CFD EXPERTS

Simulate the Future

WWW.CFDEXPERTS.NET

Ansys
2021 R2
Engineering What's Ahead.

Ansys TurboGrid User's Guide

Ansys
ANSYS, Inc.
Southpointe
2600 Ansys Drive
Canonsburg, PA 15317
ansysinfo@ansys.com
http://www.ansys.com
(T) 724-746-3304
(F) 724-514-9494

Release 2021 R2
July 2021

ANSYS, Inc. and Ansys Europe, Ltd. are UL registered ISO 9001: 2015 companies.

Published in the U.S.A.

Table of Contents

List of Figures

List of Tables

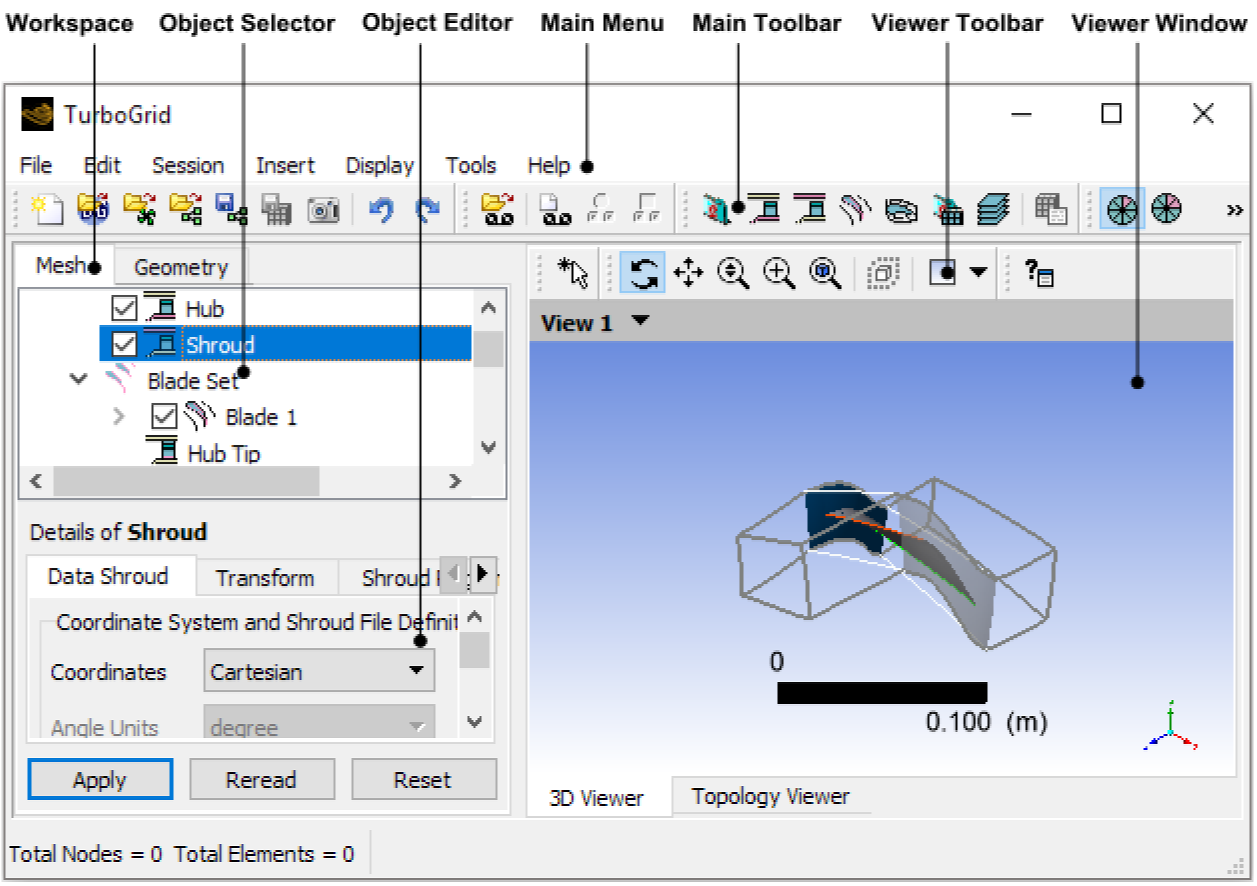
Chapter 1: Ansys TurboGrid Interface

This chapter describes:

1.1. Introduction

The Ansys TurboGrid interface is divided into several parts, as shown in Figure 1.1: TurboGrid Interface (p. 17). This chapter describes two main parts of the Ansys TurboGrid interface: the *object selector* and the *object editor*. It also introduces the two workspaces: the **Mesh** workspace and the **Geometry** workspace.

Figure 1.1: TurboGrid Interface

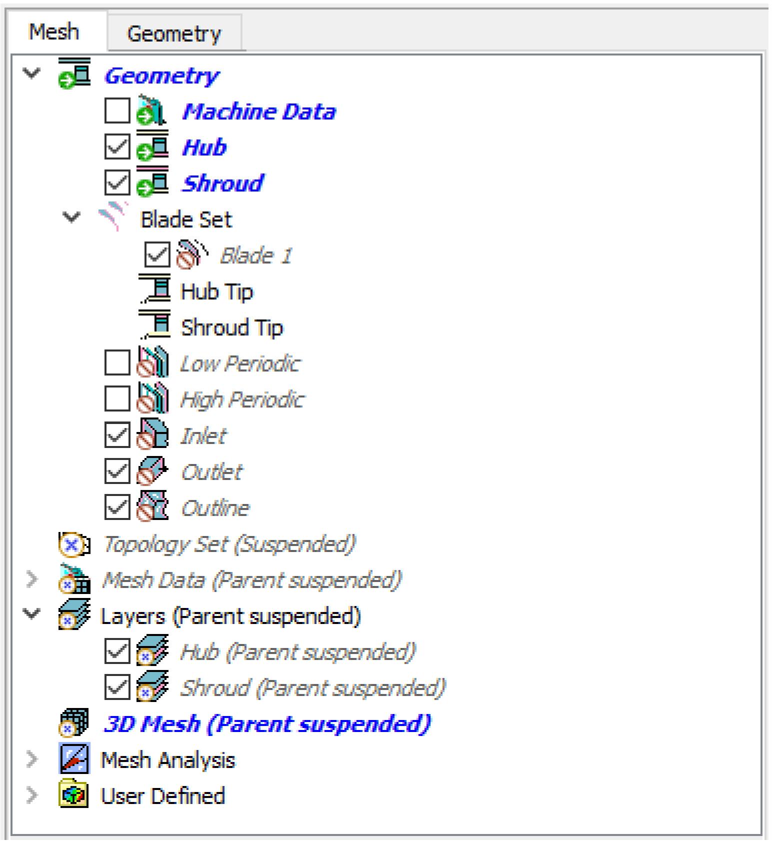


Information about main menu options and the viewer is given in subsequent chapters in this guide.

Ansys TurboGrid Workflow (p. 95) shows how to use Ansys TurboGrid.

1.2. Object Selector

The **Mesh** and **Geometry** workspaces each contain an object selector, which is a tree that lists objects in Ansys TurboGrid.

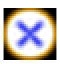


An object selector initially contains some objects in a tree format. *Objects* are data items used to drive aspects of mesh generation, visualization, and calculation.

The objects in an object selector are listed with special icons and text fonts that carry meanings as described in Table 1.1: Icon Overlays and Text Styles (p. 18).

Table 1.1: Icon Overlays and Text Styles

Icon Overlay	Font	Appended Phrase	Description
	Grey Italic		The object cannot be processed. Some other object(s) must be defined before this object can be processed.
	Blue Bold Italic		The object is ready to be defined, if applicable, and then processed. The object must be

Icon Overlay	Font	Appended Phrase	Description
			processed before a mesh can be created.
	Black		The object is complete and requires no more information before a mesh can be generated. The object can, however, be edited.
	Red Bold Italic	(Error)	The object has a problem. In the case of the `Mesh Analysis` object, a red font indicates that at least one mesh statistic falls outside the limits set in the `Mesh Limits` object.
		(Suspended)	The object will not be processed because it is suspended. You can control whether such an object is suspended from the shortcut menu. For details, see Suspend Object Updates Command (p. 176).
		(Parent suspended)	The object is suspended because a parent object is suspended. For example, whenever the `Topology Set` object is suspended, the `Mesh Data` object will also be suspended.

The object selector reflects the structure of the object definitions. For example, in the **Mesh** workspace, there is a `Hub` object in both the `Geometry` and `3D Mesh` branches. To select the geometry `Hub` object, select the `Hub` object from the `Geometry` branch.

You can open an object editor for any object by:

- Double-clicking it.

  You may need to expand a tree branch to reach a particular object. This is accomplished by clicking on the plus symbol at the root of the branch.

- Right-clicking the object and using the shortcut menu.

  Shortcut menu items will be available according to the type of object. All shortcut menu commands are described in Shortcut Menu Commands (p. 169).

An alternative way to edit an object is by using the **Command Editor** dialog box. The **Edit in Command Editor** menu item is available by right-clicking an object in the object selector. This operation opens the **Command Editor** dialog box and displays the definition of the object and its parameter settings. Edit the CCL to change the object. For further details, see Command Editor Command (p. 90).

1.2.1. Visibility Check Box

In an object selector, each object that can be displayed in the viewer window has a check box to the left of it. The check box controls the visibility of the object in the viewer. Selecting the check box turns on visibility for that object, while clearing the check box turns off the visibility. For details, see Common Options (p. 21).

1.3. Object Editor

An object editor is used to define or edit the properties of an object. It contains a set of one or more tabs that depend on the type of object being edited.

Many properties can be set via a CEL expression. To enter an expression, click in the box for a property, then click the *Enter Expression* icon that appears beside the field, then do one of the following:

- Enter an expression definition directly.
- Type the name of an existing expression.

  (This is a special case of the first point.)

You must ensure that the expression evaluates to a value having appropriate units for the property that uses the expression.

For details on CEL expressions, see Ansys CFX Expression Language in the *TurboGrid Reference Guide*.

When using Ansys TurboGrid in Ansys Workbench, you can set any property that accepts a CEL expression via a parameterized CEL expression with the parameterized CEL expression being defined by a Workbench input parameter. You can do this in either of the following ways:

- Create an expression and parameterize it using the **Expression Editor** dialog box, then use that expression as the value of the property. For details, see Expression Editor Dialog Box (p. 86).
- In the object editor, click the *Set to Workbench input parameter* icon (which appears to the right of the property) and then, in the **Enter Parameter Name** dialog box, either enter a unique name for a new Workbench input parameter (a suggested name is entered by default) or select an existing Workbench input parameter.

  If you make changes in the object editor, be sure to apply them (click **Apply**) before clicking the *Set to Workbench input parameter* icon, otherwise your changes will be lost.

  If you cannot see the *Set to Workbench input parameter* icon, click the user interface away from the property that you are trying to parameterize.

You can click *Set to value* to stop a property from being specified by a Workbench input parameter. However, the corresponding CEL expression persists, and can be managed by the **Expression Editor** dialog box. For details, see Expression Editor Dialog Box (p. 86).

A property that accepts only an integer value is also capable of being controlled by a Workbench input parameter. For such a property, the *Set to Workbench input parameter* and *Set to value* icons are used in the same way as for a property that accepts a CEL expression. Integer-value properties cannot take expressions. (No CEL expression is involved in transferring the Workbench input parameter's value from Ansys Workbench to Ansys TurboGrid, for a property that accepts only an integer value.)

1.3.1. Common Options

The options located at the bottom of the object editor are available from any of the tabs, and are common to most objects. A description of each option follows:

- The **Apply** button saves the changes made to all the tabs and, if applicable, updates the object in the viewer. Objects that depend on the edited object are also updated.
- The **Reset** button returns the settings for an object to those stored in a database for all the tabs. The settings are stored in the database each time you click **Apply**.
- The **Defaults** button restores the system default settings for all the tabs of an object. The system defaults are stored for each parameter, without regard for the types of object(s) in which the parameter may be used. For this reason, using the **Defaults** button may result in unsuitable changes to an object’s settings, and is not recommended.

1.3.2. Data Tabs

Most objects have one or more tabs that define the object. Such tabs display the definition of the object currently being edited.

1.3.3. Color Tab

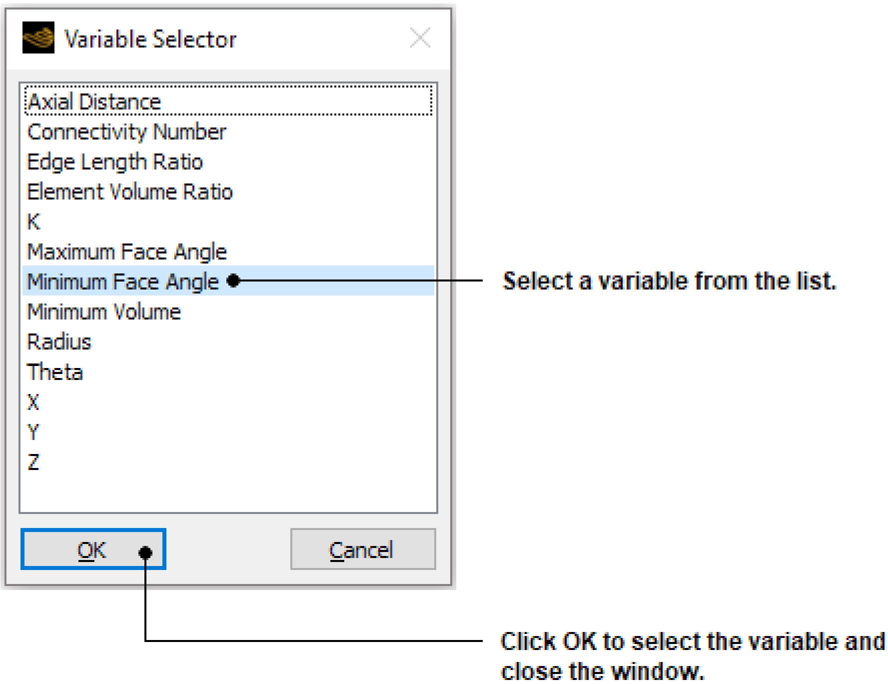
1.3.3.1. Mode

The **Color** tab controls the color of objects in the viewer. The color can be either constant or based on a variable. Select one of the two options for **Mode**.

- Select `Constant` to specify a single color for an object. To choose a color, click the ... icon to the right of the **Color** box.
- Select `Variable` to plot a variable on an object (maximum face angle on a plane, for example).
- Select `Use Plot Variable` (available for some plots, including an isosurface) to color an object by the same variable used to define it.

1.3.3.2. Variable

Choose the variable to plot from the **Variable** drop-down list. The drop-down list of variables contains the most commonly used variables. For a full list of variables, click the ... icon.



1.3.3.3. Range

Click next to the **Range** box to see the available methods for defining the range of the variable used to define the plot. This affects the variation of color used when plotting the object in the viewer. The lowest values of a variable in the selected range are shown in blue in the viewer, the highest values are shown in red.

- `Global` uses the range of the variable over all domains (regardless of the domains selected on the **Geometry** tab) to determine the minimum and maximum values.
- `Local` uses the range of the variable over the current object to determine the minimum and maximum values. This option is useful for utilizing the full color range on the object.
- Using `User Specified`, enter the minimum and maximum values for the contours. This option is useful to concentrate the full color range in a specific variable range. The variable values can be typed in, set using the embedded slider or, by clicking the icon to the right of the **Units** box, entered as an expression. Click in the box to the right of the variable value to see the available units for the variable(s).

1.3.3.4. Hybrid and Conservative Values

Select whether the plotted object is based on hybrid or conservative values of the variable used for coloring.

1.3.3.5. Undefined Color

Any areas in which the variable is not defined (when a section of an object lies outside of the computational domain, for example) use the color specified in the **Undef. Color** box. Click the  icon to the right of this box, to change the undefined color.

1.3.4. Render Tab

The exact appearance of the **Render** tab depends on the type of object plotted in the viewer window.

1.3.4.1. Draw Faces

If the **Draw Faces** check box is selected, the faces that make up an object are drawn. The faces are colored using the settings on the **Color** tab.

To change the transparency of an object, type in the transparency value or use the embedded slider (which has a maximum value of 1 and a minimum value of 0). A transparency of 0 means the shading is opaque (or having no transparency) and a transparency of 1 means the shading is invisible (completely transparent).

Shading properties can be changed to either `None`, `Flat Shading` or `Smooth Shading`.

- Select `None` so that no shading is applied to the object; it appears black.
- Select `Flat Shading` so that each rendered element is colored a constant color. Color interpolation is not used across or between rendered elements.
- Select `Smooth Shading` so that color interpolation is applied that results in color variation across a rendered element based on the color of surrounding rendered elements.

Lighting can be turned on and off by selecting/clearing the **Lighting** check box.

Specular lighting can be turned on and off by selecting/clearing the **Specular** check box. When selected, objects appear to reflect light.

Face culling turns off visibility of rendered element faces of objects that either face the viewer or point away from the viewer. Domain boundaries always have a normal vector that points out of the domain. **Face Culling** options are:

- `Front Faces`

  Selecting `Front Faces` turns off visibility of all outward-facing rendered element faces (the faces on the same side as the normal vector). This would, for example, turn off visibility of one side of a plane or the outward facing rendered elements of a cylinder locator. When applied to a volume object, the first layer of rendered element faces that point outwards are rendered invisible.

- `Back Faces`

  Selecting `Back Faces` turns off visibility of inward-facing rendered element faces (the faces on the opposite side to the normal vector). When applied to volume objects, the effect of back culling is not always visible in the viewer, since the object-rendered elements that face the outward direction obscure the culled faces. It can, however, reduce the render time when further actions are performed on the object. The effect of this would be most noticeable for large volume objects. In the same way as for front face culling, it turns off visibility of one side of surface locators.

- `No Culling`

  `No Culling` turns on the visibility of all rendered element faces.

1.3.4.2. Draw Lines

If the **Draw Lines** check box is selected, the lines that make up the object's surface are drawn. To change the line width, type in the line-width value, increase or decrease the value by 1 by clicking the up and down arrows, or use the embedded slider (which has a maximum value of 10 and a minimum value of 1). Line color can be changed by clicking on the  icon to the right of the **Color** box.

The **Edge Angle** setting is used to limit the number of visible edges in a plot. The edge angle is considered to be the angle between two faceted faces of a surface that are connected by an edge. If the angle between two adjacent faces is greater than the **Edge Angle** setting, then the edge shared by the faces is drawn. If the edge angle is 0°, the entire surface is drawn. If the edge angle is large, then only the most significant corner edges of the surface are drawn.

A sensible setting for **Edge Angle** depends on the geometry. Experiment to get a value that clearly shows where the surface is located, without displaying too much of the surface mesh. Too many lines can make it confusing when more objects are added to the geometry.

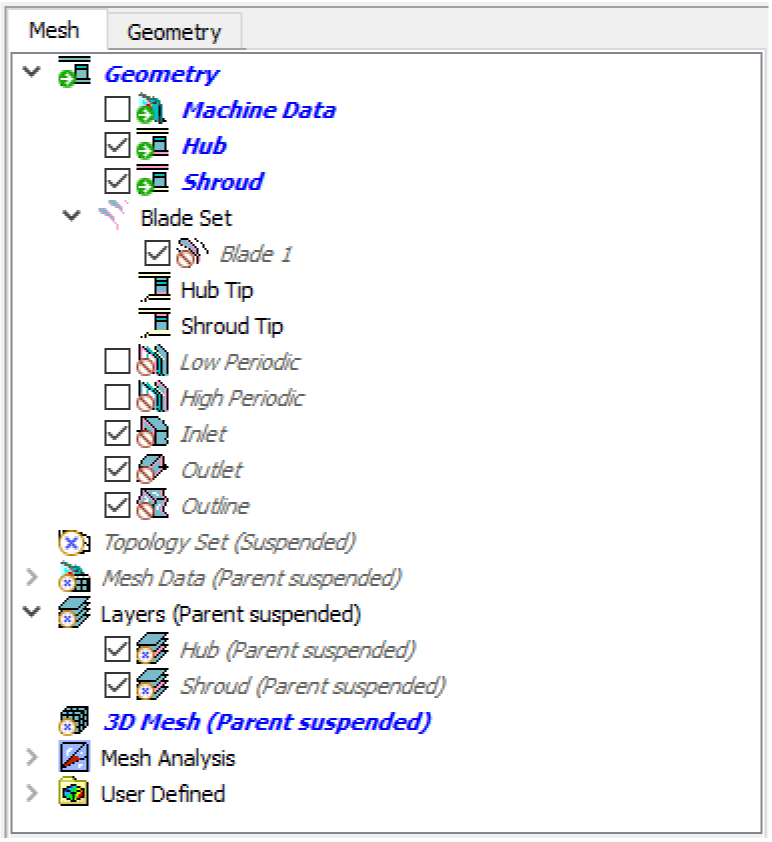
Setting an edge angle defines a minimum angle for drawing parts of the surface mesh. For example, if an edge angle of 30° is chosen, any edges shared by faces with an angle between them of 30° or more are drawn.

Reducing the edge angle shows more of the surface mesh in the viewer. When the edge angle is 0°, all of the surface mesh is shown.

1.3.4.3. Applying Instance Transforms to Objects

Instance transforms are created separately using **Insert** > **User Defined** > **Instance Transform**. For details, see Instance Transform Command (p. 67). The **Apply Instancing** check box is selected by default, and `Default Transform` is selected. To apply a different transform, it must be created and then selected from the list of existing instance transforms.

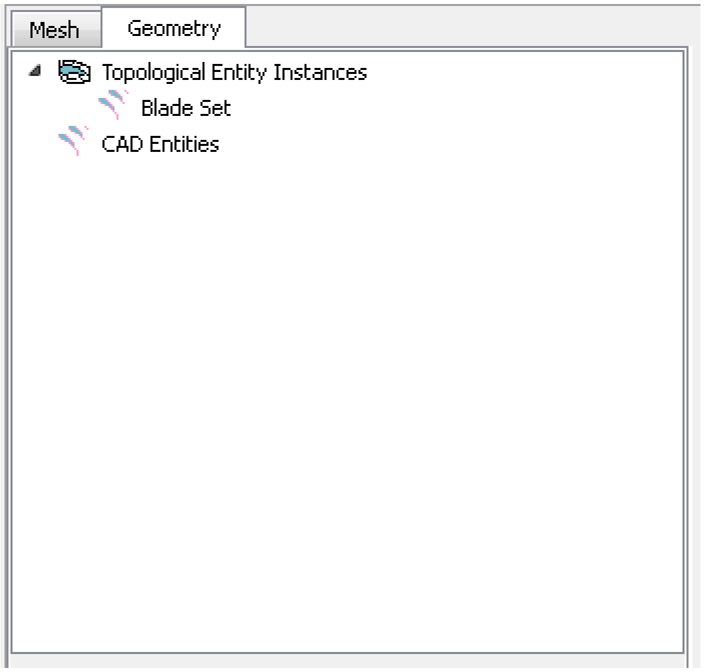
1.4. Mesh Workspace



The **Mesh** workspace holds the settings that TurboGrid needs to create a mesh.

In the **Mesh** workspace, the object selector shows objects that are used to drive aspects of mesh generation. These objects must be defined in a certain order. For details, see Steps to Create a Mesh (p. 95).

1.5. Geometry Workspace



The **Geometry** workspace can be used to:

- Select between using CAD geometry or profile point data as a geometry source. This selection is made by the **Input Mode** setting in the geometry browser, which is the main object editor of the **Geometry** workspace. To access the geometry browser, double-click a high-level object (either the `Topological Entity Instances` object or the `CAD Families` object) in the object selector.
- Using the object selector, object editors, and viewer, verify the import of CAD geometry from an upstream (BladeEditor) Geometry cell in Workbench, or manually import CAD data from another source. For details, see Defining Geometry from a CAD Source (p. 97).
- Specify a source of profile point geometry data (for example, an `*.inf` file with associated curve files). For details, see Defining Geometry from Profile Points (p. 102).

Chapter 2: File Menu

The following sections describe the commands available in the **File** menu:

2.1. Introduction

Ansys TurboGrid uses and produces the following file types:

Session File

`.tse` session files are produced by TurboGrid and contain CCL commands. Session files record the commands executed to a file for playback at a later date. Use session files to run TurboGrid in batch mode. See Session Menu (p. 49) and Batch Mode in the *TurboGrid Reference Guide* for details.

> **Note:**
>
> Because the session file is a text file of CCL commands, you can write your own session files using a text editor.

State File

`.tst` state files are produced by TurboGrid and contain CCL commands. They differ from session files in that only a snap-shot of the current state is saved to a file. Using state files, you can close TurboGrid and continue working later from the same point. See Load State Command (Import State Command) (p. 30) and Save State Command (Export State Command) (p. 31) for details.

Note:

Because the state file is a text file of CCL commands, you can write your own state files using a text editor.

Note:

State files previously had the extension `.cst`.

Topology File

`.tgt` topology files are produced by TurboGrid and define the topology. Using topology files, you can use the same topology for various cases without having to redefine it each time.

Note:

Because the topology file is a text file, you can write your own topology files using a text editor.

Mesh File

`.gtm` mesh files are produced by TurboGrid and contain the mesh in a format which can be read by Ansys CFX. For details, see Save Mesh Command (p. 34).

BladeGen.inf File

`BladeGen.inf` information files contain:

- Machine data (the rotation axis, number of blade sets, length unit)
- Names of geometry definition curve files for the hub, shroud, and blades
- Leading and trailing edge settings

The `*.inf` file is structured as follows:

```
!======  CFX-BladeGen Export  ========
Axis of Rotation: Z
Number of Blade Sets: 9
Number of Blades Per Set: 2
Geometry Units: MM       <---- Unknown|IN|MM|FT|MI|M|KM|MIL|UM|CM|UIN
Blade 0 LE: EllipseEnd   <---- EllipseEnd|CutoffEnd|SquareEnd
Blade 0 TE: CutOffEnd
Blade 1 LE: EllipseEnd
```

```
Blade 1 TE: CutOffEnd
Hub Data File: hub3.curve
Shroud Data File: shroud3.curve
Profile Data File: profile3.curve
```

(The statements following "<----" are comments that show possible values. They are not part of the format.)

For information about loading `*.inf` files, see Load TurboGrid Init File Command (p. 29).

Curve File

`.crv` and `.curve` curve files are used by TurboGrid to define machine geometry. These files contain points in free-format ASCII style and can be created:

- Using a text editor
- Using Ansys BladeGen
- Using the ExportPoints feature of BladeEditor with feature property "Export to file" = "Yes"
- By saving a modified blade geometry in TurboGrid.

Tetin File

`.tin` Tetin files are produced by TurboGrid and describe the geometry in a format which can be read by ICEM CFD products. For details, see Export Geometry Command (p. 37).

Picture Files

PNG (`.png`), AVZ (3D) (`.avz`), JPEG (`.jpg`), Windows Bitmap (`.bmp`), PPM (`.ppm`), PostScript (`.ps`), Encapsulated PS (`.eps`), and VRML (3D) (`.wrl`) files can be saved. For details, see Save Picture Command (p. 37).

2.2. New Case Command

To display the object selector and begin a new TurboGrid case, select **File** > **New Case** from the main menu or click *New Case* .

2.3. Load TurboGrid Init File Command

To load a BladeGen `*.inf` file or TurboGrid `*.tginit` initialization file, select **File** > **Load TurboGrid Init File** or click *Load TurboGrid Init File* . The **Open TurboGrid Initialization File** dialog box is displayed.

For more information, see:

2.3.1. BladeGen (`*.inf`) File Details

If the `inf` file does not specify the curve files, you may turn on an option in the dialog box (called **Guess missing curve files**) in order to guess the names of the missing curve files. The algorithm for guessing curve files looks for files in the working directory with a `.curve` or `.crv` extension, with "hub", "shroud", or "profile" in the name (using a case-insensitive search). If the curve filenames are unusual, you should verify that the correct curve files were selected by opening the appropriate `Geometry` object (for example, the `Hub` object) in the **Mesh** workspace.

If a `BladeGen.inf` file specifies multiple blades in the blade set, multiple blades will be generated.

After loading a `BladeGen.inf` file, the **Curve Type** settings for the `Hub` and `Shroud` geometry objects (in the **Mesh** workspace) are set to `Piece-wise Linear` instead of the default: `Bspline`.

2.3.2. TurboGrid Init (`*.tginit`) File Details

When opening a TurboGrid Initialization file, select a blade row to load. The selection is made in the **Open TurboGrid Initialization File** dialog box, under **Select a Blade Row**, before clicking **Open**.

