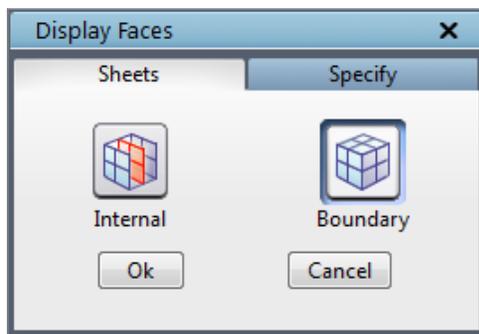


Figure 5.6: Faces dialog box

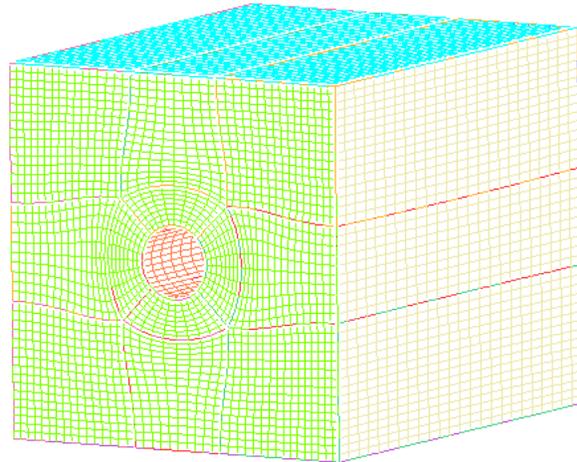


Figure 5.7: Grid blocks with boundary grid sheets

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.

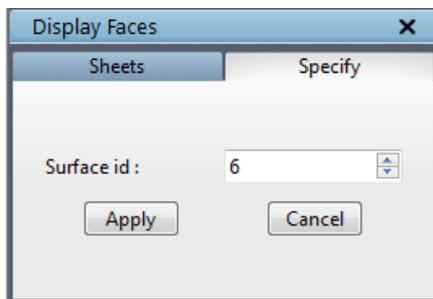


Figure 5.8: Faces dialog box for specify tab

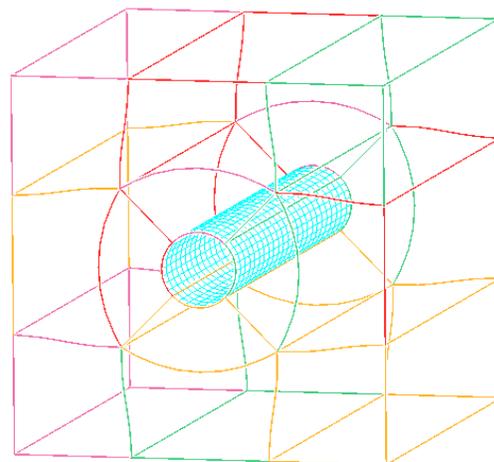


Figure 5.9: Grid blocks with specific surface grid sheet

### 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 in right-hand side to step forward or backward a sheet. The stepping process of the grid sheets is explained in *section 5.6*.

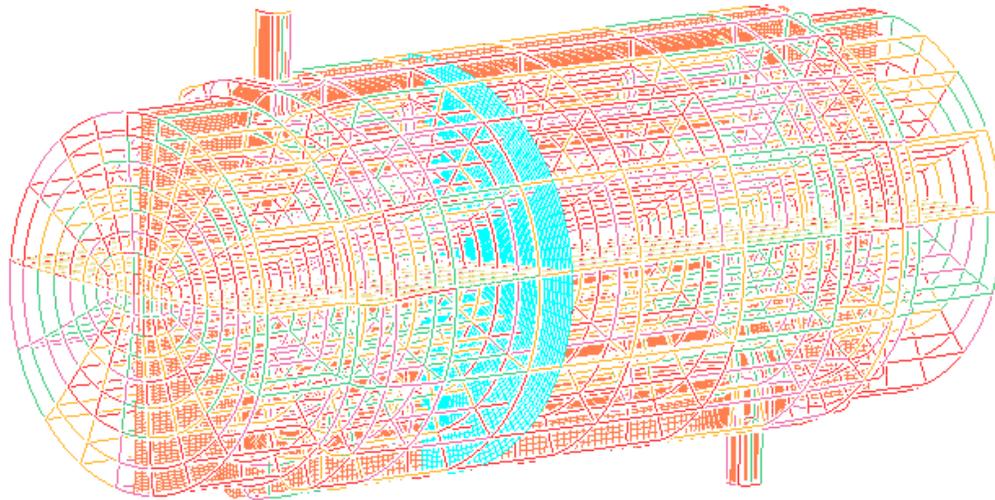


Figure 5.10: Cross-sectional grid sheets at arbitrary IJK

- **Right-click menu**

When you *right-click on an edge of a grid block* a menu appears as shown in Figure 5.11. The first option is **center of rotation**; you can pick a particular point of that edge of grid block as the center of rotation, by selecting the **center of rotation** option after right

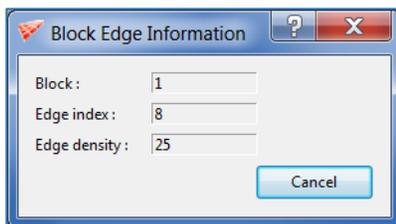


Figure 5.12: Grid block edge information

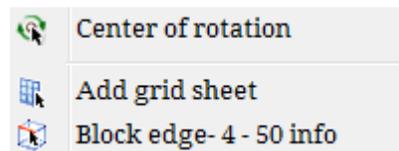


Figure 5.11: Right click menu of grid blocks

click. Similar to *slice* button, you can add grid sheet at that particular point on grid edge by selecting **add grid sheet** option. The last option shows the **grid block edge information**. It will show block index no, edge index number along with the edge density as shown in Figure 5.12.

You can also *right-click on a grid sheet* which shows the menu as in figure 5.13. Similar to picking the center of rotation on a surface, wireframe, and grid blocks, you can also select it on

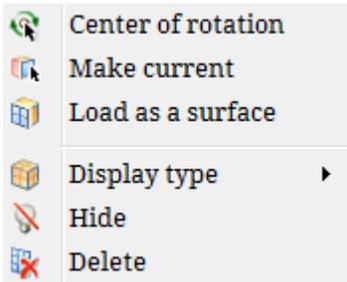


Figure 5.13: Right click menu of grid sheets

the grid sheet by selecting the *center 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, many times we 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. 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 2.61 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 global dock opens a window that allows you to find and access the file, *blk.tmp*. Note: you need not stop the gridding process in order to access the grid. In fact, since the file, *blk.tmp*, is being updated after every 100 sweeps, you can reload an updated version of *blk.tmp* by

accessing the *Update*  button in the *Edit* section. Clicking on the *Update* button will pop out a dialog box as shown in

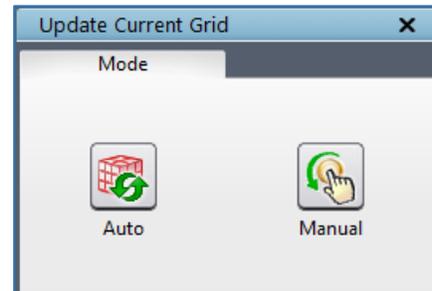


Figure 5.14: Update dialog box

Figure 5.14. If you select *Auto* option, GridPro will automatically perform the *update* operation after every 100 sweeps or every time the grid file is written until the *Auto* button is switched off. If you select *Manual*, it allows you to perform the update option once.

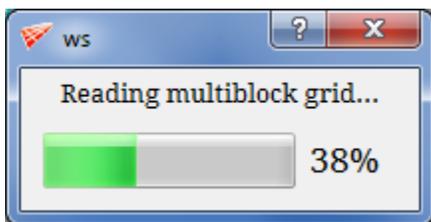


Figure 5.15: Loading/Updating grid pop-up

**Note:** Sometimes the gridding process may be faster due to less number of blocks, in such cases selecting *Auto* option may not be a wiser idea. Since the grid file will be written faster than the time it takes to load the grid, you may not be able to see the results immediately. It also keeps the *Updating grid* pop up opens all the time due to the speed at which the grid file is updating, which restricts you from accessing any

options in the UI. So in such cases, you can use *Ctrl+Shift+G* to stop the gridding process using your keyboard without even clicking on the UI. This helps to avoid you from crashing the UI forcibly.

## 5.1.5 Colour

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

The **Colour** button  pops out 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 depend on the qualities such as Skewness, Aspect ratio and Warpage. By default, the **colour by sheet** option is selected as shown in Figure 5.16.

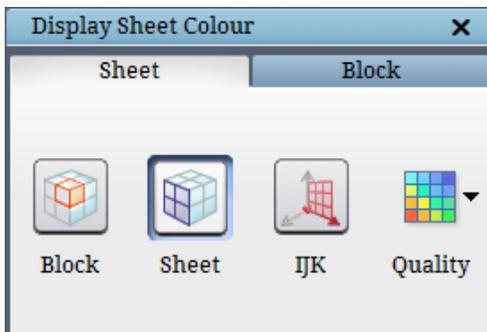


Figure 5.16: Colour dialog box with Sheet tab

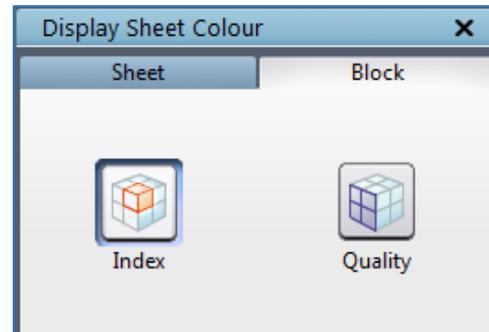


Figure 5.17: Colour dialog box with Block tab

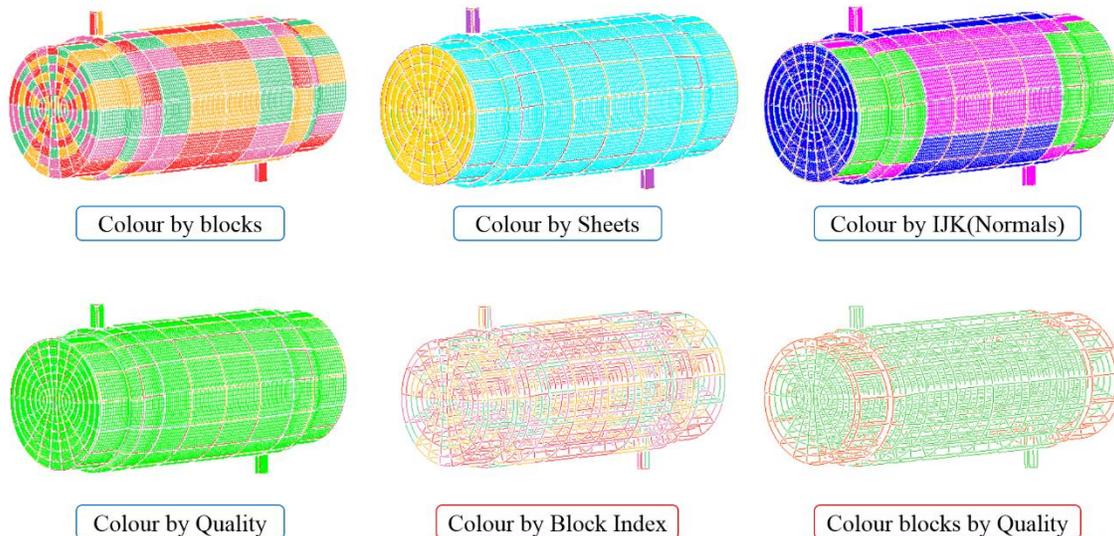


Figure 5.18: Different colour by sheets and blocks options

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 colour. The reference values for the good quality would be assumed from the default values of *Quality* button.

## 5.2 Preprocessing Utilities

### 5.2.1 Extract

In addition to visualizing the grid, in **Grid** tab you can also generate the topology from the grid. *Extract* button in the **Edit** section located in the right side panel of **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 grid connectivity file (\*.conn) along with grid. Once the button is clicked, a confirmation dialog box pops out where you need to click on ‘Ok’ to confirm the execution of the process.

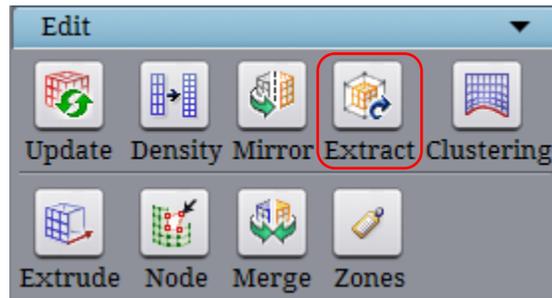


Figure 5.19: Edit Section

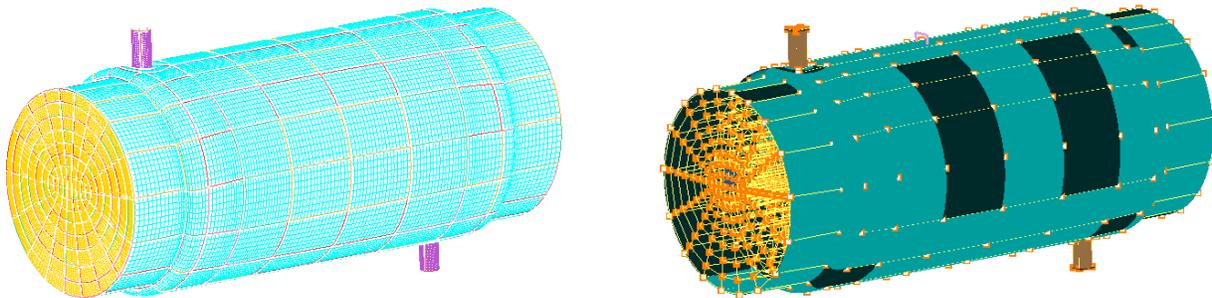


Figure 5.20: Grid and its corresponding topology extracted using Extract

The Extract button calls an external function called ‘**grid2til**’ which is 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 some minor modifications in the grid, say changing density in few edges. GridPro post-processing utilities are used, once the grid is generated. There are two post-processing utilities – *Density* and *Mirror*.

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

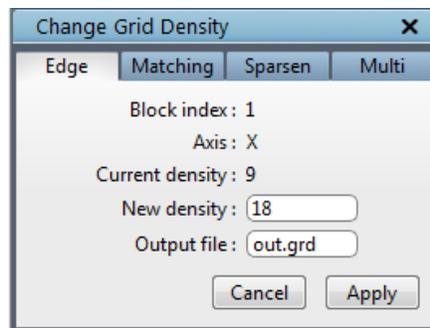


Figure 5.21: Density dialog box

*Density* button has four tabs, each performs different operations as follows. Once you click on the Density button, Change Grid Density dialog box opens as shown in Figure 5.21. 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 *Change density* option from right click menu. Selecting *Change density* option from the right-click menu of a grid block edge will open the same dialog box as shown in Figure 5.21.

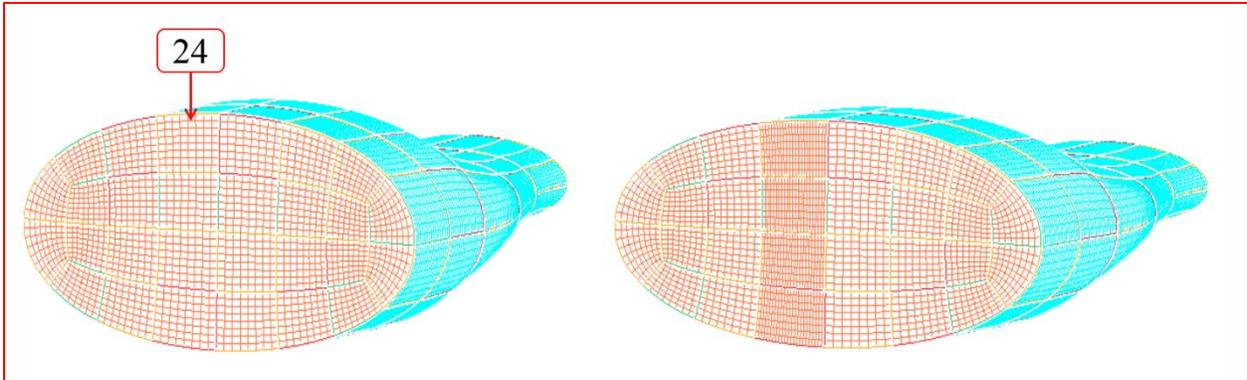


Figure 5.22: Before and after changing Edge density

The second tab in Density dialog box is **Matching**, whose dialog box is shown in Figure 5.23. A particular density of a whole grid can also be modified using this option. It will modify the density of the edges which were having 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 *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.

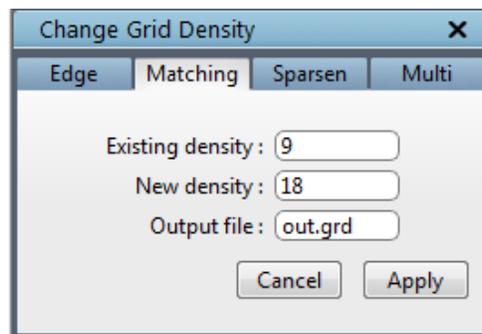


Figure 5.23: Matching tab dialog box

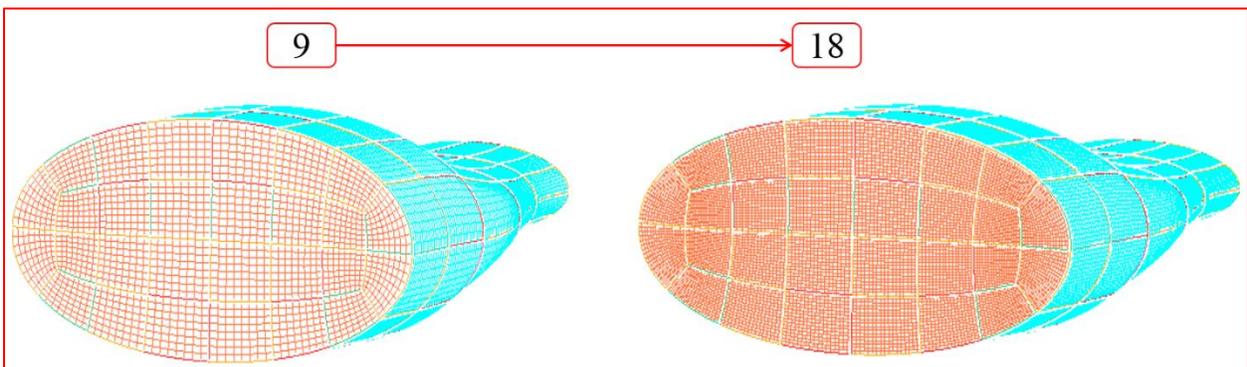


Figure 5.24: Before and after changing the density

If the density of the whole grid has to be increased or decreased by a ratio, *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 5.25. 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 sparsen the grid and more than 1 to refine the grid. Similar to *Matching* option, *Sparsen* will also create and load a new grid file and switch off the old grid file.

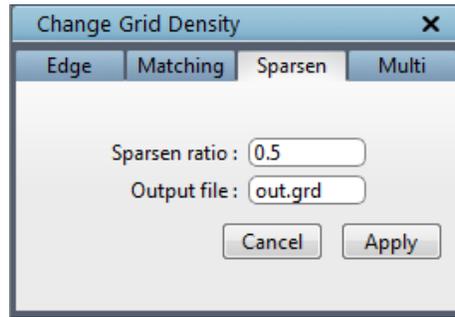


Figure 5.25: Sparsen tab dialog box

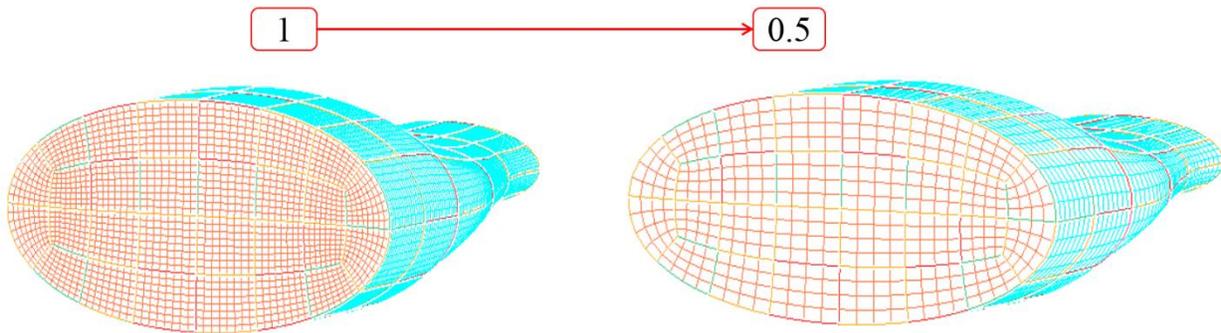


Figure 5.26: Before and after changing the density

Multiple grid files with an arithmetic or geometric increment/decrement in ratio can also be generated by using *Multi* option which is the last tab in the *Density* dialog box. The dialog box of the same is shown in Figure 5.27. Provide the ratio and number of grids 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.