2.4. Load Profile Points Command

To load new geometry curves, select **File** > **Load Profile Points** from the main menu or click *Load Profile Points*. The geometry browser is displayed with **Input Mode** set to **Profile Points**. For details, see Defining Geometry from Profile Points (p. 102).

2.5. Update Geometry Command

When running outside of Workbench, you can re-load the geometry from its source, whether profile curves or CAD data, by selecting **File** > **Update Geometry**.

Note:

If you are running in Workbench, for example, if you are updating the files in Ansys Blade-Gen/BladeModeler, and you want to re-load the updated files into TurboGrid, refresh the geometry using Workbench or the **File** > **Refresh** command in TurboGrid.

2.6. Load State Command (Import State Command)

A previously saved state file can be loaded into TurboGrid. To load a state file in stand-alone mode, select **File** > **Load State** from the main menu or click *Load State* on the toolbar; when running in Ansys Workbench, select **File** > **Import State**. The **Load State File** dialog box is displayed.

If any file specified in a state or session file cannot be found, TurboGrid will automatically search the state or session file directory for a file of the same name. If this search fails, the current working directory will be searched. As a result, state and session files will not have to be edited to change the path when state, session and curve files are moved from one directory to another.

Select the **Load as new case** option button to delete all existing objects and create new objects that are defined in the state file.

Select the **Append to current case** option button to add all objects defined in the state file to the existing objects. Existing objects are not removed unless they have the same name as an object in the state file, in which case they are replaced. Loading a state file in this way allows the use of a number of state files as building blocks for commonly used objects.

Note:

If you are running TurboGrid in Ansys Workbench, and there is a geometry provided to TurboGrid from an upstream cell, then that geometry is used and the geometry data in the state file is ignored.

Note:

Because the `Topology Set` object is now suspended by default, session and state files from version 11 or earlier may not play or load correctly. To support older session and state files, the **Play Session File** and **Load State File** dialog boxes have an option named **Unsuspend TOPOLOGY SET before loading**. Selecting this option causes the `Topology Set` object to be unsuspended before playing/loading a session/state file. When starting TurboGrid from the command line, adding the command line parameter `-u` causes the `Topology Set` object to be initially unsuspended.

2.7. Save State Command (Export State Command)

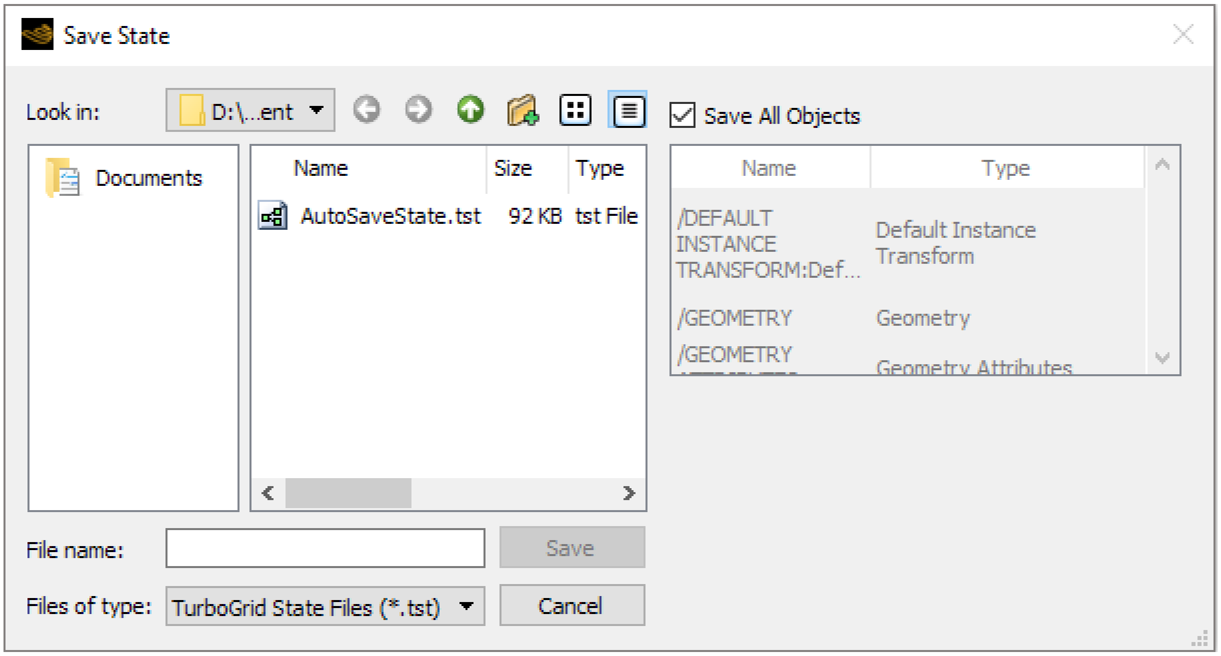
If you have not saved a state file during the current TurboGrid session, selecting **File** > **Save State** from the main menu (**File** > **Export State** when running in Ansys Workbench) opens the **Save State** dialog box where you can type a filename for your state file. For details, see Save State As Command (p. 31).

If you have already saved a state file during the current TurboGrid session, selecting **File** > **Save State** from the main menu overwrites that file. To save a state to a different filename, select **File** > **Save State As...** from the main menu. For details, see Save State As Command (p. 31).

2.8. Save State As Command

Saving a state file produces a text file containing CCL commands for the current TurboGrid state. To save a state to a new file, select **File** > **Save State As...** from the main menu or click *Save State As...*

. The **Save State** dialog box is displayed.



Set **Look in** to the directory in which you want to create the state file.

In the list on the right side, select the objects to include in the state file, or select **Save All Objects**. If **Save All Objects** is selected, the current state of all objects is written to the state file. If **Save All Objects** is cleared, select the objects to save to the state file by clicking on each object. The current state of all selected objects is written to the state file.

Enter (or select) a filename, then click **Save** to save the state file.

A state file is linked to the geometry files from which it was created by an absolute path; therefore, the location of the geometry files should not be changed. This also applies to topology files if the **From File** option is selected for a `Topology Set` object.

State files are automatically saved with a `.tst` file extension.

2.9. Save Project

When running in Ansys Workbench, you can select **File** > **Save Project** to save the entire Ansys Workbench project.

2.10. Refresh

When running in Ansys Workbench, you can select **File** > **Refresh** to refresh the associated Turbo Mesh cell.

2.11. Save Submenu

Many objects that can be saved or saved under a different name are listed under the **Save** submenu.

- Save Blade Command (p. 33)

2.11.1. Save Blade Command

If you have not saved a blade file during the current TurboGrid session, selecting **File** > **Save** > **Blade** from the main menu opens the **Save Blade** dialog box where you can type a filename for your blade file. For details, see Save Blade As Command (p. 33).

If you have already saved a blade file during the current TurboGrid session, selecting **File** > **Save** > **Blade** from the main menu overwrites that file. To save a blade to a different filename, select **File** > **Save** > **Blade As...** from the main menu. For details, see Save Blade As Command (p. 33).

2.11.2. Save Blade As Command

Saving a blade file produces a text file defining the current blade, which can then be used to define the blade for future meshes. If the same changes to the blade would otherwise be repeated, this eliminates wasted time. To save a blade to a new file, select **File** > **Save** > **Blade As** from the main menu. The **Save Blade** dialog box is displayed.

Blade files are saved with a `.crv` file extension if the file type is `All Blade Files (*.crv *.curve)`.

2.11.3. Save Periodic/Interface Surfaces Command

Saving the periodic/interface surfaces produces a text file describing the location and shape of the interfaces between adjacent blades. You can optionally set a base name for constructing the data filename. You can optionally change the units in which to save the data.

2.11.4. Save Inlet Command

Saving the inlet produces a text file describing the location and shape of the inlet region. This includes any added inlet points.

Inlet files are saved with a `.crv` file extension if the file type is `All Inlet Files (*.crv *.curve)`.

2.11.5. Save Inlet As Command

Save an inlet (see above) to an alternative filename.

2.11.6. Save Outlet Command

Saving the outlet produces a text file describing the location and shape of the outlet region. This includes any added inlet points.

Outlet files are saved with a `.crv` file extension if the file type is `All Outlet Files (*.crv *.curve)`.

2.11.7. Save Outlet As Command

Save an outlet (see above) to an alternative filename.

2.11.8. Save Topology Command

If you have not saved a topology file during the current TurboGrid session, selecting **File** > **Save** > **Save Topology** from the main menu opens the Save Topology dialog box where you can type a filename for the topology file. For details, see Save Topology As Command (p. 34).

If you have already saved a topology file during the current TurboGrid session, selecting **File** > **Save** > **Save Topology** from the main menu overwrites that file. To save a topology to a different filename, select **File** > **Save** > **Save Topology As** from the main menu. For details, see Save Topology As Command (p. 34).

2.11.9. Save Topology As Command

Saving a topology file produces a text file defining the current topology. This file can then be used to define the topology for other geometries, which may be necessary for certain analyses. To save a topology to a new file, select **File** > **Save** > **Save Topology As** from the main menu or click *Save Topology As* . The **Save Topology** window is displayed.

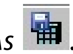
Topology files are saved with a `.tgt` file extension automatically.

2.12. Save Mesh Command

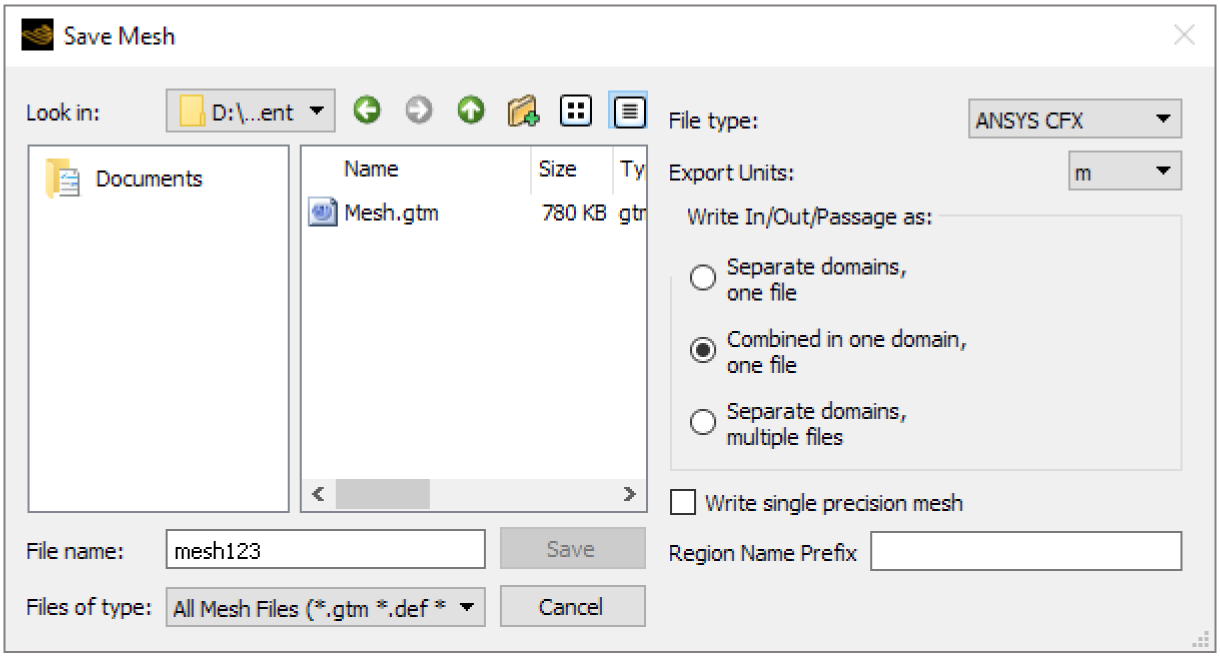
If you have not saved a mesh during the current TurboGrid session, selecting **File** > **Save Mesh** from the main menu opens the **Save Mesh** dialog box where you can type a filename for the mesh file(s). For details, see Save Mesh As Command (p. 35).

If you have already saved a mesh during the current TurboGrid session, selecting **File** > **Save Mesh** from the main menu overwrites the file(s). To save a mesh to a different name, select **File** > **Save Mesh As** from the main menu. For details, see Save Mesh As Command (p. 35).

2.13. Save Mesh As Command

Saving a mesh file produces input files for Ansys CFX or produces a CGNS file. To save a mesh to a new file, select **File** > **Save** > **Save Mesh As** from the main menu or click *Save Mesh As* . The **Save Mesh** dialog box is displayed.

2.13.1. Save Mesh Dialog Box



Set **Look in** to the directory in which you want to create the mesh file.

2.13.1.1. File Type

If **File type** is set to `Ansys CFX`, a file of type `.gtm` is saved by default (that is, if no extension is specified). You can also save a file of type `.def` by adding the `.def` extension to the specified filename. Both `.gtm` and `.def` files contain regions that can be used in CFX-Pre to set up a CFD problem.

If **File type** is set to `CGNS`, TurboGrid saves a CGNS (CFD General Notation System) file (extension `.cgns`) of version 3.3 in HDF5 1.10 format. CGNS files can be used by Ansys software such as CFX-Pre, CFD-Post, Fluent, and by third party software that supports:

- The features that TurboGrid writes,
- CGNS Version 3.0,
- HDF5 1.10.

If you change **File type**, any existing file extension (for example, `.def` or `.cgns`) is changed automatically in the specified filename.

2.13.1.2. Export Units

Set **Export Units** to the length unit for the exported mesh.

2.13.1.3. Write In/Out/Passage As

If **File type** is set to `Ansys CFX`, the following options are available:

- **Separate domains, one file**

  The inlet and outlet domains remain separate from the passage domain. Three separate assemblies appear in CFX-Pre. This choice is ideal when you want to place the inlet and outlet domains in a different frame of reference from the passage.

- **Combined in one domain, one file**

  The inlet and outlet domains are combined with (merged with) the passage domain. One combined assembly appears in CFX-Pre. This choice is ideal when you want to keep the inlet and outlet domains in the same frame of reference as the passage.

- **Separate domains, multiple files**

  Up to three separate files are written: `<basename>.gtm` for the passage domain, `<basename>_Inlet.gtm` for the inlet domain (if applicable), and `<basename>_Outlet.gtm` for the outlet domain (if applicable). Here, `<basename>` represents the specified name and `.gtm` is an example file extension.

2.13.1.4. Write Single Precision Mesh

If **File type** is set to `Ansys CFX`, you can select the **Write single precision mesh** check box to cause a single-precision mesh file to be written instead of a double-precision file. The default is double-precision. There is little benefit to using single-precision other than to reduce the size of the mesh file.

2.13.1.5. Region Name Prefix

This property specifies a string of characters that is prefixed to all mesh region names when the mesh is written to file. This property is blank by default.

For information on how CFX-Pre handles duplicate mesh region names, see Importing Multiple Meshes.

Note:

CGNS mesh region names have a 32–character limit. This limit sometimes causes the mesh region name to be truncated or replaced with a unique name (if necessary) when saved in TurboGrid.

For example,

`ABCDEF0123456789ABCDEF0123456789regionA` would be truncated to

`ABCDEF0123456789ABCDEF0123456789`, and

`ABCDEF0123456789ABCDEF0123456789regionB` would be renamed `Boundary 3`.

If your mesh region names have been truncated or renamed, you may want to manually rename the affected mesh regions in CFX-Pre or Fluent.

2.13.2. Region Naming

The regions saved to the `.gtm` and `.def` files are BLADE, PER1, PER2, HUB, SHROUD, INFLOW and OUTFLOW.

If the mesh has an inlet or outlet 3D region, additional regions will be created in the files for these domains with the suffix **Inlet** or **Outlet**. For example, the regions for a case with an inlet 3D region are PER1 Inlet, PER2 Inlet, HUB Inlet, SHROUD Inlet, INFLOW and OUTFLOW Inlet. The same applies for an outlet 3D region and for the Passage Region. For the parts specific to a region (BLADE for Passage, INFLOW for Inlet, and OUTFLOW for Outlet), no suffix is added.

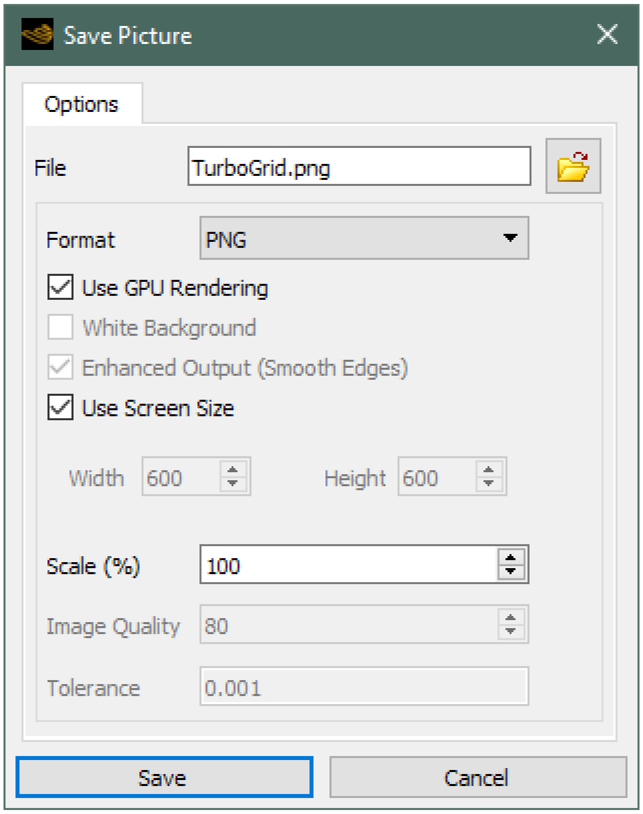
2.14. Export Geometry Command

The geometry in TurboGrid can be exported to a tetin file format which can then be read by ICEM CFD products. To export a geometry to a tetin file, select **File** > **Export Geometry** from the main menu. The **Export Geometry** dialog box is displayed.

Geometry files are saved with a `.tin` file extension automatically.

2.15. Save Picture Command

To save the current contents of the viewer window to a file, select **File** > **Save Picture** from the main menu or click *Save Picture* . The **Save Picture** dialog box is displayed.



Options Tab

The **Options** tab has the following settings:

File

Enables you to specify the filename of the file. You may enter the filename and path into the **File** field, or click the *Browse* icon and search for the directory in which the file is to be saved.

Files are always saved with the file extension corresponding to the selected graphics format.

Format

To choose a file format, click . When creating a new image file, the file format you choose affects the quality of the image:

PNG

Portable Network Graphics is a raster file format (`*.png`) that supports lossless image compression.

AVZ (3D)

Ansys Viewer Format is a file format (`*.avz`) used to present interactive three-dimensional views. It can be displayed using the Ansys Viewer.

JPEG

A compressed file format (`*.jpg`) developed for compressing raw digital information. File sizes are relatively small. Due to compression artifacts, this format is not recommended for line drawings.

Windows Bitmap

A file type (`*.bmp`) that is usually large and does not adjust well to resizing or editing. This file type does retain all of the quality of the original image and can be easily converted to other formats.

PPM

Portable Pixel Map is a file format (`*.ppm`) similar to a `Windows Bitmap`. It is an uncompressed format and is not recommended for large images.

PostScript and Encapsulated PS

PostScript (`*.ps`) and Encapsulated PS (`*.eps`) are generally recommended for output to a printer or line drawings. However, some graphics objects and features can cause the PS/EPS to output as a very large bitmap file, in which case a PNG file would be a more efficient alternative. Note that the Ansys logo and the axis do not cause the PS/EPS output to become a bitmap.

VRML (3D)

Virtual Reality Modeling Language is a file format (`*.wrl`) used to present interactive three-dimensional views. The output is VRML 2.0.

Note:

Existing CFD-Viewer State (`*.cvf`) files previously generated can be converted by using an application supplied as part of the CFX, TurboGrid and CFD-Post installations:

```
C:\Program Files\ANSYS Inc\v$ANSYSCFXRELEASE\CFX\bin\cfx5cvfconvert <cvf-file>
```

Use GPU Rendering Check Box

If your graphics hardware is compatible with GPU rendering, selecting this option is strongly recommended. You can set the default value for this option using the following preference: **Edit** > **Options** > **Common** > **Viewer Setup** > **Use GPU Rendering for Printing**.

If **Use GPU Rendering** is selected, GPU rendering is used unless compatible GPU hardware is not found, in which case all saved pictures are instead software rendered (by the CPU).

When you save a picture with GPU rendering, your graphics hardware renders an image that closely matches what is shown in the viewer.

If GPU rendering is not used, software rendering is used instead. Software rendering is relatively slow and does not always render as shown in the viewer. One benefit of software rendering is that it has no graphics hardware requirements.

White Background Check Box

You can save the current image with a white background by selecting **White Background**.

When the White Background check box is selected, certain white objects may be colored black and certain black objects may be colored white in the image file. Objects that are not affected can usually be manually colored by editing them.

Enhanced Output (Smooth Edges) Check Box

When **Enhanced Output (Smooth Edges)** is selected, the image is processed by antialiasing.

Use Screen Size Check Box

When **Use Screen Size** is selected, the output has the same width and height, measured in pixels, as shown in the viewer. You can clear the check box to specify the width and height manually.

Width/Height

You can specify the width and height of the image in pixels by entering values for **Width** and **Height**. In order to use these settings, the **Use Screen Size** check box must be cleared.

Scale (%)

Scale (%) is used to scale the size of bitmap images to a fraction (in percent) of the current viewer window size. This option is available when **Use Screen Size** is selected.

Image Quality

Image Quality is available only for the JPEG format. A value of 100 specifies the highest image quality; a value of 1 specifies the lowest image quality.

Tolerance

Tolerance is a non-dimensional value used in face sorting when generating pictures. Larger values result in faster generation times, but may cause defects in the resulting output.

Click **Save** to save the current viewer contents to an image file.

2.16. Recent State Files Submenu

TurboGrid saves the file paths of the last several state files opened. To re-open a recently used state file, select **File** > **Recent State Files** from the main menu and then select the file from the **Recent State Files** submenu.

2.17. Recent Session Files Submenu

TurboGrid saves the file paths of the last several session files opened. To re-open a recently used session file, select **File** > **Recent Session Files** from the main menu and then select the file from the **Recent Session Files** submenu.

2.18. Quit Command

To exit from TurboGrid select **Quit** from the file menu. Objects created during the TurboGrid session are not automatically saved. If you want to save the objects before quitting, create a state file. For details, see Save State Command (Export State Command) (p. 31).

Chapter 3: Edit Menu

The following sections describe the commands available in the **Edit** menu:

3.1. Undo and Redo Commands

Ansys TurboGrid includes an infinite Undo feature, limited only by the available memory on the machine and a few other restrictions described below. Select **Edit** > **Undo** from the main menu or click *Undo*  on the toolbar to return to the state immediately prior to when the last **Apply** action was executed.

Select **Edit** > **Redo** from the main menu or click *Redo*  on the toolbar to reapply changes that were undone. A **Redo** command must follow an undo or Redo command.

The undo function has a few limitations:

- When a mesh is created, the undo stack is cleared, meaning that the mesh creation process itself, and all commands before it, cannot be undone using the **Undo** command.
- While creating a session file, the undo/redo commands are not available.
- Any action that does not affect the state cannot be undone. For example, creating a mesh cannot be undone, nor can saving a topology file because neither of these actions changes the state.
- The undo function cannot return to the initial state of a default object. For example, after defining the Hub object for the first time, you cannot click *Undo* to return to the undefined state of the Hub object.
- Undo also reverses geometry manipulation when the named view icons located at the top of the Viewer window have been used. Rotation, zoom and translation actions performed using the mouse are not affected by selecting **Edit** > **Undo** from the main menu or by clicking on the  icon on the toolbar. You can, however, undo some viewer manipulations. For details, see Viewer Hotkeys (p. 77).

The redo feature is used to reverse an undo action. Selecting **Edit** > **Redo** from the main menu or clicking on the  icon on the toolbar can be done repeatedly to reverse as many undo actions as have been applied.

3.2. Options Command

Select **Edit** > **Options** from the main menu to set various viewer and appearance options in Ansys TurboGrid.

3.2.1. TurboGrid Options

The TurboGrid options are:

- **Enable Advanced Features**

  Some advanced features are hidden in the user interface. You can select this option to unhide those advanced features.

- **Enable Beta Features**

  Some Beta features are hidden in the user interface. You can select this option to "unhide" those features. When selected, such features are identified by "(Beta)" in the user interface. Note that Beta features are unofficial and not well tested.

- **Enable regions based on high and low blade geometry**

  This preference is on by default. Turning off this preference prevents writing `GEO HIGH` and `GEO LOW` versions of the `HIGHBLADE` and `LOWBLADE` surface groups, enabling Fluent in Workbench to read meshes from TurboGrid via CGNS files; Fluent does not currently support reading the `GEO HIGH`- and `GEO LOW`-based surface groups. For details on surface groups, see Surface Groups in the *TurboGrid User's Guide* (p. 64).

- **Use ATM3D Meshing By Default (Advanced)**

  Turning on this preference causes new cases to use the ATM3D meshing approach by default. This preference takes effect the next time you start TurboGrid.

  You can override the default setting for the current case using the **Use ATM3D Mesh Generation (Advanced)** option in the `Topology Set` object. For details, see Use ATM3D Mesh Generation (Advanced) (p. 136).

- **Opening Region Tolerance**

  This value is multiplied by the hub curve length (in a given case) in order to generate the default value for the **Hub Region Proximity Tolerance** setting. Similarly, this value is multiplied by the shroud curve length in order to generate the default value for the **Shroud Region Proximity Tolerance** setting. For details, see Hub Regions and Shroud Regions Tab (p. 110).

3.2.1.1. Viewer

Highlight Type controls how an object is highlighted in the viewer window while in picking mode when highlighting is on. For details, see Viewer Toolbar (p. 75).

- If `Bounding Box` is selected, the object is always highlighted with a red box surrounding the Object.
- If `Wireframe` is selected, the object is traced with a red line if the object contains surfaces.

3.2.1.1.1. Background

The following background options are available:

- Color: A constant color can be chosen.
- Image: One of a list of Predefined images or a Custom image can be selected. When setting a custom image, you must choose an image file and a type of mapping. The image types that are supported are: bmp, jpg, png, ppm. Mapping options are Flat and Spherical.

3.2.1.1.2. Text Color and Edge Color

In the stand-alone version, set text color and edge color as appropriate.

3.2.1.1.3. Axis Visibility

If the **Axis Visibility** check box is selected, the axis appears in the lower left corner of the viewer window. The axis is useful for reference when the geometry is rotated. The axis labels change when the viewer coordinates are transformed.

3.2.1.1.4. Ruler Visibility

If the **Ruler Visibility** check box is selected, a ruler appears in the viewer to show the length scale.

3.2.1.1.5. Stereo

See Stereo Viewer (p. 80).

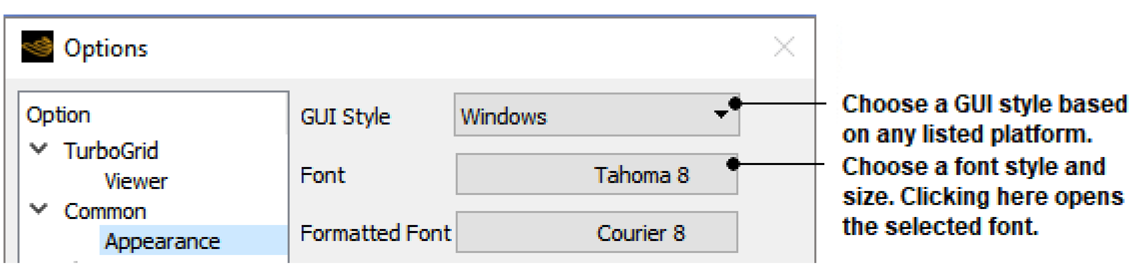
3.2.2. Common Options

Ansys TurboGrid includes an auto-save function that backs up work at set time intervals by saving a state file. Select the frequency of the auto save by picking a value from the drop-down list. **Auto Save** can be disabled by selecting Never from the list.

To change the directory in which auto-saved state files are saved, type a filename and path into the **Temporary Dir** box, or click the browse icon beside the **Temporary Dir** box to open the **Temporary Directory** dialog box.

3.2.2.1. Appearance

Click Appearance in the options box to control the appearance of the user interface.



Ansys TurboGrid sets the GUI Style to that of the machine's platform, by default. For example, on Windows the user interface has a Windows look to it. If you prefer a Linux appearance to the user interface, then select Motif from the drop-down list. Any of the following appearances can be used on any platform:

- `Windows`
- `Motif`
- `CDE`
- `Plastique`
- `Cleanlooks`

The font used within the user interface can be changed by clicking on the font button to open the **Select Font** window.

3.2.2.2. Viewer Setup: General

Viewer options under the `Common` branch are `Double Buffering` and `Unlimited Zoom` toggles.

Note:

Using Unlimited Zoom will allow the exceeding of the depth-buffer accuracy, resulting in rendering artifacts.

3.2.2.2.1. Double Buffering

`Double Buffering` is a feature supported by most OpenGL implementations. It provides two complete color buffers that swap between each other to animate graphics smoothly. If your implementation of OpenGL does not support double buffering, you can clear this check box.

3.2.2.2.2. Unlimited Zoom

By default, zoom is restricted to prevent graphics problems related to depth sorting. Selecting `Unlimited Zoom` allows an unrestricted zoom.

3.2.2.2.3. Use GPU Rendering for Printing

If your graphics hardware is compatible with GPU rendering, selecting this preference is strongly recommended.

When selected, this preference causes saved pictures in CFD-Post to use GPU rendering and saved pictures in CFX-Pre and TurboGrid to (by controlling the default value of a setting in the **Save Picture** dialog box) use GPU rendering by default.

When you save a picture with GPU rendering, your graphics hardware renders an image that closely matches what is shown in the viewer.

If GPU rendering is not set or cannot be used (for example, if compatible GPU hardware is not found), software rendering is used instead. Software rendering is relatively slow and does not always render as shown in the viewer. One benefit of software rendering is that it has no graphics hardware requirements.

Note:

Batch mode printing in CFX-Pre and TurboGrid does not support GPU rendering.

3.2.2.3. Viewer Setup: Mouse Mapping

The mouse mapping options enable you to assign viewer actions to mouse actions and keyboard/mouse combined actions.

A description of each action follows.

- Rotate: rotate the view about the screen X and Y axes.
- Object Zoom: drag the mouse up to zoom out and down to zoom in.
- Camera Zoom: drag the mouse up to zoom in and down to zoom out.
- Translate: drag the mouse to translate the view in the plane of the screen.
- Zoom Box: drag a rectangle around the area of interest. The selected area will fill the viewer when the mouse button is released.
- Zoom In: click the mouse button to zoom in step-by-step centered on the location of the mouse pointer.
- Zoom Out: click the mouse button to zoom out step-by-step centered on the location of the mouse pointer.
- Rotate Z: drag the mouse up to rotate the view clockwise about the screen Z axis, and down to rotate the view counterclockwise.
- Set Pivot Point: click an object to set the point about which the Rotate and Rotate Z actions pivot.
- Move Light: drag to move the angle of the virtual light source in the viewer. Drag the mouse left or right to move the horizontal lighting source and up or down to move the vertical lighting source. The lighting angle holds two angular values, each between 0° and 180°.

3.2.2.4. Units

Click `Units` in the options box to control the units that are presented in the user interface for each Quantity Type.

Select a pre-defined Units System such as `SI`, `English Engineering` or `British Technical`. The predefined Units for any quantity types in these units systems cannot be changed. Select the **Custom** units system to specify any valid units for each quantity type. For example, you may want to display **Length** in `mm` (an SI unit), but **Angle** in `radians` (not an SI unit). Click **More Quantity Types** to display the **Custom Quantity Types** form and set custom units for more quantity types.

The units set on this panel define the units used in the user interface for all quantity types. The units you select for a quantity type appear wherever that quantity type is used. For example, if you choose `mm` as the unit of **Length** and create a plot in Ansys TurboGrid colored by **Length**, you must specify a user specified length range in units of mm. If you created a Legend for the plot, the values on the Legend would be in units of mm.

If the **Always convert units to Preferred Units** check box is selected, all quantities entered in the user interface are converted into the preferred units.

Note:

Setting units in the variable editor will override the actual setting for that quantity type.

3.2.2.5. Threading

The **Enable Threading** check box, when selected, enables TurboGrid to accelerate processing by making use of multiple CPU cores. The **Maximum concurrent threads** setting controls the maximum number of process threads that TurboGrid is permitted to use. It is recommended that you set a value that is not higher than the number of available CPU cores.

Chapter 4: Session Menu

The following sections describe the commands available in the **Session** menu:

4.1. Introduction

Session files contain a record of the commands issued during a TurboGrid session. Actions that cause commands to be written to a session file include:

- Creation of new objects and changes to existing objects committed by clicking **Apply** in the object editor.
- Creation of a mesh.
- Commands issued in the **Command Editor** dialog box.
- Calculations performed in the built-in calculator.
- Viewer manipulation performed using the icons located at the top of the viewer window. (Viewer manipulation performed using the mouse and keyboard are not recorded to a session file.)
- Creation of new cameras and selecting a camera view.
- All actions available from the **File** menu.
- Creation of expressions and user variables.
- Creation of chart lines and viewing charts.

Session files can be used to run TurboGrid in batch mode. For details, see Batch Mode in the *TurboGrid Reference Guide*.

Note:

Since the session file is a text file of CCL commands, you can write your own session files using a text editor.

Note:

Do not end your session file with the `quit` command.

4.2. Play Session Command

A previously recorded session file can be played in TurboGrid. To play a session file, select **Session** > **Play Session** from the main menu or click *Play Session*  on the toolbar. The **Play Session File** dialog box is displayed.

The commands listed in the selected session file are run. Existing objects with the same name as objects defined in the session file are replaced by those in the session file.

For any file specified in a state or session file, if the file cannot be found TurboGrid will automatically search the state or session file directory for a file of the same name. If this procedure fails, the current working directory will be searched. As a result, state and session files will not have to be edited to change the path when state, session and curve files are moved from one directory to another.

Note:

A session file cannot be played if it contains the `Undo` command. To run a session file that contains the `Undo` command, edit the session file first to remove the command.

Note:

Since the `Topology Set` object is now suspended by default, session and state files from version 11 or earlier may not play or load correctly. To support older session and state files, the **Play Session File** and **Load State File** dialog boxes have an option named **Unsuspend TOPOLOGY SET before loading**. Selecting this option causes the `Topology Set` object to be unsuspended before playing/loading a session/state file. When starting TurboGrid from the command line, adding the command line parameter `-u` causes the `Topology Set` object to be initially unsuspended.

Note:

Workbench journal files for TurboGrid and TurboGrid session files from 12.0/12.1 may not work with 14.0. If the recorded file does not explicitly set the topology type, the following CCL block must be added:

```
TOPOLOGY SET:
ATM Topology Optimizer = off
END
```

For Workbench journal files, this CCL block should be added immediately before the following line occurs:

```
> um mode=normal, object=/TOPOLOGY SET
```

The line above appears only in journal files that were recorded while the `Topology Set` object was processed (unsuspended).

For session files, this CCL block should be added to the beginning of the session file.

For information about using Workbench journaling and scripting with TurboGrid, see Using Ansys Workbench Journaling and Scripting with TurboSystem in the *TurboSystem User's Guide*.

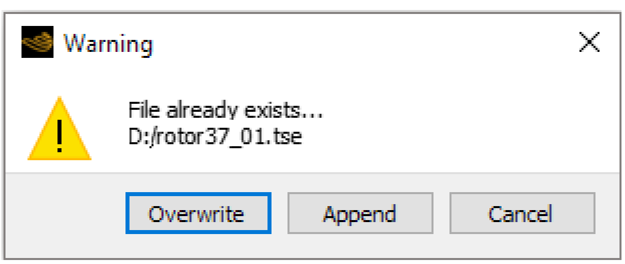
4.3. New Session Command

To create a new Session file, select **Session** > **New Session** from the main menu or click *New Session*  on the toolbar. The **Set Session File** dialog box is displayed.

Upon clicking **Save** in the dialog window, that file becomes the current session file. Commands are not written to the file until recording begins. For details, see Start Recording Command (p. 51).

Session files should be saved with a `.tse` file extension. The extension is added to a filename if `TG Session Files (*.tse)` is selected as the file type.

If you create more than one session file during a TurboGrid session, the most recently created file is by default the current session file. To set a different file to be the current session file, select an existing file from the **Set Session File** dialog box and then click **Save**. The following message then appears:



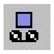
Click **Overwrite** to delete the existing session file and create a new file in its place. Click **Append** to add selected commands to the end of the existing session file when recording begins.

4.4. Start Recording Command

To start recording a session file, select **Session** > **Start Recording** from the main menu or click *Start Recording*  on the toolbar. This activates recording of CCL commands issued to the current session file. A session file must be set before recording can begin. For details, see New Session Command (p. 51).

While recording a session file, the *Undo*  and *Redo*  icons are disabled.

4.5. Stop Recording Command

To stop recording a session file, select **Session** > **Stop Recording** from the main menu or click *Stop Recording*  on the toolbar. This terminates recording of CCL commands issued to the current session file. Start and stop recording to a session file as many times as necessary.

Chapter 5: Insert Menu

The following sections describe the commands available in the **Insert** menu.

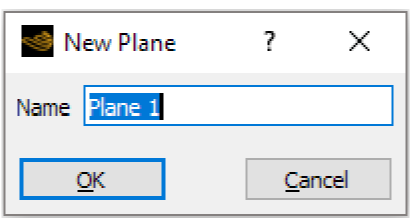
Note:

Some of the commands in the **Insert** menu are not available until you create a mesh.

5.1. Introduction

The **Insert** menu contains a list of objects that can be used as tools to analyze the mesh and geometry (although they are not necessary for mesh generation).

When you select any of the objects from the **Insert** menu, a dialog box appears in which you can either accept the default name or type a new one for the object (in this case a plane). The name should be different from any current object of the same type to avoid overwriting the existing object. Ansys TurboGrid does not enable you to create different types of objects with the same name.



Click **OK** or press **Enter** to open the relevant object editor. The object does not exist in the database until you click **Apply** in the object editor. For information on how to create objects from the command line see Object Creation and Deletion in the *TurboGrid Reference Guide*

5.2. Mesh Command

The flow path through the rotating machine is divided into small but discrete volumes. These volumes are primarily hexahedral elements, which have six sides and eight corners (wedge elements may be included in the blade tip region). There are nodes placed on each corner and the assembly of these nodes forms the mesh. The mesh fills the entire flow path and is used by Ansys CFX solvers. The quality of the solution depends partly on the quality of the Mesh. To create a mesh, select **Insert** > **Mesh** from the main menu, click *Insert Mesh*  on the toolbar, or right-click anywhere in the viewer or object selector and select **Insert Mesh** from the shortcut menu.

Ansys TurboGrid creates the mesh using the current state of the `Topology Set` and `Mesh Data` objects. The time it takes to create the mesh depends on its size and the number of smoothing iterations chosen. Displayed in the status bar at the bottom left corner of the application window is an estimate of the total number of nodes and the total number of elements in the mesh. After the mesh is created, the color and rendering settings in the viewer window can be controlled for each individual mesh surface.

If changes are made to any of the `Geometry`, `Topology Set`, `Mesh Data`, or `Layers` objects, the mesh must be recreated in order for these changes to be included in the mesh.

5.3. User Defined Submenu

The **User Defined** submenu lists commands that create new objects.

Each command invokes a dialog box to prompt you for the name of the object to create. The name of a new object must be different from any existing object. When you click **OK** or press **Enter** to accept the name of the object, the relevant object editor opens, but the object is not created until you click **Apply** on the object editor.

5.3.1. Point Command

A point can exist anywhere within or outside the domain. To create a new point, select **Insert** > **User Defined** > **Point** from the main menu.

5.3.1.1. Point: Geometry Tab

5.3.1.1.1. Point Definition

The available options for defining a point are described next:

- `XYZ`

  Set the point coordinates. To set the coordinates, first click one of the point coordinate boxes. The point coordinates will be displayed with a yellow background, and the viewer will switch to picking mode. You can then pick a point directly by clicking a visible object in the viewer. The point can lie outside of the domain.

  You can alternatively set the coordinates one at a time by typing in the coordinates and/or using the embedded slider that appears beneath the coordinate boxes.

- `Node Number`

  Create a point at a nodal location. Use the **Domains** drop-down list to select the domain(s) in which the locator exists. After choosing the domain(s) in which to select the node, enter the node number. Type in the node number or set it using the embedded sliders. When more than one domain is selected, a point is created for the specified node number in each domain (if the node number exists). If the node number does not exist in one domain but exists in another, select only the domain in which the node exists or an error message is displayed.

- `Variable Minimum` and `Variable Maximum`

Create the point where a variable is at its maximum or minimum value on any named locator. The **Domains** drop-down list is used to select the domain(s) in which the locator exists. After choosing the domain(s), select the locator name and the variable of interest. When more than one domain is selected, a point is created for the maximum/minimum value of the variable within each domain.

5.3.1.1.2. Symbol Definition

The **Symbol Size** must be between `0` and `10` (`10` being a similar scale to the geometry). Type in the value or set it using the embedded slider.

Set **Symbol** to one of the available symbols.

Picking Mode can be used to select and/or translate points in the viewer. For details, see Viewer Toolbar (p. 75).

Note:

You cannot move points that have been defined using `Node Number`, `Variable Minimum`, or `Variable Maximum`.

5.3.1.2. Point: Color Tab

See Color Tab (p. 21) for details.

5.3.1.3. Point: Render Tab

See Render Tab (p. 23) for details.

5.3.2. Line Command

A line locator can exist between two points anywhere within or outside the domain. To create a new line, select **Insert** > **User Defined** > **Line** from the main menu.

5.3.2.1. Line: Geometry Tab

5.3.2.1.1. Domains

Select the domain(s) in which the line will exist.

5.3.2.1.2. Line Definition

Lines are created by defining two points. To set the coordinates, first click one of the point coordinate boxes. The point coordinates will be displayed with a yellow background, and the viewer will switch to picking mode. You can then pick a point directly by clicking a visible object in the viewer. The point can lie outside of the domain.

You can alternatively set the coordinates one at a time by typing in the coordinates and/or using the embedded slider that appears beneath the coordinate boxes.

5.3.2.1.3. Line Type

Set the **Line Type** to either **Cut** or **Sample**.

- **Cut** extends the line in both directions until it reaches the edge of the domain. Points on the line correspond to points where the line intersects a mesh element face. As a result, the number of points on the line is indirectly proportional to the mesh spacing.

- **Sample** creates the line between the two specified points. The sample line is a set of evenly-spaced sampling points that are independent of the mesh spacing. The number of points along the line corresponds to the value in the **Samples** box.

Picking Mode can be used to select and/or translate lines in the viewer. For details, see Viewer Toolbar (p. 75).

5.3.2.2. Line: Color Tab

See Color Tab (p. 21) for details.

5.3.2.3. Line: Render Tab

See Render Tab (p. 23) for details.

5.3.3. Plane Command

A plane locator is a two-dimensional area that exists only within the boundaries of the geometry. To create a new plane, select **Insert** > **User Defined** > **Plane** from the main menu.

5.3.3.1. Plane: Geometry Tab

5.3.3.1.1. Domains

Select the domain(s) in which the plane will exist.

5.3.3.1.2. Plane Definition

The available methods for defining a plane are described next:

- `YZ Plane`

  Create a plane normal to the X axis and at a specific X value. Type in a value for **X**, set it using the embedded slider, or click in the box beside **X**, then click *Enter Expression* to the right of the **X** setting and enter its value as an expression. For details, see Expressions Command (p. 86).

- `ZX Plane`

  Create a plane normal to the Y axis and at a specific Y value. Type in a value for **Y**, set it using the embedded slider, or click in the box beside **Y**, then click *Enter Expression* to the right of the **Y** setting and enter its value as an expression. For details, see Expressions Command (p. 86).

- `XY Plane`

  Create a plane normal to the Z axis and at a specific Z value. Type in a value for **Z**, set it using the embedded slider, or click in the box beside **Z**, then click *Enter Expression*  to the right of the **Z** setting and enter its value as an expression. For details, see Expressions Command (p. 86).

- `Point and Normal`

  Create a plane using a single point on the plane and a vector normal to the plane.

  To set the point coordinates, first click one of the point coordinate boxes. The point coordinates will be displayed with a yellow background and the viewer will switch to picking mode. You can then pick a point directly by clicking a visible object in the viewer. The point can lie outside of the domain.

  You can alternatively set the coordinates one at a time by typing in the coordinates and/or using the embedded slider that appears beneath the coordinate boxes.

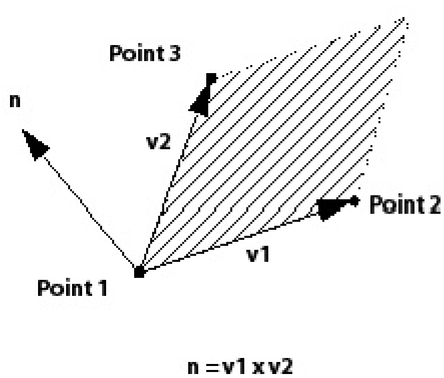
  When picking a point from the viewer for the normal vector, the vector is taken from the origin to the point picked.

- `Three Points`

  Create a plane using three points. To set the point coordinates, first click one of the point coordinate boxes. The point coordinates will be displayed with a yellow background, and the viewer will switch to picking mode. You can then pick a point directly by clicking a visible object in the viewer. The point can lie outside of the domain.

  You can alternatively set the coordinates one at a time by typing in the coordinates and/or using the embedded slider that appears beneath the coordinate boxes.

  The normal vector to the plane is calculated using the right-hand rule. The first vector is from Point 1 to Point 2 and the second is from Point 1 to Point 3 as shown in the diagram below.



5.3.3.1.3. Plane Bounds

The available types of bounds are described next:

- When `None` is selected the plane cuts through a complete cross-section of each domain specified in the **Domains** list. The plane is only bounded by the limits of the domains.
- The `Circular` option defines the bounds of a plane as a circle centered at the point used in the **Plane Definition**. Enter the value of the radius of the circle or click in the box beside **Radius**, then click *Enter Expression*  to the right of the **Radius** setting to enter an expression for the radius of the circle. The plane is undefined in areas where the circle extends outside of the domains specified in the **Domains** list.
- For the `Rectangular` option, the plane bounds are defined by a rectangle centered about the point selected in the **Plane Definition** with lengths in the x and y-directions of **X Size** and **Y Size**, respectively. You may have to specify lengths in other directions, depending on the **Plane Definition**. The size is determined with reference to the plane center (that is, the plane is resized around its center). The **X Angle** value rotates the plane counter-clockwise about its normal by the specified number of degrees. The plane is undefined in areas where the rectangle extends outside of the domains specified in the **Domains** list.

Both the circular and rectangular options have an **Invert Plane Bound** check box. If this check box is selected, the area defined by the rectangle or circle is used as a cut-out area from a slice plane that is bounded only by the domain(s). The area inside the bounds of the rectangle or circle does not form part of the plane, but everything on the slice plane outside of these bounds is included.

5.3.3.1.4. Plane Type

Set **Plane Type** to either **Slice** or **Sample**.

Slice extends the plane in all directions until it reaches the edge of the domain. Points on the plane correspond to points where the plane intersects an edge of the mesh. As a result, the number of points in a slice plane is indirectly proportional to the mesh spacing.

Sample creates the plane with either circular or rectangular bounds, depending on the plane bounds selected. For the `Circular` option, the density of points on the plane corresponds to the radius of the plane specified in the **Plane Bounds** frame, and the values for the **Radial** and **Circumferential** settings in the **Plane Type** frame. For rectangular bounds, the density of points on the plane corresponds to the size of the bounds for the plane in each of the plane directions, and the sample values provided in the **Plane Type** frame. A sample plane is a set of evenly-spaced points that are independent of the mesh spacing.

Picking Mode can be used to select and/or translate planes in the viewer. For details, see Viewer Toolbar (p. 75).

5.3.3.2. Plane: Color Tab

See Color Tab (p. 21) for details.

5.3.3.3. Plane: Render Tab

See Render Tab (p. 23) for details.

5.3.4. Turbo Surface Command

A *turbo surface* is a surface that exists only within the boundaries of the geometry and is defined using variables specific to rotating machinery. To create a new turbo surface, select **Insert** > **User Defined** > **Turbo Surface** from the main menu.

5.3.4.1. Turbo Surface: Geometry Tab

5.3.4.1.1. Domains

Select the domain(s) in which the turbo surface will exist.

5.3.4.1.2. Turbo Surface Definition

Ansys TurboGrid creates a turbo surface using a defined variable and its value. Set the domain to either `All Domains` or `DOMAIN:Passage`. Select a variable and set a value to define the location of the turbo surface. (The variable is "K" by default, which is the mesh plane index that varies from hub to shroud for the H-Grid and J-Grid topologies.) You may enter the value directly, or, for the "K" variable, use the up/down arrows, or, for any other variable, click in the **Value** box, then click *Enter Expression*  (to the right of the **Value** box) to enter an expression for the value, or use the slider.

5.3.4.2. Turbo Surface: Color Tab

See Color Tab (p. 21) for details.

5.3.4.3. Turbo Surface: Render Tab

See Render Tab (p. 23) for details.

5.3.5. Volume Command

A *volume* object is a collection of mesh elements; it can be created anywhere within the domain. To create a new volume, select **Insert** > **User Defined** > **Volume** from the main menu.

5.3.5.1. Volume: Geometry Tab

5.3.5.1.1. Domains

Select the domain(s) in which the volume will exist.

5.3.5.1.2. Volume Definition

The available methods for defining a volume are described next:

- `Sphere`

  Create a volume in the shape of a sphere by defining the center point and radius for the sphere.

  To set the coordinates of the center points, first click one of the point coordinate boxes. The point coordinates will be displayed with a yellow background, and the viewer will switch to picking mode. You can then pick a point directly by clicking a visible object in the viewer. The point can lie outside of the domain.

  You can alternatively set the coordinates one at a time by typing in the coordinates and/or using the embedded slider that appears beneath the coordinate boxes.

  Type in the radius value or set it using the embedded slider. Ensure that the units are set correctly. When **Mode** is set to `Intersection`, the volume consists of all elements that intersect the sphere surface. When **Mode** is set to `Below Intersection` or `Above Intersection`, the volume consists of all elements that are "below" or "above" the sphere surface, respectively. If the **Inclusive** check box is selected, all elements that intersect the sphere surface are added to the volume.

- `From Surface`

  Create a volume using an existing surface. Set **Location** to one or more of the available surfaces for defining the volume (use **Ctrl** to multi-select). When **Mode** is set to `Intersection`, the volume consists of all elements that intersect the sphere surface. When **Mode** is set to `Below Intersection` or `Above Intersection`, the volume consists of all elements that are "below" or "above" the sphere surface, respectively. If the **Inclusive** check box is selected, all elements that intersect the sphere surface are added to the volume.

- `Isovolume`

  Create a volume based on the value(s) of a variable. Click next to the variable box to see the available variables for defining the volume. Type in the value(s), set the value(s) using the embedded slider, or click in the box, click *Enter Expression* to the right of the box, then enter the value(s) as an expression. Set the units for the variable(s) appropriately. When **Mode** is set to `At Value`, the volume consists of all elements having the defined variable value. When **Mode** is set to `Below Value` or `Above Value`, the volume consists of all elements that are "below" or "above" the defined variable value, respectively. When **Mode** is set to `Between Values`, the volume consists of all elements that are "between" the defined variable values. If the **Inclusive** check box is selected, the elements having the defined variable value are added to the volume.

5.3.5.1.3. Hybrid/Conservative

For volumes made with **Method** set to `Isovolume`, the option of using hybrid or conservative values is available. Unless you are postprocessing CFD results using Ansys TurboGrid, this option can be ignored. For details, see Hybrid and Conservative Variable Values (p. 90).

> **Note:**
>
> Volumes are not displayed as perfect shapes (for example, a perfect sphere) because mesh elements are either included or excluded from the volume.

5.3.5.2. Volume: Color Tab

See Color Tab (p. 21) for details.

5.3.5.3. Volume: Render Tab

See Render Tab (p. 23) for details.

5.3.6. Isosurface Command

An *isosurface* object is a surface upon which a particular variable has a constant value, called the "level". To create a new Isosurface, select **Insert** > **User Defined** > **Isosurface** from the main menu.

5.3.6.1. Isosurface: Geometry Tab

5.3.6.1.1. Domains

Select the domain(s) in which the isosurface will exist.

5.3.6.1.2. Isosurface Definition

Set **Variable** to one of the available variables for defining the isosurface. Type in the variable value, set it using the embedded slider, or click in the box, click *Enter Expression* (to the right of the box), and enter an expression. Set appropriate units for the variable. The isosurface connects all locations with the specified variable value.

5.3.6.1.3. Hybrid/Conservative

The option of using hybrid or conservative values is available. Unless you are postprocessing CFD results using Ansys TurboGrid, this option can be ignored. For details, see Hybrid and Conservative Variable Values (p. 90).

5.3.6.2. Isosurface: Color Tab

See Color Tab (p. 21) for details.

Note:

You may color the isosurface using any variable or choose a constant color. You should not select the `Local` range option when coloring an isosurface with the variable used to define it. In this case the local range is zero by definition, so a plot would be colored according to the effects of round-off error.

5.3.6.3. Isosurface: Render Tab

See Render Tab (p. 23) for details.

5.3.7. Polyline Command

5.3.7.1. Polyline: Geometry Tab

A *polyline* object is a set of connected line segments that connect a series of points. To create a polyline, select **Insert** > **User Defined** > **Polyline** from the main menu.

Available methods of creating a polyline are:

- `From File`
- `Boundary Intersection`

The file format for a polyline is shown below:

```
[Name]
Polyline 1
[Data]
X [ m ], Y [ m ], Z [ m ], Area [ m^2 ], Density [ kg m^-3 ]
-1.04539007e-01, 1.68649014e-02, 5.99999987e-02, 0.00000000e+00, ...
-9.89871025e-02, 3.27597000e-02, 5.99999987e-02, 0.00000000e+00, ...
.
.
.
[Lines]
0, 1
1, 2
.
.
.
[Name]
Polyline 2
.
.
.
```

In this example, the two lines containing data are shown word-wrapped onto the next line. In the actual file, all data for a given point must be on a single line.

The name of each locator is listed under the **Name** section. Point coordinates and the corresponding variable values are stored in the **Data** section. Line connectivity data is listed in the **Lines** section; this data makes reference to the points that are defined in the **Data** section. For this purpose, point numbering in the **Data** section is consecutive, starting at zero.

Comments in the file are preceded by # (or ## for the CFX-5.6 polyline format) and can appear anywhere in the file.

Blank lines are ignored and can appear anywhere in the file (except between the line "`[Data]`" and first line of data following it).

5.3.7.2. Polyline: Color Tab

See Color Tab (p. 21) for details.

5.3.7.3. Polyline: Render Tab

See Render Tab (p. 23) for details.

5.3.8. Surface Command

To create a user surface, select **Insert** > **User Defined** > **Surface** from the main menu.

5.3.8.1. User Surface: Geometry Tab

5.3.8.1.1. Method

Choose one of the available methods for defining a user surface:

- `From File`

  Select the `From File` alternative when you are not able to create the required surface using the Boundary Intersection method. This more versatile option reads data describing the surface from a file. The points may have *path variables* (variables that are only defined on the surface) associated with them. For details on the required format for a surface data file, see Surface Data Format (p. 63). Click the browse icon to open an **Import** dialog box and browse to the surface data file. Alternatively, you can type the path and filename into the **Input File** box.

- `Boundary Intersection`

  `Boundary Intersection` forms a user surface where a given locator intersects a specified list of boundaries. The **Domains** setting specifies the domains in which this user surface exists.

  To select multiple boundaries for the **Boundary List**, click ... and hold **Ctrl** as you select each boundary. Set **Location** to one of the existing graphic objects.

  The user surface consists of mesh element faces on the specified boundaries, for the mesh elements that are intersected by the locator. The user surface is usually narrow with a varying width. The fluctuation in width is more noticeable in a coarser mesh.

5.3.8.1.2. Surface Data Format

When the `From File` method is selected, an external file must exist that defines the surface. A set of surfaces can be defined in a simple text file with the following format.

```
## Comment line - optional.
## List of path variables
# <varName1>
# <varName2>
# ...
## List of point locations with path variable values
## Each line in the following list is numbered 0,1,2...
<X> <Y> <Z> <Var1Value> <Var2Value> ...
<X> <Y> <Z> <Var1Value> <Var2Value> ...
...
## Next line is a keyword that starts the definition of faces
# Faces
## List of 3 - 6 point numbers to define faces.
<Point0> <Point1> <Point2>
<Point1> <Point2> <Point3> <Point4> <Point5> <Point6>
...
```

Comments in the file are preceded by ## and can appear anywhere in the file. A single # does *not* indicate a comment; words appearing after a single # are keywords such as `Faces`.

The start of the file should begin with a list of path variables (up to 256 characters, spaces allowed). These are variables that are only defined on the user surface. Ensure that the names of these variables do not conflict with the names of existing variables. There is no need to define any path variables (if you just want to define the location of a user surface), in which case the file begins with the point location values.