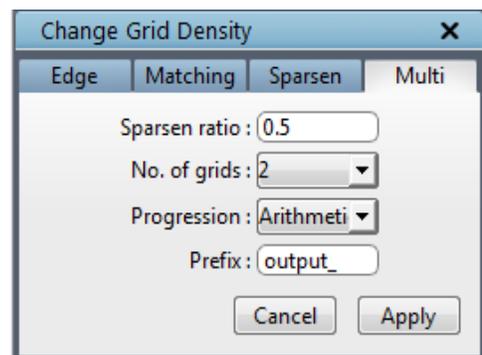


Figure 5.27: Multi tab dialog box

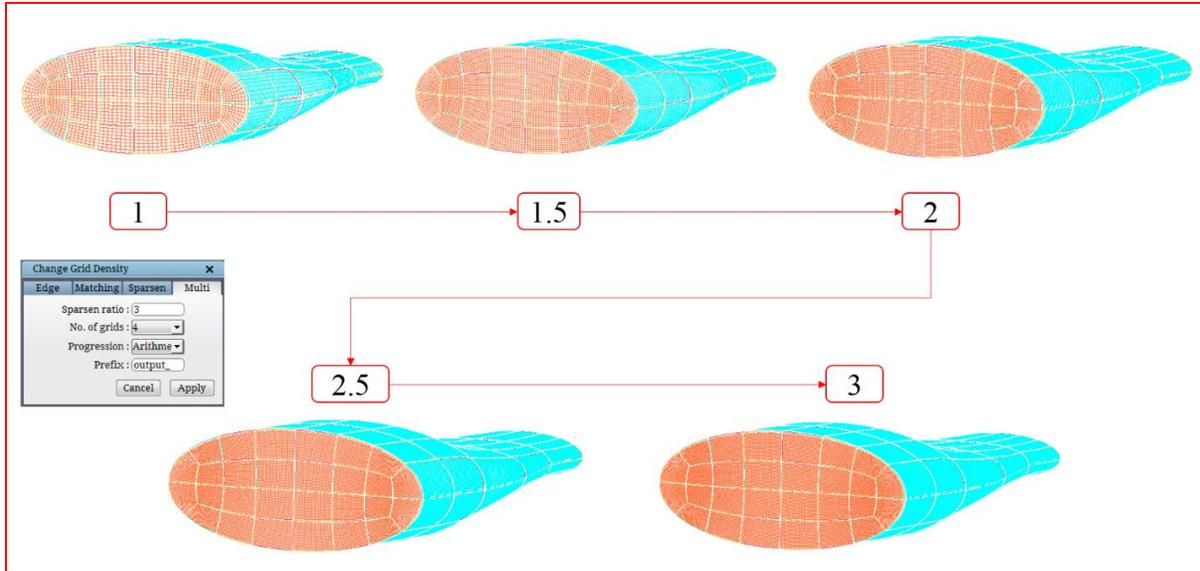


Figure 5.28: Multi-Grids output

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 the current grid is. If you are using the Density button for the second time simultaneously, make sure that you have selected the desired grid as *current*.

### 5.3.2 Mirror

Another feature which can create reflective or periodic copies of a grid is **Mirror**. This option is also located in the **Edit** section. If clicked on **Mirror** option, a dialog box will open in the right side panel as shown in Figure 5.29. 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 upon its periodicity and merge all the copies 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 UI in addition to previous grid file. If **workplane** is selected, the center 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 center 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, original grid and a newly created merged grid. Suppose if a wrong axis is selected, undo the process by clicking on the **Undo** button and redo the process by selecting the correct axis.

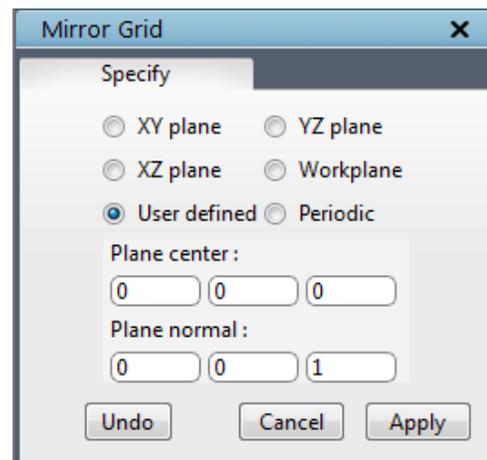


Figure 5.29: Mirror dialog box

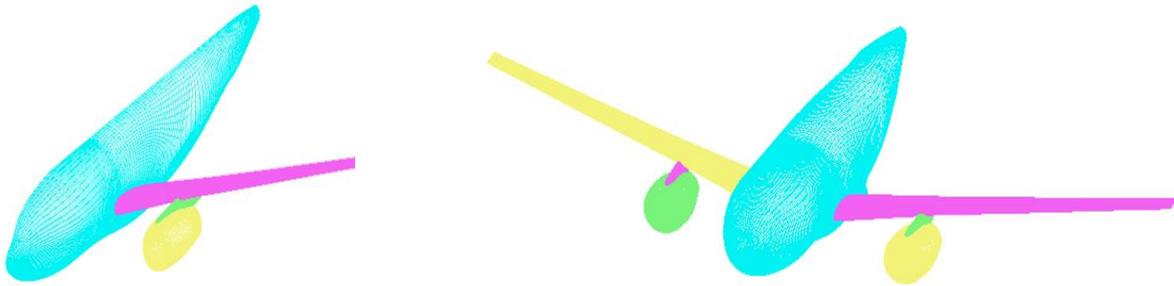


Figure 5.30: Before and after Mirror grid

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

### 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 given no. of cells in the normal direction. It can be found in the **Edit** section of the **Grid** tab on the right side tab. Clicking on **Extrude**  button pops a dialog box as shown in Figure 5.31.

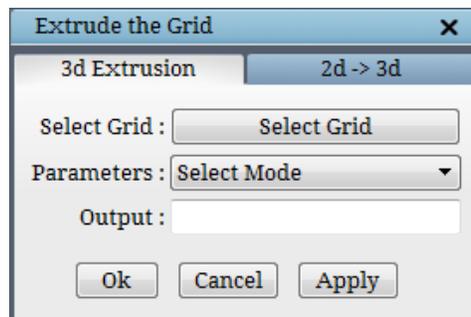


Figure 5.31: Extrude dialog box

- **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 5.32.