The point location list in `X Y Z` format follows the optional path variable list. You must also include a value for each path variable that you have defined at the start of the file (if any). Surfaces are defined by typing `# Faces` followed by lists of 3 (triangle) to 6 (hexagon) points to define each surface. Each surface is automatically closed by connecting the last point to the first point. The list of point locations are numbered 0,1,2....n-1 where n is the number of points in the list. When defining faces, use these numbers to reference the points in the point location list. The faces specification is *not* optional.

Blank lines are ignored and can appear anywhere in the file.

The following example defines one quadrilateral face with two path variables at each point of the face:

```
# Time
# MyVar
1 1 1 1.2 500
1 2 1 2.1 200
2 2 1 3.4 300
2 1 1 4.65 400
# Faces
0 1 2 3
```

5.3.8.2. User Surface: Color Tab

See Color Tab (p. 21) for details.

5.3.8.3. User Surface: Render Tab

See Render Tab (p. 23) for details.

5.3.8.4. Surface Groups

Surface groups are produced automatically when a mesh is generated. They are found in the `3D Mesh` branch of the object selector. Each surface group shows a surface of the mesh. The available surface groups vary according to the number of blades, the types of leading and trailing edges, and a user preference. A representative list follows:

- `HUB`, `SHROUD`
- One of the following two sets of surface groups, depending on user preference **Enable regions based on high and low blade geometry**, which is found under **Edit** > **Options** > `TurboGrid`:
  - With the preference selected:

`[blade name] LOWBLADE GEO LOW`, `[blade name] LOWBLADE GEO HIGH`, `[blade name] HIGHBLADE GEO LOW`, `[blade name] HIGHBLADE GEO HIGH`

– With the preference not selected:

`[blade name] HIGHBLADE`, `[blade name] LOWBLADE`

If the blade set contains more than one blade, the blade name is included in each surface group name. `LOWBLADE` and `HIGHBLADE` refer to sides of the blade, as divided according to the mesh topology. For example, the superset of all `LOWBLADE` surface groups for a particular blade forms a surface that is mainly on the low-theta side of the blade. `GEO LOW` and `GEO HIGH` refer to sides of the blade, as divided by the leading and trailing edges, considering the centroid of each mesh element face on the blade surface. For example, the superset of all `GEO LOW` surface groups for a particular blade forms a surface, the face centroids of which are all on the low-theta side of the blade between the leading and trailing edges.

- `[blade name] BLADE LE`, `[blade name] BLADE TE`

  If there is one blade in the blade set, then the surface groups `BLADE TE` and `BLADE LE` will be available, as applicable, after generating a mesh. These surface groups are applicable only for cut-off or square leading/trailing edges. If there are two or more blades in the blade set, the surface group names start with the blade name. For example, if there are two blades named `Main` and `Splitter`, and the trailing edge of `Main` is cut-off, and the leading and trailing edges of `Splitter` are cut-off, then surface groups `Main BLADE TE`, `Splitter BLADE LE` and `Splitter BLADE TE` will be available after generating a mesh.

- `LOWPERIODIC`, `HIGHPERIODIC`
- `INLET`, `OUTLET`

"LOW" refers to low Theta value and "HIGH" refers to high Theta value.

A visibility check box next to each surface group enables you to control which are displayed.

Besides the visibility, you may also change the color and render properties for each surface group.

5.3.8.4.1. Surface Group: Definition Tab

You may view the **Domains** and **Locations** information on the **Definition** tab.

5.3.8.4.2. Surface Group: Color Tab

See Color Tab (p. 21) for details.

5.3.8.4.3. Surface Group: Render Tab

See Render Tab (p. 23) for details.

5.3.9. Contour Command

A *contour plot* is a series of lines linking points with equal values of a given variable. For example, contours of height exist on geographical maps and give an impression of gradient and land shape. To create a contour, select **Insert** > **User Defined** > **Contour** from the main menu.

5.3.9.1. Contour Plot: Definition Tab

5.3.9.1.1. Domains

Select the domain(s) in which the contour object will exist.

5.3.9.1.2. Locations

Select the locator(s) on which to plot the contours. To select multiple locators, click the icon, hold down **Ctrl**, and select each locator.

5.3.9.1.3. Variable

Choose the plot variable for the contour plot.

5.3.9.1.4. Range

Set **Range** to one of the available methods for defining the range of the contour plot. This affects the variation of color used when plotting the contours in the viewer. The lowest values of a variable in the selected range are shown in blue in the viewer, the highest values are shown in red.

- `Global` uses the range of the variable over all domains (regardless of the domains selected on the **Geometry** tab) to determine the minimum and maximum values for the contours.
- `Local` uses the range of the variable over the selected locator(s) to determine the minimum and maximum values for the contours.
- When using `User Specified`, enter the minimum and maximum values for the contours. Type in the variable values, set them using the embedded slider or, by clicking *Enter Expression* to the right of the **Units** box, enter them as an expression.
- Using `Value List`, a list separated by commas, specify the actual values at which contours should be plotted. For example, if plotting minimum face angle, try a value list of `5, 10, 15, 20, 25` degrees. It should be noted that entering a value list overrides the number specified in the **# of Contours** box (see below).

5.3.9.1.5. Hybrid/Conservative

The option of using hybrid or conservative values is available. Unless you are postprocessing CFD results using Ansys TurboGrid, this option can be ignored. For details, see Hybrid and Conservative Variable Values (p. 90).

5.3.9.1.6. Number of Contours

Set the **# of Contours** to appear in the plot. This is the number of bands plus one. This number is overridden if the **Range** setting is set to `Value List`.

5.3.9.2. Contour Plot: Labels Tab

The **Show Numbers** check box determines whether numbers corresponding to the number of contours are displayed on the plot. To view the values of the plotted variable at each contour, create a legend of the contour plot. See Legend Command (p. 68) for more details. To change the size of the text that appears on the contour plot, type a new value into the **Text Height** box or use the embedded slider (which has a maximum value of 1 and a minimum value of 0). The **Text Height** number is a fraction of the viewer height. **Text Font** controls which font is used for the numbers that appear in the contour plot. To change the text color, set **Color Mode** to `User Specified` then choose a color by clicking in the **Text Color** box or by clicking the icon, which invokes a color selector.

5.3.9.3. Contour Plot: Render Tab

See Render Tab (p. 23) for details.

Note:

When the **Draw Faces** check box is selected, the area between contour lines is shaded with a color that corresponds to a value midway between the upper and lower contour line value. For example, for a contour line at 1000 Pa and a contour line at 1200 Pa, the shaded area has a color that corresponds to 1100 Pa. If you have created a legend for the contour plot, the legend adopts flat shading between 2 contour levels. By referring to the legend, the variable values can quickly be associated with the shaded regions of the plot.

Note:

To view the contour lines as a single color, select the **Constant Coloring** check box. With this option selected, you can either leave the color set to default or you can set **Color Mode** to `User Specified` then choose a color by clicking in the **Text Color** box or by clicking the icon, which invokes a color selector.

5.3.10. Instance Transform Command

Due to rotational symmetry, only one blade passage must be meshed, reducing computing cost and time. *Instance transforms* are used to replicate sections of the computational domain in the viewer for viewing purposes. For example, you may use a rotational transform to copy a blade passage to produce a plot of an entire rotor or stator, or a fraction thereof. However, for most purposes, it is sufficient to make use of the special toolbar icons dedicated to viewing 1, 2, or all instances (, , and respectively). Instance transforms are more flexible than these special toolbar icons in that they can be applied to single objects rather than to only all (qualified) objects at the same time.

To create an instance transform, select **Insert** > **User Defined** > **Instance Transform** from the main menu.

5.3.10.1. Instance Transform: Definition Tab

> **Note:**
>
> In this release of Ansys TurboGrid, instancing is purely visual. This means that quantitative calculations can be carried out only for the original geometry.

5.3.10.1.1. Number of Copies

The **# of Copies** is the number of times the domain is replicated in the viewer. Type in the value, increase or decrease the value by 1 by clicking or respectively.

To see the whole rotating machine, the number of copies must equal the number of blade sets, as defined in the `Machine Data` object.

5.3.10.1.2. CCL Editing

If you edit an instance transform in the **Command Editor** dialog box, changes to settings, other than **Number of Copies**, will be lost the next time the **Apply** button is clicked in the instance transform editor. This happens because the instance transform editor overwrites all of the ccl parameters, except **# of Copies**, with the values stored in the default instance transform object.

5.3.11. Legend Command

A legend can be created for any object that plots a variable. The legend gives an approximate quantitative value to the colors representing the variable on a locator. To create a legend, select **Insert** > **User Defined** > **Legend** from the main menu. After a legend object is created, it is listed in the object selector under `User Defined`.

5.3.11.1. Legend: Definition Tab

5.3.11.1.1. Plot

Set **Plot** to the object for which to create a legend.

> **Note:**
>
> Any existing object can be selected, but if there is not a variable for which to create a legend, one is not created.

5.3.11.1.2. Location

The exact position of the legend in the viewer can be controlled using the **Location** settings. Set **X Justification** to `Left`, `Center`, `Right`, or `None`. Similarly, set **Y Justification** to `Top`, `Center`, `Bottom`, or `None`. In both cases, if `None` is selected, type in the position value(s) (or use the embedded sliders, which have a maximum value of 1 and a minimum value of 0) in the appropriate **Position** box (the left box is for **X Justification** and the right box is for **Y Justification**). The position values represent a fraction of the viewer width from the left side for **X Position**, or a fraction of the viewer height from the bottom for **Y Position**. The position entered is the bottom-left corner of the legend.

5.3.11.2. Legend: Appearance Tab

5.3.11.2.1. Sizing Parameters

The size of the legend can be set as a fraction of the viewer window height. Increasing the value of the **Size** setting increases both the height and width of the legend. Type in the size value or use the embedded slider (which has a maximum value of 1 and a minimum value of 0).

The **Aspect** setting controls the width of the color range bar displayed in the legend. Type in the aspect value or use the embedded slider (which has a maximum value of 0.2 and a minimum value of 0).

5.3.11.2.2. Text Parameters

The **Precision** setting controls the number of digits after the decimal point, for numbers displayed in the legend. Set **Precision** to a format of either `Scientific` or `Fixed`.

The **Value Ticks** setting specifies the number of graduations (with labels) displayed in the legend. For example, for a scale ranging from 0 to 10, a setting of 3 ticks produces graduations at 0, 5 and 10. For a contour plot, each number assigned to a contour line is displayed on the legend, along with its associated variable value. The **Value Ticks** setting has no effect when the legend is for a contour plot.

5.3.12. Text Command

Text can be added to the viewer, for example for annotation or comments. To create text, select **Insert** > **User Defined** > **Text** from the main menu. After a text object is created, it is listed in the object selector under `User Defined`.

5.3.12.1. Text: Definition Tab

5.3.12.1.1. Location

Set **Position Mode** to one of the available methods for defining the location of the text.

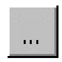
- Using **Two Coords**, position the text at a fixed location in the viewer. Set **X Justification** to `Left`, `Center`, `Right`, or `None`. Similarly, set **Y Justification** to `Top`, `Center`, `Bottom`, or `None`. In both cases, if `None` is selected, type in the position value(s) (or use the embedded sliders, which have a maximum value of 1 and a minimum value of 0) in the appropriate **Position** box (the left box is for **X Justification** and the right box is for **Y Justification**). The position values represent a fraction of the viewer width from the left side for **X Position**, or a fraction of the viewer height from the bottom for **Y Position**. The position entered is the bottom-left corner of the text.

- Using **Three Coords**, position the text using Cartesian coordinates attached to the geometry. The text moves with the geometry, but its orientation remains fixed. The X, Y, and Z coordinates are required to set the text location. Type in the position values or use the embedded sliders. Type in the rotation value, or set it using the embedded slider (which has a maximum value of 360 degrees and a minimum value of 0 degrees), or click in the **Rotation** box, click *Enter Expression*  (to the right of the **Rotation** setting), and enter an expression. A rotation angle of 0 degrees positions the text horizontally; a positive angle is measured counter-clockwise from that position.

5.3.12.2. Text: Appearance Tab

5.3.12.2.1. Text Properties

Type in the text height or use the embedded slider (which has a maximum value of 1 and a minimum value of 0).

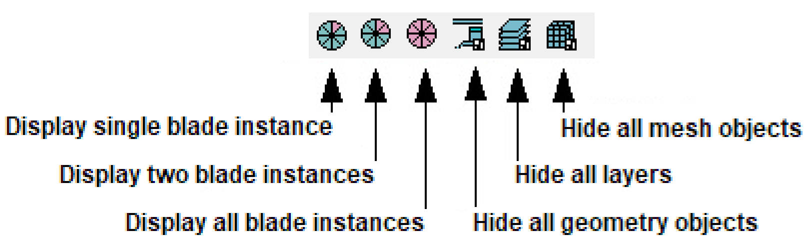
To change the text color, change **Color Mode** to `User Specified` then choose a color by clicking in the **Color** box or by clicking the  icon, which invokes a color selector.

Chapter 6: Display Menu

The following sections describe the commands available in the **Display** menu:

6.1. Introduction

The **Display Menu** options are also available from a toolbar located above the viewer.



6.2. Display One Instance Command

The default setting, **Display One Instance**, will show one passage of the geometry in the viewer. Because only one instance must be solved, displaying only one instance enables Ansys TurboGrid to work more quickly because less rendering is required.

For further information, see Instance Transform Command (p. 67).

6.3. Display Two Instances Command

When **Display Two Instances** is set, two passages of the geometry are shown in the viewer. This can give a better idea of the full machine without having Ansys TurboGrid slowed by rendering the full machine.

For further information, see Instance Transform Command (p. 67).

6.4. Display All Instances Command

When **Display All Instances** is set, the full machine is shown in the viewer. This setting shows the full geometry; the mesh can be seen on the entire geometry. It is not recommended to work with this setting because the constant rendering of the full machine will slow down processing speed.

For further information, see Instance Transform Command (p. 67).

6.5. Hide/Unhide Geometry Objects Commands

Select **Hide Geometry Objects** to turn off the visibility of all geometry objects (that is, hub, shroud, blade, and so on) and give a clear view of the remaining objects. Select **Unhide Geometry Objects** to do the opposite.

6.6. Hide/Unhide Layers Commands

Select **Hide Layers** to turn off the visibility of all of the layers. This is much more efficient than turning off the visibility for each individual layer in the object selector. Select **Unhide Layers** to do the opposite.

6.7. Hide/Unhide Mesh Objects Commands

Select **Hide Mesh Objects** to turn off the visibility of the `3D Mesh` objects. This can be useful for viewing the geometry after a mesh has been created. Select **Unhide Mesh Objects** to do the opposite.

6.8. Blade-to-Blade View Submenu

In the blade-to-blade view, some parts of the mesh might appear to be distorted, wavy, or overlapping[1]. You might be able to reduce the amount of distortion by selecting an appropriate command in the **Blade-to-Blade View** submenu. Each command affects the portion of the geometry on which the transform is based. The optimal choice depends on the blade geometry.

The **Blade-to-Blade View** submenu commands are:

- **Use Default Transform**

  The **Use Default Transform** command chooses the transform method automatically by effectively choosing either the **Use Full Transform** command or the **Use Passage Transform** command.

- **Use Full Transform**

  The **Use Full Transform** command causes the blade-to-blade coordinates to be calculated using the complete hub and shroud curves.

- **Use Passage Transform**

---

[1] By contrast, the Cartesian view does not typically exhibit distortion, except at extremely high zoom levels.

The **Use Passage Transform** command causes the blade-to-blade coordinates to be calculated using the portion of the hub and shroud curves that fall within the passage mesh, truncated at the inlet and outlet (that is, excluding the portions of the hub and shroud curves that lie within the inlet and outlet domains).

- **Use Passage Excluding Tip Transform**

  The **Use Passage Excluding Tip Transform** command is similar to the **Use Passage Transform** command, except that the tip regions are excluded in the spanwise direction. For example, if a blade has no hub tip and a profile-based shroud tip, the blade-to-blade coordinates are calculated using the portion of the hub curve that falls within the passage and the profile curve at the shroud tip. This transform may be the best choice if the hub/shroud tip is defined by a profile that varies significantly from a constant span when viewed in the other transforms.

The Passage and Passage Excluding Tip transforms:

- usually exhibit less distortion than the Full transform,
- are available only after the topology has been created,
- cause geometry objects to be omitted from the blade-to-blade view (even if these transforms are used indirectly via the default transform).

Chapter 7: Viewer

7.1. Introduction

The viewer in Ansys TurboGrid plays a central role in the mesh creation process. Its interactive interface, including the mouse, toolbars and hotkeys, allows you to inspect and alter your work.

The viewer toolbar and hotkeys are described in this chapter. Mouse controls are described in Viewer Setup: Mouse Mapping (p. 47).

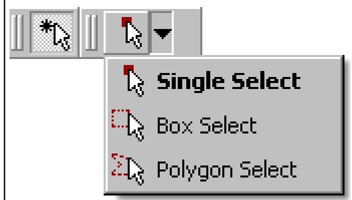
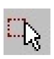
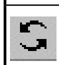
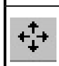
Note:

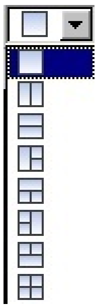
Depending on the graphics card and driver version, you may experience problems with the accuracy of mouse clicks in the viewer. For example, after dragging an inlet point, the point might move to a location far from where you dragged the mouse pointer. If you experience such problems, try lowering the hardware acceleration setting of your graphics card.

7.2. Viewer Toolbar

The viewer has the following tools:

Tool	Description
	Enters picking mode. To choose the behavior of this mode, use the drop-down arrow on the adjacent toolbar icon (which becomes visible only after you click *Select* ):

Tool	Description
	
	This is the **Single Select** option for the *Select*  tool. You can use it to select objects or drag certain objects to new locations, using the mouse. When a number of objects overlap, the one closest to the camera is picked. The text at the bottom of the viewer window shows which object would be picked at the mouse's current location. If you cannot pick the object you want because other objects overlap it, turn off the visibility of the overlapping objects, or adjust the camera to make it possible to pick the object.
	This is the **Box Select** option for the *Select*  tool. You can use it to select objects using a box. Drag a box around the object(s) you want to select.
	This is the **Polygon Select** option for the *Select*  tool. You can use it to select objects using an enclosed polygon. Click to drop points around the object(s). Double-click to complete the selection.
	Rotates the view as you drag with the mouse. Alternatively, hold down the middle mouse button to rotate the view.
	Pans the view as you drag with the mouse. Alternatively, you can pan the view by holding down **Ctrl** and the middle mouse button.
	Adjusts the zoom level as you drag with the mouse vertically. Alternatively, you can zoom the view by holding down **Shift** and the middle mouse button.
	Zooms to the area enclosed in a box that you create by dragging with the mouse. Alternatively, you can drag and zoom the view by holding down the right mouse button.
	Centers all visible objects in the viewer.
	Toggles highlighting. Highlighting makes it easier to select the correct object from the viewer. While in picking mode, highlighting by default puts a red box around the object that would be selected at the mouse's current location. The style of highlighting is

Tool	Description
	controlled by **Edit** > **Options** > **TurboGrid** > **Viewer** > **Highlight Type**.
	Selects the viewport arrangement; by default, each view displays a different transform. Independent zoom, rotation, and translate options can be carried out in each viewport.
	Displays the **Viewer Key Mapping** dialog box. See Viewer Hotkeys (p. 77) for details.

7.3. Viewer Hotkeys

A number of hotkeys are available to carry out common viewer tasks. Before using a viewer hotkey, place the mouse focus on the viewer window.

Key	Action
Space	Toggle between picking and viewing mode.
Up or **Down** or **Left** or **Right**	Rotate about horizontal and vertical axes.
Ctrl+**Up** or **Ctrl**+**Down**	Rotate about an axis normal to the screen.
Shift+**Up** or **Shift**+**Down** or **Shift**+**Left** or **Shift**+**Right**	Moves the light source.

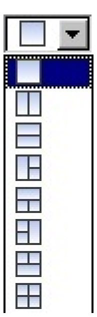
Key	Action
Ctrl+Shift	When in viewing mode, you can hold **Ctrl+Shift** to select objects in the viewer. Releasing **Ctrl+Shift** returns you to viewing mode.
1	Switches to one viewport.
2	Switches to two viewports.
3	Switches to three viewports.
4	Switches to four viewports.
c	Centers the graphic object in the viewer window.
n	Toggles the projection between orthographic and perspective.
r	Resets the view to the initial orientation.
u	Undoes transformation.
U	Redoes transformation.
x	Sets view from +X axis.
X	Sets view from -X axis.
y	Sets view from +Y axis.
Y	Sets view from -Y axis.
z	Sets view from +Z axis.
Z	Sets view from -Z axis.

7.4. Multiple Viewports

Initially the viewer contains one single viewport. The viewer can be divided into more than one window or multiple viewports, ranging from one to four.

By default, the viewer shows the current objects in a 3D Cartesian view.

The currently active viewport layout is shown in the box to the right of the *Highlighting* icon on the viewer toolbar. Click to the right of the current viewport layout picture to select a new viewport layout from the drop-down list on the viewer toolbar.



Change the active viewport by placing the mouse pointer over a viewport and clicking with any of the three mouse buttons. The viewport that contains the mouse pointer is then set as the active viewport.

Mouse-controlled transformations are applied automatically to the viewport that contains the mouse pointer. When the transformation is complete the viewport also becomes the active viewport.

Hard copy plots (postscript or other image file formats) always show all visible viewports in the viewer (a verbatim copy of the viewer window).

If the viewer consists of four viewports, each with object(s), and the viewport layout is changed to have less than four viewports, the graphic in the now "hidden" viewports remains intact, but not visible. When the viewport layout returns to the former layout all graphic objects are present, unchanged.

Each viewport has a default coordinate system.

- Viewport 1: Cartesian
- Viewport 2: 2D Blade-to-blade
- Viewport 3: 2D Meridional
- Viewport 4: 3D Turbo

You can change the coordinate system for each viewport by right-clicking on a blank area in the viewer, and selecting one of the **Transformation** commands.

Note:

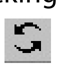
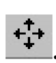
If a viewport initially appears to be empty when it is used for the first time, try clicking *Fit View* . This will center the objects and reset the zoom level.

7.4.1. Selecting, Adding, and Deleting Views

Each viewport can use any of the 4 pre-defined views or user-defined views. To switch between views, use the drop-down menu in the upper-left corner of the viewport. To add a new view based on the current state of the viewer, right-click in the viewer and select **Create New View** from the shortcut menu. To delete an existing user-defined view, right-click in the viewer and select **Delete View** while the view is selected.

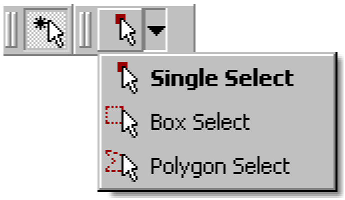
7.5. Selecting and Dragging Objects while in Viewing Mode

Picking mode allows you to select and move objects, such as points, curves, and planes.

To enter picking mode, click *Select* . To leave picking mode (that is, switch back to *viewing mode*), click *Rotate*  or *Pan* .

To temporarily enter picking mode when you are in viewing mode, hold **Ctrl**+**Shift**. When you release the **Ctrl**+**Shift** keys, you will return to viewing mode.

When you click *Select*  to enter picking mode, a toolbar icon appears to enable you to change the selection method:



For details, see Viewer Toolbar (p. 75).

7.6. Stereo Viewer

If you:

1. Have a standard stereo display

2. Have a graphics card that supports quad buffering OpenGL output

3. Have set your graphics card to "Stereo"

4. Have set your view to Perspective mode (right-click in the Viewer and select **Projection** > **Perspective**)

...you can view output in stereo. To enable this functionality:

1. Select **Edit** > **Options**.

2. In the **Options** dialog box, select **TurboGrid** > **Viewer**.

3. On the **Viewer** panel:

    a. Set the Stereo **Mode** to **Stereo**.

    b. Set the **Stereo Effect**. The value of the "stereo effect" that is required is related to the distance between the observer and the display. If the stereo effect is too strong, either move away from the display, or move the slider towards **Weaker**.

4. Click **OK** to save the settings.

Chapter 8: Tools Menu

The following sections describe the commands available in the **Tools** menu:

8.1. Calculator Command

The built-in calculator provides quantitative information about the geometry and mesh. To view the calculator, select **Tools** > **Calculator** from the main menu. The **Function Calculator** dialog box is displayed.

> **Note:**
>
> In this release of Ansys TurboGrid, quantitative calculations can only be carried out for the original geometry. Any applied instance transforms are purely visual and do not affect the calculation results.

8.1.1. Function Calculator Dialog Box

8.1.1.1. Function

Click  to select a function from the drop-down list. The table below outlines the available quantitative functions.

Function Name	Operation
area (p. 82)	Area of location
areaAve (p. 82)	Area-weighted average
areaInt (p. 82)	Area-weighted integral (can be projected to a direction)
ave (p. 83)	Arithmetic average
count (p. 83)	Number of calculation points
length (p. 83)	Length of a curve
lengthAve (p. 83)	Length-weighted average
lengthInt (p. 84)	Length-weighted integration

Function Name	Operation
maxVal (p. 84)	Maximum Value
minVal (p. 84)	Minimum Value
probe (p. 84)	Value at a point
sum (p. 84)	Sum over the calculation points
volume (p. 85)	Volume of a 3D location
volumeAve (p. 85)	Volume-weighted average
volumeInt (p. 85)	Volume-weighted integral

8.1.1.1.1. area

The `area` function is used to calculate the area of a 2D location. The following example demonstrates use of the function.

- **Function:** `area`, **Location:** `Plane1`. This example calculates the total area of the locator `Plane1`.

8.1.1.1.2. areaAve

The `areaAve` function calculates the area-weighted average of an expression on a 2D location. The area-weighted average of a variable is the average value of the variable on a location when the mesh element sizes are taken into account. Without the area weighting function, the average of all the nodal variable values would be biased towards variable values in regions of high mesh density. The following examples demonstrate use of the Function.

- **Function:** `areaAve`, **Location:** `Outlet`, **Variable:** `Velocity`. This example calculates the average magnitude of the velocity on the `outlet` location. Note that flow direction is not considered since the magnitude of a vector quantity at each node is calculated. Use the scalar components of velocity (for example, `Velocity u`) to include a directional sign, for example:
- **Function:** `areaAve`, **Location:** `Outlet`, **Variable:** `max(Velocity u, 0.0[m s^-1]`). This example calculates the area-weighted average value of `Velocity u`, with negative values of the variable replaced by zero. Note that this is not the average positive value since zero values contribute to the average.

8.1.1.1.3. areaInt

The `areaInt` function integrates a variable over the specified 2D location. To perform the integration over the total face area, the **None** option should be selected from the Direction drop-down list. If a direction is selected, the result is an integration over the projected area of each face onto a plane normal to that direction. Each point on a location has an associated area that is stored as a vector and therefore has direction. By selecting a direction in the calculator you are using only a single component of the vector in the area-weighting function. Since these components can be positive or negative, depending on the direction of the normal on the location, it is possible for areas to cancel out. An example of this would be on a closed surface where the projected area is always zero (the results returned are not in general zero since the variable values differ over the closed surface). On a flat surface the normal vectors always point in the same direction and never cancel out. The following examples demonstrate use of the function.

- **Function:** `areaInt`, **Location:** `Plane1`, **Variable:** `Pressure`, **Direction:** `None` This example integrates pressure over Plane1. The result returned is the total pressure force acting on Plane1. The magnitude of each area vector is used and so the direction of the vectors is not considered.

- **Function:** `areaInt`, **Location:** `Plane1`, **Variable:** `Pressure`, **Direction:** `Global X`. This example integrates pressure over the projected area of Plane1 onto a plane normal to the X axis. The result is the pressure force acting in the X direction on Plane1. This differs slightly from using the force function to calculate the X-directional force on Plane1 — the force function includes forces due to the advection of momentum when calculating the force on an internal arbitrary plane or a non-wall boundary (such as inlets).

8.1.1.1.4. ave

The `ave` function calculates the arithmetic average (the mean value) of a variable or expression on the specified location. This is the sum of the values at each node on the location divided by the number of nodes. Results are biased towards areas of high nodal density on the location. To obtain a mesh-independent result, use the `lengthAve`, `areaAve`, `volumeAve` or `massFlowAve` functions. The following example demonstrates use of the function.