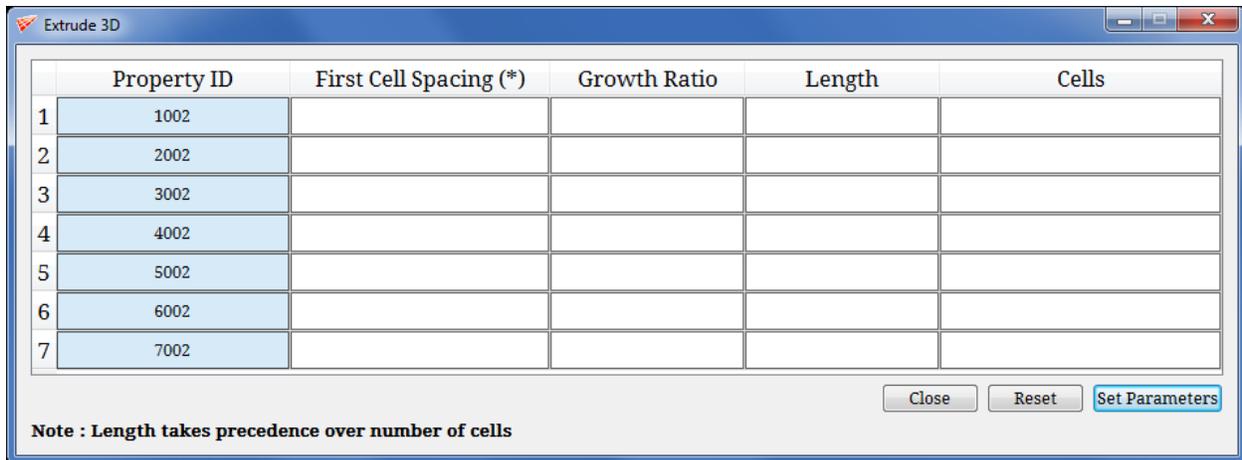


Figure 5.32: Property id tabular form

**Property ID** is an id which 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 end of the file. The properties would be listed as shown in Figure 5.33. The number prior to the property labels (surface\_1, etc) is called as property id.

```

#2 Blocks
## bid b-pty b-lbid f_i-pty f_i-lbid f_I-pty f_I-lbid ...
# 1 1 -1 2002 -1 1 -1 5002 -1 4002 -1 1 -1 6002 -1
# 2 1 -1 1 -1 1 -1 5002 -1 4002 -1 1 -1 6002 -1
# 3 1 -1 1 -1 3002 -1 5002 -1 4002 -1 1 -1 6002 -1
# 4 1 -1 5002 -1 4002 -1 1 -1 2002 -1 1 -1 1 -1
# 5 1 -1 1 -1 2002 -1 5002 -1 4002 -1 1 -1 7002 -1
# 6 1 -1 4002 -1 5002 -1 1 -1 3002 -1 1 -1 1 -1
# 7 1 -1 1 -1 1 -1 5002 -1 4002 -1 1 -1 7002 -1
# 8 1 -1 3002 -1 1 -1 5002 -1 4002 -1 1 -1 7002 -1
# 9 1 -1 5002 -1 4002 -1 1 -1 1 -1 1002 -1 1 -1
# 10 1 -1 1 -1 1 -1 5002 -1 4002 -1 1002 -1 1 -1
# 11 1 -1 1 -1 1 -1 5002 -1 4002 -1 1002 -1 1 -1
# 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 5.33: 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 the precedence. Once it is set, click on 'set parameters' button and click on Apply in the main dialog box as shown in Figure 5.31.

Similarly, if the ‘**surface ids**’ option is selected, a dialog box as shown in Figure 5.34 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.

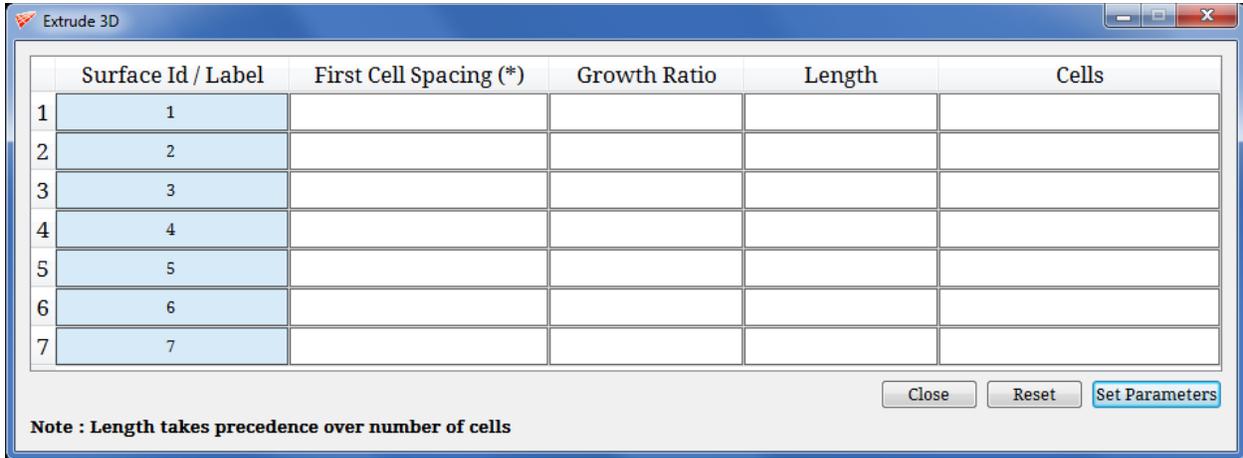


Figure 5.34: Surface id tabular form

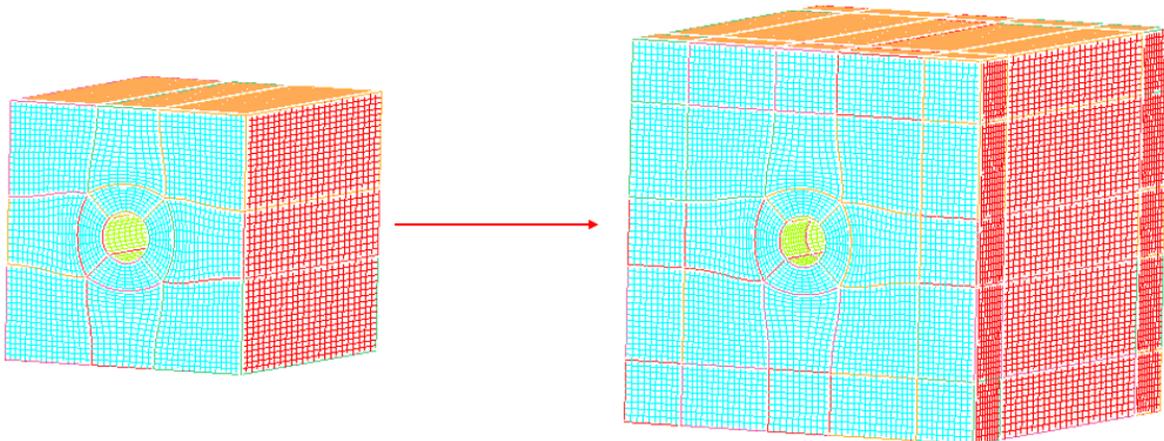


Figure 5.35: Before and After Extrude on all 6 sides

- **2D → 3D**

A 2D grid can be extruded to generate a 3D grid using the 2D→3D dialog box as shown in Figure 5.36. Since the 2D grids are always generated on XY plane, it 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 Density button. Once the required input

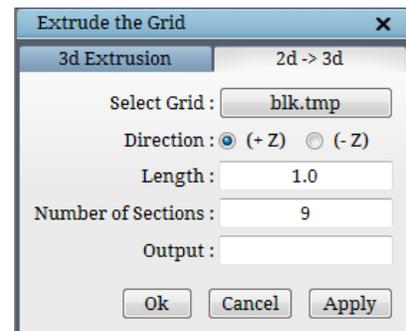


Figure 5.36: 2D → 3D dialog box

and output file name is given, it generates a 3D grid and loads it into the UI.

This operation is primarily helpful when your geometry has same cross-section along the lengths of it. For e.g. analyzing an infinite wing geometry where the cross-section of the wing has same airfoil profile along its span.

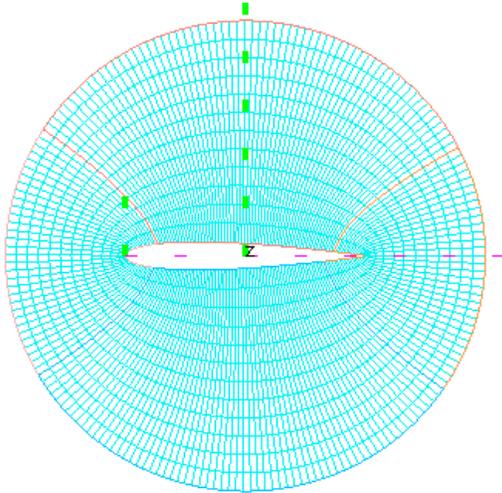


Figure 5.37: 2D grid

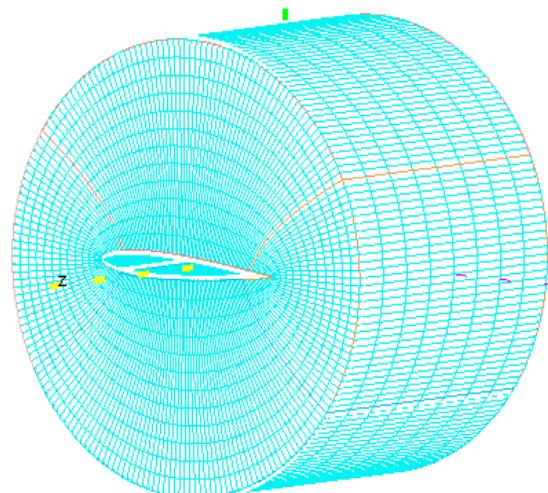


Figure 5.38: 3D grid after extrusion

### 5.3.4 Clustering

Boundary layer clustering  is one of the most important steps in grid generation. It was accomplished using two ways either setting 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 nice tabular form as shown in Figure 5.40.

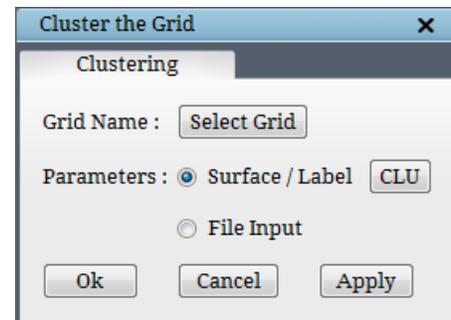


Figure 5.39: Clustering dialog box

If clicked on the **Clustering** button, a dialog box opens in the pop\_up section as shown in Figure 5.39. Grid Name will have the list of grid file names which are loaded into the UI. If the UI has only one grid, then it automatically reads the same. The clustering parameters can be provided in two ways.

1. Using Surfaces or labels
2. Using a text file

By default, **Surface/Label** option is selected. In order to set the parameters, click on the **CLU** button which will open a tabular form dialog box as shown in Figure 5.40.

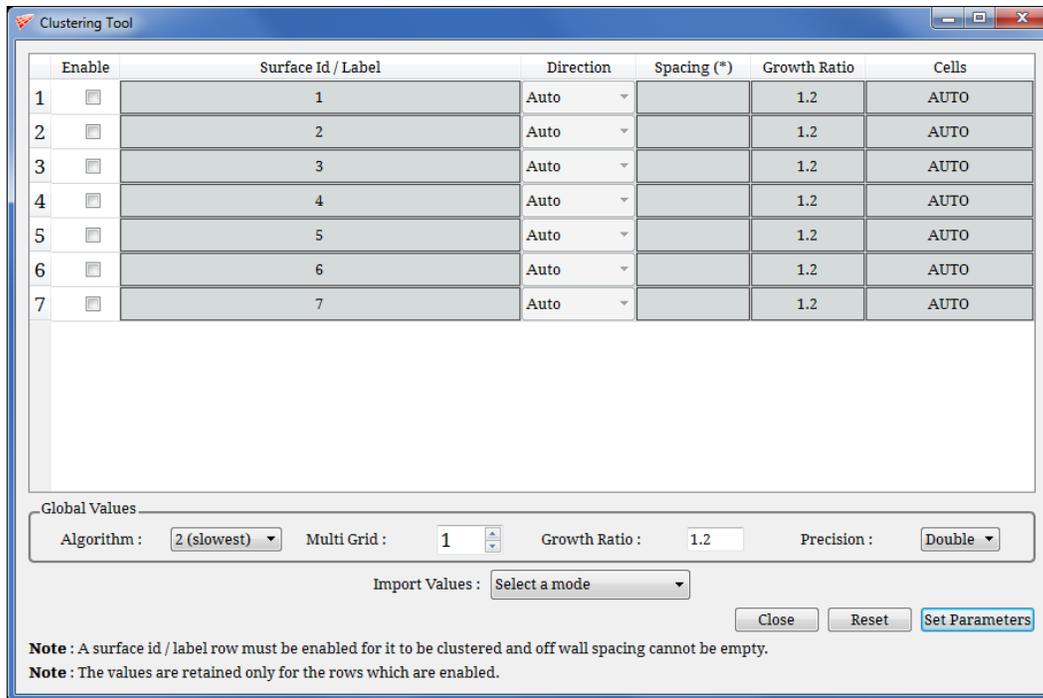


Figure 5.40: Clustering tabular form

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. In Growth ratio, the ratio is mentioned and the no. of cells for the block is being determined internally whereas in 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 such as 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 average spacing
- 1 = Smoother algorithm with 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 or 20 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 ‘*set parameters*’ button at the bottom and click ‘*Apply*’ in the main dialog box as in Figure 5.39.

The parameters can also be imported into the table from a fra file or text file using the Import values drop down. It can also be imported by selecting the ‘**File Input**’ option in the Clustering dialog box as in Figure 5.39.

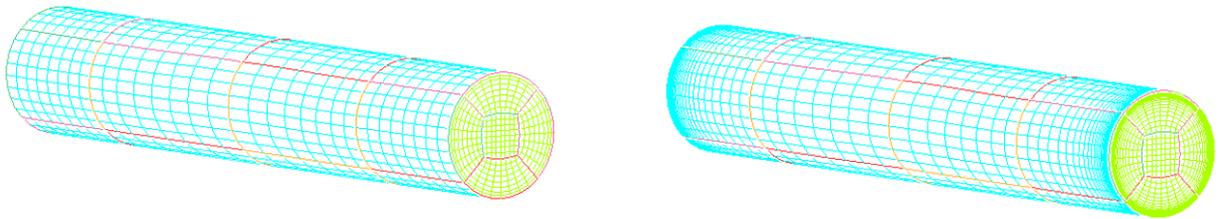


Figure 5.41: Before and After Clustering

### 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 the Figure 5.42.

You can either merge the blocks within a grid using Merge→Blocks tab or merge two different grids into a single grid using Merge→grids tab.

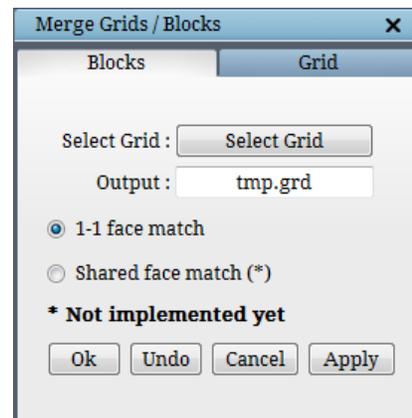


Figure 5.42: Merge →Blocks tab

By default, the dialog box opens with **Merge→Blocks** tab, where the input grid has to be selected. If multiple grids are loaded into the UI, it will pop\_up a dialog box with the list of grids in the UI, 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 which has one to one face match.

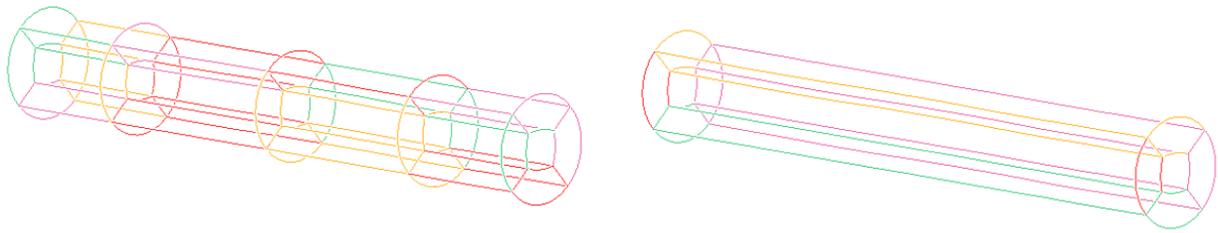


Figure 5.43: Before and After merging

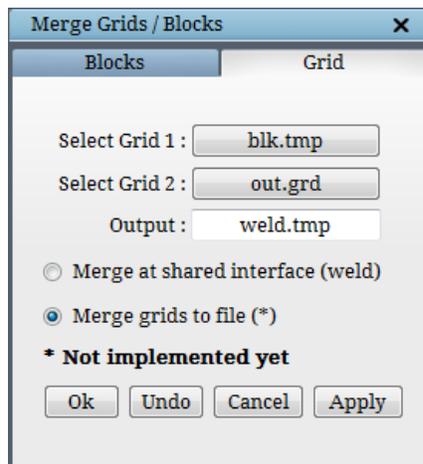


Figure 5.44: Merge →Grid tab

**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 same block pattern and no. of cells at all the welding interfaces. Currently, it only executes the basic welding. For advanced merging options, refer to the command line tool called ‘**weld**’. Similar to Merge→Blocks, here also user need to select the grids from the loaded ones.

### 5.3.6 Node

**Node** button helps in improving 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 output a grid without any folds. Resolving the singularities is a time-consuming process hence we came up with a tool which can move the nodes at 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 a dialog box and highlights the grid nodes of the current grid sheet which aids the cursor to move the desired nodes as shown in Figure 5.45.

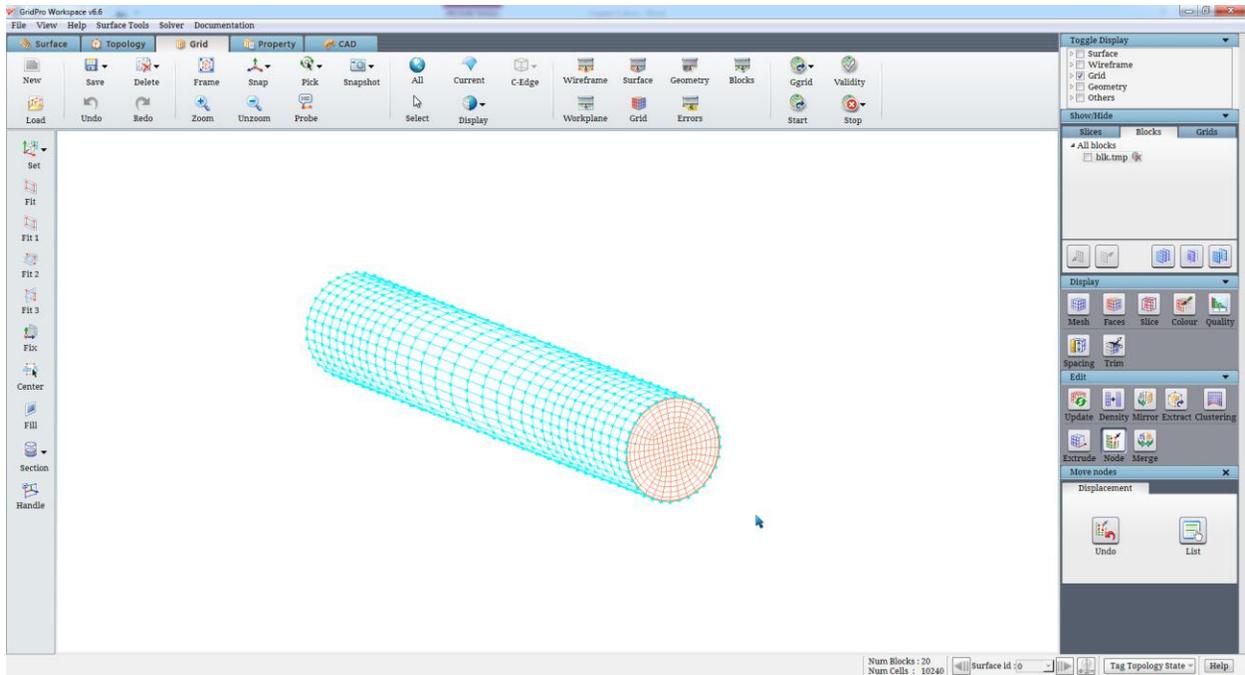


Figure 5.45: Node dialog box

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 integral part of a grid. There are so 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 condition 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 centers manually in a complex grid like aircraft. In order to ease the process of setting the volume properties to a complex grid, a new function called '**Zones**' is implemented in v6.7.

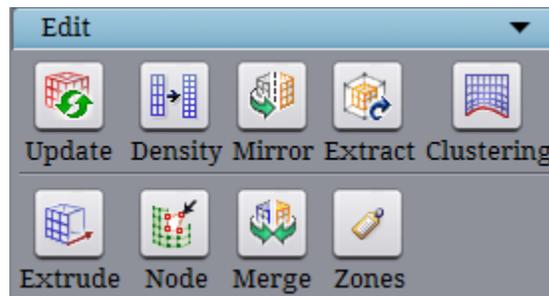


Figure 5.46: Edit section

**Zones** is located under **Edit** section of the **Grid** tab as shown in Edit section Figure 5.46. 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 set 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.

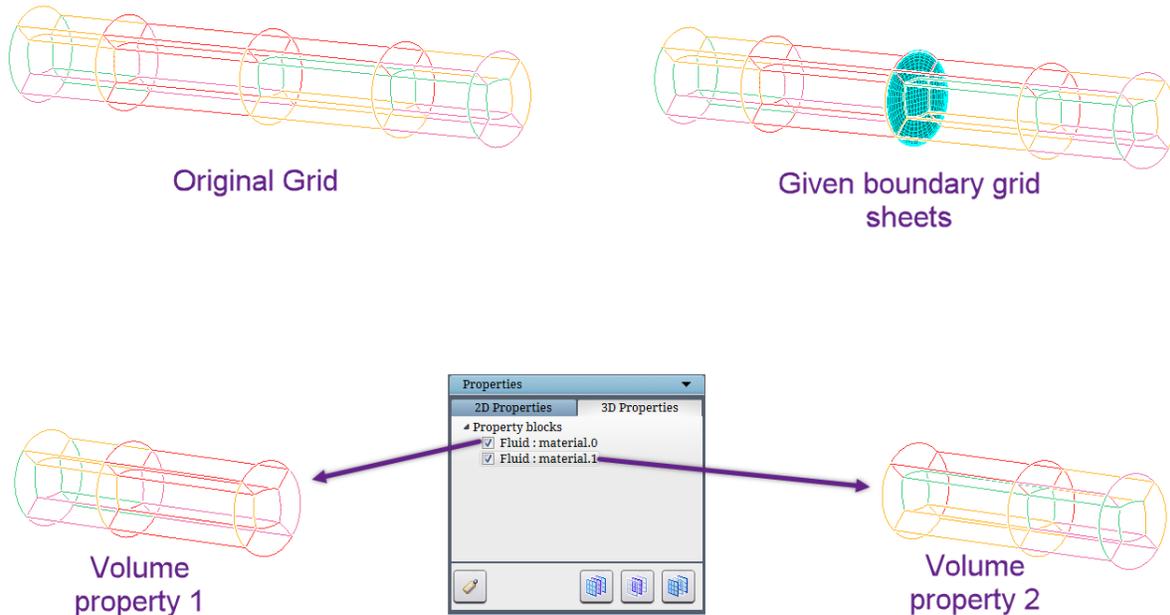


Figure 5.47: Create 3D property

If the input sheet does not have any internal boundary sheet or if it has volume sheets, then the UI will report a warning message.

## 5.4 Grid Probing Utility

### 5.4.1 Spacing

Sometimes, the user will need to test the smoothness of a grid or to measure 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 of the right side panel in the **Grid** tab, then moving your cursor to the line connecting two nodes and by clicking the left mouse button. Color display mode must be in the *color by sheet* mode in order to use *spacing* because it works only on the *current* grid sheet. When a grid line is clicked after pressing the *spacing* button, a green line connecting two nodes will be shown. The length of the line, or the node spacing, is displayed next

to the line as shown in Figure 5.48. The *spacing* button proves useful when clustering is used in the grid. It allows to precisely check the off-wall spacing near a boundary.

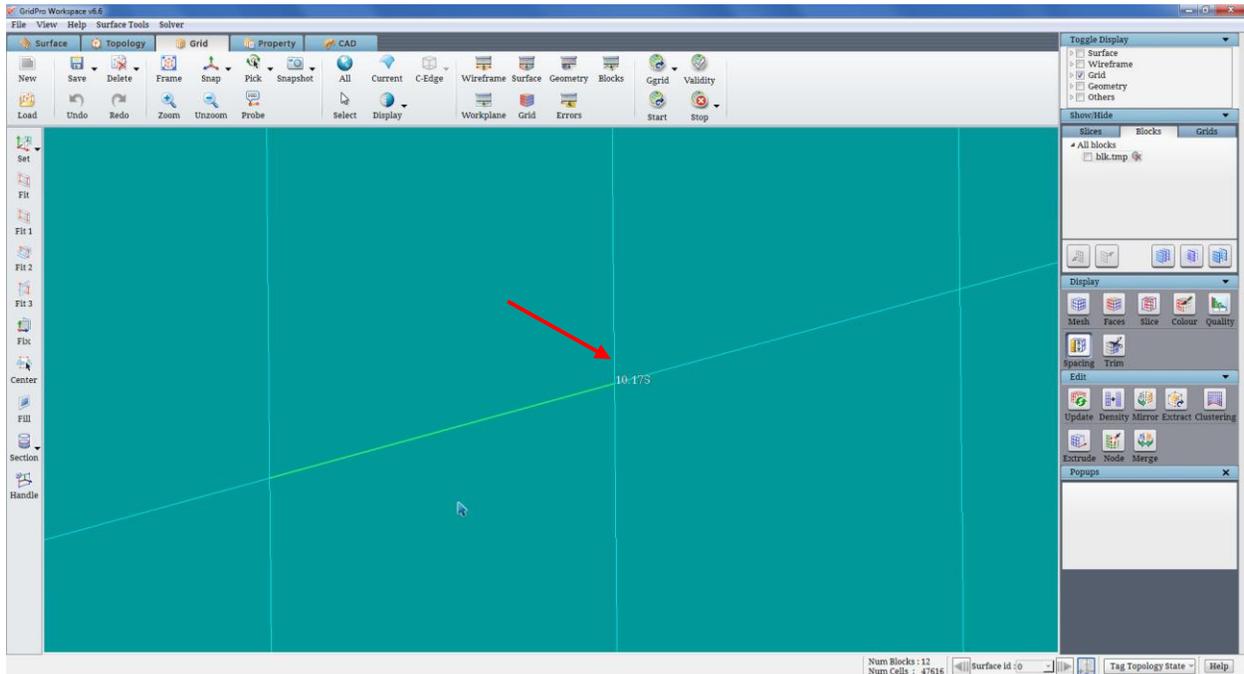


Figure 5.48: Spacing line with the distance pop\_up

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 won't be accurate in curvy regions.

## 5.5 Quality Check Parameters

Quality check is one of the most important post-processing tool in the mesh generation. Using a good quality grid will give more accurate results from the solver. GridPro has a quality checking tool called *Qcheck* which is accessible from both command line and from GUI. To check the quality of the grid, click on the *Quality* button in the *Display* section of right side panel of the **Grid** tab. Once clicked, Quality parameters dialog box opens as shown in Figure 5.51. 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) and also adds the topology faces of the folded cells to QFAIL group. A checkbox is provided for each of the options. User can check/uncheck while running the quality check depends on the preferences. By default, block orientation check is enabled.

If there are any left-handed blocks, it pops out a dialog box as shown in Figure 5.49. Clicking on 'yes' will realign the blocks to right-handed and overwrites the same grid file.

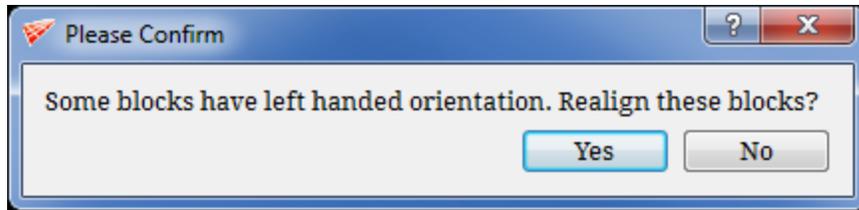


Figure 5.49: Confirmation dialog box

Display folded cells options loads the all the cells which are folded in the grid as blocks for better visualization. Add folded faces to QFAIL group option, takes the folded cells in the grid & finds out its corresponding topology faces and add them to the QFAIL group.

The report will only give the maximum value on each parameter but more often the interest is to see a range of values and also the location of those bad cells in the grid. To find the bad cells in a given range, click on the **Advanced** tab. Four parameters are provided as shown in Figure 5.50, with which the bad cell's location within a given range can be examined. They are Folds, Aspect ratio, Skewness and Warpage. You are flexible enough to check either one of the parameters or all the parameters in a single check by enabling or disabling the checkbox before each parameter. Once it is enabled, the minimum and maximum range of that parameter can be given. The cells which fall under that range will be displayed in the UI, with a new grid consisting of only these cells under the name, 'qcheck.hex.grd'. Note that, for each parameter, the range of min and max values varies which are already displayed in their respective fields.

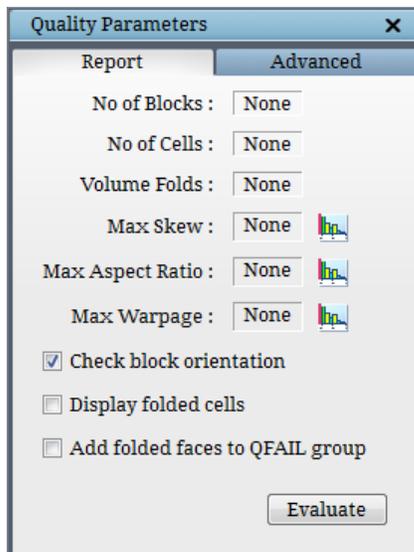


Figure 5.51: Quality check dialog box

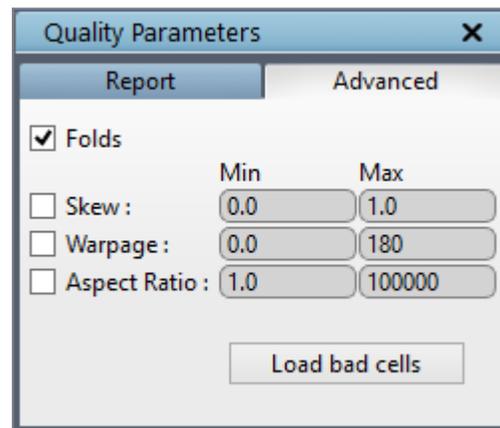


Figure 5.50: Advanced tab dialog box

- **Folds**

Folds are defined as the inconsistency in the positioning of the grid nodes of a cell, with respect to the neighboring cells.

- **Aspect ratio**

Aspect ratio 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 neighboring blocks. The aspect ratio value can range from 1 to infinity. Ideally, it should be equal to 1 to ensure best results. By default, it takes the range from 1 to 100000.

- **Skewness**

Skewness is always considered as the aptest measure of the mesh quality. Skewness determines how close a cell or face 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 degenerate cell (worst). A skewness up to 0.9 is widely accepted for computation.

- **Warpage**

The warpage is 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 faces to define the warpage angle of the cell. The warpage angle is measured from 0 to 180 degree.

## 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 how the surfaces are selected as *current*, grids can also be selected as, *current*. User can toggle between grids, grid blocks and sheets from the *Show/Hide* section located in the right side panel below *Toggle display section* as shown in Figure 5.52. Local toggle section has three tabs, one for *Grids*, one for *Grid blocks* and the other for *Grid slices*. By default grid slices tab is visible, Click on **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 which is 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 its respective sheets if any, while in *blocks* tab it is performed on the blocks of the *current* grid and not on its respective sheets and in *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 delete icon next to the grid name. Once the grid is deleted, it cannot be retrieved back. If it has to be retrieved, it has to be loaded again using *Load* option in *Global dock*.

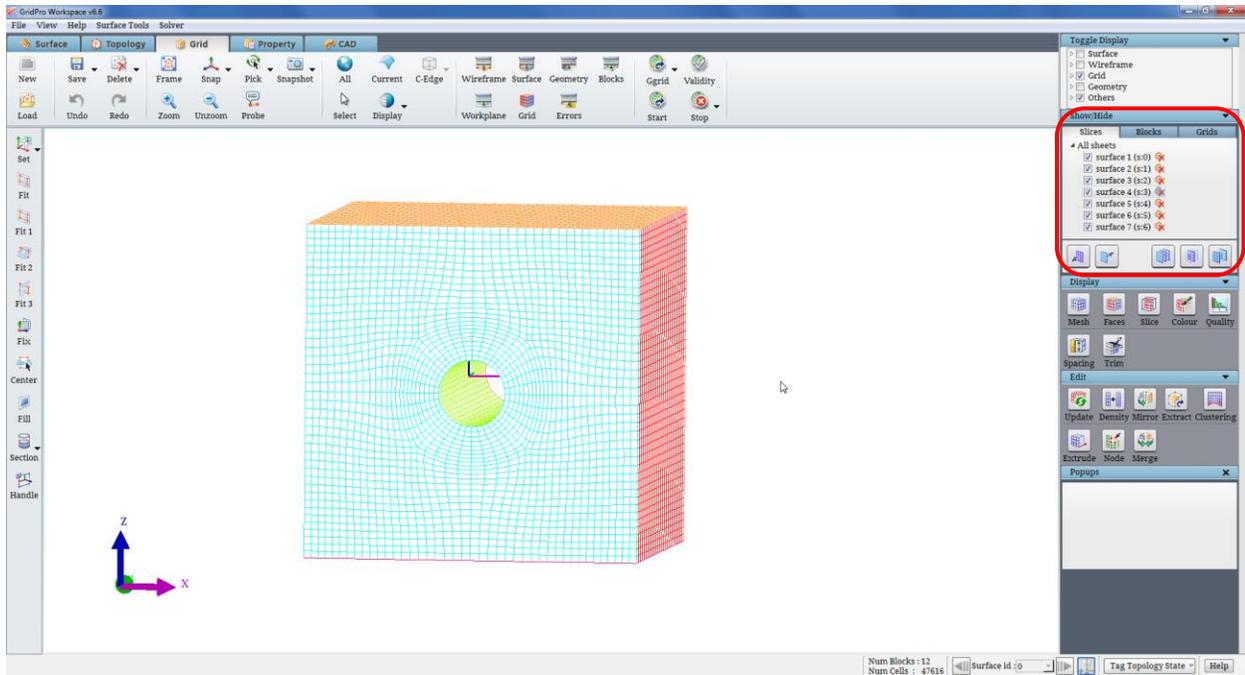


Figure 5.52: GUI highlighting the Show/Hide section

As discussed before, *current* grid can be deleted using **Current grid** option under **Delete** button in **Global dock**. Below the grid names list, there are five buttons placed as shown in the Figure 5.53. The First two buttons are enabled for only **Grid slices**. The rest three are enabled for all the three tabs. With respect to **Grids** tab, the third button i.e. **Show all Grids**  button switches on the 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 which are displayed will be hidden and grids which were hidden will be visible. Once the **Current** grid is selected, user can create as many grid sheets as he wants. The grid sheets can be toggled from **Slices** tab. It will show only the sheets belonging to the *current* grid. Similar to 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. Using the **delete** icon next to the grid sheet name, the respective grid sheet can be deleted.

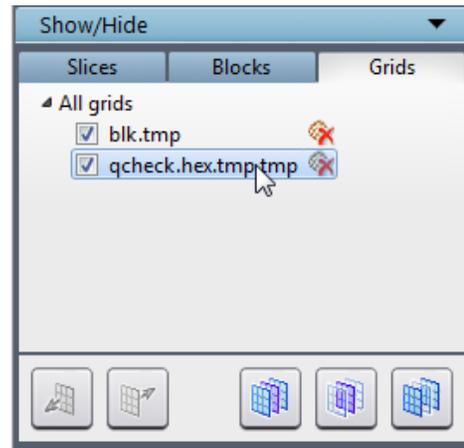


Figure 5.53: Grids tab

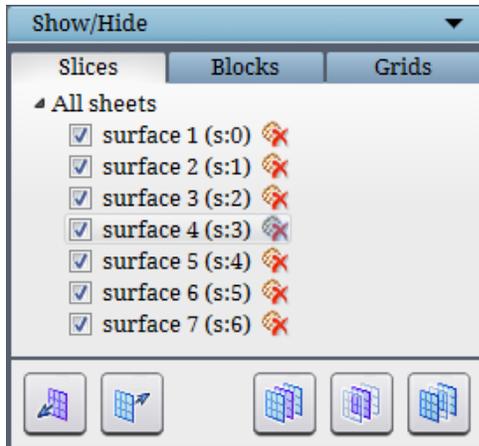


Figure 5.54: Slices tab

The first two buttons i.e. forward step and backward step, below the grid sheet list allows the user to scroll the *current* sheet through all the positions within that block. Forward  button steps the sheet forward, while backward  button steps backward. It is useful when a sheet is created using *slice* button, and the user want 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 current grid and will show only *current* sheet of *current* grid. The fifth button, *toggle all grid sheets*  will flip the visibility of all the grid sheets.

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 works 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 will show only *current* grid blocks. The fifth button is *toggle all grid blocks*  will flip the visibility of all the grid blocks.

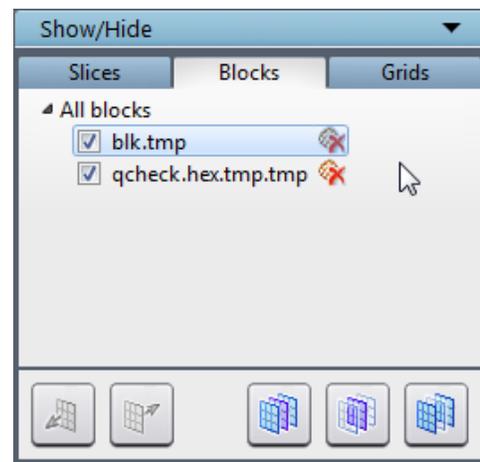


Figure 5.55: Blocks tab

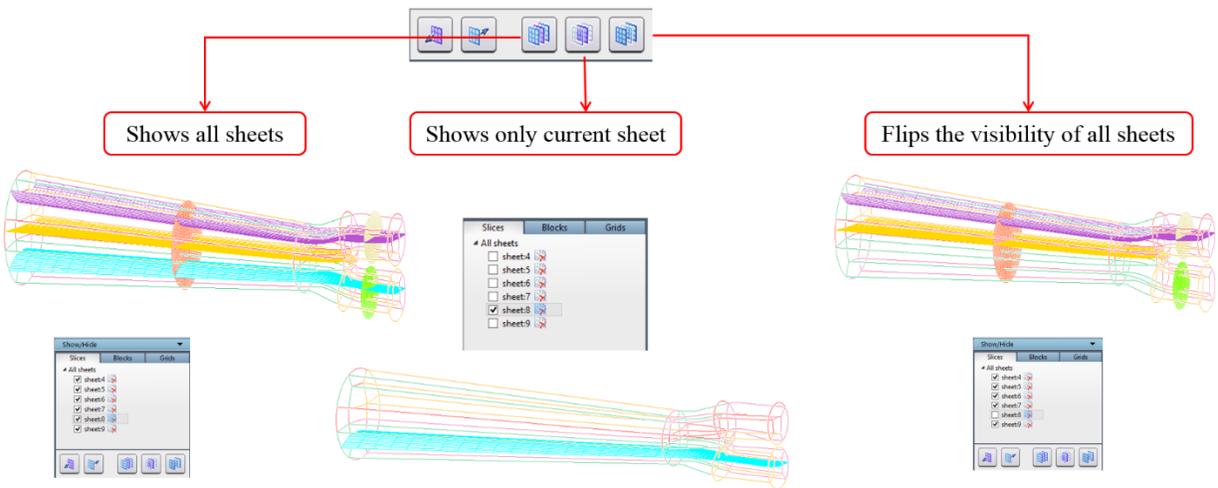


Figure 5.56: Toggling options

NOTE: Switching off the grid sheets or grid blocks in toggle display section will also switch off the corresponding object in the **Grid sheets** section but the 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 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 **Trim** located in the **Tools** section. Clicking on the **Trim** button will open a dialog box as shown in Figure 5.57. User can select the trimming tools for blocks or for faces by selecting one of the tabs in **Trim** dialog box. Select **Blocks** for trimming blocks and **Faces** for trimming sheets. By default, **Blocks** have been selected for trimming.

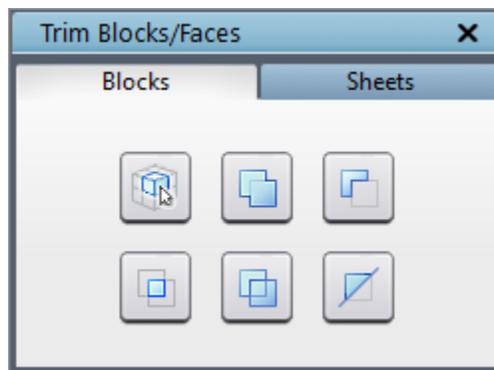


Figure 5.57: Trim dialog box

## 5.7.1 Blocks

Each grid block has a center, 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 center of each block is the center of mass of each block and is represented by a small square. The skeleton of a block is the lines connecting the indexed center of the block with the centers of the block's bounding edges (2D) or faces (3D). Each block skeleton is drawn in its own color. When collected, each block skeleton forms a skeleton for the entire multiblock grid. The skeleton represents the minimal amount of information required to show a broad representation of a multiblock 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 5.58 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 and/or a collection of skeletons. However, only one of these display forms can be used at a time in order to avoid the screen clutter.

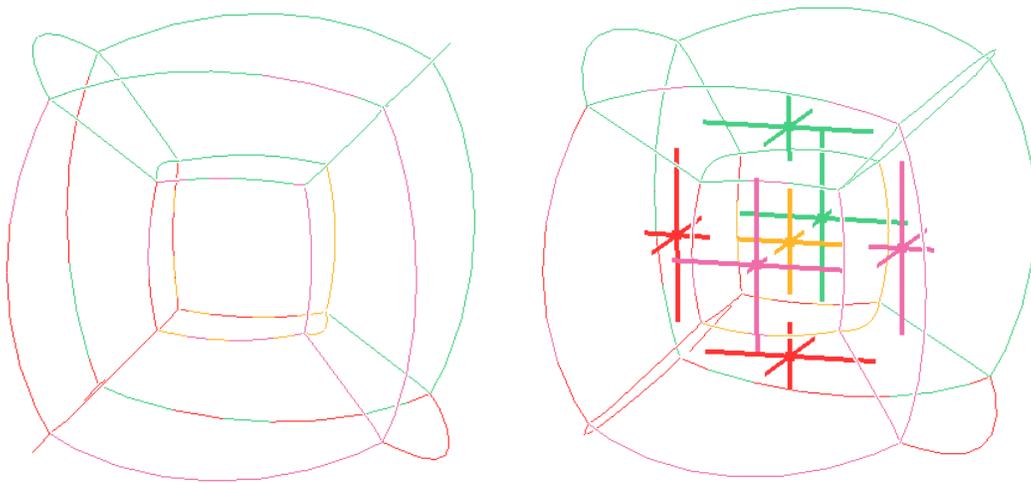


Figure 5.58: Blocks outline and skeleton of a sphere

Grid skeleton will appear whenever the user tries to do trimming operations on the blocks. If add or subtract or intersection in *Trim*  $\rightarrow$  *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*  $\rightarrow$  *Blocks* deals with activating and deactivating blocks. In order to deactivate a block, deactivate the skeletal center of the block. By default, all the blocks will be on. If the block's center 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 slices of the grid which do not propagate beyond the active blocks. If a block

skeletal center is a filled, colored square, the center is activated. If a center is a hollow, colored square, the center is deactivated. The *Mesh* button places a mesh on the exterior boundaries of the activate blocks. The result of pressing *Mesh* is the display of the active blocks as if they were a solid object. The propagation is limited to the active blocks.

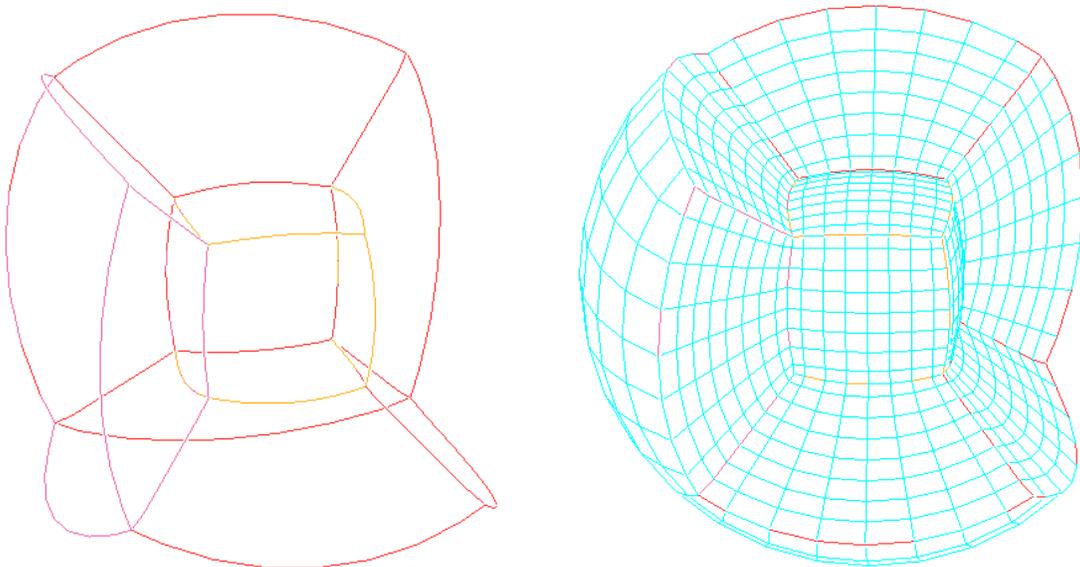


Figure 5.59: Trimmed blocks and trimmed blocks with grid cells

Because most grids will contain numerous blocks, 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 centers 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 centers 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 centers that have to be deactivated. The  button is an intersection operation. Similar to the  and  operations, the  operation allows the user to create a purple box and enclose it around centers. Any center which is not enclosed by this box will be deactivated. Activated centers enclosed by this box remain activated and deactivated centers enclosed by this box remain deactivated. The  button, when depressed, simply activates all the centers. The  button, when pressed, activates all the deactivated centers and deactivates all the active centers.