The average of a vector value is calculated as an average of its magnitudes, not the magnitude of component averages. As an example, for velocity, $\left|v\right|_{\mathrm{ave}} = \frac{\left|v_1\right| + \left|v_2\right|}{2}$

where $\left|v_i\right| = \sqrt{\left(v_{xi}^2 + v_{yi}^2 + v_{zi}^2\right)}$

- **Function:** `ave`, **Location:** `MainDomain`, **Variable:** `Temperature`. This example calculates the mean temperature at all nodes in the selected domain.

8.1.1.1.5. count

The `count` function returns the number of nodes on the specified location. The following example demonstrates use of the function.

- **Function:** `count`, **Location:** `MainDomain`. This example returns the number of nodes in the specified domain.

8.1.1.1.6. length

Computes the length of the specified line as the sum of the distances between the points making up the line. The following example demonstrates use of the function.

- **Function:** `length`, **Location:** `Polyline1`. Calculates the length of the Polyline.

8.1.1.1.7. lengthAve

Computes the length-based average of the variable on the specified line. This is the 1D equivalent of the areaAve function. The result is independent of the nodal distribution along the line since a weighting function assigns a higher weighting to areas of sparse nodal density. The following example demonstrates use of the function.

- **Function:** `lengthAve`, **Location:** `Polyline1`, **Variable:** `Velocity`. This calculates the average velocity on the location Polyline1 using a length-based weighting function to account for the distribution of points along the line.

8.1.1.1.8. lengthInt

Computes the length-based integral of the variable on the specified line. This is the 1D equivalent of the areaInt function. The following example demonstrates use of the function.

8.1.1.1.9. maxVal

Returns the maximum value of the specified variable on the specified locator. Create a user variable if you want to find the maximum value of an expression. The following example demonstrates use of the function.

- **Function:** `maxVal`, **Location:** `Default`, **Variable:** `Yplus`. This returns the maximum `Yplus` value on the `Default` wall boundaries.

8.1.1.1.10. minVal

Returns the minimum value of the specified variable on the specified locator. Create a user variable if you want to find the minimum value of an expression. The following example demonstrates use of the function.

- **Function:** `minVal`, **Location:** `MainDomain`, **Variable:** `Temperature`. These settings return the minimum temperature in the domain.

8.1.1.1.11. probe

Returns the value of the specified variable on the specified point object. The following example demonstrates use of the function.

- **Function:** `probe`, **Location:** `Point1`, **Variable:** `Density`. Returns the density value at `Point1`.

> **Important:**
>
> This calculation should only be performed for point locators described by single points. Incorrect solutions are produced for multiple point locators.

8.1.1.1.12. sum

Computes the sum of the specified variable values at each point on the specified location. The following example demonstrates use of the function.

- **Function:** `sum`, **Location:** `SubDomain1`, **Variable:** `Volume of Finite Volume` Returns the sum of the finite volumes assigned to each node in the location `SubDomain1`. In this case this sums to the volume of the subdomain.

8.1.1.1.13. volume

The `volume` function is used to calculate the volume of a 3D location. The following example demonstrates use of the function.

- **Function:** `volume`, **Location:** `Volume1`. Returns the sum of the volumes of each mesh element included in the location `Volume1`.

8.1.1.1.14. volumeAve

The `volumeAve` function calculates the volume-weighted average of an expression on a 3D location. This is the 3D equivalent of the `areaAve` function. The volume-weighted average of a variable is the average value of the variable on a location weighted by the volume assigned to each point on a location. Without the volume weighting function, the average of all the nodal variable values would be biased towards values in regions of high mesh density. The following example demonstrates use of the function.

- **Function:** `volumeAve`, **Location:** `Volume1`, **Variable:** `Density`. This example calculates the volume-weighted average value of density in the region enclosed by the location `Volume1`.

8.1.1.1.15. volumeInt

The `volumeInt` function integrates the specified variable over the volume location. This is the 3D equivalent of the areaInt function. The following example demonstrates use of the function.

- **Function:** `volumeInt`, **Location:** `Volume1`, **Variable:** `Density`. This calculates the integral of density (the total mass) in `Volume1`.

8.1.1.2. Location

Click to select a location from the drop-down list. Only locations valid for the selected function are available.

8.1.1.3. Variable

Click to select a variable from the drop-down list. Only variables valid for the selected function are available.

For most functions, click in the **Variable** box and enter an expression to use as the variable. The expression can include other variables and any valid CEL (Ansys CFX Expression Language) function (see CEL Functions, Constants and System Variables in the *TurboGrid Reference Guide*). For example, `abs(Velocity u)` could be entered so that the calculation is performed using the absolute values of the variable `Velocity u`.

8.1.1.4. Direction

The `areaInt` function requires a direction to be specified before the calculation can be performed. The `areaInt` function projects the location onto a plane normal to the specified direction (if the direction is not set to `None`), and then performs the calculation on the projected location (direction specification can also be `None`). The direction of the normal vectors for the location is important and cancels out for surfaces such as closed surfaces.

8.1.1.5. Hybrid and Conservative Variables

In **Ansys TurboGrid** there is no difference between hybrid and conservative variables. Leave all controls for selecting between them at their default values.

8.2. Expressions Command

The expression editor is used to create new expressions and modify existing expressions in Ansys TurboGrid. Ansys TurboGrid can use expressions in place of any numeric value (as long as the correct units are returned by the expression). For example, you can use expressions to define object properties or new variables.

Note:

- When a setting is defined by an expression, and the latter evaluates to a quantity that has no units, the software internally applies the default units for that setting.
- In an expression, a term that has no units can be added to a term that has angular units, in which case the software internally applies radians to the term that has no units.

To create or edit an expression, select **Tools** > **Expressions** from the main menu. The **Expression Editor** dialog box is displayed.

8.2.1. Expression Editor Dialog Box

The **Expression Editor** dialog box contains:

- A tree view for viewing and managing expressions (The tree view has a shortcut menu with commands for creating/deleting/modifying expressions, as described below.)
- A details view for creating/editing/displaying the definition of the new/selected expression

  (You can find a list of valid CEL expressions and constants in Ansys CFX Expression Language in the *TurboGrid Reference Guide*.)
- A **Value** box to show the value of the selected expression when the expression does not contain variables and evaluates to a single value.

You can right-click an existing expression or a blank area in the tree view portion of the **Expression Editor** dialog box in order to access a shortcut menu with the following commands:

Table 8.1: Expression Editor Shortcut Menu Commands

Command	Description
Edit	Displays the definition of the selected **Expression** in the details view portion of the **Expression Editor** dialog box, and gives it the keyboard focus. An alternative way to do the same is to double-click an expression in the tree view.

Command	Description
Insert > Expression	Opens the **New Expression** dialog box where you can enter a name for a new expression. After you enter a name, you must provide and apply a definition in order to finish creating the new expression. When selecting an expression name: • You cannot create an expression with the same name as an object. • You cannot create an expression with the same name as a variable.
Edit in Command Editor	Invokes the **Command Editor** dialog box populated with the selected expression.
Delete	Deletes the selected expression.
Duplicate	Creates a duplicate of the selected expression after obtaining the name of the new expression from the **New Expression** dialog box.
Use as Workbench Input Parameter [a]	Makes the selected expression a Workbench input parameter so that it is available in, and controllable by, Ansys Workbench. The Workbench input parameter is initialized by the expression definition specified in Ansys TurboGrid. The definition must be a function of only constants; units are permitted. It is not possible to edit the value of an expression in Ansys TurboGrid while the expression is a Workbench input parameter.
Use as Workbench Output Parameter[a] [b]	Makes the selected expression a Workbench output parameter, causing it to be output from Ansys TurboGrid for use in Ansys Workbench.
Deselect as Workbench Parameter[a][b]	Removes the Ansys Workbench parameter status from the selected expression.

[a] You can enable/disable the use of a Workbench input parameter directly for many settings in the object editor. For details, see Object Editor (p. 20).

[b] For details on Workbench parameters, see Working with Parameters and Design Points in the *Workbench User's Guide*

When defining an expression:

- You cannot use a user variable.
- Constants require units inside square brackets.

- You can right-click in the details view to access a shortcut menu for inserting variables, locators, expressions, and functions into your equation.

8.2.1.1. Expression Editor Example

To define an expression for distance from the X axis:

1. Select **Tools** > **Expressions** from the main menu to open the **Expression Editor** dialog box.
2. When the **New Expression** dialog box appears, enter the name `radial` and click **OK**.
3. In the details view portion of the **Expression Editor** dialog box, enter the following expression definition:

   ```
   sqrt(Y^2+Z^2)
   ```

4. Click **Apply** to create the expression.

Note that the **Value** box does not show a value for the expression because the expression does not evaluate to a single value.

The expression created in this example (`radial`) is used in the Variable Editor Example (p. 90).

8.3. Variables Command

The variable editor is used to create new user variables and modify existing variables in Ansys TurboGrid.

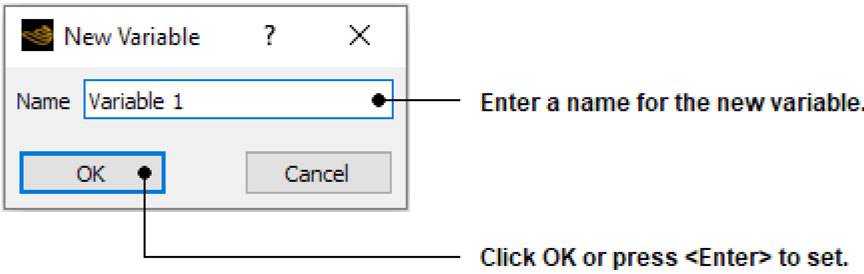
To create or edit a variable, select **Tools** > **Variables** from the main menu. The **Variable Editor** dialog box is displayed.

8.3.1. Variable Editor Dialog Box

All icons become active when you select a variable from the **Variable** list. Alternatively, you can right-click a variable in the list to see the same options. Each icon and its function is described in the following table.

Icon	Description
	*New* opens the **New Variable** dialog box where you can enter a name.
	*Edit* displays the selected **Variable** in the **Expression** and/or **Units** box where you can edit it.
	*Copy* makes a duplicate of the selected variable and opens the **New Variable** dialog box where you can enter a name for the copy.
	*Delete* removes the selected variable from the list.
	*Use Hybrid Values* sets all variables to hybrid values.
	*Use Conservative Values* sets all variables to conservative values.

When you click *New* to create a new variable, the **New Variable** dialog box is displayed.



8.3.1.1. Name

There are a few guidelines to follow when selecting a variable name.

- You cannot create a variable with the same name as an object.
- You cannot create a variable with the same name as an expression. For example, if you have an expression named `Radius`, you must choose a different variable name for that expression.
- Within the Ansys CFX Expression Language some variables are known by short names to save typing in the full variable name. For example, `p` refers to `Pressure`. Although it is possible to create a variable with the same name as an abbreviated variable, it is ignored. For example, if you use a variable with the name `p` in an expression, it returns the value `Pressure` in all cases, no matter what the definition of the variable is.

8.3.1.2. Type

Click *Edit* to edit both the fundamental and user variables. Select the expression used to define the variable from a list of existing expressions for user variables. The expressions available from this list are those which you have created in the expression editor. For details, see Expressions Command (p. 86).

For fundamental variables, the units are changeable. Click next to the Units box to see the available units for the selected variable. This means, for example, that you could create a legend which uses alternative angle units (such as degrees or radians) by clicking on the icon and selecting new units.

Note:

These settings override the global units setting, defined in the Edit Menu. For details, see Options Command (p. 43).

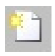
The variable type used affects all quantitative calculations and plots in Ansys TurboGrid.

8.3.1.3. Hybrid and Conservative Variable Values

This is useful only for advanced postprocessing and is not relevant for Ansys TurboGrid. It is included to maintain consistency with other Ansys CFX products. Refer to the CFD-Post documentation for more information on the differences between hybrid and conservative variable values.

8.3.1.4. Variable Editor Example

In this example, an isosurface that has a fixed radial distance from the X axis is created using the expression defined in the Expression Editor Example (p. 88).

1. Select **Tools** > **Variables** from the main menu to open the variable editor.
2. Click *New*  to create a new variable. When the **New Variable** dialog box appears enter the name `Radial Distance` and click **OK**.
3. In the variable editor, use the **Expression** drop-down list to select the expression `radial` which you created earlier. Click **Apply** to create the new variable.

This variable now appears in the list of available variables and can be used like any other variable. Notice that the variable type is listed as **User**.

You can now create an isosurface using this variable. For details on creating isosurfaces, see Isosurface Command (p. 61).

1. Select **Insert** > **User Defined** > **Isosurface** from the main menu, enter a name, then click **OK** on the **New Isosurface** dialog box.
2. In the **Geometry** tab for the isosurface, set **Variable** to `Radial Distance`.
3. Set **Value** to `20 m`. This is a suitable value for the Rotor 37 geometry used in Tutorial 1. You may need to alter this value to something more sensible depending on the geometry you are viewing.
4. Click the **Color** tab and set **Mode** to `Variable`.
5. Select a sensible variable (for example, `Maximum Face Angle` or `Axial Distance`) with which to color the isosurface.
6. Set **Range** to `Local` so that the full color range is used on the isosurface.
7. Click **Apply** to create the isosurface object.

You should now see the isosurface in the viewer. All points on the isosurface are a distance of 20 m (or whatever value you used in the **Value** box) from the X-axis.

8.4. Command Editor Command

The **Command Editor** dialog box can be used to create or modify any of the objects in Ansys TurboGrid using the CFX Command Language. For further details, see CFX Command Language in the *TurboGrid Reference Guide*.

Power Syntax can be entered and processed in the **Command Editor** dialog box. For details, see Power Syntax in the *TurboGrid Reference Guide*. Power Syntax commands should be proceeded by the ! symbol.

In addition, any valid CCL command can be typed and processed in the **Command Editor** dialog box. For details, see Command Actions in the *TurboGrid Reference Guide*. Action Commands should be preceded by the > symbol.

To create an object using CCL, select **Tools** > **Command Editor** from the main menu. The **Command Editor** dialog box is displayed.

To edit an existing object using CCL, right-click the object in the object selector and select **Edit in Command Editor** from the shortcut menu. The CCL definition of that object is automatically displayed and can be edited to alter the object properties.

You can access the following basic editing tools by right-clicking in the **Command Editor** dialog box:

- **Undo**

  Undoes the last edit action.

- **Redo**

  Redoes the most recently undone edit action.

- **Cut**

  Cuts the selected text and places it on a clipboard.

- **Copy**

  Places the selected text on a clipboard.

- **Paste**

  Pastes the clipboard text at the insertion point, or replaces the selection.

- **Clear**

  Clears all of the contents of the **Command Editor** dialog box.

- **Select All**

  Selects all of the contents of the **Command Editor** dialog box.

- **Find**

  Makes a search tool appear at the bottom of the **Command Editor** dialog box. Enter a search term and click either **Next** or **Previous** to search upwards or downwards from the insertion point or text selection. To hide the search tool, press <Esc>.

8.5. Reset Inlet/Outlet Points Command

This command moves the inlet and outlet points to their default locations.

Chapter 9: Help Menu

The **Help** menu has the following commands:

TurboGrid

Opens TurboGrid User's Guide (p. 1).

Contents

Opens a page that lists various help resources associated with this product.

[various help resources]

Each of these commands goes directly to a particular help resource.

Ansys Product Improvement Program

Provides a brief description of, and enables you to control participation in, the Ansys Product Improvement Program.

About TurboGrid

This gives the point releases and software patches that are installed.

Help on Help

Opens documentation about the help system: Ansys TurboGrid Help and Conventions in the *TurboGrid Introduction*.

Chapter 10: Ansys TurboGrid Workflow

This chapter describes:

10.1. Introduction

This chapter explains the steps required to create a mesh. A brief overview is given in the next section. The remaining sections of this chapter cover each major step in detail.

10.2. Steps to Create a Mesh

To create a mesh:

1. Define the geometry.

   For details, see Geometry (p. 97).

2. Define the topology object(s) by selecting a method and optionally changing other settings.

   For details, see Topology (p. 128).

3. Optionally modify the `Mesh Data` object to adjust the number of nodes.

   If you plan to make a fine (high-resolution) mesh, you can optionally set the mesh density at a later time in order to minimize processing time while establishing the topology. Keep in mind that changing the mesh density can affect the mesh quality.

   For details, see Mesh Data (p. 145).

4. Optionally modify the layer objects to improve mesh quality.

For details, see Layers (p. 159).

5. Generate the 3D mesh.

   By default, the `3D Mesh` object is initially unsuspended, so mesh generation happens automatically whenever the `Layers` object is (re)processed. For details, see 3D Mesh (p. 165).

6. Optionally investigate the mesh and refine any of the above objects as necessary.

   To help identify problem areas of the mesh, mesh analysis tools are available. For details, see Mesh Analysis (p. 166).

   If you change any objects that affect the mesh, you must generate the mesh again.

7. Save the mesh to a file.

   For details, see Save Mesh Command (p. 34).

In general, you should proceed from top to bottom in the object selector, or from left to right on the toolbar.

Figure 10.1: Object Selector

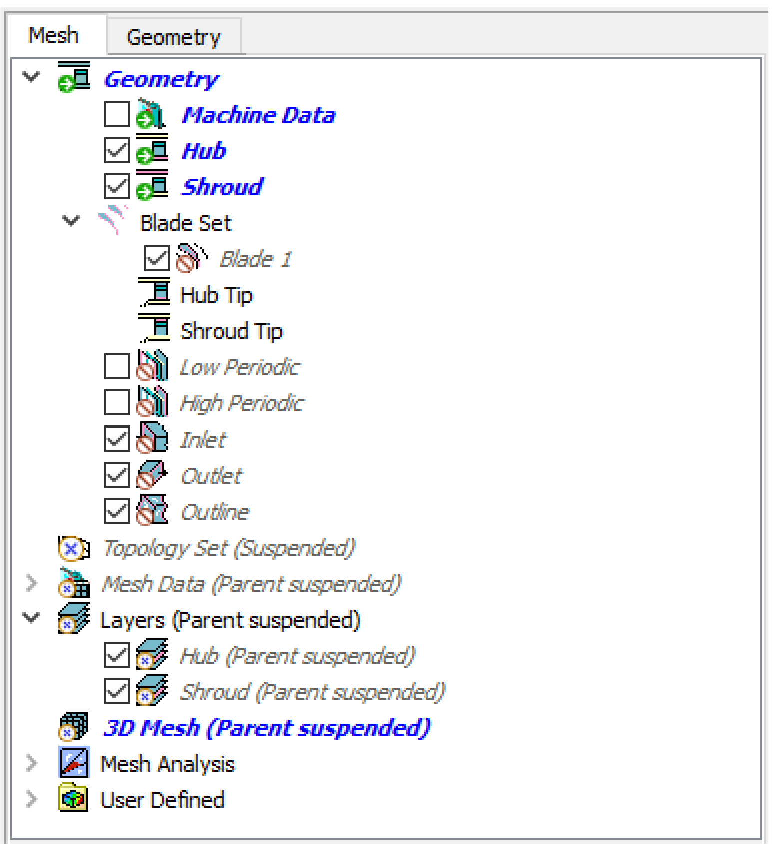
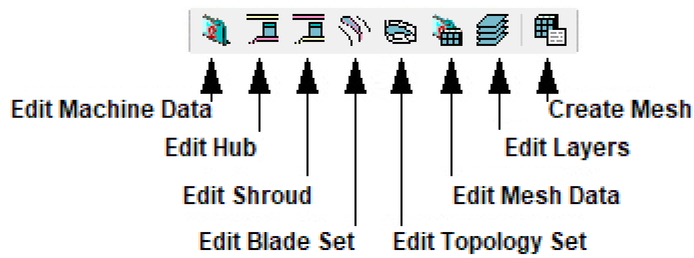


Figure 10.2: Part of Toolbar



The other objects in the object selector are used for visualization and inspection purposes.

Not every object in the object selector requires a definition before a mesh can be created. However, every object represented by an icon on the toolbar must be defined.

10.3. Geometry

Geometry can be defined using two types of source data:

- CAD data (for example, from BladeEditor or a Parasolid file)
- Profile point data (for example, from BladeEditor or BladeGen)

The geometry of the blades and main passage can be made using either type of source data. The geometry of any secondary flow paths must be made using CAD data.

After you have finished creating the geometry, you may want to hide some or all of it in the **Mesh** workspace to allow you to see the topology clearly. You can control the visibility of a geometry object by toggling the check box next to the object in the object selector. You can also right-click an object and use shortcut menu commands. The toolbar allows you to hide and unhide all geometry objects via the *Hide all geometry objects*  and *Unhide_geometry_objects*  icons.

The following topics are discussed.

10.3.1. Defining Geometry from a CAD Source

CAD data can be used for the geometry of the blades and main passage, and for secondary flow paths.

You can load CAD data from:

- A BladeEditor Geometry cell in Workbench, or

- A Parasolid or other CAD file in stand-alone mode.

Note:

If TurboGrid crashes while loading CAD geometry, there might be a failure in creating internal representations of the CAD blade surfaces for meshing. A possible workaround is to perform one or more of the following steps, using the **Command Editor** dialog box to make CCL changes:

- Before loading the geometry set Stop Before Blade Approximate Surface Generation in BLADE SET to true.
- Load the geometry.
- Increase Surface Parameterization Tolerance Multiplier in BLADE SET to 10.0.
- Increase Surface Parameterization Number Of Splits in BLADE SET to 200.
- Change Parametric Surface Generation Smoother in the Blade object(s) to Angle-based.
- Change Display Trimmed Surfaces in the Blade object(s) to false.
- Reset Stop Before Blade Approximate Surface Generation in BLADE SET to false.

The following topics are discussed.

10.3.1.1. Loading CAD Data From BladeEditor

To load CAD data from a BladeEditor Geometry cell:

1. In Ansys Workbench, attach a Geometry cell (containing a BladeEditor geometry) upstream of the Turbo Mesh cell.
2. Specify which exported blade(s) to load into TurboGrid by editing the Turbo Mesh cell properties. See TurboGrid Turbo Mesh Cell Properties in the *TurboSystem User's Guide* for more information.

Note:

- This method is applicable only for native BladeEditor blades (not general CAD geometry imported into DesignModeler). If a geometry that is to be imported from BladeEditor fits within a cube of edge length 10 cm, the DesignModeler session unit (see Units Menu in the *Ansys DesignModeler User's Guide*) should be set to **Micrometer** for better geometry precision. For details, see Export to Ansys TurboGrid in the *TurboSystem User's Guide*.

- When a blade is passed from BladeEditor to TurboGrid (so that a CFD mesh can be made by TurboGrid), two versions of the geometry are passed:
  - The "display geometry", which is the true geometry designed in BladeEditor, and
  - The "meshing geometry", which is used by TurboGrid in constructing the mesh.

    The meshing geometry is slightly different from the display geometry because it must meet TurboGrid's mesh generation requirements. For example, the meshing geometry has extended surfaces that overlap so that TurboGrid can reliably locate surface intersections.

  In the **Mesh** workspace, the display geometry can be visually compared with the mesh. In the **Geometry** workspace, you can view both the display geometry and the meshing geometry by turning on visibility of the corresponding CAD objects listed in the tree.

  In particular, for filleted blades from BladeEditor, there will typically be a noticeable difference between the mesh and the display geometry shown in the **Mesh** workspace, and a noticeable difference between the meshing geometry and display geometry shown in the **Geometry** workspace. The overall difference in fillet geometry (for example, the amount of flow blockage) can be affected by changing fillet-related parameters in BladeEditor.

10.3.1.1.1. Troubleshooting Fillets from BladeEditor

- If you experience a problem with mesh quality near a cut-off edge (LE or TE) and near a blade fillet, then adjusting the following CCL parameters might help:
  - `Number Of Hub To Shroud Constant Span Splits`
  - `Mesh Curves Tesselation Number Of Power Samples = 8`
  - `Mesh Curves Tesselation Number Of Power Samples Override = true`
- If a blade fillet looks wavy where it meets the hub/shroud (more likely for cases with high hub/shroud curvature), increasing the following CCL parameters might help:
  - `Turbo Transform Background Mesh Size`
  - `Turbo Transform Background Mesh Size For Topology`

10.3.1.2. Loading CAD Data From a CAD File

To load CAD data from a file (for example, a Parasolid file):

- In TurboGrid in stand-alone mode, in the **Geometry** workspace, open the geometry browser, set **Input Mode** to **CAD** (or **Profile Points With CAD**) and configure the applicable settings in the geometry browser.

Note:

When a CAD blade is loaded from a file into TurboGrid, two versions of the geometry might be displayed:

- When available, the "display geometry", which is the best representation of the true geometry, is displayed in the **Geometry** workspace.
- The "meshing geometry", which is used by TurboGrid in constructing the mesh, is displayed in the **Mesh** workspace.

  The meshing geometry is slightly different from the display geometry because it must meet TurboGrid's mesh generation requirements. For example, the meshing geometry has extended surfaces that overlap so that TurboGrid can reliably locate surface intersections.

In the **Mesh** workspace, the display geometry can be visually compared with the mesh. In the **Geometry** workspace, you can view both the display geometry (when available) and the meshing geometry by turning on visibility of the corresponding CAD objects listed in the tree.

10.3.1.3. Objects in the Geometry Workspace (CAD Mode)

The **Geometry** workspace tree holds CAD objects and can be used to visually check that the correct CAD geometry is imported.

The main objects in the **Geometry** workspace are:

The Topological Entity Instances Object	This object holds other objects that each define a key geometric feature needed by TurboGrid. The objects under `Topological Entity Instances` hold references to CAD objects and/or other settings.
The CAD Families Object	This object holds other objects that each define a geometric feature that was imported from a CAD file. You can select different features to view them in the **3D Viewer** and to see their properties in the object editor. Specific features from the CAD file can be used by TurboGrid after they are assigned to objects under the `Topological Entity Instances` object.

10.3.1.4. Geometry Browser Settings (CAD Mode)

To access the geometry browser, switch to the **Geometry** workspace and then, if necessary, double-click a high-level object (either the `Topological Entity Instances` object or the `CAD Families` object) in the object selector.

The geometry browser CAD-related settings, which are available for **CAD** mode and for **Profile Points With CAD** mode, are described next:

10.3.1.4.1. CAD Input Definition

Set **File Name** to the name of a CAD file (for example, `*.x_b` exported from BladeEditor).

Optionally select **Named Selection Processing** to have TurboGrid create CAD families for named selections in the CAD file. You should select this option for Parasolid and ICEM CAD files, except for Parasolid files written by SpaceClaim; for the latter, you should clear this check box in order to have CAD entities organized into families corresponding to the organization in SpaceClaim.

Optionally select **Specify CAD Geometry Units** and specify units for **CAD Length Units**. If you do not specify units, TurboGrid will attempt to determine the units from the file contents; in such cases you should confirm that the units are correct by observing the resulting geometry in the **3D Viewer**.

10.3.1.4.2. Geometry Setup (CAD Mode)

10.3.1.4.2.1. Rotation (CAD Mode)

These settings are similar to the **Rotation** settings of the Machine Data object. For details, see Rotation (p. 107).

10.3.1.4.2.2. # of Bladesets (CAD Mode)

A rotating machine component is made up of adjacent blades that are equally spaced around the circumference of the machine. The Theta extent of one blade set is calculated as 360 degrees divided by the number of main blades. Many rotating machine components have secondary and tertiary blades that are placed between the main blades. These are often called splitter blades. A blade set contains one main blade and optional splitter blades that repeat cyclically around the axis of the rotating machine component.

For rotating machine components without any splitter blades, the number of blade sets equals the total number of blades.

TurboGrid creates a mesh for one blade set only. The mesh can be copied and rotated using an Ansys CFX Pre-processor, if necessary, before it is solved in an Ansys CFX Solver.

10.3.1.5. Limitations of CAD Geometry Usage

TurboGrid uses a specific set of geometric inputs. Because CAD geometry is more general, there are some CAD features that cannot be used by TurboGrid. A list of some such features follows:

- You cannot transform imported CAD geometry.
- Inlet and outlet trimming is governed by selected CAD curves.

10.3.2. Defining Geometry from Profile Points

Profile point data can be used for the geometry of the blades and main passage.

Note:

Profile point data cannot be used to define secondary flow paths. However, you can add secondary flow paths from a CAD source.

To define the geometry in TurboGrid using profile point data, you can do any of the following:

- Load profile point data from a BladeEditor Geometry cell (in Workbench):

  1. In BladeEditor, create an ExportPoints feature. For details, see Export to Ansys TurboGrid in the *TurboSystem User's Guide*.
  2. In Workbench, attach a Geometry cell upstream of the Turbo Mesh cell.
  3. Set the Turbo Mesh cell properties. For details, see TurboGrid Turbo Mesh Cell Properties in the *TurboSystem User's Guide*.
  4. In TurboGrid, in the **Geometry** workspace, open the geometry browser, set **Input Mode** to **Profile Points** (or **Profile Points With CAD**) and configure the applicable settings in the geometry browser.
  5. Review the settings in the `Geometry` objects in the **Mesh** workspace, making and applying changes as required. For details, see The Geometry Objects in the Mesh Workspace (p. 105).

- Load profile point data from a BladeGen Blade Design cell (in Workbench):

  1. In Ansys Workbench, attach a Blade Design cell upstream of the Turbo Mesh cell.
  2. In TurboGrid, review the settings in the geometry browser in the **Geometry** workspace, making and applying changes as required. Note that **Input Mode** should be set to **Profile Points**.
  3. Review the settings in the `Geometry` objects in the **Mesh** workspace, making and applying changes as required. For details, see The Geometry Objects in the Mesh Workspace (p. 105).

  Note that, in this case, some of the geometry objects in the **Mesh** workspace (such as Machine Data, Hub, Shroud, Blade Set, and Blade) have their settings taken from the upstream cell, and changes to these settings cannot be made in Ansys TurboGrid. Other settings in geometry objects are taken from the upstream cell by default, but can be overridden. To override a particular setting, select the appropriate check box in the object editor or the geometry browser. For example, although the machine rotation axis is, by default, taken from the upstream cell, you can reverse the rotation axis direction by first selecting the **Override Theta Direction for Topology** check box in the **Machine Data** object editor or the geometry browser, and then selecting **Right Handed** or **Left Handed**.

- Load profile point data from a BladeGen `*.inf` file:

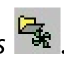
  1. Select **File** > **Load TurboGrid Init File** or click *Load TurboGrid Init File*  then select and open an `*.inf` file.

     The `*.inf` file contains machine data, the names of geometry definition files, the choice of coordinate system, and leading/trailing edge settings. For details on `*.inf` files, see BladeGen.inf File (p. 28).

  2. Review the settings in the geometry browser in the **Geometry** workspace, making and applying changes as required. Note that **Input Mode** should be set to **Profile Points**.

  3. Review the settings in the `Geometry` objects in the **Mesh** workspace, making and applying changes as required. For details, see The Geometry Objects in the Mesh Workspace (p. 105).

- Load profile point data from separate curve (`.curve` or `.crv`) files:

  1. Select **File** > **Load Profile Points** or click *Load Profile Points* .

  2. In the geometry browser in the **Geometry** workspace, specify curve files and make and apply other changes as required. Note that **Input Mode** should be set to **Profile Points**.

  3. Review the settings in the `Geometry` objects in the **Mesh** workspace, making and applying changes as required. For details, see The Geometry Objects in the Mesh Workspace (p. 105).

  For details on curve files, see Curve File (p. 29).

- In the **Mesh** workspace, define the `Machine Data` object, then each of the remaining geometry objects separately.

  Note that, in the geometry browser in the **Geometry** workspace, **Input Mode** must be set to **Profile Points**.

Using the geometry browser with **Profile Points** mode selected is a convenient way to define multiple geometry objects, compared to editing each geometry object separately in the **Mesh** workspace. It is useful when you first open TurboGrid and when you want to use the same settings on a slightly modified geometry.

The information specified in the geometry browser is used to overwrite the corresponding information in the appropriate `Geometry` objects in the **Mesh** workspace. No other previously-defined settings are affected. For example, if a control angle is defined for the `Hub` object, it will not be changed by using the geometry browser.

The curve file for the blades can specify one or more blades for the blade row; one blade is generated for each blade defined in the curve file. If blade names are not included in the curve file, then names will be generated in the form Blade *n*, where *n* is an integer.

Note:

If a curve file that specifies multiple blades (defining a blade set) is used to define a particular blade object (stored in an object under the `Blade Set` object in the **Mesh** workspace), only the first blade in the file is used.

The following topics are discussed:

10.3.2.1. Objects in the Geometry Workspace (Profile Points Mode)

The objects in the **Geometry** workspace are not used when loading geometry from profile points. They are used only to verify that the geometry is imported correctly from a CAD source. For details, see Objects in the Geometry Workspace (CAD Mode) (p. 100).

10.3.2.2. Geometry Browser Settings (Profile Points Mode)

To access the geometry browser, switch to the **Geometry** workspace and then, if necessary, double-click a high-level object (either the `Topological Entity Instances` object or the `CAD Families` object) in the object selector.

The geometry browser settings for Profile Points mode are described next:

10.3.2.2.1. Point Data Definition

10.3.2.2.1.1. TurboGrid Curve Files

To specify a curve file, set a filename using a path relative to the working directory. You can click the corresponding *Browse* icon to select a curve file using a browser.

10.3.2.2.1.2. Coordinates and Units

Click next to the corresponding boxes to select the coordinates, angle units, and length units used in the hub, shroud, and blade files. If any of these are not the same for all three geometry files, use the object selector in the **Mesh** workspace to individually define them.

10.3.2.2.2. Geometry Setup (Profile Points Mode)

10.3.2.2.2.1. Rotation (Profile Points Mode)

These settings are similar to the **Rotation** settings of the Machine Data object. For details, see Rotation (p. 107).

10.3.2.2.2.2. # of Bladesets (Profile Points Mode)

A rotating machine component is made up of adjacent blades that are equally spaced around the circumference of the machine. The Theta extent of one blade set is calculated as 360 degrees divided by the number of main blades. Many rotating machine components have secondary and tertiary blades that are placed between the main blades. These are often called splitter blades. A blade set contains one main blade and optional splitter blades that repeat cyclically around the axis of the rotating machine component.

For rotating machine components without any splitter blades, the number of blade sets equals the total number of blades.

TurboGrid creates a mesh for one blade set only. The mesh can be copied and rotated using an Ansys CFX Pre-processor, if necessary, before it is solved in an Ansys CFX Solver.

10.3.2.2.2.3. Leading/Trailing Edge Definition on the Blade (Profile Points Mode Only)

Optionally select `Cut-off or square`. For details, see Cut-off or square (p. 117).

10.3.3. Defining the Geometry from Profile Points and a CAD Source

In stand-alone mode, you can use a combination of geometry sources:

- Profile point data for the blades and main passage
- A CAD file, such as a Parasolid file, for secondary flow passages.

To do this:

- Optionally open a TurboGrid Initialization File (**File** > **Load TurboGrid Init File**).

  This step can save time by configuring file names in the settings of the geometry browser.

- In the **Geometry** workspace, open the geometry browser and select **Profile Points With CAD**.
- Configure the settings in the geometry browser.

  These settings are a superset of those for the **CAD** and **Profile Points** modes, which are detailed in the following sections:

  – Defining Geometry from a CAD Source (p. 97)

  – Defining Geometry from Profile Points (p. 102)

10.3.4. The Geometry Objects in the Mesh Workspace

The **Mesh** workspace can be used to set up geometry from a source of profile points. It can also control the visibility of the geometric objects that TurboGrid generates, regardless of the source of geometric data (profile points or CAD data).

The `Geometry` objects in the **Mesh** workspace are outlined below with a summary of the function performed by each.

The Machine Data Object (p. 106)	This object defines the geometry and rotation properties of the rotating machine.
The Hub and Shroud Objects (p. 108)	These objects define the hub of the rotating machine.
The Blade Set Object and Blade Objects (p. 112)	These objects define the blade(s) of the rotating machine.
The Hub Tip and Shroud Tip Objects (p. 120)	These objects define the hub tip and shroud tip of the blade(s).
The Low Periodic and High Periodic Objects (p. 122)	These objects control the methods for creating the low and high periodic surfaces.
The Inlet and Outlet Objects (p. 122)	These objects define the locations and profiles of the inlet and outlet domains of the mesh.
The Outline Object (p. 128)	This object controls the display of the outline of the geometry.

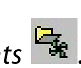
10.3.4.1. The Machine Data Object

The `Machine Data` object contains geometric data that applies to the entire turbo machine (for example, the location of the rotation axis). Defining the machine data is an essential step in creating a mesh in Ansys TurboGrid, and can be accomplished by one of the following methods:

- Editing the `Machine Data` object.
- In Ansys Workbench, attaching a Geometry or Blade Design cell upstream of the Turbo Mesh cell.

  Note that, in this case:

  - The **Pitch Angle** setting is taken from the upstream cell, and cannot be changed in Ansys TurboGrid.
  - The **Rotation** settings are taken from the upstream cell by default, but can be overridden by selecting the **Rotation** check box.
  - The **Base Units** setting can be changed and is not affected by changes to the upstream cell.
- Loading a BladeGen `.inf` file by selecting **File** > **Load TurboGrid Init File** or clicking *Load TurboGrid Init File* .

  You can define machine data, hub, shroud, and blade objects by loading a BladeGen `.inf` file. For details, see Load TurboGrid Init File Command (p. 29).
- Loading curve files by selecting **File** > **Load Profile Points** or clicking *Load Profile Points* .

You can define the machine data, hub, shroud, and blade objects by loading curve files. For details, see Load Profile Points Command (p. 30).

To edit the machine data, open the `Machine Data` object from the object selector or click *Edit Machine Data* .

10.3.4.1.1. Data Tab

10.3.4.1.1.1. Pitch Angle

A rotating machine component is made up of blade sets that are identical, adjacent to each other, and equally spaced around the entire circumference. In the context of Ansys TurboGrid, the *pitch angle* is the Theta extent of one blade set. To calculate the pitch angle based on the number of blade sets per 360°, set **Method** to `Bladeset Count`; the pitch angle is then calculated as 360° divided by the number of blade sets. If you need to specify the pitch angle directly (as in the Deformed Turbine tutorial), set **Method** to `Specified Angle`.

Ansys TurboGrid creates a mesh for one blade set only. The mesh can be copied and rotated using an Ansys CFX Pre-processor, if necessary, before it is solved in an Ansys CFX Solver.

10.3.4.1.1.2. Rotation

The **Method** for defining the rotation axis can be set to one of the following:

- **Principal Axis** - select the X, Y, or Z axis. Select **Right Handed** or **Left Handed** to indicate the direction of Theta.

  If you are running TurboGrid in Workbench and need to change the direction of Theta (for example, if the available topologies do not match the geometry in terms of blade ordering within a blade set), select **Override Theta Direction for Topology** and then select the appropriate setting: **Right Handed** or **Left Handed**.

- **Rotation Axis** - define a custom axis by entering the Cartesian coordinates of two points on the axis. The coordinates of the points can be typed in or selected using the  icon in the object editor.

10.3.4.1.1.3. Units

Base units are applicable only when using profile point data as a geometry source.

It is necessary to specify base units because the geometry kernel has an optimal range for storing numbers between 1 and 500. If, for example, the geometry data is given in mm, but the machine is on the order of 10 m in size, the numbers stored in mm would be outside of the optimal range. In this case, the base units should be cm (default), to reflect the scale of the machine.

The base units specified in the `Machine Data` object are used only to adjust the range of numbers to suit the geometry kernel; they are not used to interpret geometric data contained in files (for example, the hub curve file), nor are they used to specify the units of an exported mesh.

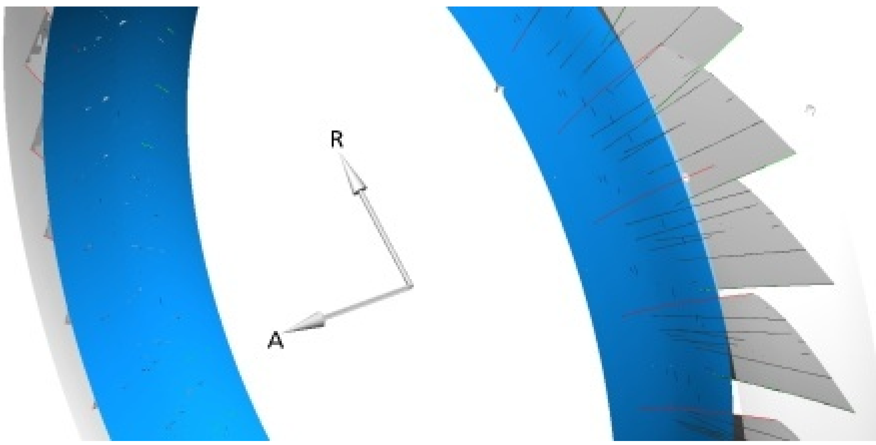
10.3.4.1.1.4. Machine Type

Set **Machine Type** to an appropriate value or leave it set to `Other`. The value will be used by TurboGrid to help choose appropriate topology templates.

10.3.4.1.2. Rotation Axis Visibility

By turning on the visibility of the `Machine Data` object, the rotation axis becomes visible in the viewer, along with a sample radial direction vector. An example is shown in Figure 10.3: Rotation Axis (p. 108).

Figure 10.3: Rotation Axis



The visibility of the `Machine Data` object is off by default. For details on setting visibility, see Visibility Check Box (p. 20).

10.3.4.2. The Hub and Shroud Objects

The hub is the surface of the machine closest to the axis of rotation. It defines the inner fluid flow surface. To edit or define the machine component's hub, edit the `Hub` object from the selector or click *Edit Hub* .

The shroud is the surface of the machine farthest from the axis of rotation. It defines the outer fluid flow surface. To edit or define the machine component's shroud, edit the `Shroud` object from the selector or click *Edit Shroud* .

The hub and shroud can be defined only after the machine data has been defined, unless all of these objects are defined together using the **Profile Points** settings in the geometry browser (in the **Geometry** workspace). For details, see Defining Geometry from Profile Points (p. 102).

10.3.4.2.1. Data Hub and Data Shroud Tabs

10.3.4.2.1.1. Coordinate System and Hub File Definition

Set **Coordinates**, **Angle Units**, and **Length Units** according to the data in the hub/shroud file. The hub/shroud file must contain data points that define the hub/shroud curve. To specify a hub/shroud file, set **File Name** using a path relative to the working directory. You can click the *Browse* icon to select a curve file using a browser.

The hub/shroud curve runs upstream to downstream and must extend upstream of the blade leading edge and downstream of the blade trailing edge. The points must be listed in the file line by line, in free-format ASCII style, in order from upstream to downstream.

For example:

```
-1.0  0.0  1.0
2.0  0.0  1.0
```

No names or labels may be present in the file.

Note:

If the hub/shroud file was used in earlier versions of CFX-TurboGrid, you can use the file again in this release without any modifications. However, any data in the fourth field formerly used for parametric values is now ignored.

The **Curve Type** setting has the following options for defining the type of hub/shroud curve.

- `Bspline`, the default, means that a smooth curve is interpolated using the points listed in the hub/shroud file. This method may be necessary if the hub/shroud curve is defined with a small number of points.
- `Piece-Wise Linear` means that the points listed in the hub/shroud file are connected to one another with straight lines.

10.3.4.2.1.2. Curve or Surface Visibility

By default, the hub/shroud surface is visible while the hub/shroud curve is not visible. To see the curve, the following steps could be taken in the **Data Hub** tab:

1. Select the **Curve or Surface Visibility** > **Show Curve** check box.
2. Clear the **Curve or Surface Visibility** > **Show Surface** check box.
3. Click **Apply**.
4. Right-click the `Hub/Shroud` object and select **Render (Properties)** > **Edit Options**.
5. On the **Render** tab, select the **Draw Lines** check box.
6. Click **OK**.

10.3.4.2.1.3. Reread Button

If the file that defines the hub/shroud has been modified since it was last loaded, clicking the **Reread** button will cause the hub/shroud data to be reloaded. The `Blade Set` object and individual blade objects also have this feature.

10.3.4.2.2. Transform Tab

This tab contains specifications for rotational and translational transforms that adjust the geometry. The settings on this tab adjust the hub/shroud geometry only. The `Blade Set` object and individual blade objects can also have their coordinates transformed using the corresponding tabs of those objects.

10.3.4.2.2.1. General Rotation

Specify an axis via two points and a rotation angle about the axis.

10.3.4.2.2.2. Translation

Specify the direction and distance to translate the geometry. The direction may be any one of: `Machine Axis`, `X Axis`, `Y Axis`, `Z Axis`, or `Vector` (any specified direction).

10.3.4.2.2.3. Axial Rotation

Specify the angle to revolve the geometry about the machine axis.

10.3.4.2.3. Hub Regions and Shroud Regions Tab

You can create regions on the hub and/or shroud within the inlet and/or outlet domains. Each region is defined either by two points on the hub curve or two points on the shroud curve.

Hub/shroud regions can be used to precisely position (in the streamwise direction) mesh elements at the hub/shroud so that mesh nodes are placed at key locations, facilitating the subsequent addition of geometric features at those locations (using other software). The hub/shroud regions affect the distribution of mesh elements along the inlet/outlet domain (in the streamwise direction), generally causing elements to be distributed finely (in the streamwise direction) near the region borders. This fine distribution could potentially help to resolve flow details near subsequently added geometric features.

To add a new region, click *New*  or right-click in the list and select **New**. (To delete an existing region, select the region in the list and click *Delete*  or right-click a region in the list and select **Delete**.)

Each region is specified according to pairs of region boundary points on the hub or shroud curve. Although the points within a pair are named "Start" and "End", there is no directionality; the points can be swapped without effect. When specifying the boundary points for a region, choose the **Method** most appropriate for the geometry:

- `Set A (Estimate R)`

  Choose this method if the hub or shroud curve extends mainly in the axial direction (or at least not very closely in the radial direction).

- `Set R (Estimate A)`

  Choose this method if the hub or shroud curve extends mainly in the radial direction (or at least not very closely in the axial direction).

Enter point coordinates using the keyboard or, after clicking on a coordinate field, click (with *Select*  active) a point on the hub/shroud curve in the **3D Viewer**.

When you click **Apply**, the points are defined. Each point is represented visually in the **3D Viewer** by an octahedron on the low-Theta edge of the hub/shroud.

When a point is visible in the **3D Viewer**, you can adjust its location with the mouse by holding **Ctrl**+**Shift** while dragging.

The **Hub Region Proximity Tolerance**/**Shroud Region Proximity Tolerance** setting controls the minimum element size in the streamwise direction. This setting can normally be left at its default value, which is controlled by a preference. For details, see TurboGrid Options (p. 44).

After you click **Apply**, the hub/shroud region is defined.

When hub/shroud regions are defined on the inlet and/or outlet domain, the `Mesh Data` object has additional settings on its **Inlet** and/or **Outlet** tabs, respectively. Those Mesh Data settings control:

- Whether the defined hub and shroud regions are enabled. By default, they are enabled, in which case they are used to divide their respective inlet/outlet domains into segments.
- The mesh element count and distribution (in the streamwise direction) within each segment of the inlet/outlet domain.

For details, see Inlet/Outlet Tab (p. 155).

As just mentioned, when the defined hub and shroud regions are enabled, segments are formed within the inlet/outlet domain. Some segments directly correspond to named hub/shroud regions. Other segments are formed automatically in order to cover any remaining portions of the hub/shroud within the inlet/outlet domain. The resulting 3D mesh includes 2D regions that correspond with the segments.

Figure 10.14: Inlet Domain Segments (p. 158) shows an example of a `3D Mesh` object with 2D regions that correspond with the segments inside the inlet domain.

The source of a segment determines how the corresponding 2D region is named:

- For a segment that originates from a defined hub/shroud region, the name of the hub/shroud region (as specified in the **Hub Regions** or **Shroud Regions** tab) is used for the corresponding 2D region.
- For any other segment, the corresponding 2D region is automatically given a name like "HUB1", "HUB2", "SHROUD1", "SHROUD2", with numbers increasing for segments further from the passage domain.

Note:

- A region cannot have one end in the inlet domain and the other end in the outlet domain. Regions on the hub must not overlap with other regions on the hub. Regions on the shroud must not overlap with other regions on the shroud.

- A region defined on the hub affects the mesh distribution through the inlet/outlet domain to the shroud and vice versa. A topological line segment is created between each specified region boundary point and its corresponding point on the opposite side (hub or shroud). The corresponding point is created by projecting across the domain, unless such a projected point would be too close (according to the **Hub Region Proximity Tolerance**/**Shroud Region Proximity Tolerance** specification) to an existing region boundary point, in which case the latter serves as the corresponding point.
- If you have trimmed the inlet/outlet domain (see Trim Inlet Tab or Trim Outlet Tab - Changing the Inlet or Outlet Boundary (p. 127)), a region that crosses the trim location is adjusted so that the defining point that lies within the trimmed region is effectively applied at the trim location.

10.3.4.3. The Blade Set Object and Blade Objects

To edit or define the machine component's blades as a set, edit the `Blade Set` object from the object selector or click *Edit Blade Set*. To define or modify a single blade, edit the corresponding object stored under the `Blade Set` object. You can only define the blade set or individual blades after the hub and shroud have been defined, although all of these can be defined in one step. For details, see Steps to Create a Mesh (p. 95).

The count and naming of the individual geometry (and topology, and mesh data) objects for each blade in the blade set are managed automatically by Ansys TurboGrid. If a profile file containing multiple blades is specified for the `Blade Set` object (whether directly or indirectly, for example, by loading a BladeGen `.inf` file), then a blade object will be created for each set of data in the file and named according to the name specified in the file (unless a name is not specified, in which case a generic name of the form "Blade *n*" will be given, where *n* is an integer starting at 1).

The individual blade objects contain a profile curve file specification and a setting to indicate which single blade in the file applies; numbering starts at zero.

10.3.4.3.1. Blade Tab

10.3.4.3.1.1. Apply Settings To All Blades Check Box

When **Apply Settings To All Blades** is selected, `Blade Set` settings overwrite the corresponding blade object settings. If **Apply Settings To All Blades** is cleared, the settings that apply to all blades are disabled.

If you change a setting for a blade object, and the `Blade Set` object has a corresponding setting, then the **Apply Settings To All Blades** check box will be cleared to help prevent accidentally losing the blade-specific settings in case you reapply the `Blade Set` object.

10.3.4.3.1.2. Coordinate System and Blade File Definition

10.3.4.3.1.2.1. File Name

Set **File Name** to the name of the blade set file. To open a file selector, click *Browse*.

The blade file must contain data points that define the blade profile (or rib) curves and should have the file extension `.curve` or `.crv`.

The profile points must be listed line by line, in free-format ASCII style in a closed-loop surrounding the blade.

A minimum of two profiles are required in the blade file: one which lies close to the hub surface and one which lies close to the shroud surface. The profile is not required to lie exactly on the surface. If it lies between the hub and shroud surfaces, it must be within 8% of the span from the surface. If a profile lies outside of the passage, its distance from the surface has no maximum limit.

Profiles can be used to define tips. For details, see The Hub Tip and Shroud Tip Objects (p. 120). Profiles are allowed, and sometimes required, beyond a tip.

The profiles must be listed in the file in order from hub to shroud. Individual profile datasets are separated by a line beginning with the "#" character. Any text following the "#" character is used as the name associated with the subsequent profile.

For example:

```
# Hub Profile
0.0  0.0  1.0 le
1.0  0.0  1.0
1.0  1.0  1.0 te
0.0  1.0  1.0
0.0  0.0  1.0
# Intermediate Blade Profile
0.0  0.0  2.0 le
1.0  0.0  2.0
1.0  1.0  2.0 te
0.0  1.0  2.0
0.0  0.0  2.0
# Shroud Tip Profile
0.0  0.0  2.75 le
1.0  0.0 2.75
1.0  1.0 2.75 te
0.0  1.0 2.75
0.0  0.0 2.75
# Shroud Profile
0.0  0.0  3.0 le
1.0  0.0  3.0
1.0  1.0  3.0 te
0.0  1.0  3.0
0.0  0.0  3.0
```

There is no restriction on the number of profiles or the number of data points in a profile dataset. However, as noted above, a minimum of two profiles is required.

10.3.4.3.1.2.2. Coordinates, Angle Units, and Length Units

Set **Coordinates**, **Angle Units**, and **Length Units** according to the data in the blade file.

10.3.4.3.1.3. Geometric Representation

Ansys TurboGrid generates blade surfaces using a two-step process:

1. Curves are generated.

2. A surface is created by lofting across the set of curves (that is, sweeping from one curve to the next).

The **Geometric Representation** settings control how these steps are performed.

10.3.4.3.1.3.1. Method

The **Method** setting allows two preset methods, and one general method, for controlling the settings that govern the geometric representation of blade surfaces (that is, the **Lofting**, **Curve Type**, and **Surface Type** settings, described shortly).

The available **Method** options are:

- `BladeModeler`

  The `BladeModeler` option sets **Lofting** to `Spanwise`, **Curve Type** to `Bspline`, and **Surface Type** to `Ruled`.

- `Flank Milled`

  The `Flank Milled` option sets **Lofting** to `Streamwise`, **Curve Type** to `Piece-wise Linear`, **Surface Type** to `Ruled`.

- `Specify`

  The `Specify` option makes the **Lofting**, **Curve Type**, and **Surface Type** settings available for direct specification.

The following rules are followed by Ansys TurboGrid for selecting the geometric representation method:

- If you load a blade from a BladeModeler `.inf` file, the `BladeModeler` option will be selected automatically.
- If the `BladeModeler` option is selected and there are only 2 blade profile curves (for the applicable blade), the selected option will change to `Flank Milled` (which is equivalent, in this case).
- If the `Flank Milled` option is selected and there are more than 2 blade profile curves (for the applicable blade), the selected option will change to `BladeModeler` and a warning message will be issued.

10.3.4.3.1.3.2. Lofting

The direction in which lofting occurs is set by the **Lofting** setting. The available **Lofting** options are:

- `Streamwise`

  The curves that are swept in the process of streamwise lofting are formed by connecting corresponding points between adjacent blade profiles. For this method of lofting, you must ensure that:

  – each blade profile has the same number of points

  – the points are ordered in the same direction

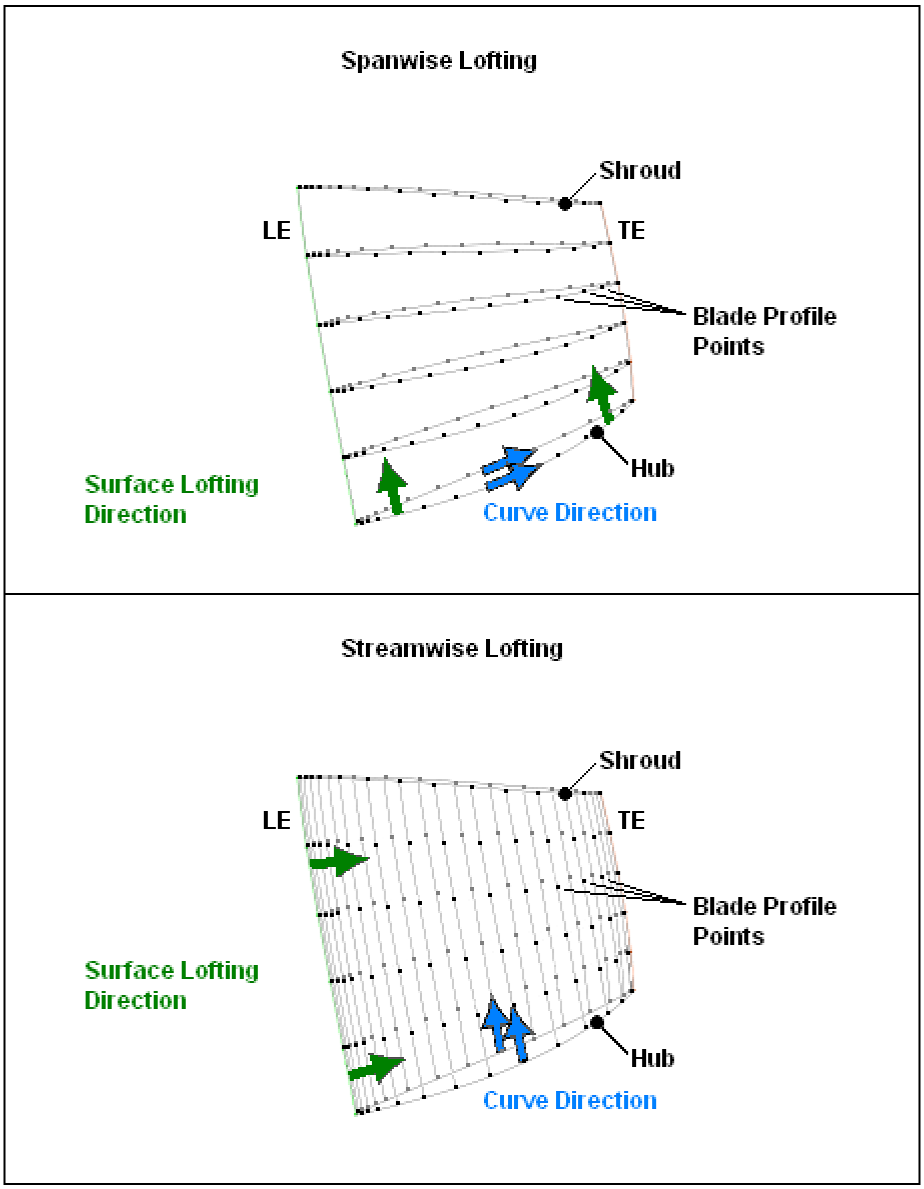
  – the first, *n*th, and last point of each blade profile are at similar locations around the blade

- `Spanwise`

  The curves that are swept in the process of spanwise lofting are the blade profile curves.