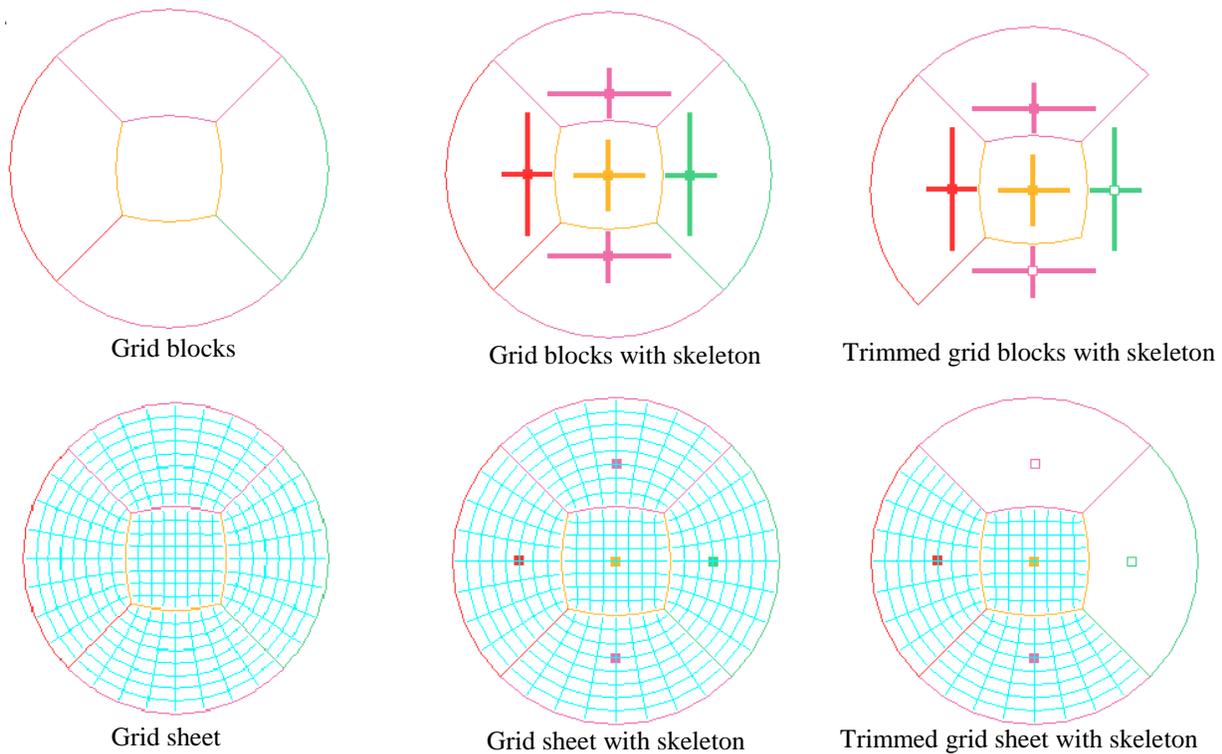


Figure 5.60: Grid trimming

## 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 5.61. When the trim mode is set to *Sheets*, all the trimming buttons works 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 have 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 centers of the given sheets. The other buttons are the same and work the same way for the sheet faces instead of blocks depends on the mode set in the trim section.

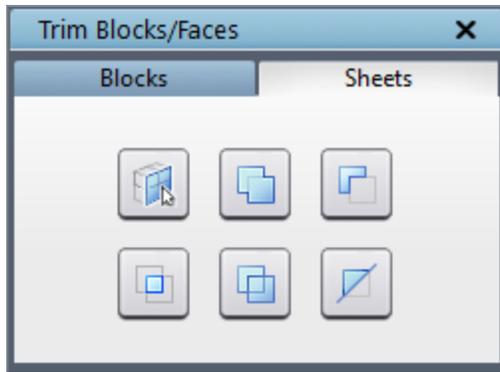


Figure 5.61: Sheets tab dialog box

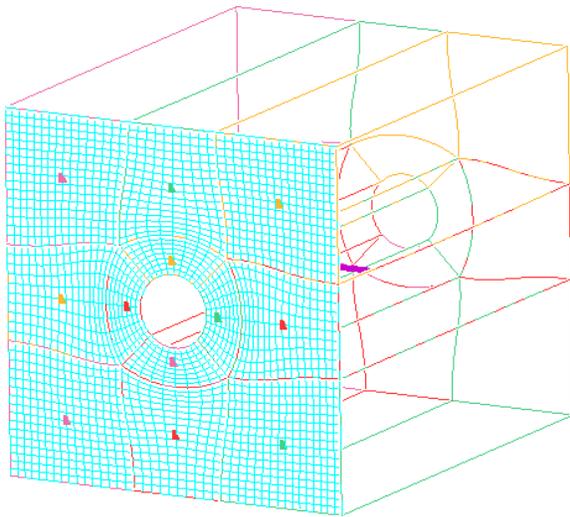


Figure 5.63: Skeleton for the grid sheet

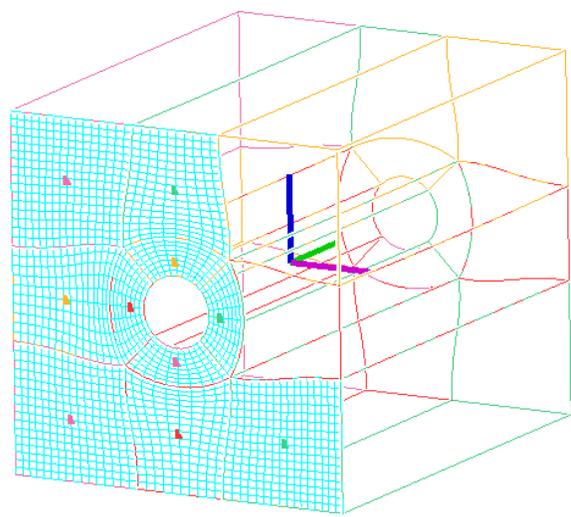


Figure 5.62: After trimming two faces of the grid sheet

## 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 either set it on the surface which will be reflected in the grid or directly 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 **Solver** menu in the top menu bar. Each solver has a separate list of properties, depends on the selected solver the corresponding property list can be seen in the property drop-down 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 do as well as what each operation displays in the drawing area.

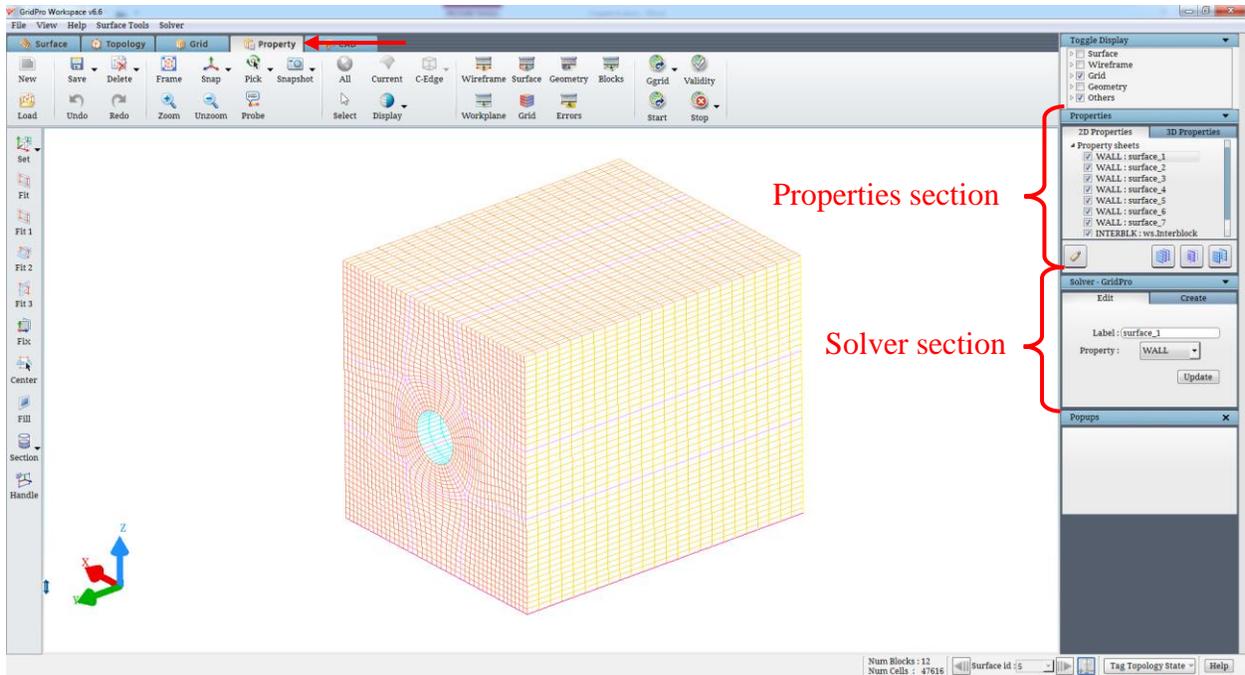


Figure 6.1: Property tab

## 6.1 Properties Section

In *properties* section, which is located below *the toggle display* section in **Properties** tab on the right side panel, the properties assigned to the grid sheets and grid blocks can be viewed. *Properties* section is split into two tabs i.e. 2D properties and 3D properties as shown in Figure 6.2 & Figure 6.3.

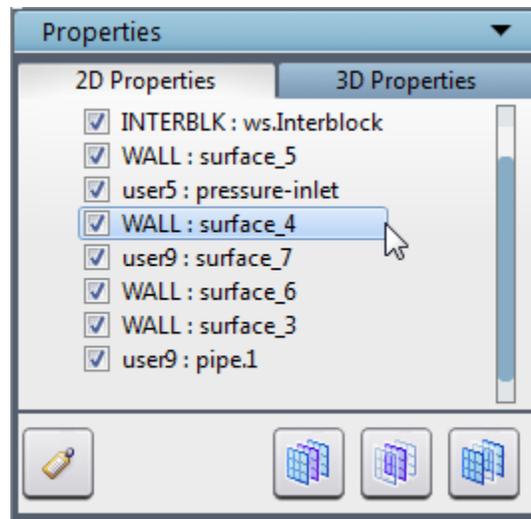


Figure 6.2: 2D properties tab

In the *2d properties* tab, assigned property sheets are shown and in the *3D properties* tab, assigned property blocks are shown. It also gives an access to toggle on/off a property. By default,

each grid has two 2D properties such as 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.

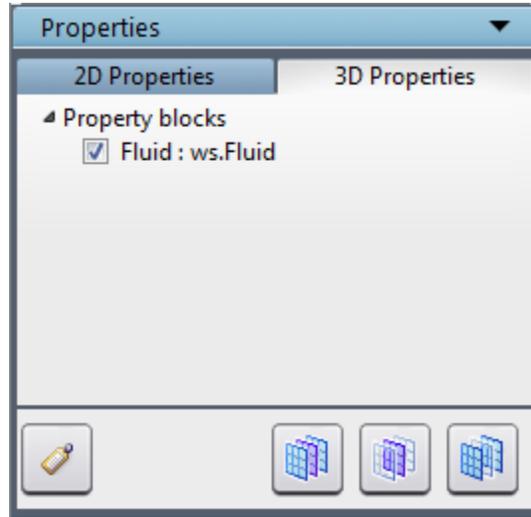


Figure 6.3: 3D properties tab

Note that there are two names for every property sheet in figure 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 **2D properties** tab is selected, properties are assigned to grid sheets and in **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 which is highlighted with blue shade is the *current* property displayed as a sea blue color 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.

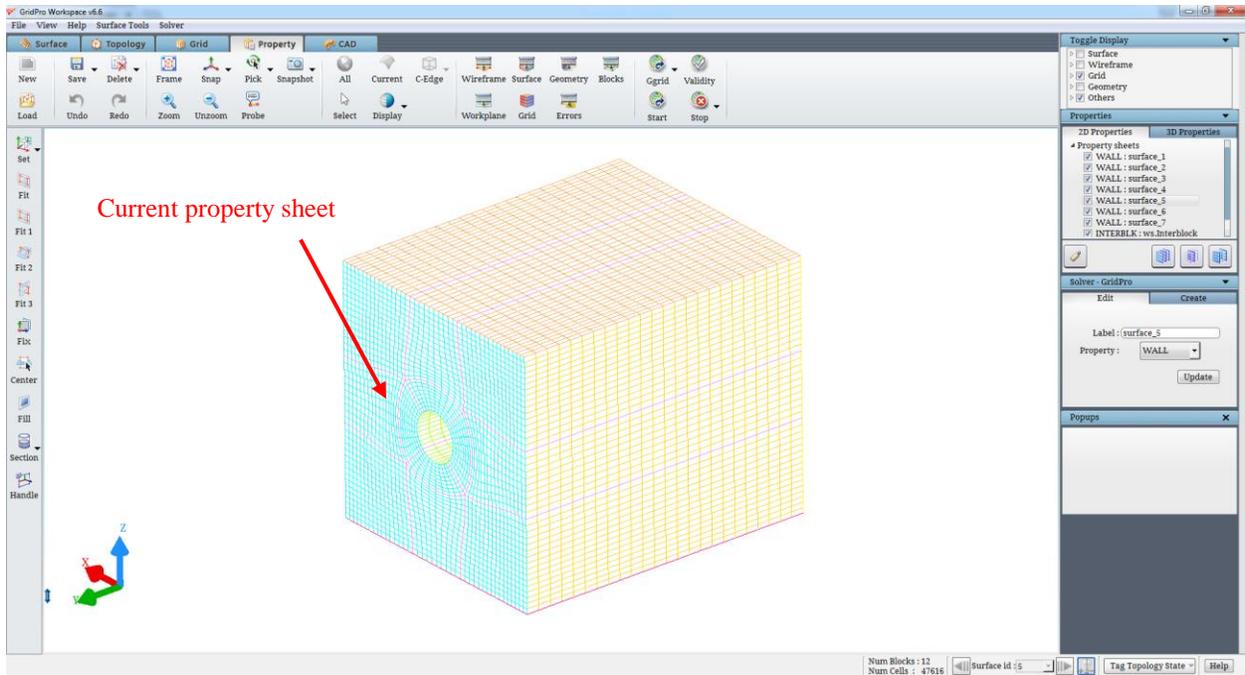


Figure 6.4: 2D properties section with Inlet as current property

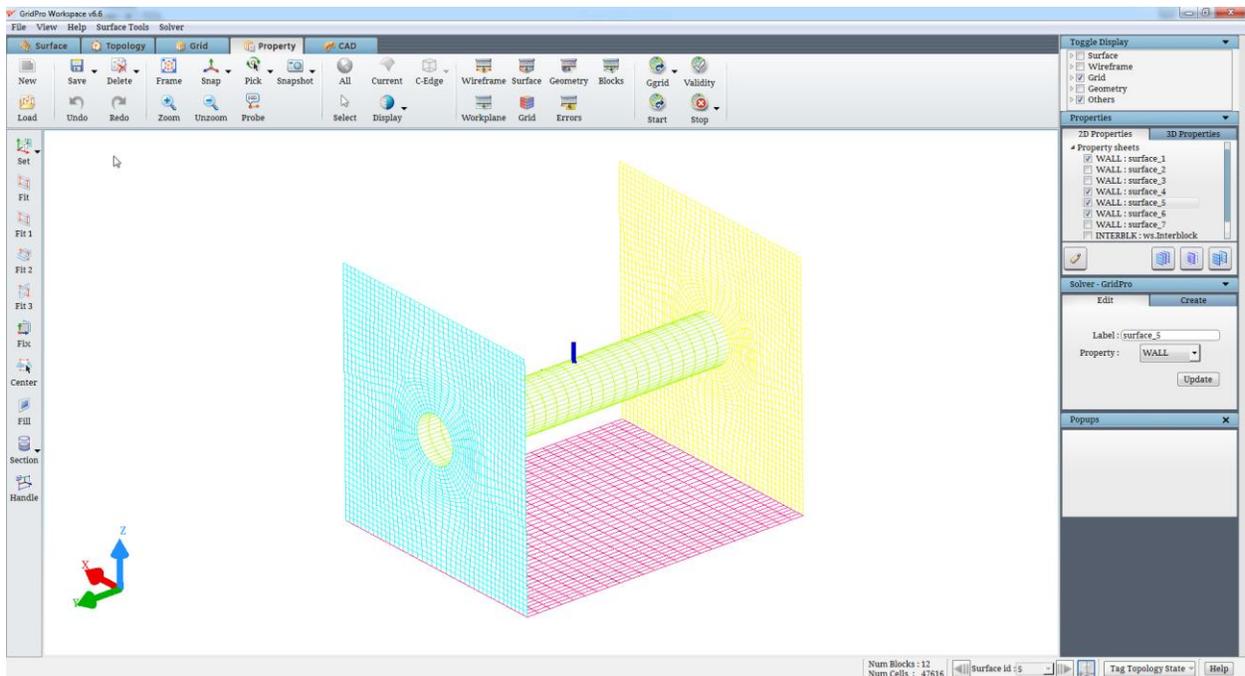


Figure 6.5: Property sheets switched off in the 2D properties section

The first button located at the right side bottom 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.

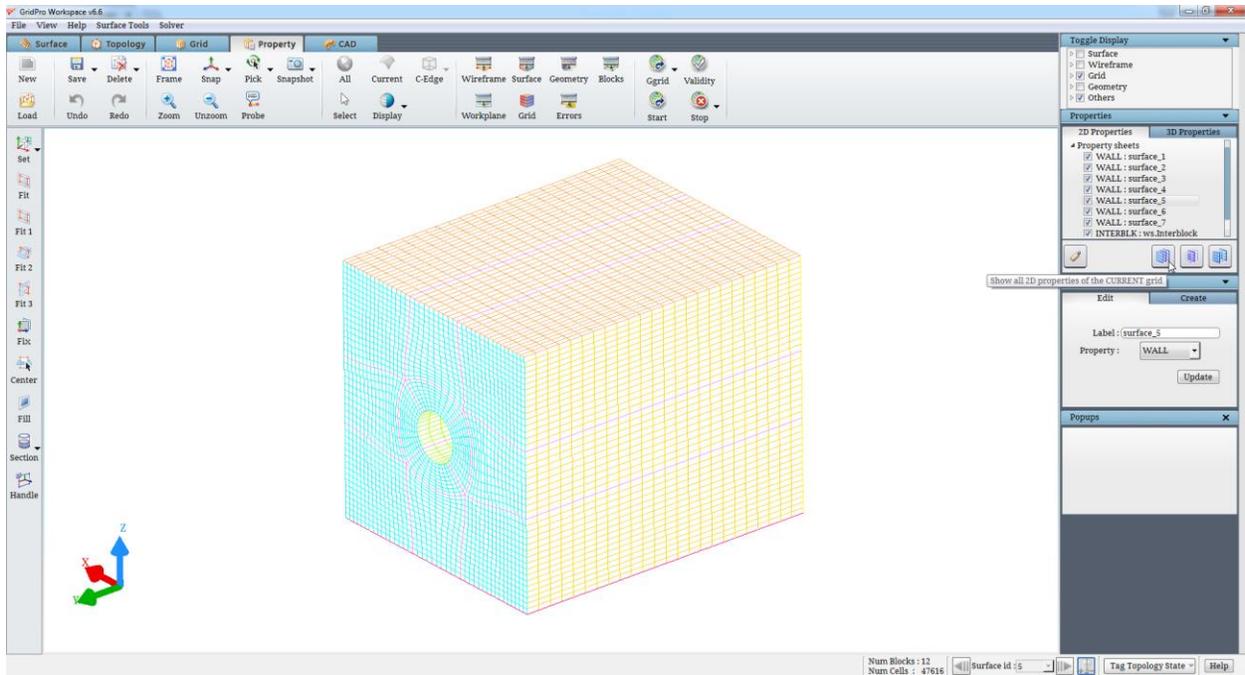


Figure 6.6: After clicking on Show all property button

The second button i.e. **Show only Current**  button will hide all other properties except **current** property.

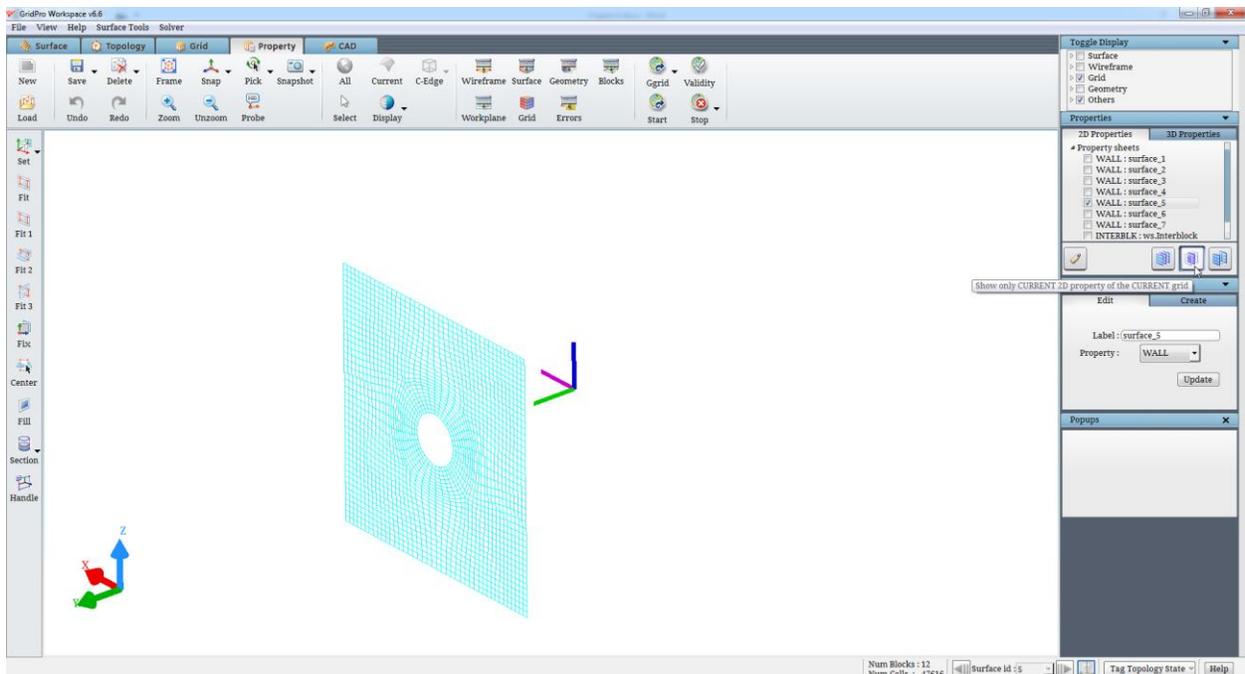


Figure 6.7: After clicking on Show only current property button

The third button i.e. **toggle all property**  button will flip the visibility of all the properties in that tab.

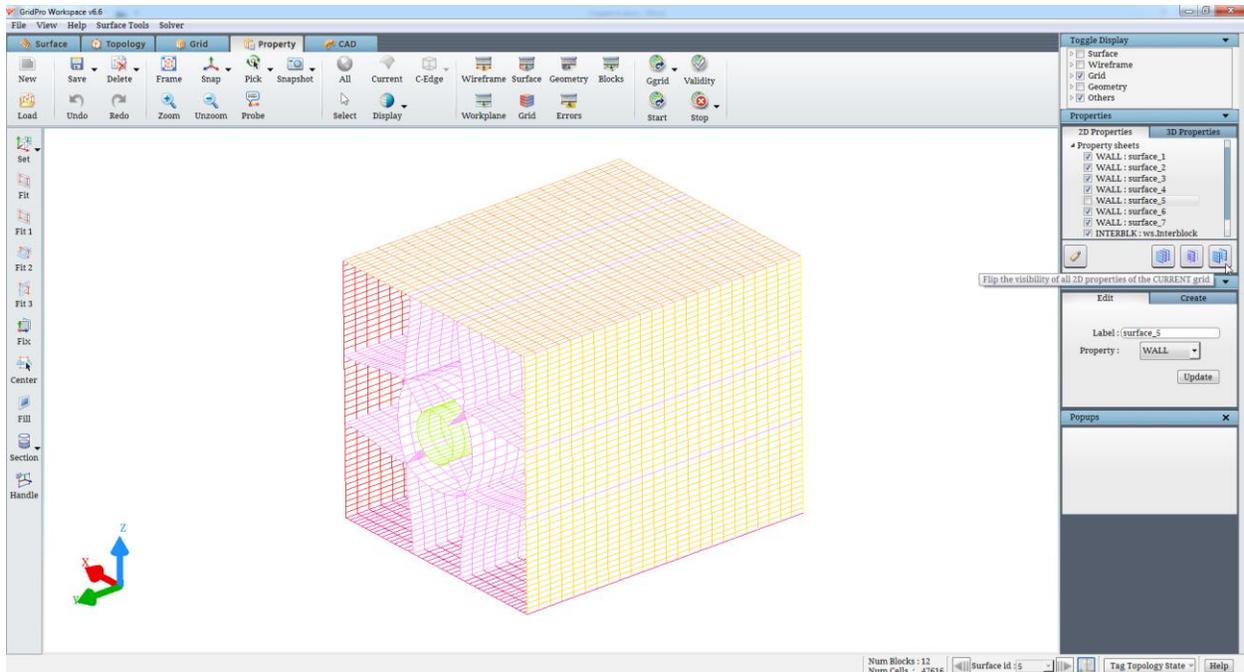


Figure 6.8: After clicking on Toggle all property button

In the **3D properties** tab, all these buttons work in a similar way 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 id's. 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.

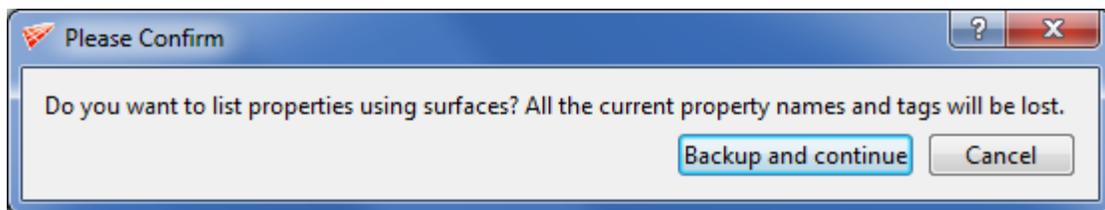


Figure 6.9: Labels Confirmation dialog box

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. Firstly, a CFD solver format has to be selected. Various solver formats are available in the drop-down of *Solver* menu in the menu bar as shown in Figure 6.10. By default, GridPro format is selected.

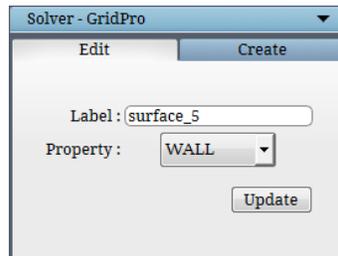


Figure 6.11: Solver section

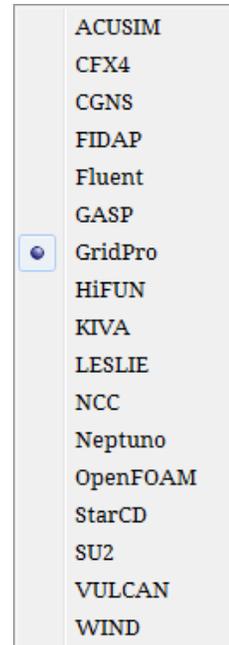


Figure 6.10: List of solver formats

Each format has unique properties. The list of property for each solver can be viewed in the field next to *the property* in *solver* section. Figure 6.12 provides an example of properties that are listed under property drop-down list.

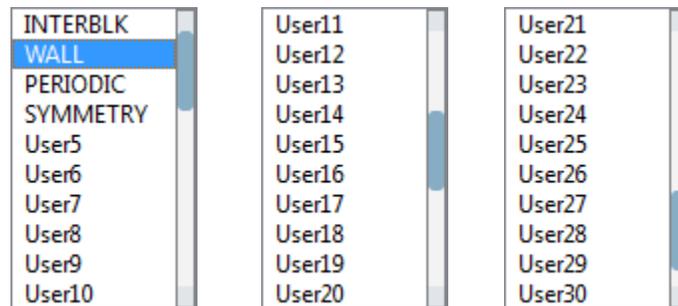


Figure 6.12: List of properties for the GridPro format

The **Property** tab makes extensive use of grid skeletons for the purpose of efficient property assignment. As previously mentioned, the skeleton of a block is the lines connecting the indexed center of the block with the centers of the block's bounding edges (2-d) or faces (3-d). Each block skeleton is drawn in its own color and represents the minimal amount of information required to show a broad representation of a multiblock grid.

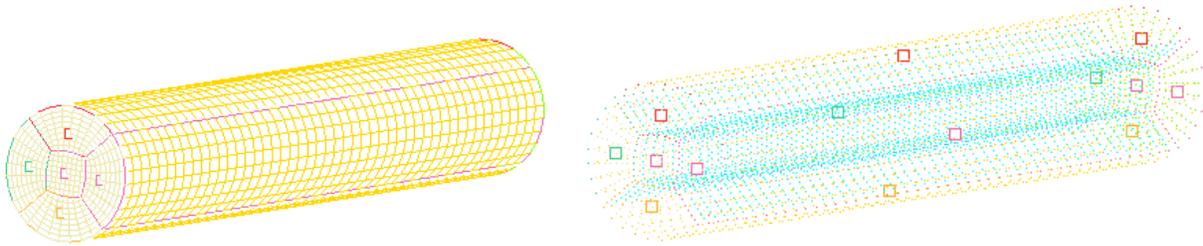


Figure 6.13: Block skeleton

Switch to the 2D properties tab in the property section, after selecting the desired solver format in *solver* menu. Remember, in **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 *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.

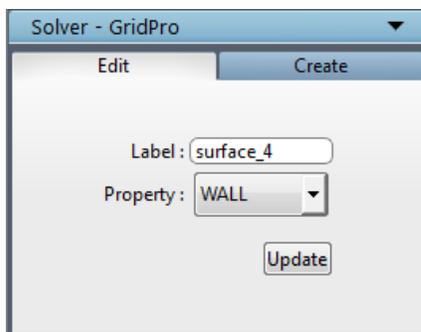


Figure 6.14: Edit tab

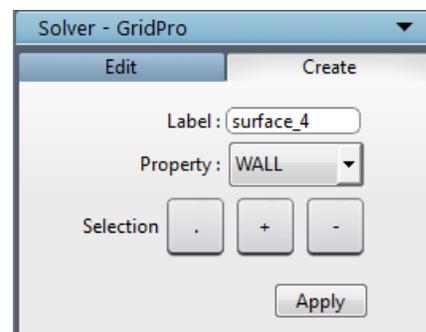


Figure 6.15: Create tab

New properties can be created by using the options available in the *Create* tab under *Solver* Section as shown in Figure 6.15.

Select the property to be assigned in the property field in the *Create* tab of Solver section. Give the desired label for the property. The four buttons in the selection row (the *Create* , *toggle* , *add* , *subtract* ) 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 center of the face skeleton will be deactivated. Conversely, if the face skeleton center has to be activated, the face has to be displayed first. Firstly, select the desired property and add its corresponding label and Click on *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 center of that face's skeleton.

Depressing the  button will display the skeletons of all the faces and allows the user to assign/un-assign the current property to any one of the faces by clicking (with the left mouse button) on the center of the face's skeleton. Assigned face centers will be displayed as solid blocks and unassigned face centers will be displayed as transparent blocks. Depressing the  button will display the skeletons of all faces which do not contain the current property and allows the user to assign the current property to many faces by dragging a red box (with the right mouse button) over their skeletons' centers. Depressing the  button will display the skeletons of all faces which contain the current property and allows the user to un-assign the current property from many faces by dragging a red box over their skeletons' centers. Once the property is unassigned from the faces, the un-assigned faces will be automatically 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. Here, the *2D properties* tab has to be switched to *3D Properties* tab. The chosen property format determines the list of 3d properties to be displayed, when the drop down next to *properties* in *solver* section is clicked.

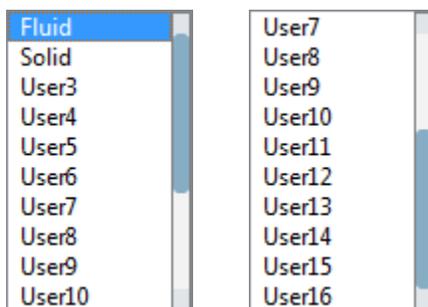


Figure 6.16: 3D Property list of PDC format

- **Right-click menu**

When you *right-click on an edge of a property block* a menu appears that is similar to the right-click menu of an edge of grid block in Grid tab as shown in Figure 6.17. All the options are discussed under section 5.1.3.

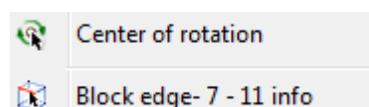


Figure 6.17: Right-click menu of grid blocks

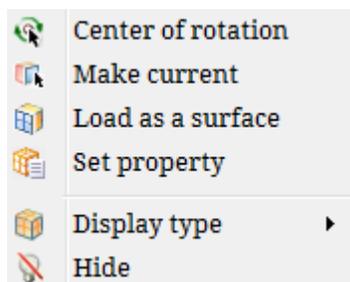


Figure 6.18: Right-click menu of grid sheets

Similarly, right-click menu for property sheet is also similar to the right-click menu of grid sheet in grid tab as shown in Figure 6.18. All the options except *Set Property* are explained in section 5.1.3. Set Property option can be used to 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.

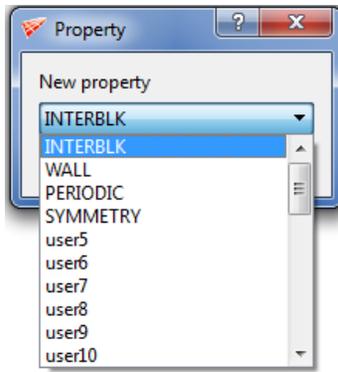


Figure 6.19: Set property dialog box

Create 2D/3D Property	Edit 2D/3D Property
<ol style="list-style-type: none"> <li>1. Select the desired Property from drop down</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, 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 in order to differentiate each, property labels are introduced recently.

Another major 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 chapter 7 in the User\_Guide\_for \_TIL manual for more information on how the block information is written.