Figure 10.4: Spanwise Lofting versus Streamwise Lofting (p. 115) shows how spanwise lofting and streamwise lofting differ.

Figure 10.4: Spanwise Lofting versus Streamwise Lofting



The appropriate lofting setting depends on how the geometry was produced. For example, BladeModeler can produce blades via streamwise or spanwise lofting. The appropriate setting is stored in the `BladeGen.inf` file. As another example, a flank-milled blade should be generated and interpreted using two blade profiles (one at (or near) the hub and one at (or near) the shroud) with streamwise lofting.

Note:

The method of "sweeping" through the curves in the lofting process is controllable via the **Surface Type** setting, which can be set to `Bspline` (default) or `Ruled`. For details, see Surface Type (p. 116). For streamwise lofting, the `Bspline` method may cause unexpected results. In this case, the workaround is to use the `Ruled` method.

10.3.4.3.1.3.3. Curve Type

The set of curves that are used to loft the blade surface may be constructed in one of two ways, as determined by the **Curve Type** setting. The available **Curve Type** settings are:

- `Piece-wise linear`

  The `Piece-wise linear` setting causes the points listed for each profile in the blade file to be connected to one another with straight lines. You should use this setting when you have a large number of points (1000 or more) in a profile. This setting is always used when **Method** is set to `Flank Milled`.

- `Bspline`

  The `Bspline` setting results in the interpolation of a smooth curve for each profile using points listed in the blade file. This setting may be necessary if the profile curves are defined with a small number of points. This setting is discouraged when you have a large number of points (1000 or more) in a profile, due to undesirable effects on the shape of the spline, and due to limitations of the software. The `Bspline` setting is the default when **Method** is set to `BladeModeler` or `Specify`.

10.3.4.3.1.3.4. Surface Type

The manner in which the set of curves is lofted is controlled by the **Surface Type** setting. The available **Surface Type** options are:

- `Ruled`

  A ruled surface is created by sweeping along linear paths that connect one curve to a corresponding place on the next curve. The `Ruled` method for surfaces is similar to the `Piece-wise linear` method for curves, but is done in 2 dimensions instead.

- `Bspline`

  A B-spline surface is created by sweeping along curvilinear paths that connect one curve to a corresponding place on the next curve. The `Bspline` method may be necessary if the blade is defined with a small number of profile curves.

10.3.4.3.1.4. Leading and Trailing Edge Definitions

Ansys TurboGrid uses an algorithm to determine the location of the leading edge curve(s) and trailing edge curve(s) and in most cases gets the correct locations (double edges are possible in the case of cut-off or square edges). In some instances, the location of one or more of the points may need to be adjusted. See Leading Edge (p. 119) and Trailing Edge (p. 120) for information about adjusting the leading and trailing edges.

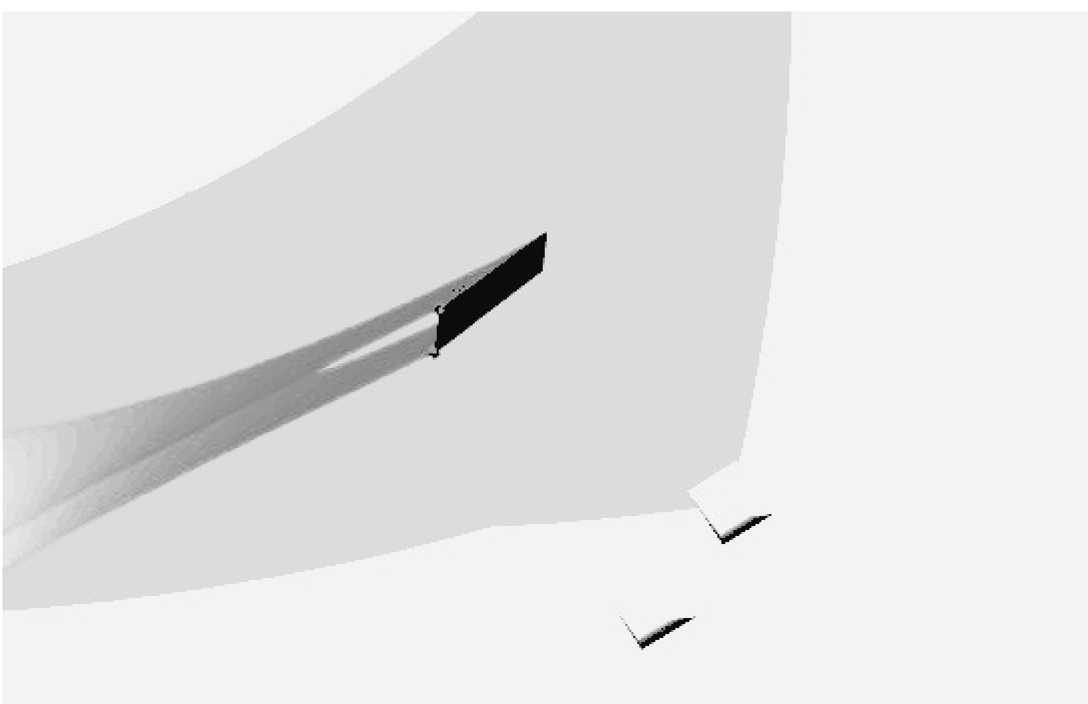
If you know which points on each profile should lay on the leading and trailing edge curves, you can optionally designate them in the blade file using a fourth field. This ensures the correct positioning of the leading and trailing edge curves in the beginning. The leading edge point on a profile should have **le1** (or **le2** for the second leading edge in the case of a double leading edge) in the fourth field and the trailing edge point on a profile should have **te1** (or **te2**) in the fourth field. These points are then guaranteed to be listed in the leading edge or trailing edge objects. For an example of the format of a blade file containing leading and trailing edge designations, try the following:

1. Load a blade file into Ansys TurboGrid.
2. Optionally change the type of leading or trailing edge between single and double (using the **LE Type** and **TE Type** settings, which are described below).
3. Save the blade file and examine it in a text editor, noting the **le1**, **te1** (and, for double edges **le2** and **te2**) entries in the fourth field.

10.3.4.3.1.4.1. Cut-off or square

This is an option that, when not using ATM3D, may be used to create a flat leading/trailing edge between two edge curves. See Figure 10.5: Trailing Edge with Pair of Edge Curves (p. 117) for an example of a "cut-off or square" trailing edge.

Figure 10.5: Trailing Edge with Pair of Edge Curves



The **Cut-off or square** option is intended for use on blade profiles that have discontinuities in slope at the edges. Use of this option on rounded or blunt leading/trailing edges may result in edge curves that are not placed in optimal locations (that is, further adjustments may be necessary).

Note:

Using the **Cut-off or square** option prevents the use of the ATM3D meshing approach.

10.3.4.3.1.5. Bias of Blade towards High Periodic

The **Bias of Blade towards High Periodic** > **Factor** setting, available in the `Blade Set` object, controls the location of the periodic surface in the passage, with respect to the first blade of the blade set. Set **Factor** to a number somewhere between `0` and `1`. A higher value brings the high periodic surface closer to the first blade (causing the low periodic surface to move further from the blade). A lower value brings the low periodic surface closer to the blade. The main purpose of this feature is to avoid having a periodic surface too close to (especially touching), the blade surface. Such a situation can occur for some blade geometries.

10.3.4.3.1.6. Curve or Surface Visibility

The visibility of curves and surfaces for a blade object works in the same way as for the hub. For details, see Curve or Surface Visibility (p. 109).

10.3.4.3.1.7. Reread Button

If the file that defines the blade profile has been modified since it was last loaded, clicking the **Reread** button will cause the blade profile data to be reloaded. The hub and shroud objects also have this feature.

10.3.4.3.1.8. Save & Load Button

If a leading or trailing edge point has been moved, the **Save & Load** button appears in place of the **Reread** button. Selecting this button causes the modified blade profile data to be saved to a profile file (you will be prompted for the name), then reloaded (from the same file) in order for the new geometry to take effect. The leading edge and trailing edge objects also have this feature.

10.3.4.3.2. Transform Tab

The **Transform** tab contains specifications for rotational and translational transforms that adjust the geometry. The settings on this tab adjust the blade geometry only. The hub and shroud can also have their coordinates transformed using the corresponding tab of those objects.

10.3.4.3.2.1. General Rotation

Specify an axis via two points and a rotation angle about the axis.

10.3.4.3.2.2. Translation

Specify the direction and distance to translate the geometry. The direction may be any one of: `Machine Axis`, `X Axis`, `Y Axis`, `Z Axis`, or `Vector` (any specified direction).

10.3.4.3.2.3. Axial Rotation

Specify the angle to revolve the geometry about the machine axis.

10.3.4.3.3. Leading Edge

The leading edge curve is the most upstream part of the blade. To modify the blade's leading edge, select the leading edge object from the object selector. The leading edge can only be defined after the blade has been defined.

10.3.4.3.3.1. Leading Edge Tab

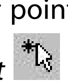
10.3.4.3.3.1.1. Curve

The leading edge curve appears green in the viewer window and each point on the curve (one for each blade profile, or rib, curve) is marked with an octahedron symbol. The points are listed from hub to shroud. Edit the individual leading edge point locations to create the correct leading edge curve for the blade.

The quality of the mesh is dependent on the accuracy of the leading edge curve. In most cases, the initial placement of leading edge points is adequate. However there are some cases when some points may need to be moved.

Any change to the leading edge changes the blade surfaces, which changes the periodic surfaces as well as the hub and shroud surfaces. In order to avoid the delays associated with recreating the entire geometry after any modification to the leading edge, another step is required. The blade must be saved after the modifications to the leading edge are complete. For details, see Save Blade As Command (p. 33). The saved blade file contains data in the fourth field to designate the leading edge points. To complete the leading edge modification, the blade object must be updated with this new blade file. For details, see The Blade Set Object and Blade Objects (p. 112).

There are several steps required to edit the location of a point on the leading edge curve.

1. Select the point to edit by clicking on it in the list of points. Alternatively, you can pick the point to edit from the viewer after clicking *Select*  (on the viewer toolbar).
2. Click in any coordinate widget and then click the new location for the point in the viewer. The coordinates of the picked point are displayed in the object editor. Alternatively, type the coordinates of the new location for the point or use the embedded sliders. The units of the coordinates are the base units or solution units. For details, see Units (p. 107).
3. Click **Apply** to save the new location of the current point.

Picking Mode can also be used to select and translate leading edge points in the viewer. For details, see Selecting and Dragging Objects while in Viewing Mode (p. 79). After moving the leading edge point, it will snap to the nearest point on the current profile curve.

If the leading edge points were designated in the blade file, it may not be necessary to adjust them since, in that case, the leading edge curve may already be acceptable. For details, see Coordinate System and Blade File Definition (p. 112).

10.3.4.3.3.1.2. Save & Load Button

If a leading or trailing edge point has been moved, the **Save & Load** button appears. Clicking this button causes the modified blade profile data to be saved to a profile file (you will be prompted for the name), then reloaded (from the same file) in order for the new geometry to take effect. The blade and trailing edge objects also have this feature.

10.3.4.3.4. Trailing Edge

The trailing edge curve is the most downstream part of the blade. The trailing edge curve appears red in the viewer window. Because of the similarity of the trailing edge with the leading edge, see Leading Edge (p. 119) for details.

10.3.4.4. The Hub Tip and Shroud Tip Objects

The shroud tip is the portion of the blade surface that exists as a result of the blade not extending all the way to the shroud. By default, the blade extends all the way to the shroud (that is, no shroud tip exists). To create a shroud tip, edit the `Shroud Tip` object after the blade has been defined.

The hub tip is similar to the shroud tip, except that it is at the hub end of the blade. The settings for the hub tip are stored in the `Hub Tip` object.

Note:

It is possible to specify a hub tip and a shroud tip for the same blade.

Note:

Using a tip clearance on one blade but not another is possible by changing the CCL for individual blades, but could produce an invalid mesh near the tip of any blade that does not have tip clearance.

10.3.4.4.1. Hub Tip and Shroud Tip Tabs

The following sections describe the object editor settings for the `Hub Tip` and `Shroud Tip` objects:

10.3.4.4.1.1. Override Upstream Geometry Options

Selecting the **Override Upstream Geometry Options** check box enables you to change the hub/shroud tip settings for the case of TurboGrid running in Workbench with an upstream geometry provided by BladeEditor. If you do not select the **Override Upstream Geometry**

Options check box, the hub/shroud tip settings are displayed but are not editable, being controlled by the upstream BladeEditor geometry.

> **Note:**
>
> The shroud tip geometry, but not the hub tip geometry, can be defined in BladeEditor.

10.3.4.4.1.2. Clearance Type

The **Tip Option** setting has the following options:

- `None`

  This option causes the blade to extend from hub to shroud.

- `Constant Span`

  This option creates a tip at the span location that you specify. A span of 0.0 represents the hub and a span of 1.0 represents the shroud. The span value for the hub tip must be between 0 and 0.5; the span value for the shroud tip must be between 0.5 and 1.0.

- `Normal Distance`

  This option creates a tip so that the gap distance is whatever you specify.

- `Variable Normal Distance`

  This option creates a tip so that the gap distance varies from leading edge to trailing edge using values that you specify. The gap distance varies linearly with m-coordinate as calculated on the hub (for a hub tip) or shroud (for a shroud tip).

- `Profile Number`

  This option creates a tip at the blade profile that you specify. The available profiles are defined in the blade file. For details, see The Blade Set Object and Blade Objects (p. 112). The profile closest to the hub is profile 1.

  > **Note:**
  >
  > The blade surface and resulting mesh can be adversely affected when a blade shroud tip is defined by the second-last profile, and the last profile has a shape that is inconsistent with where the blade would exist if the blade were extended in the spanwise direction without the last profile. To avoid this problem, you can either ensure that the last profile is consistent with the second last profile, or manually remove the last profile from the curve file. A similar statement applies for the hub end of a blade having a hub tip.

10.3.4.5. The Low Periodic and High Periodic Objects

The low periodic surface extends from hub to shroud and inlet to outlet along the side of the mesh that has the lowest Theta values. (The direction of Theta is determined by the machine rotation axis and the right-hand rule.) The high periodic surface is on the opposite side of the passage from the low periodic surface (that is, on the high-Theta side).

The periodic surfaces can be modified only after the blade has been defined.

10.3.4.5.1. Data Tab

Set **Method** to one of the following:

- `Automatic`

  This option creates the periodic surface automatically. The shape of the periodic surface is based on the geometry of the adjacent blades. The position of the automatically-generated periodic surface is governed by the **Bias of Blade towards High Periodic** setting found in the `Blade Set` object, as well as the **# of Bladesets** setting of the `Machine Data` object.

- `From File`

  This option allows you to load a periodic surface that was saved to a file. The **Length Units** setting controls the units that will be used to interpret the coordinate data in the file. The **Rotation Angle** setting transforms the loaded periodic surface by rotating by the specified angle about the machine rotation axis. Note that the **# of Bladesets** setting of the `Machine Data` object will not change automatically due to changes to the **Rotation Angle** setting.

10.3.4.5.2. Rendering Properties

To modify the color or rendering of the low or high periodic surface, right-click the `Low Periodic` or `High Periodic` object (respectively), then select **Render (Properties)** > **Edit Options**.

10.3.4.6. The Inlet and Outlet Objects

The `Inlet` object defines a surface that governs the meridional shape of the inlet end of the passage mesh. This end of the passage mesh can represent a stage interface. Note that if you subsequently add an inlet domain (via the `Mesh Data` object), this end of the passage mesh serves as an interface between the inlet domain and the passage mesh.

Note:

In this documentation, the upstream and downstream ends of the passage may be referred to as interfaces, even though they might represent the ends of the mesh, because they have the potential to become interfaces if you subsequently add inlet or outlet domains (via the `Mesh Data` object).

The `Outlet` object is similar to the `Inlet` object, except that it applies to the opposite end of the passage, and has an extra option for how it can be defined.

You can modify the `Inlet` and `Outlet` objects only after the blade has been defined.

The following topics are discussed:

10.3.4.6.1. Inlet Tab or Outlet Tab

The **Inlet** tab enables you to move and shape the interface between the inlet domain and the passage domain. The **Outlet** tab acts similarly for the interface between the outlet domain and the passage domain.

The following **Interface Specification Method** options are available, as applicable:

- **Parametric**

  This is the default option. You specify hub and shroud locations parametrically. You have the option of making the interface conic or allowing it to bend according to the geometries of the blade, hub, and shroud. For details, see Interface Specification Method: Parametric (p. 123).

- **Points**

  You specify a set of points that controls the shape and position of the interface. For details, see Interface Specification Method: Points (p. 124).

- **Adjacent blade**

  You load a curve file that describes the shape of the next blade upstream or downstream of the passage, depending on whether you are defining an inlet or outlet object, respectively. TurboGrid then positions the interface automatically, taking into account the geometry of the blade being meshed and the geometry of the adjacent blade. For details, see Interface Specification Method: Adjacent blade (p. 126).

- **Fully extend**

  This option causes all inlet points to move to the very beginning of the passage (as far away from the blade as possible), and all outlet points to move to the very end of the passage. This option can be used to prevent the blade from crossing the passage inlet or passage outlet.

- **Meridional splitter** (available only on the **Outlet** tab)

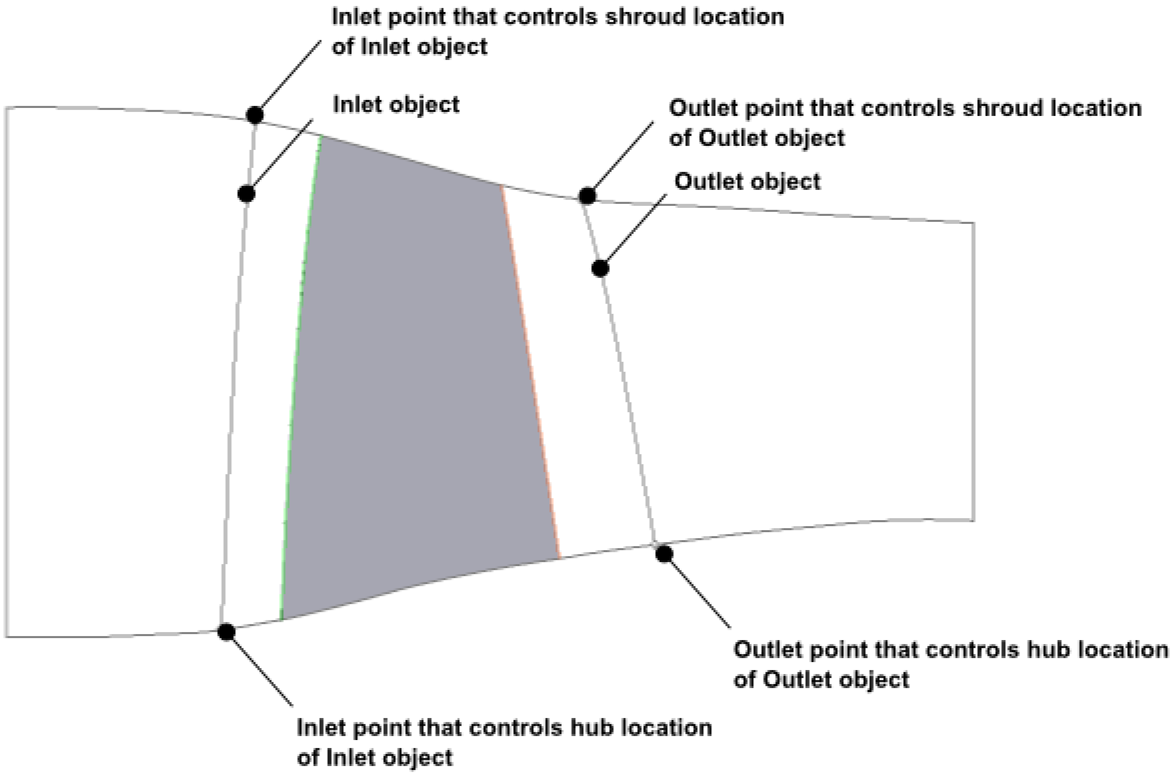
  You load a curve file that describes the shape of a meridional splitter that is immediately downstream of the passage. TurboGrid then positions the interface automatically, taking into account the geometry of the blade being meshed and the geometry of the meridional splitter. For details, see Interface Specification Method: Meridional splitter (p. 126).

10.3.4.6.1.1. Interface Specification Method: Parametric

You can specify the hub and shroud locations of the interface in parametric space in one of two ways:

- By providing a number between 0 (all the way towards the blade) and 1 (all the way away from the blade) for each of the following settings: **Parametric Location Parameters** > **Hub**, **Parametric Location Parameters** > **Shroud**.

- By using the mouse to drag the inlet or outlet points in the viewer.



By default, the interface follows a straight path in parametric space between the specified locations on the hub and shroud. Viewed in A-R space, the interface appears to bend in accordance with the geometries of the blade, hub, and shroud. By selecting the **Use simple conic surface** option, you can override this default behavior and instead have the interface follow a straight line in A-R space between the specified locations on the hub and shroud. In this case, the interface forms a conic section in Cartesian space.

10.3.4.6.1.2. Interface Specification Method: Points

10.3.4.6.1.2.1. Automatically generate required intermediate points

The **Automatically generate required intermediate points** option is available only before the hub, shroud, and blade are loaded. It is on by default. This option causes a set of intermediate points to be generated between the low hub point and low shroud point upon loading the geometry. TurboGrid aims to add an appropriate number of points, placed at locations that follow the leading or trailing edge of the blade.

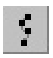
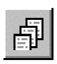
10.3.4.6.1.2.2. Curve

The inlet/outlet curve consists of a set of points that guide the meridional shape of the inlet end of the passage mesh as a function of the local spanwise direction. By default, a set of points is automatically generated (see Automatically generate required intermediate points (p. 124)).

The inlet/outlet curve can be defined by creating, modifying, or deleting points in the list of points. There must be a point at both the hub and shroud while additional points are optional.

The points are connected linearly and must be listed in order from hub to shroud, or they will be ignored.

A point can be selected by clicking on it in the list of points or by picking it in the viewer window. For details on picking points in the viewer, see Selecting and Dragging Objects while in Viewing Mode (p. 79). The available icons become active when a point is selected. Alternatively, you can right-click an existing point in the list of points to see the same options. Each icon and its function is described in the following table:

Icon	Description
	*New* adds a point in the list of points below the selected point. The initial location of the new point on the Inlet is between the locations of the selected point and the following point in the list of points. You cannot create a new point after the low shroud point.
	*Generate intermediate points* removes all points except for the low hub point and the low shroud point, then automatically generates a set of zero or more points between them.
	*Delete* removes the selected point from the list of points. You cannot delete the low hub point or the low shroud point.
	*Read from File* opens a load file window. Select the file with the inlet points and click **Open**.

You can modify the location of the selected point by:

1. Typing in the coordinates (axial, radial, or both, depending on the selected **Method** option), or

2. Selecting and dragging the point in the viewer using the left mouse button. For details, see How to Select and Move a Point (p. 126).

Set **Method** to one of the available options:

- `From A and R`
- `Set A`
- `Set R`

Regardless of the method used to modify the location of the selected point, the location is restricted. The low hub point snaps to the line of intersection between the hub and low periodic surfaces. The low shroud point snaps to the line of intersection between the shroud and low periodic surfaces. Use the OUTLINE object as a guide when dragging the points. Any other points between the hub and shroud snap to the low periodic surface and are not constrained in the spanwise or streamwise directions. Inlet points cannot be moved further downstream on the low periodic surface than a position level with the leading edge of the blade. After you have clicked **Apply**, or dragged a point, the **Selected Point** setting will show the actual position to which the point has moved.

Note:

The points are defined using the A-R coordinate system as opposed to the X-Y coordinate system.

The visibility of the inlet or outlet points is controlled by the **Point Visibility** check box.

10.3.4.6.1.2.3. How to Select and Move a Point

You can select and move points in the following ways:

- Select a point by holding **Ctrl**+**Shift**, and clicking the mouse while pointing at a point.
- Select and move a point by holding **Ctrl**+**Shift** and dragging using the left mouse button.
- To select and drag points without holding down **Ctrl**+**Shift**, you can click the *Select*  icon, then select and drag points with the left mouse button.

10.3.4.6.1.3. Interface Specification Method: Adjacent blade

Specify a curve file that defines the shape of the blade in the adjacent stage.

Note:

The hub and shroud curves are required to extend past this blade.

10.3.4.6.1.4. Interface Specification Method: Fully extend

The **Fully extend** option causes all inlet points to move to the very beginning of the passage (as far away from the blade as possible), and all outlet points to move to the very end of the passage. This option can be used to prevent the blade from crossing the passage inlet or passage outlet.

10.3.4.6.1.5. Interface Specification Method: Meridional splitter

Specify a curve file that defines the shape of a meridional splitter. Also specify **Length Units** and **Curve Type** to help interpret the curve file.

Note:

A hub tip is not allowed when using the **Meridional splitter** interface specification method.

10.3.4.6.1.5.1. Interface parametric locations

You can move the passage interface upstream or downstream by decreasing or increasing the **Hub** and **Shroud** values, which control where the interface meets the hub and shroud, respectively.

10.3.4.6.1.6. Point Visibility

The **Point Visibility** option controls the visibility of the control points that govern the shape of the inlet or outlet end of the passage mesh.

10.3.4.6.1.7. Control Angle

By default, TurboGrid automatically adjusts the geometry upstream and downstream of the blade. You can override the angle of the periodic surface relative to the rotation axis. This can be useful for improving the quality of the mesh and/or guiding the mesh in the general flow direction.

10.3.4.6.2. Trim Inlet Tab or Trim Outlet Tab - Changing the Inlet or Outlet Boundary

The **Trim Inlet** tab enables you to trim the hub and shroud curves to effectively move the upstream end of the inlet domain. The **Trim Outlet** tab acts similarly for the downstream end of the outlet domain.

Note:

The upstream end of the inlet domain and downstream end of the outlet domain remain surfaces of revolution, formed by revolving the line segments that connect corresponding ends of the hub and shroud curves, whether or not those ends were moved due to trimming.

The following methods of choosing the hub and shroud curve trim locations are available:

- **Set A**

  Specify two axial coordinates: one to trim the hub curve, the other to trim the shroud curve. This method offers precise control of trim locations for cases where the hub and shroud curves are aligned primarily in the axial direction near their respective trim locations. Conversely, this method is not suitable (and may fail) for cases where the hub or shroud curve is aligned closely with the radial direction near its trim location.

  Ideally, each specified axial coordinate exists in exactly one place on the corresponding (hub or shroud) curve.

  If a specified axial coordinate exists in multiple locations, TurboGrid then also considers the radial coordinate (which is displayed beside the specified axial coordinate) and chooses the nearest location. Note that the radial coordinate is changeable by selecting a point via the mouse (for example, by clicking near the intended trim location); it can also be changed using the **Command Editor** dialog box.

If a specified axial coordinate does not exist on the corresponding (hub or shroud) curve, a warning message is given and TurboGrid resets the specified axial coordinate (and associated radial coordinate) to the location of the curve end.

- **Set R**

  The description for this method is the same as for `Set A`, except with the words "axial" and "radial" swapped.

- **AR Intersection**

  Specify two points in the axial-radial plane. The line that passes through these points is used to trim the hub and shroud curves where it intersects those curves.

  If there are multiple intersection points on a given curve, TurboGrid chooses one based on proximity to the line segment that connects the specified points.

  If there are no intersections on a given curve, a warning message is given and TurboGrid resets one or both specified points to the location of a curve end.

10.3.4.6.3. Using Stage Interfaces

In order to produce a proper stage interface, you should follow these guidelines:

- Use the same hub curve for both stages, and the same **Curve Type** setting: `Bspline` or `Piece-Wise Linear`. The hub curve must extend through both stages to do this.

  The same applies to the shroud curve.

  As a less preferable alternative, you can use hub curves for each stage. The curves should meet at a point, and not overlap. When using this method, there is a risk of a discontinuity in slope where the curves meet.

- Use the same interface points for stage interfaces. Do this by saving the interface (inlet or outlet geometry object, as applicable), then loading the interface for the adjacent stage.