```

#2 Blocks
## 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)

```

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 which 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 and STEP format. 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 toggling functions provided. The Geometric Faces section works similar to Show/Hide and Properties section.

Under the Tools section, you are provided with the tools to repair and mesh the geometry. Section 7.2 explains about the tools in detail.

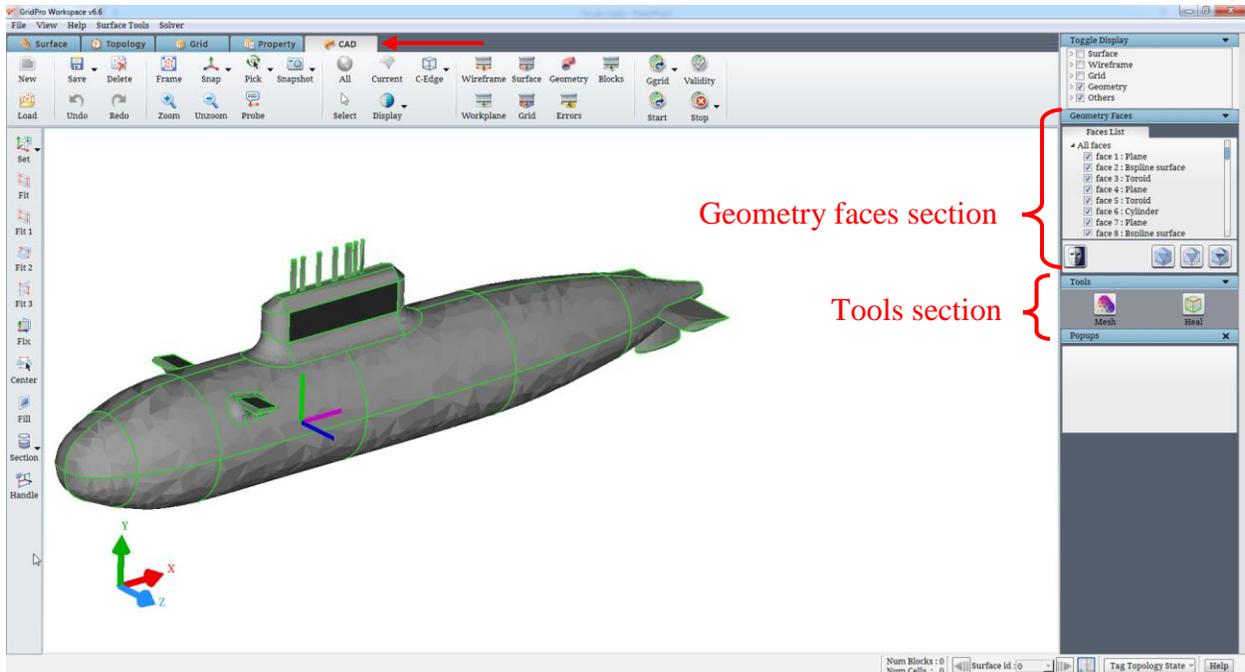


Figure 7.1: CAD tab

## 7.1 Geometry Faces

In *geometric faces* section, which is located below *the toggle display* section in **CAD** tab on the right side panel, the individual face definitions of the geometry can be viewed. Figure 7.2 shows the Geometric Faces section.

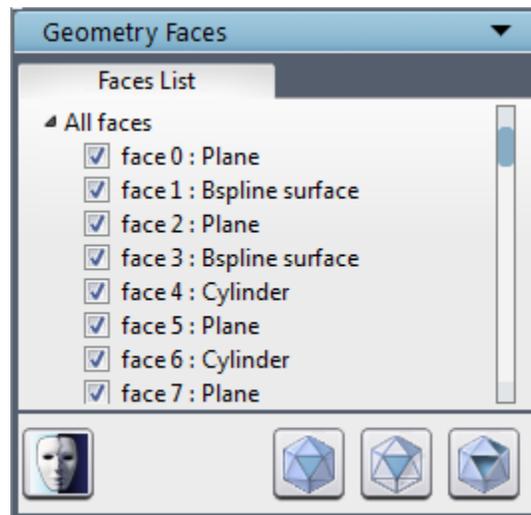


Figure 7.2: Geometry Faces section

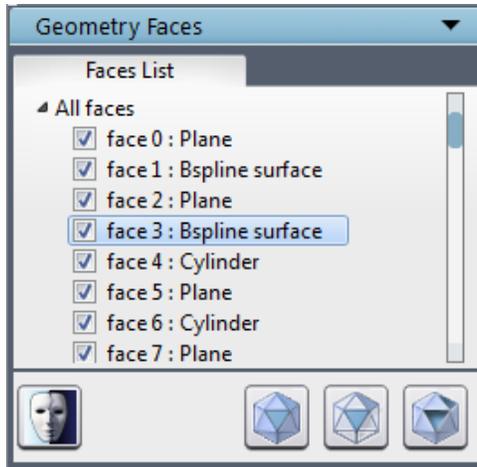


Figure 7.3: Geometry Faces section with as a current face

The face which is highlighted with blue shade in the *Geometry Faces* section is the *current* face. Unlike other tabs, there are no different color codes to differentiate the *current* geometry face. Any face can be selected as a *current* face by clicking on that face name. To hide a face uncheck the box before the face name. If the box is checked, the face will be visible.

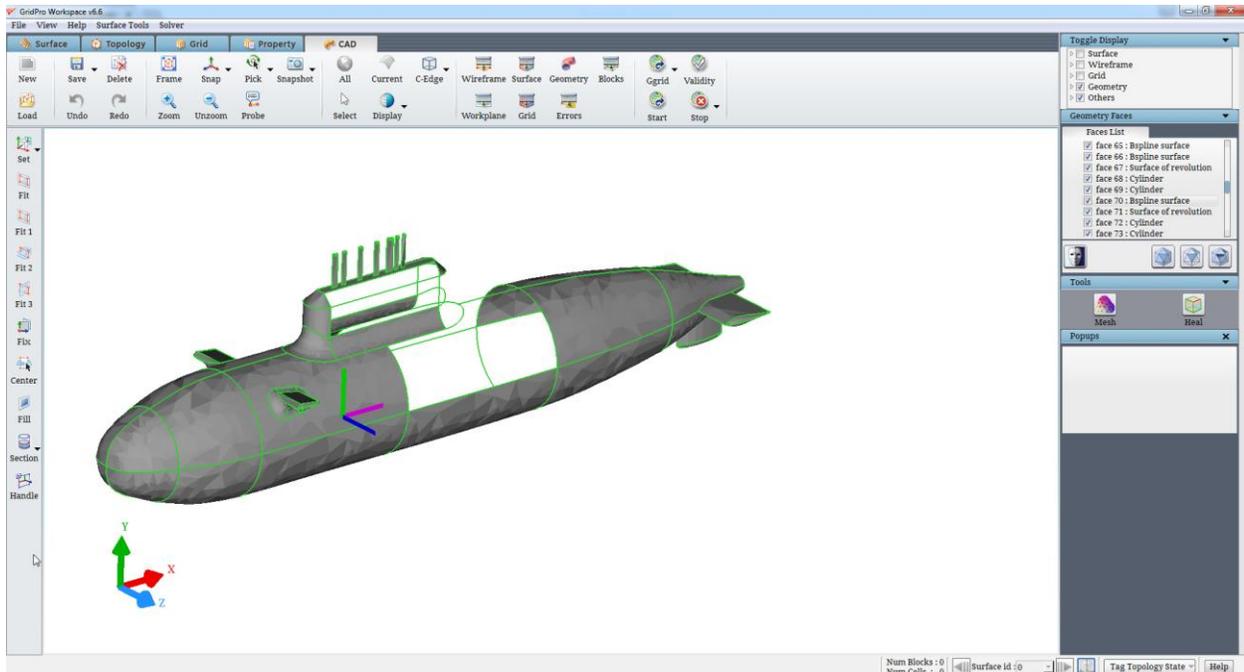


Figure 7.4: After a few of the Geometry faces being switched off

The first button located at the right side bottom of the *Geometric Faces* section i.e. *show all faces*  button will switch on the visibility of all faces in the list.

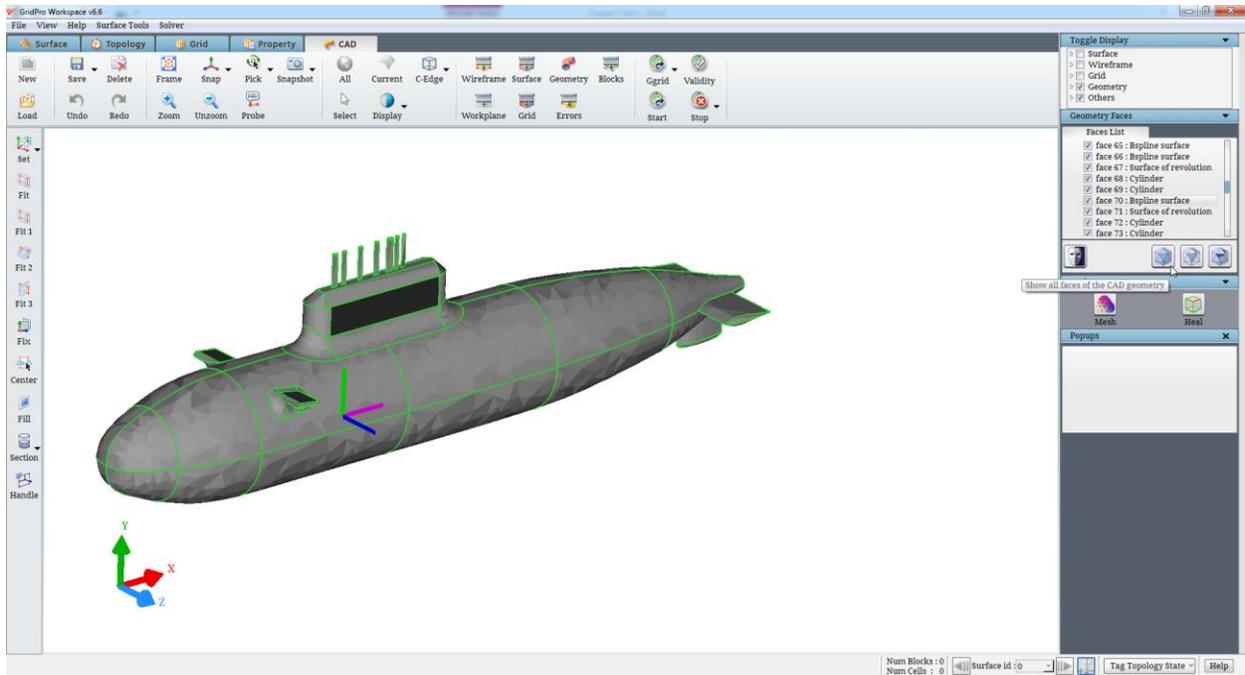


Figure 7.5: After clicking on Show all face button

The second button i.e. **Show only Current**  button will hide all other faces except the *current* face.

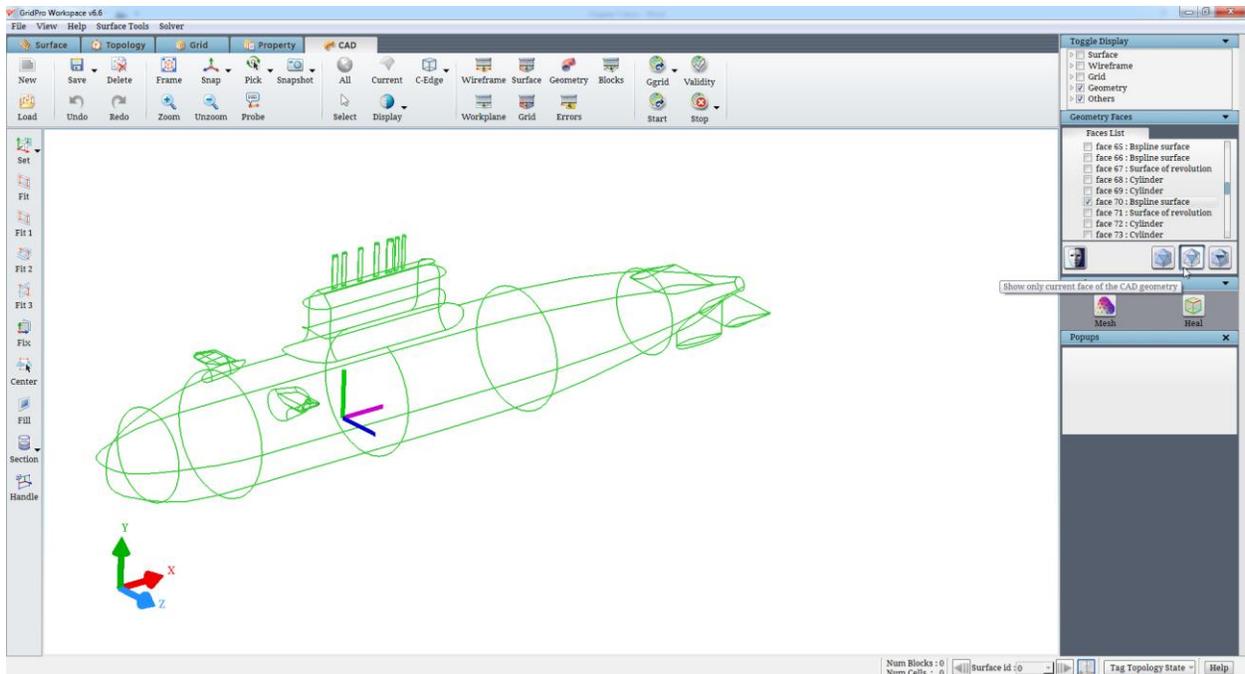


Figure 7.6: After clicking on Show only current face button

The third button i.e. **toggle all faces**  button will flip the visibility of all the faces in the list.

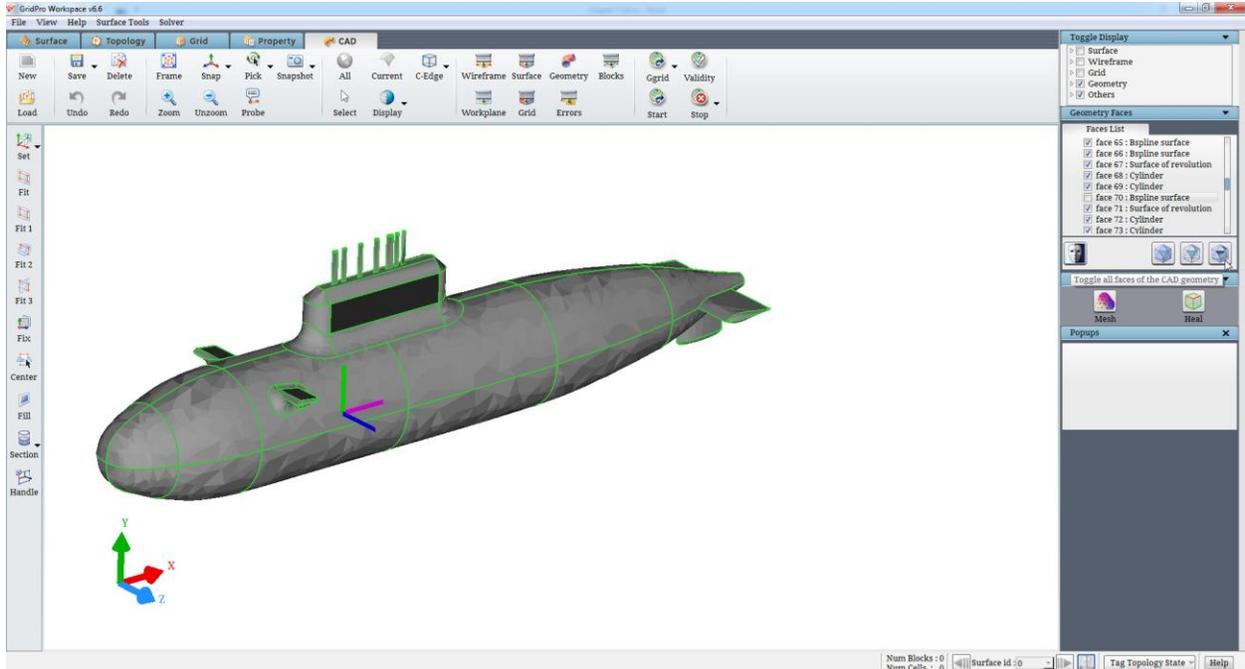


Figure 7.7: After clicking on Toggle all faces button

The Hide button  provides advanced control to toggle the faces. You can hide the desired faces by picking on the faces one by one after switching ON the Hide button. Switch OFF the button manually once the desired faces are hidden by clicking on it again.

## 7.2 Tools Section

There are two buttons provided under Tools section; Mesh and Heal. Mesh creates a surface triangulation mesh for the given geometry which can be exported to other surface formats such as IGES, STEP, STL and TRIA. Heal helps in fixing basic surface repairs.

### 7.2.1 Mesh

Once the Mesh button is clicked, a dialog box opens as shown in Figure 7.8.

Two kinds of mesh can be created for each geometry. One is basic mesh which creates a mesh with a minimum number of triangles. Maximum sag and turn angle values are used to create a basic mesh. Maximum sag represents the maximum allowable distance between a facet and its underlying surface. The default maximum facet sag is set according to the size of the model. Turn angle specifies the maximum angle that an edge facet can turn through before it is too large and must be refined.

The second one is Quality mesh, which has advanced options to control the mesh quality. A brief explanation of all the options is as follows. Maximum edge length represents the maximum length of a triangle edge. Minimum edge length represents the minimum length that a triangle edge can take, it will not create edges lower than the specified value.

Fineness ratio is the ratio which defines how fine the mesh should be, near the edges of a face compared to the middle of a face.

Grading ratio is the ratio which defines how slowly or fast, the fine mesh at the edges should be transformed to a coarser mesh at the middle of a face.

Facets/Edge specifies the minimum number of faces to be created on each edge and Facets/Curvature specifies the minimum number of faces to be created on each curvature region.

Refine close edges checkbox decides whether to refine the mesh around the smaller edges of the geometry. Checking the checkbox enables the refinement factor option where the number of refinements can be mentioned.

Once the desired mesh is created, it can be exported using Export options at the bottom of the dialog box. It can export the mesh to STL, Tria and 2D linear file format. You can also choose to export and load it as a GridPro surface in a single step.

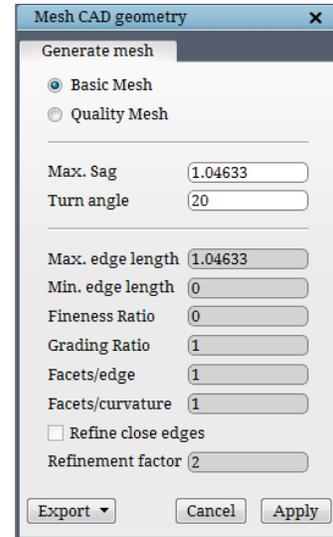


Figure 7.8: Mesh dialog box

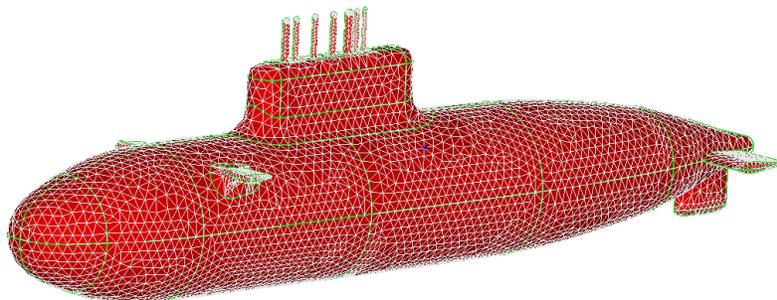


Figure 7.9: Mesh created on a geometry

## 7.2.2 Heal

Heal provides the options to fix the basic surface repairs such as small edges, spot faces, strip faces and sew faces.

Once the heal button is clicked, a dialog box opens as shown in Figure 7.10.

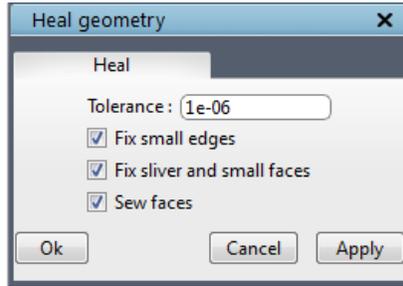


Figure 7.10: Heal dialog box

Small edges represent all the edges whose edge length falls under the given tolerance level.

Sliver faces are the faces whose area is less than the limiting area  $W*((P/2)-W)$ , where  $W$  is the width (tolerance) specified by the user, and  $P$  is the face perimeter.

Small faces are the faces whose edges are below the given tolerance and the area of the face is less than the square of the specified length of an edge (tolerance).

Sew faces, in other words, stitch two disjoint faces whose gap falls under the given tolerance value.

- **Right-click menu**

When you right-click on a geometry face, a menu appears as shown in Figure 7.11. **Center of rotation** picks the right-clicked position on the geometry as the center of rotation of the screen. Clicking on **Hide face** will hide the selected face of the geometry.

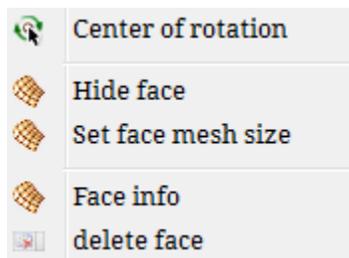


Figure 7.11: Right-click menu of geometry

**Set face mesh size** helps in setting the mesh size of the face locally. Clicking on the options opens a dialog box as shown in Figure 7.12. The input value cannot be more than the max sag value mentioned in the 'Mesh' dialog box. A warning message pops out as shown in Figure 7.13 and reset it to the original value. This value will be considered only when you create a surface mesh next time. You can also reset the modified face mesh size for the selected face or all faces using the reset drop-down menu.

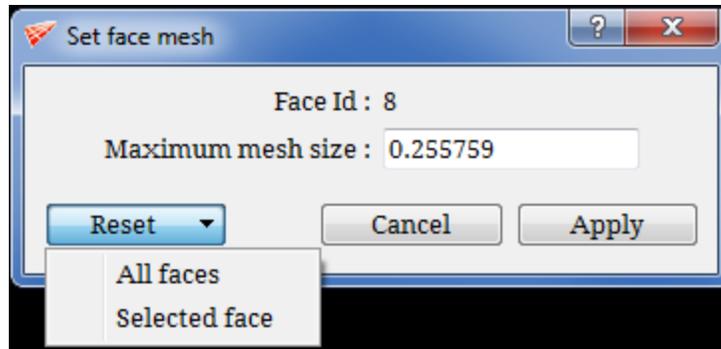


Figure 7.12: Face mesh size dialog box

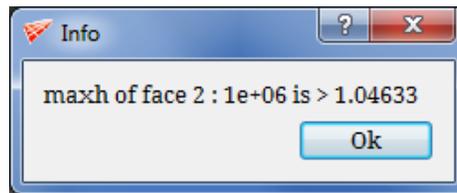


Figure 7.13: Warning message

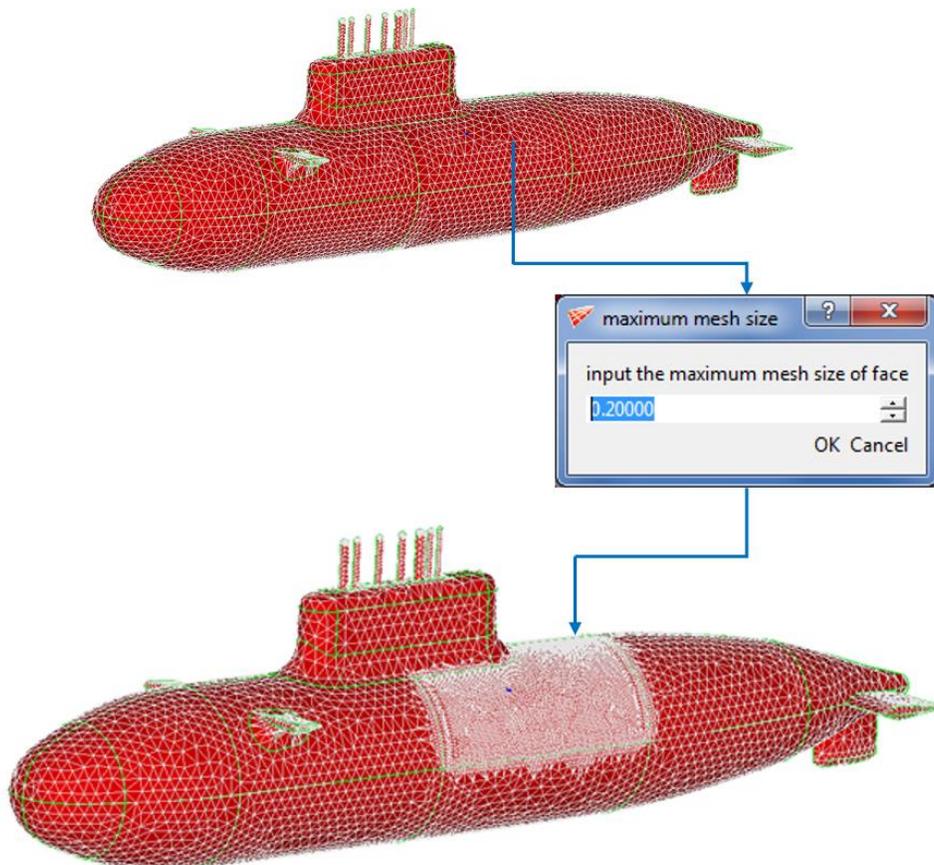


Figure 7.14: Local face mesh

*Face info* displays the information of the selected face as shown in Figure 7.15.

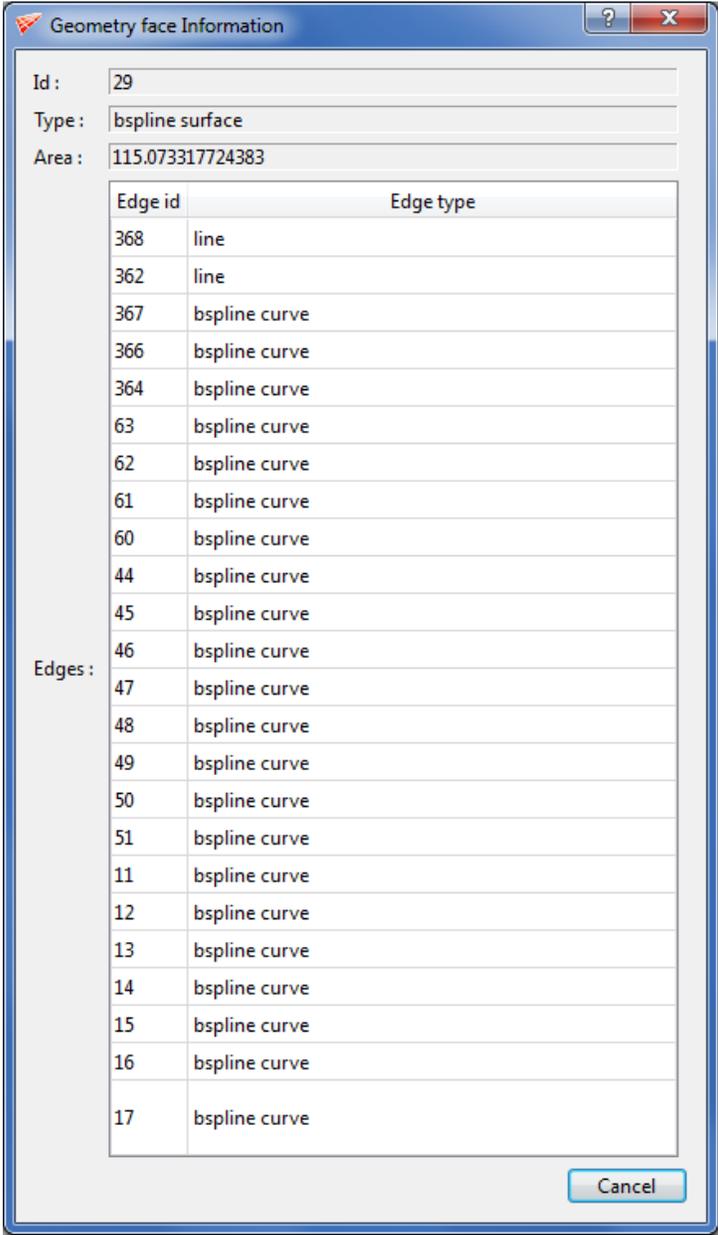


Figure 7.15: Face info

*Delete Face* deletes the selected face from the UI.

## APPENDIX A

### A1. Adding your Solver name to Solver menu:

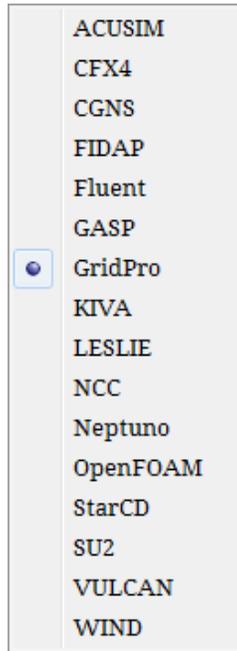


Figure A1: List of Solver formats

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
FIDAP &pty=FIDAP & ws_ptymap.fidap
Fluent &pty=FLUENT & ws_ptymap.fluent
GASP &pty=GASP & ws_ptymap.gasp
GridPro &pty=PDC & ws_ptymap.default
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 which 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 “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 to map the properties from GridPro to other solvers. The second column is the property name or boundary conditions of the respective solver. The third column is the default name or default tags; this is used when there is no labels/tags defined by the user for any given property. If there is no default name 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 for it).

Note that there are ‘32’ 2d properties and ‘16’ 3d properties. This is 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
Acusim/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
FIDAP & Unstructured & outU_fidap.script
Fluent & Unstructured & outU_fluent_new.script
FluentOld (for US3D users) & Unstructured & outU_fluent_old.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
KIUA & Unstructured & outU_kiua.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
# 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 in order to access it while exporting the grid to the user-specific solver. The script file should contain the information on how GridPro grid should be converted into the user-specific solver. The script file can also make use of the commands available in GridPro such as mrgb, chfmt, change\_format, trf etc.

## APPENDIX B

### B.1 Export to CFX format

#### How to export the *GridPro* grid format to CFX format?

There is no direct export for CFX format like other formats. Please follow the procedure to import a *GridPro* grid into CFX UI successfully without losing any parameters or properties.

Step 1: Generate the *GridPro* grid and assign properties if required.

Step 2: Open CFX Pre-processor UI

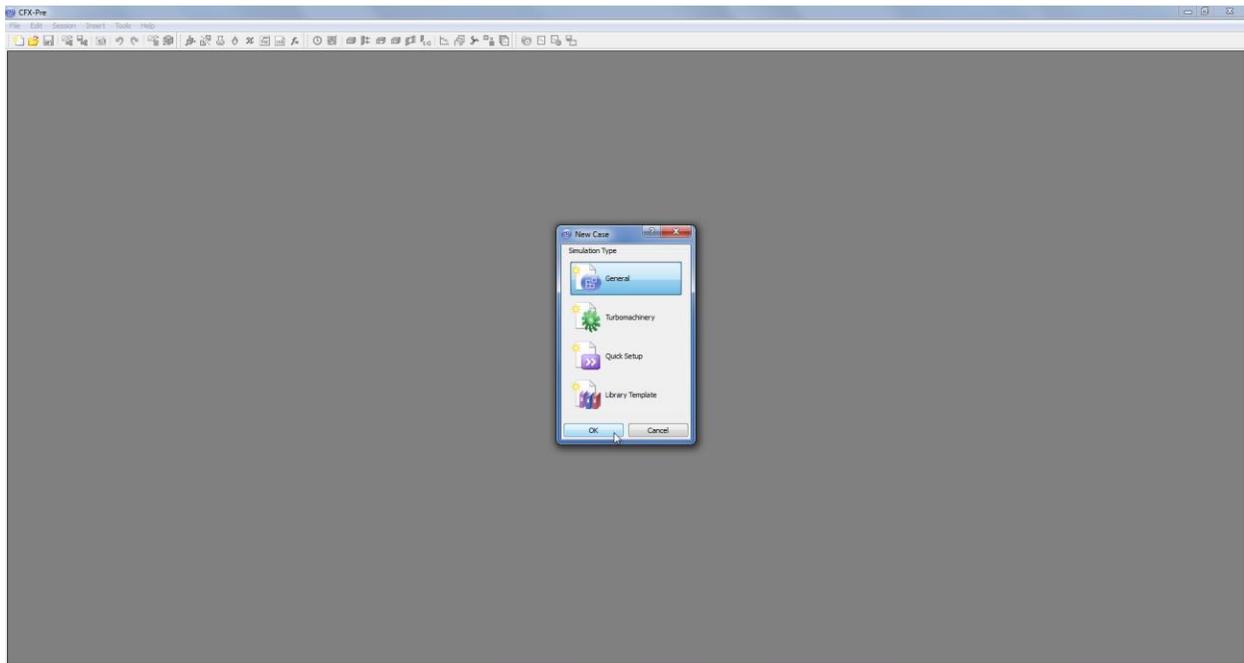


Figure B1: CFX-Pre UI

Step 3: Go to File→Import→Mesh

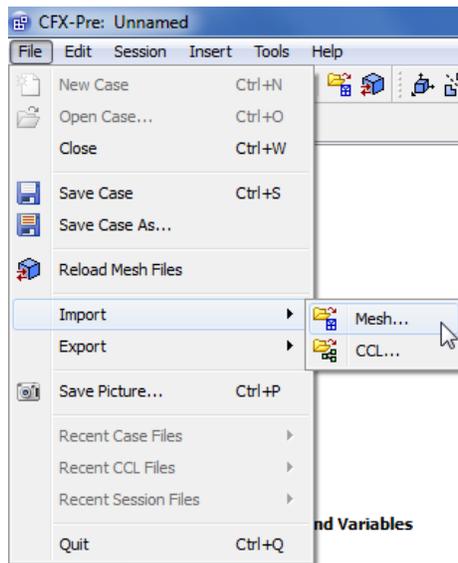


Figure B2: File menu

Step 4: Navigate the folder to the working directory and select the File type to “User Import(\*)” as shown below.

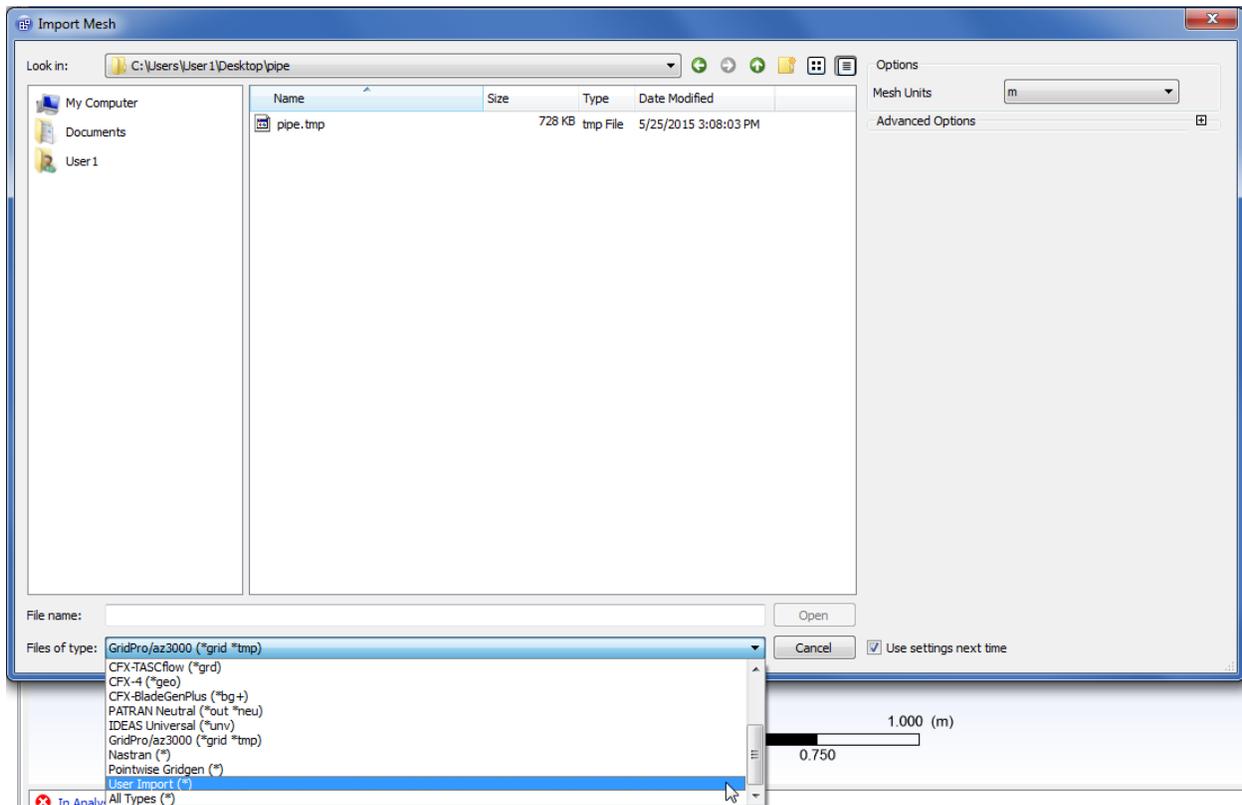


Figure B3: Import Mesh dialog box

Step 5: Navigate the “Exec Location” field to GridPro installation directory and select ‘ws.exe’ file.

For Windows users: C:/Program Files/GridPro/bin/ws.exe

For Linux/Mac users: /HOME/GridPro/az3000/ws

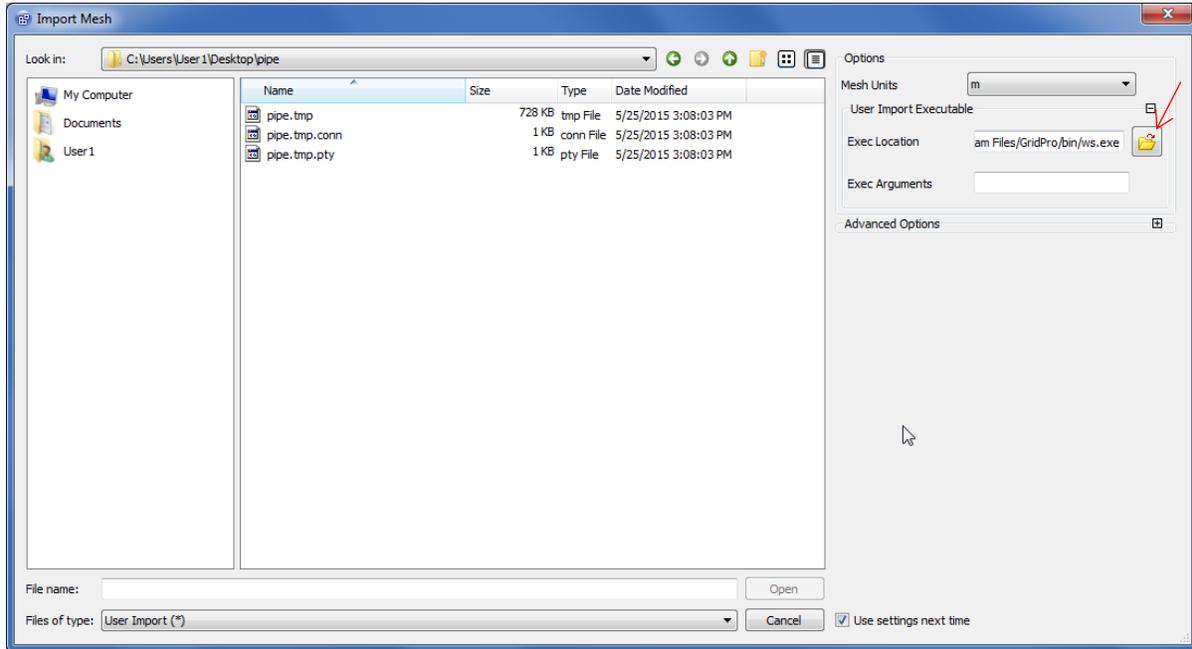


Figure B4: User Import options

Step 6: Enter the following syntax in the “Exec Arguments” field.

“change\_format -ifn”

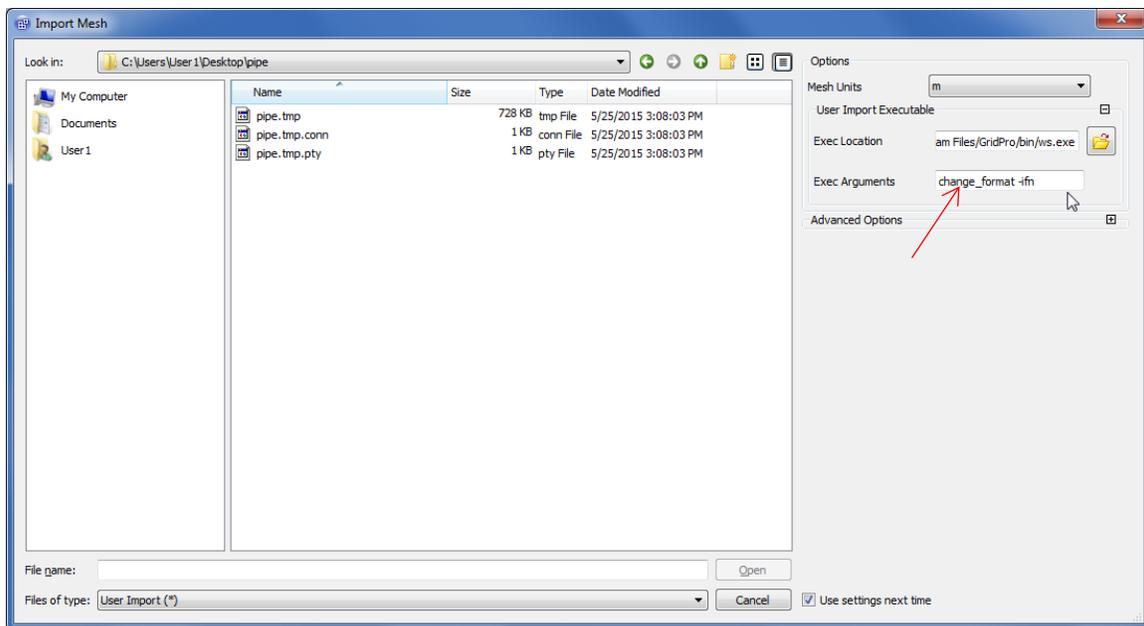


Figure B5: Input parameters

Step 7: Select the GridPro grid in \*.tmp or \*.grd format and click 'Open'

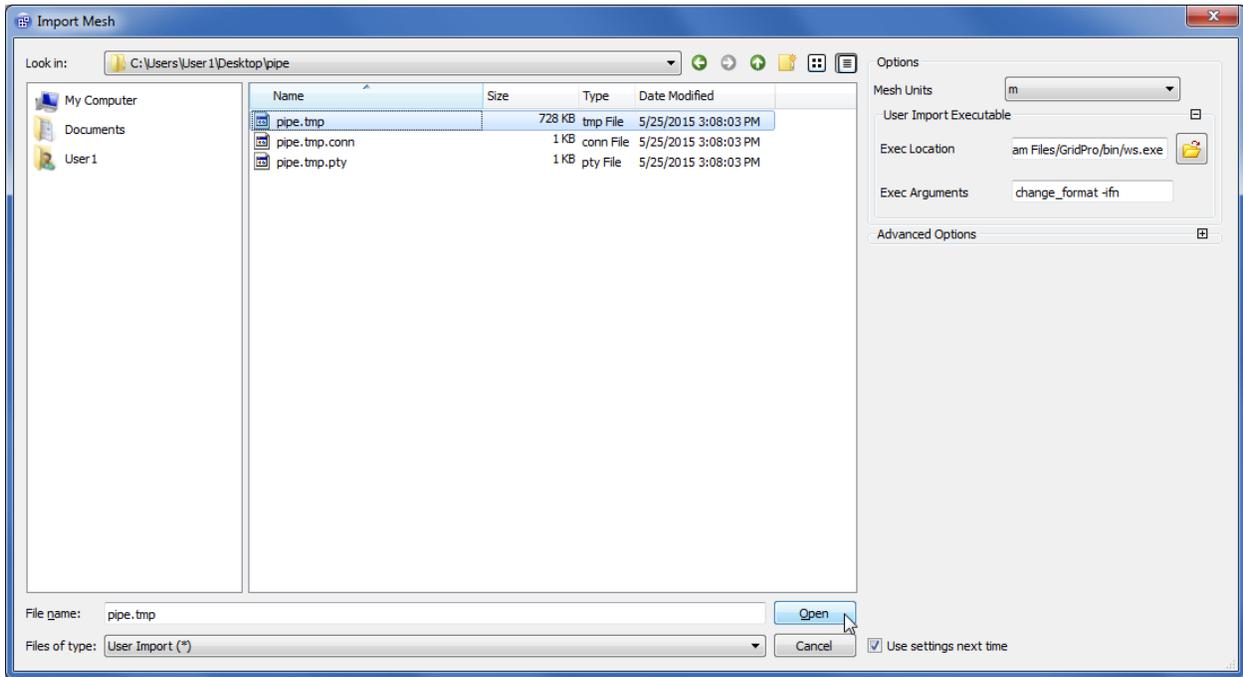


Figure B6: File selection

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.

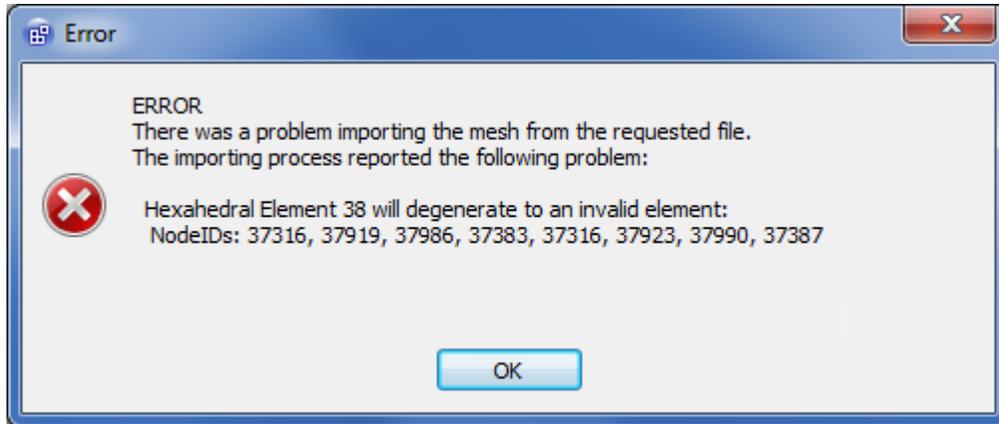


Figure B7: Error message

Step A: Go to “Duplicate node checking” under Advanced Options.

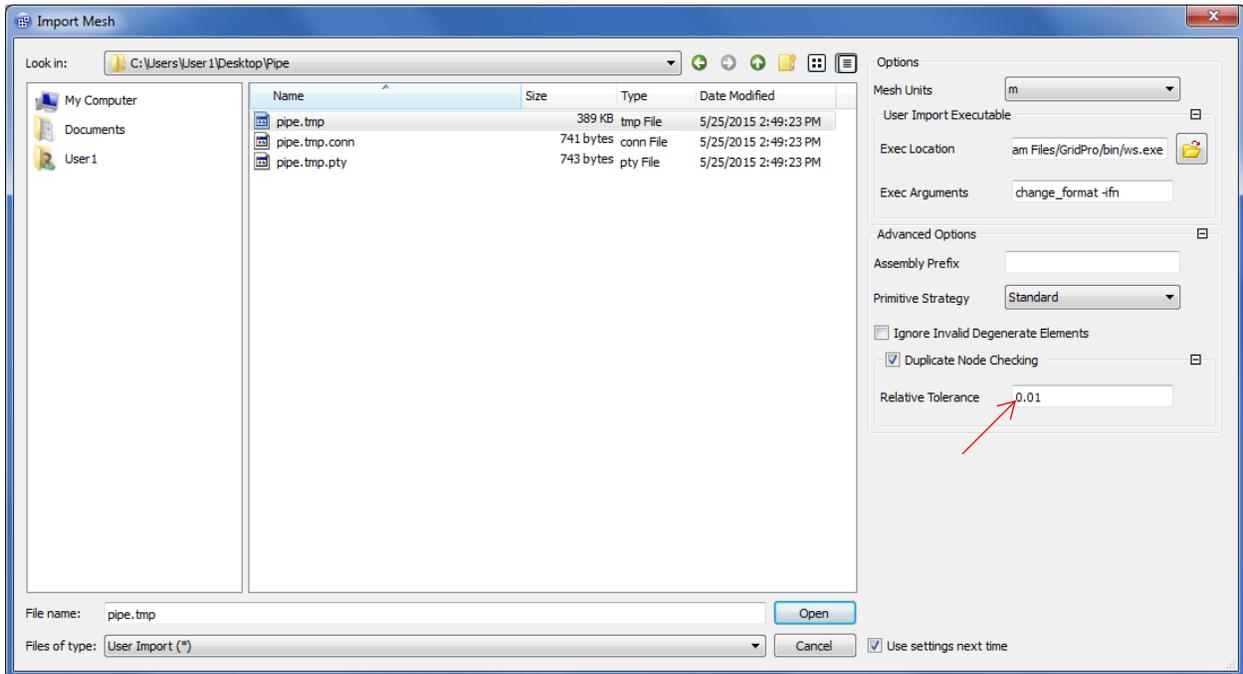


Figure B8: Advanced options

Step B: Decrease the relative tolerance value and retry

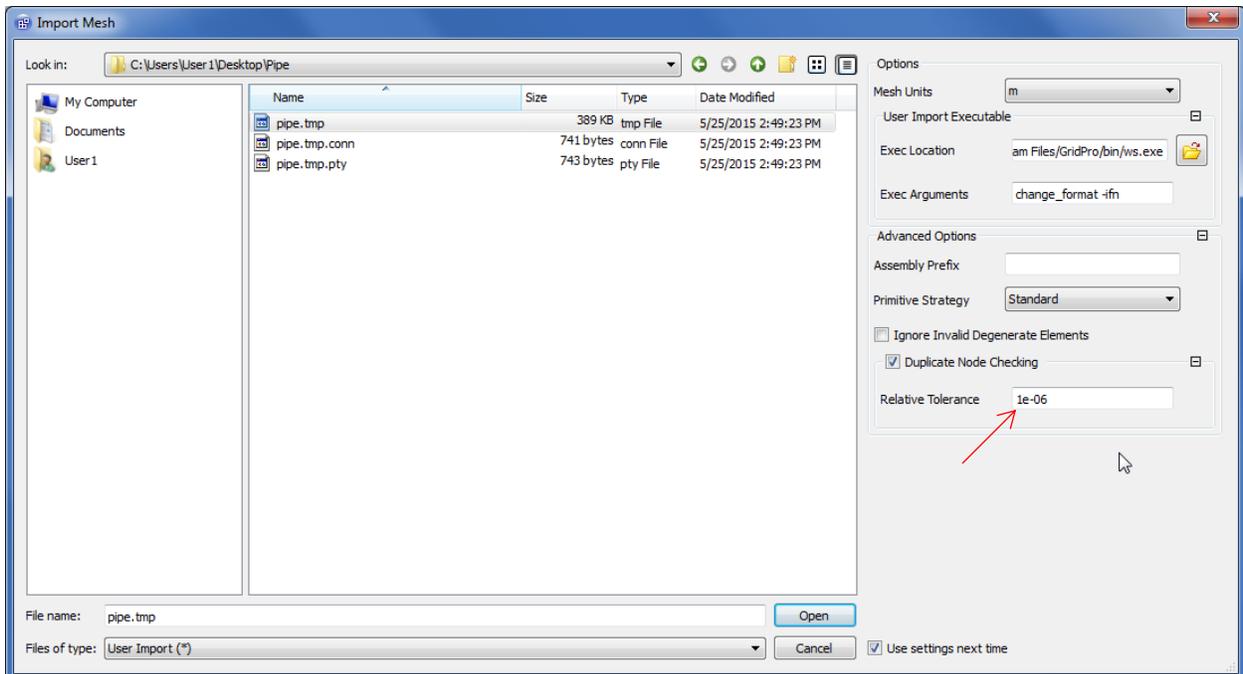


Figure B9: Relative tolerance

*If you are trying to import a clustered grid, then the relative tolerance value should be lesser than the clustering parameter. For e.g. If the clustering parameter is  $1e-05$ , then 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 2.3.24, 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

Schedule file is composed of a sequence of steps which is 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 Euler grid, remove this syntax from the schedule file.

*-C all 1.0 24* → Internal clustering for cell smoothing.

*-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* needs to 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`

Many a time, the schedule file is edited in order 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 which 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)

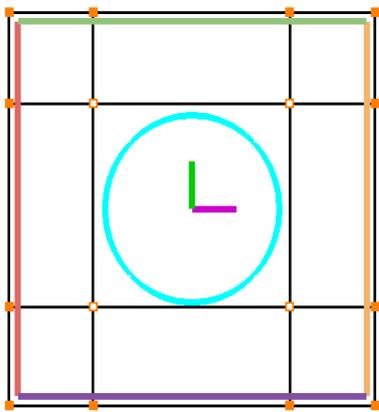
write -f blk.tmp
write -D 0 -f dump.tmp
```

Figure C2: Example schedule file

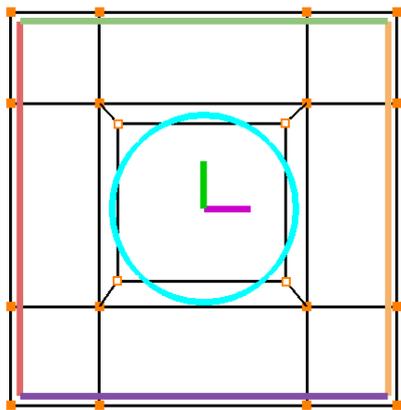
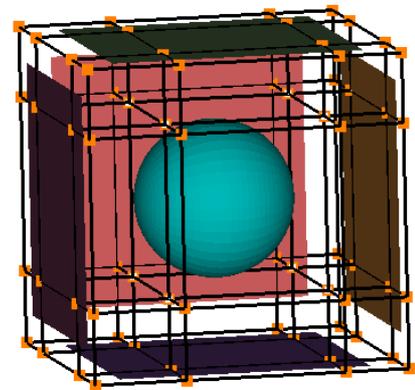
## APPENDIX D

### D.1 Surface Assignment Rules

1. If three or more blocks are emerging from an edge in 3D (or corner in 2D), it cannot be assigned to a boundary surface.



Incorrect Assignment



Correct Assignment

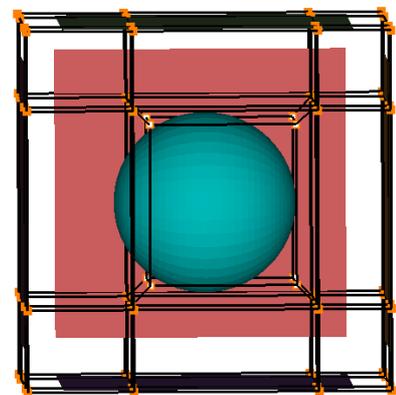


Figure D1

- Two edges/faces of the same block cannot be assigned/associated with a single surface.

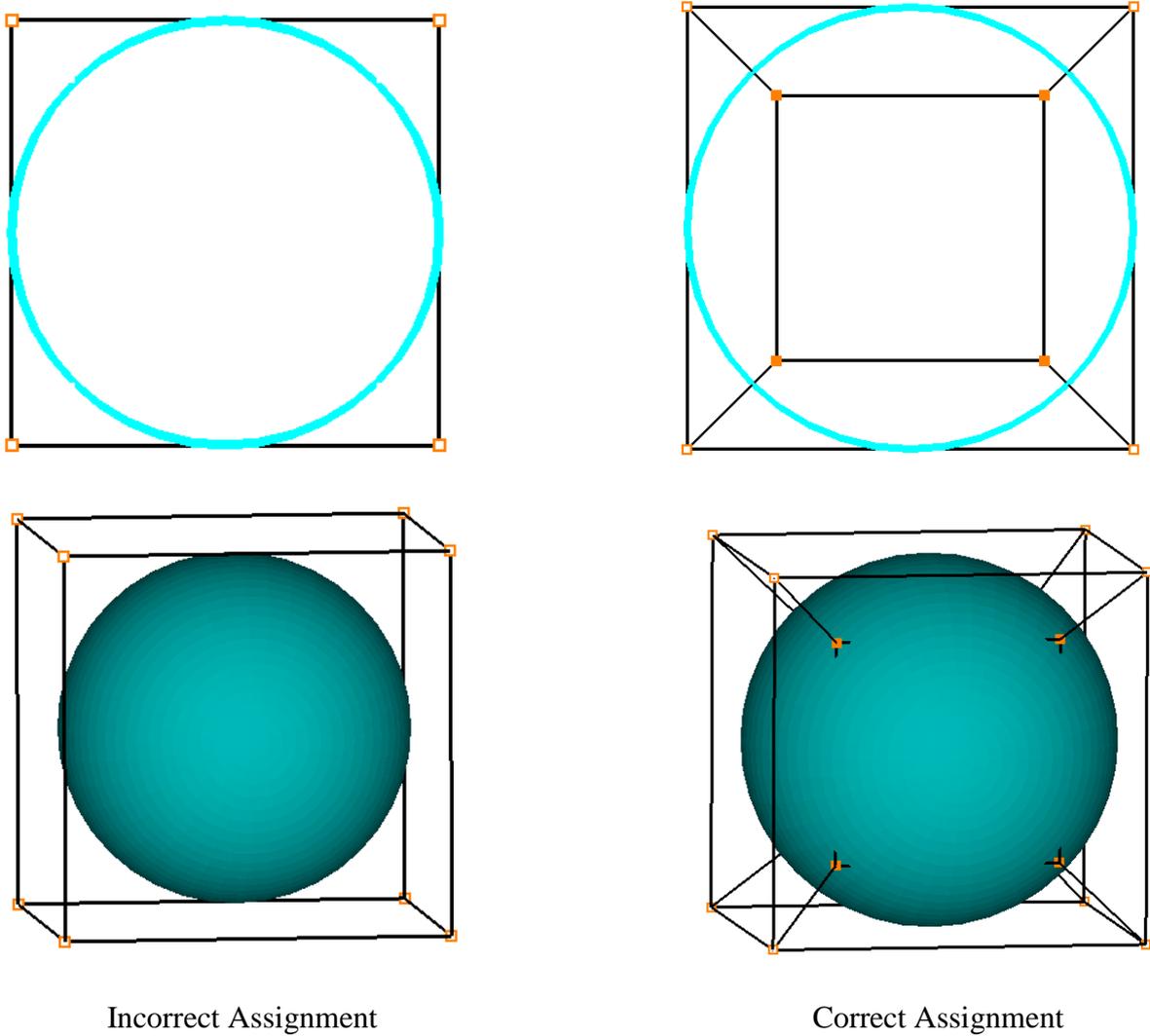
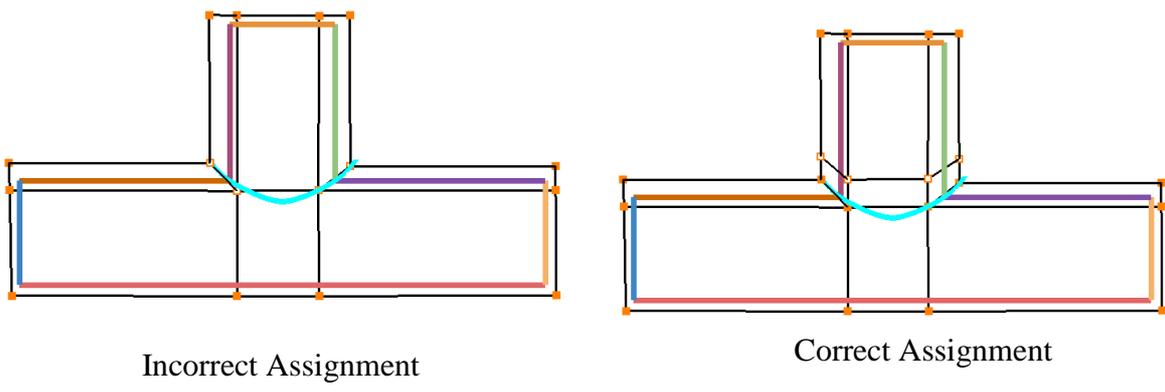


Figure D2

- The corners/edges which are only assigned to an internal surface should share two blocks on either side.



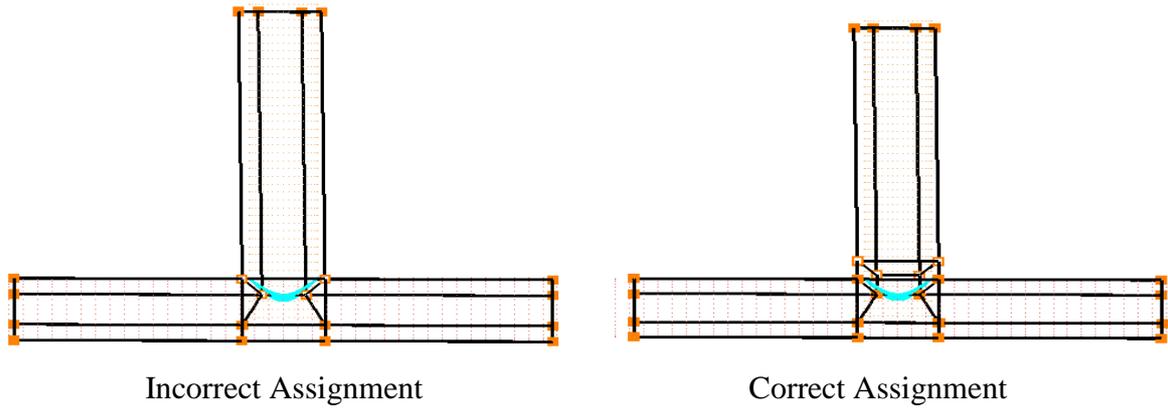


Figure D3

4. The slope discontinuity in a single surface should be represented by two distinct surfaces.

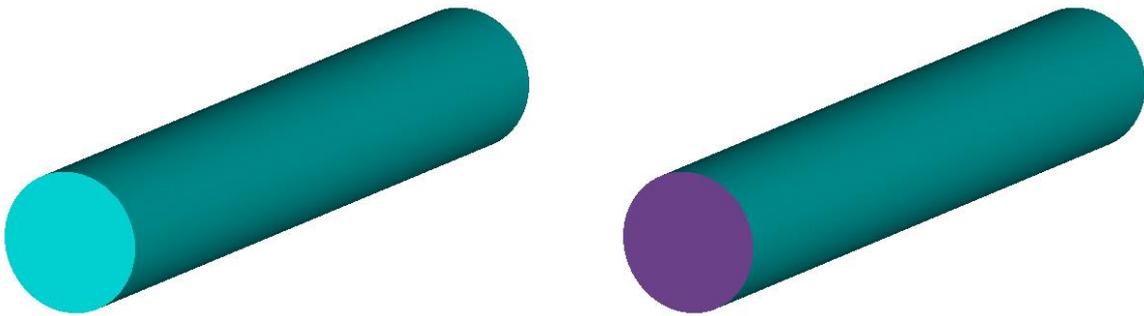
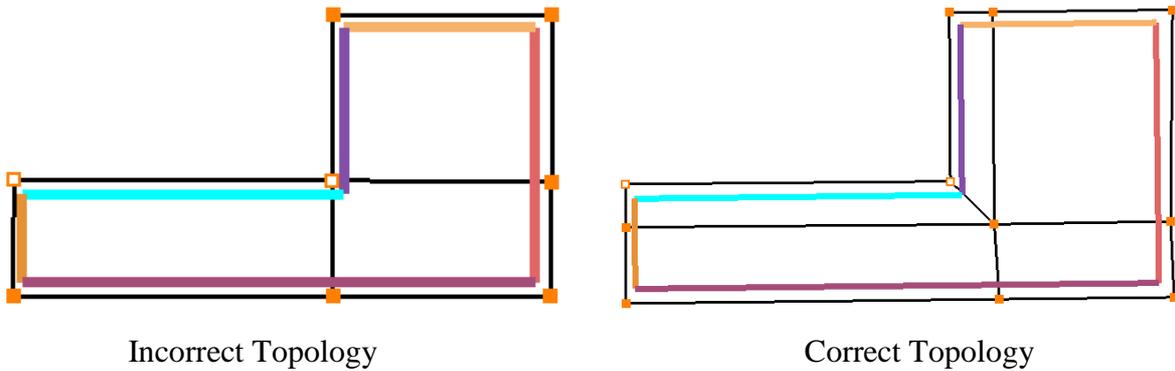
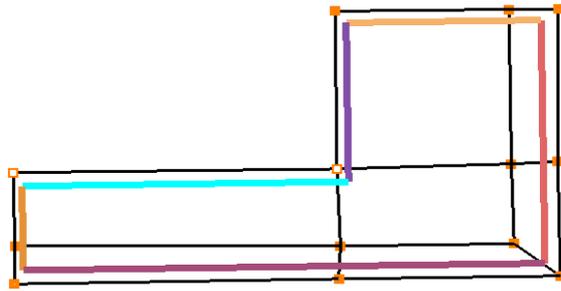


Figure D4

5. When the surface slope discontinuity is convex in nature, the blocks emerging out of an edge (or corner in 2D) should be 2 and if it is concave in nature, the blocks emerging out of an edge (or corner in 2D) should be 1.





Incorrect Topology

Figure D5