10.3.4.7. The Outline Object

The outline is a group of curves on the outer extents of the geometry. The outline includes the full extent of the hub and shroud read from the files, independent of the inlet and outlet locations. To modify the color or rendering of the geometry outline, right-click the `OUTLINE` object, then select **Render (Properties)** > **Edit Options**. You can only modify the outline after the blade has been defined.

10.4. Topology

The topology is a structure of blocks that acts as a framework for positioning mesh elements. The topology affects how the mesh is made, and can influence the mesh quality.

TurboGrid uses ATM and ATM3D optimized topology.

The object selector contains two types of topology-related objects:

- Topology Set

  The `Topology Set` object handles general topology settings, and often handles all the topology settings.

- Topology

  Each topology object handles details of the topology surrounding a particular blade. These details can be inherited from the `Topology Set` object, or (if applicable) can serve to fine tune the topology around a particular blade. Topology objects are given different names, to help distinguish the blade to which they apply; for example, the topology object for the main blade might be named "Main Blade".

This section discusses:

10.4.1. About Topology

Topology blocks represent sections of the mesh that contain a regular pattern of hexahedral (hex) elements. They are laid out adjacent to each other without overlap or gaps, with shared edges and corners between adjacent blocks, such that the entire domain is filled. By using topology blocks to control the placement of hex elements, a valid hex mesh can be generated to fill a domain of arbitrary shape. The topology is invariant from hub to shroud and is viewed on 2D layers that are located at various spanwise stations (see Layers (p. 159)).

The topology should be investigated at various layers (especially the hub and shroud layers) to check its quality since the mesh quality is directly dependent.

Note:

To visualize the topology on a layer, turn on the visibility using the visibility check box for that layer in the object selector and ensure that at least one topology visibility setting for the layer is turned on. The **Topology Visibility** setting controls the visibility of the yellow line segments that outline the topology block edges. For details, see Topology Visibility (p. 164). The **Master Topology Visibility** setting controls the visibility of the violet line segments that outline the master topology edges. For details, see Master Topology Visibility (p. 164).

A key feature of Ansys TurboGrid is the visibility of the surface mesh on the topology. As you adjust the topology, Ansys TurboGrid adjusts the surface mesh in real time so that the true effect of topology changes is visible. To help identify problem areas in the surface mesh before you generate the full 3D mesh, you can visualize mesh statistics on the layers. For details, see Mesh Statistics (p. 167)

Topology blocks contain a number of mesh elements along each side. The mesh elements vary in size across topology blocks in a way that produces a smooth transition within and between blocks. This is accomplished by shifting the nodes ("node biasing"[1]) toward, or away from, certain block edges. The topology blocks are positioned by default so that the mesh element sizes vary as smoothly as possible, given the constraints.

10.4.2. Basic Usage

The basic steps to using ATM Optimized topology are as follows:

1. Load the blade geometry and set the geometry parameters (for example, inlet/output points) as necessary.

2. Open `Topology Set`, set **Method** to an appropriate value, and click **Apply**.

3. Right-click **Topology Set** and clear **Suspend Object Updates**.

Note:

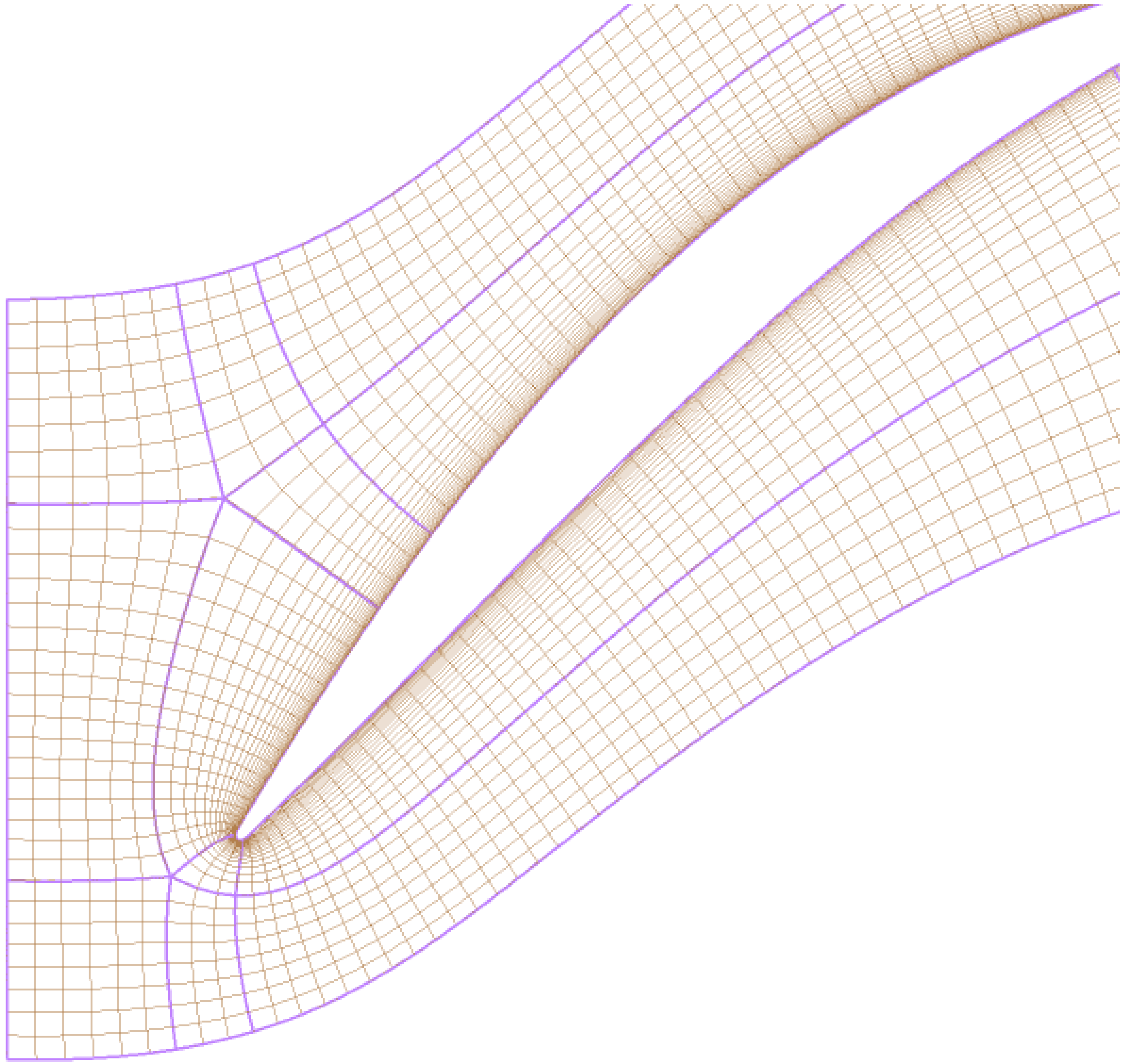
The **Undo** command cannot properly undo a suspension change (for example, to the `Topology Set` object); attempting to undo a suspension change can lead to an inconsistent state.

Ansys TurboGrid computes the master topology, topology, and refined mesh (the final mesh for the layer) for the hub and shroud layers based on the current Mesh Data specification.

After the topology has been generated, you can view the 2D mesh on a layer. If you open a layer object (for example, for the hub) in the object editor, you can control the visibility of the master topology, the topology, and the refined mesh; you can also see the mesh statistics for the layer. By default, the master topology and refined mesh are visible, as shown in the following figure.

[1] Node biasing is an important aspect of mesh generation because it helps to reduce the number of mesh elements.

Figure 10.6: Master Topology and Refined Mesh on the Hub Layer for an Axial Compressor Blade



The master topology is shown with heavy lines. The topology (not shown) serves as a background mesh that guides creation of the refined mesh. The refined mesh is shown with fine lines.

10.4.3. The Topology Set Object

This section includes:

10.4.3.1. Definition Tab for the Topology Set Object

The following topics are discussed:

10.4.3.1.1. ATM Topology

The **Method** setting enables you to generate an ATM topology in three basic ways.

- `Automatic`

  Automatically selects the appropriate topology based on the blade style (cut-off or rounded edges) and blade angles. The topology family chosen by TurboGrid appears beside **ATM Topology** > **Used**.

- *`[Topology Family]`*

  Allows you to manually select one of several topology packages, called families. If you have chosen a topology family that does not fit your geometry, you will receive an error message, and the topology will not be generated. For details, see Manually Selecting a Topology Family (p. 132).

- `Manual (Advanced)`

  For details, see Advanced Topology Control (p. 133).

10.4.3.1.1.1. Manually Selecting a Topology Family

By manually selecting a topology family, you are selecting one of the families that TurboGrid uses to generate a topology.

1. Load the blade geometry and set the geometry parameters (for example, inlet/output points) as necessary.
2. Open the **Topology Set** object.
3. Click **Apply**.
4. Click the **Topology Viewer** tab.
5. In **ATM Topology** > **Method**, scroll through the options in the drop-down menu to see the different topology families. A description of the family appears in the Topology Viewer (not the **3D Viewer**) when the family is selected.
6. Select the topology family you want to use.
7. Click **Apply**.
8. Right-click **Topology Set** and clear **Suspend Object Updates**.

> **Note:**
>
> The **Undo** command cannot properly undo a suspension change (for example, to the `Topology Set` object); attempting to undo a suspension change can lead to an inconsistent state.

If you have chosen a topology that fits your geometry, the topology will be generated.

You can view the details of the topology, regardless of whether it has been generated, by clicking the **Details** tab in the **Topology Set** editor. For more details, see Details Tab for the Topology Set Object (p. 136).

If you choose a topology that does not fit your geometry, the list of topology families in **ATM Topology** > **Method** shrinks to include only the topology that will fit your geometry. You may select one of the topology families displayed in the **Method** menu and click **Apply** to generate a topology that fits your geometry.

10.4.3.1.1.2. Advanced Topology Control

To use advanced topology control:

1. Click **Edit** > **Options**.
2. Select **Enable Advanced Features** and click **OK**.
3. Open the **Topology Set** object and set **ATM Topology** > **Method** to `Manual (Advanced)`.

> **Note:**
>
> Advanced features are tested but require an advanced level of understanding to be used.

A list of the available topology templates is displayed in the object editor. Any templates found in the local working directory (instead of the TurboGrid template directory) will be listed in bold text.

Table 10.1: Topology Template Naming Conventions (p. 133) describes the naming conventions used for topologies.

Table 10.1: Topology Template Naming Conventions

Naming Convention	Description
Prefixed with "S1"	Applies to geometries consisting of a main blade and a single splitter blade
Prefixed with "LP"	Low passage topology pieces
Prefixed with "MP"	Main passage topology pieces
Prefixed with "HP"	High passage topology pieces
Contains "LE"	Applies to the leading edge of a single bladed geometry
Contains "TE"	Applies to the trailing edge of a single bladed geometry

Naming Convention	Description
Contains "Passage"	Applies to the area between blades

Note:

- These terms can appear together. For example, the template LEcircleLowPassage covers both the leading edge and passage areas.
- The topology templates are "passage centric", that is, they are designed to fit between the blade and a rotated copy of the blade. Before the topology is used, it is transformed to "blade centric" where it is placed around the blade.

You must select a combination of templates that covers the entire passage without overlap.

A preview of each topology indicating vertex and edge numbers is provided to help with the template selection process. Previews appear in the Topology Viewer when you double-click a topology or when you select a topology and then click *Display Topology*. For more details, see Details Tab for the Topology Set Object (p. 136).

In the Topology Viewer, drag the mouse to pan across the image. To zoom in and out, roll the mouse wheel or drag the mouse vertically using the middle mouse button.

10.4.3.1.2. Tip Topology Option

By default, TurboGrid produces a mesh that has a non-conformal mesh interface in the blade tip region. When not using ATM3D, you can select **Tip Topology Option** > **Conformal Tip Enabled** to produce a conformal mesh in the tip region that is filled with unstructured wedge elements and that involves no mesh interface.

Figure 10.7: Conformal Tip Not Enabled

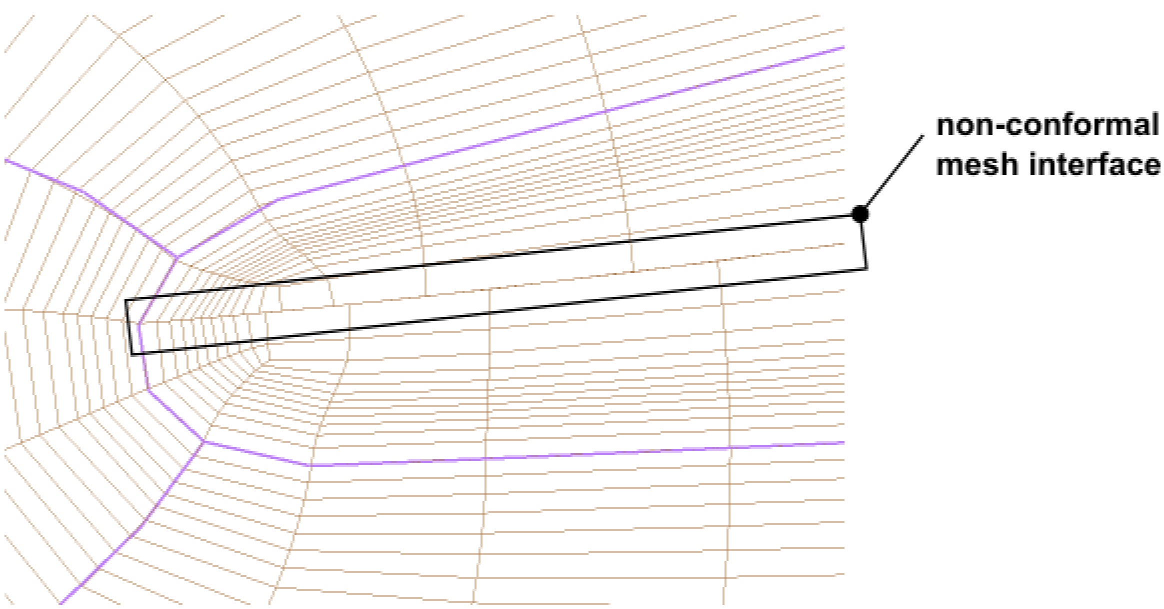
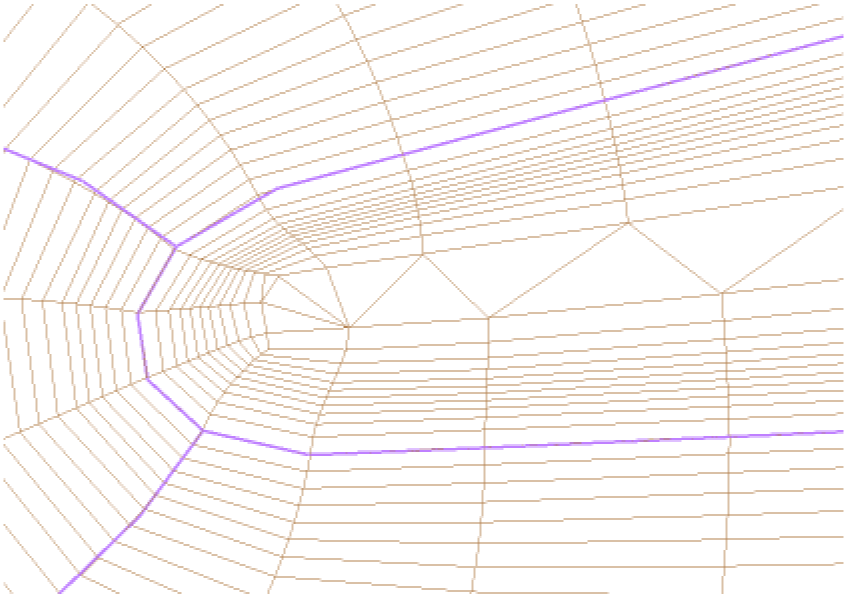


Figure 10.8: Conformal Tip Enabled



Note:

CCL parameter `Conformal Tip Minimum Quad Angle` (which can be found by editing the `Topology Set` object using the **Command Editor** dialog box) controls where the unstructured wedge region starts and ends along the blade. Starting from the leading edge and moving towards the trailing edge (and vice versa), quadrilateral elements (hexahedral elements in the 3D mesh) are used to fill in the space until the minimum inside angle of a quadrilateral element would drop below the specified value, at which point triangular elements (wedge elements in the 3D mesh) are used to continue filling in the space. Note that the interior angles of triangular elements may be less than the specified value of `Conformal Tip Minimum Quad Angle`.

Note:

The **Conformal Tip Enabled** option is not available when using the ATM3D meshing approach.

10.4.3.1.3. Split Mesh Regions

The **Split Mesh Regions At Leading Edge** option generates a line of rotation from each cut-off point of a "cut-off or square" (see Cut-off or square (p. 117)) blade leading edge. The **Split Mesh Regions At Trailing Edge** option generates a line of rotation from each cut-off point of a "cut-off or square" blade trailing edge.

A "cut-off or square" edge having the same axial and radial coordinates at both cut-off points will have lines of rotation that coincide. In this case, the applicable option will divide the passage mesh, hub, and shroud at the cut-off edge by a surface of revolution about the machine axis. Additional 2D and 3D regions will be created as a result of this division of the passage. This allows, for example, the mesh to be used in a simulation where the portion of the passage mesh surrounding the blade rotates, while the rest of the passage mesh is stationary.

These options may still be used if the cut-off points do not have the same axial and radial coordinates, although the interface between the divisions will not be a surface of revolution.

10.4.3.1.4. Use ATM3D Mesh Generation (Advanced)

The **Use ATM3D Mesh Generation (Advanced)** option enables the ATM3D meshing approach.

Note:

You can set this approach to be the default by setting a preference. For details see TurboGrid Options (p. 44).

For details on the ATM3D approach, see ATM3D (Advanced) (p. 144).

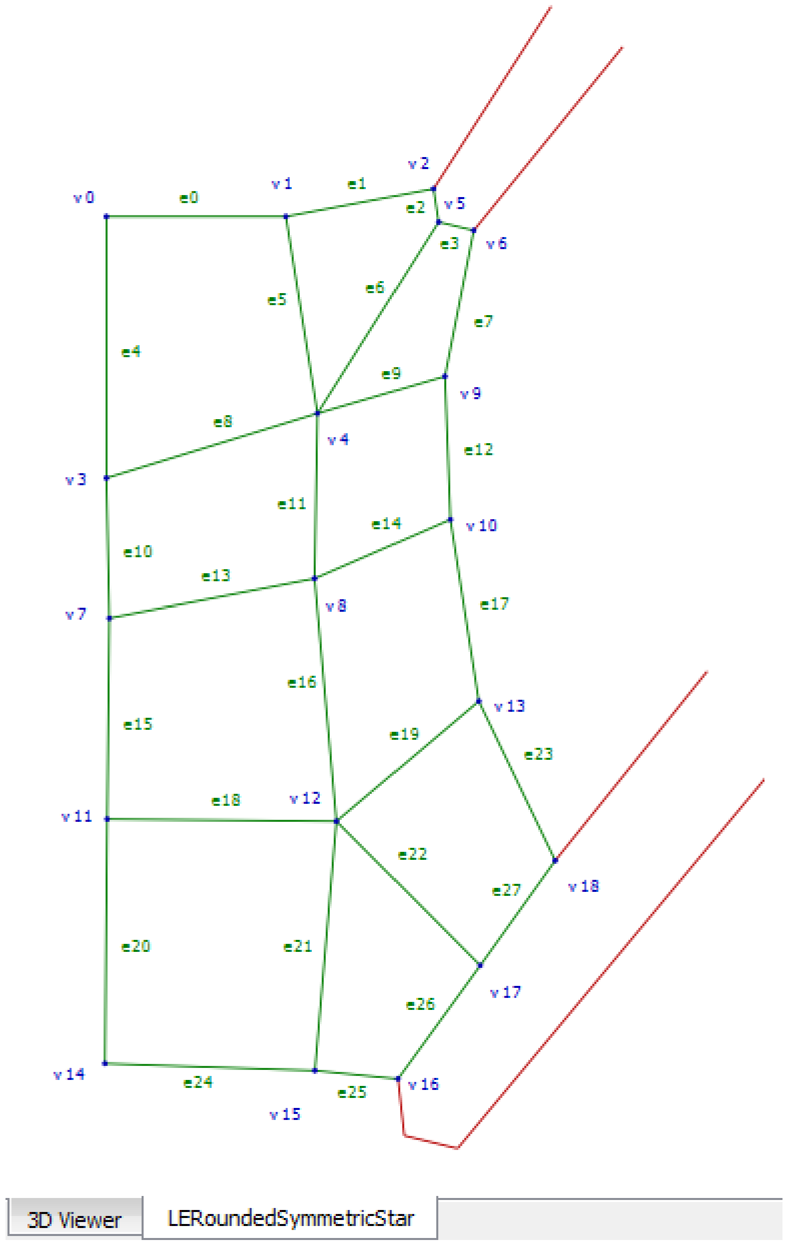
10.4.3.2. Details Tab for the Topology Set Object

The **Details** tab lists the topology templates that are being used in the current case.

To display one of the listed topology templates in the Topology Viewer, you can do any of the following:

- Right-click a listed topology template and select the **Display Topology** context menu command.
- Select a listed topology template and then click *Display Topology* .
- Double-click a listed topology template.

In the Topology Viewer, blue labels are used to indicate topology nodes and green labels are used to indicate topology edges. Blades are outlined in red. As an example, the figure below shows the LERoundedSymmetricStar topology template:



10.4.4. Using Splitter Blades with ATM

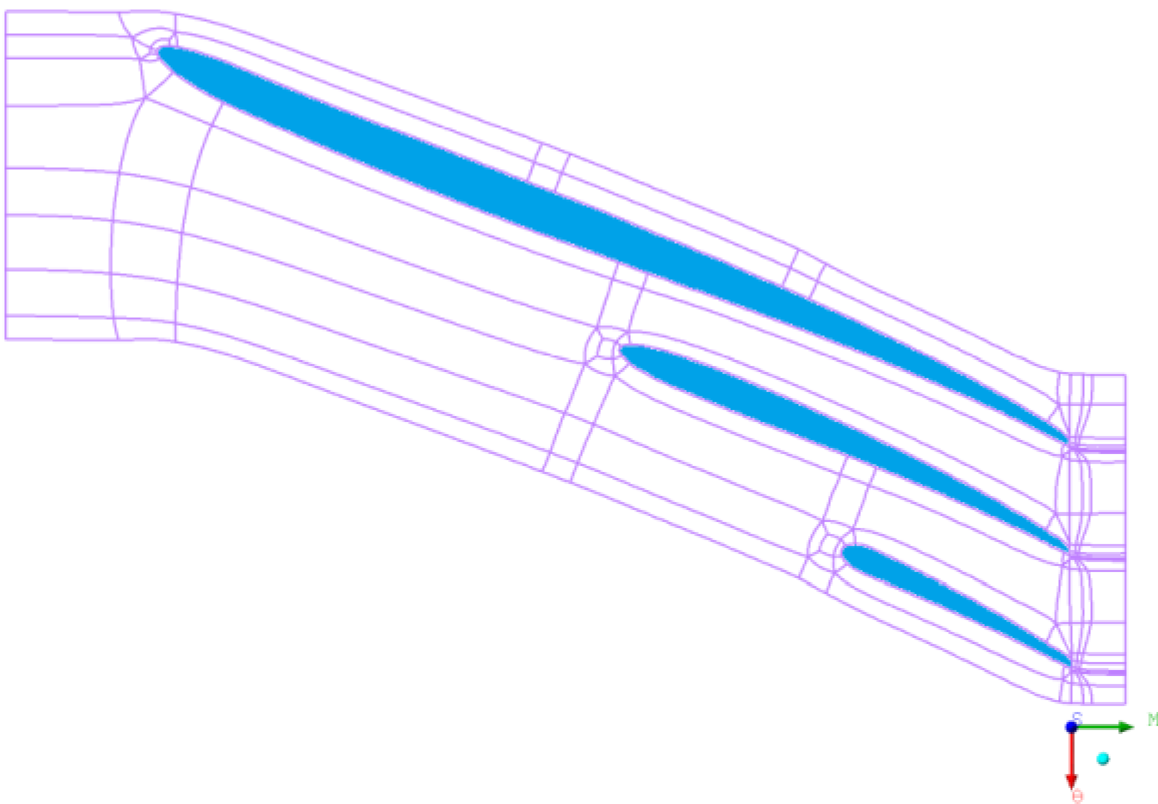
The ATM method supports one or two splitter blades provided that the following criteria are met:

- The leading edges of the main and splitter blades are both rounded.
- The trailing edges of the main and splitter blades are both rounded or both cut-off.
- The leading edge of the main blade is positioned, in the meridional direction, significantly ahead of the leading edge of the splitter blade.
- The trailing edges of the main and splitter blades, if cut-off, are at the same meridional position.

- The main blade is positioned, in the theta direction, lower than the splitter blade relative to the selected theta direction. You can specify the theta direction in the **Machine Data** object. For details, see Rotation (p. 107).

- In the case of two splitter blades, the three blades must be ordered by meridional length, with the longest blade at the lowest theta position and the shortest blade at the highest theta position.

Figure 10.9: Example Template for Dual Splitter Blades



10.4.5. Using Tandem Vanes with ATM

The ATM method supports the following tandem vane templates:

Table 10.2: Tandem Vane Templates

Template Name	Description
TandemVaneAlignedHigh TandemVaneAlignedLow	The leading edge of the second blade is positioned, in the meridional direction, near or behind the trailing edge of the first blade. The second blade should be offset by approximately half a blade pitch (in the theta direction) compared to the first blade.

Template Name	Description
	The leading and trailing edges of the first blade and second blade are rounded. You must choose one of the two templates. For details, see Choosing the Appropriate TandemVaneAligned Template (p. 139).
TandemVanesCustomizedNr1	The leading edge of the first blade is positioned, in the meridional direction, near or behind the trailing edge of the second blade. The leading edge of the first blade is positioned at a lower Theta value than the trailing edge of the second blade. The leading edge of the first blade is rounded. The trailing edge of the first blade is cut-off. The leading and trailing edges of the second blade are rounded. For details, see The TandemVanesCustomizedNr1 Template (p. 142).

To access these templates, enable advanced features. For details, see Advanced Topology Control (p. 133).

10.4.5.1. Choosing the Appropriate TandemVaneAligned Template

The tandem vane template has a "Low" and "High" version. To determine which tandem vane template is appropriate:

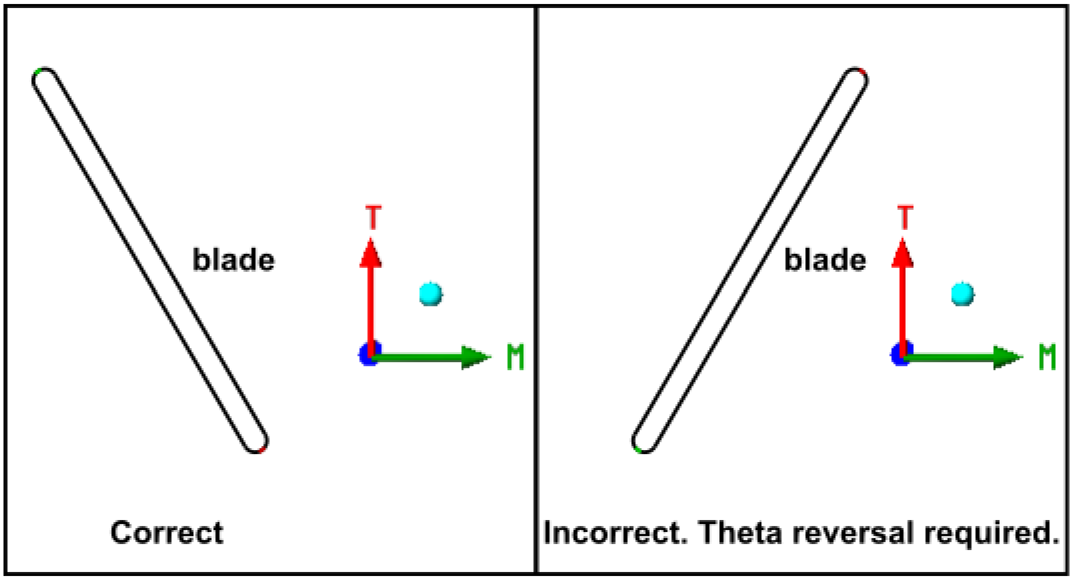
1. Right-click in the viewer and select **Transformation** > **3D Turbo (Theta-M'-Span)**.

2. Orient the view so that the meridional coordinate (M) increases to the right and Theta (T) increases upward.

   After you do this, the leading edges are on the left.

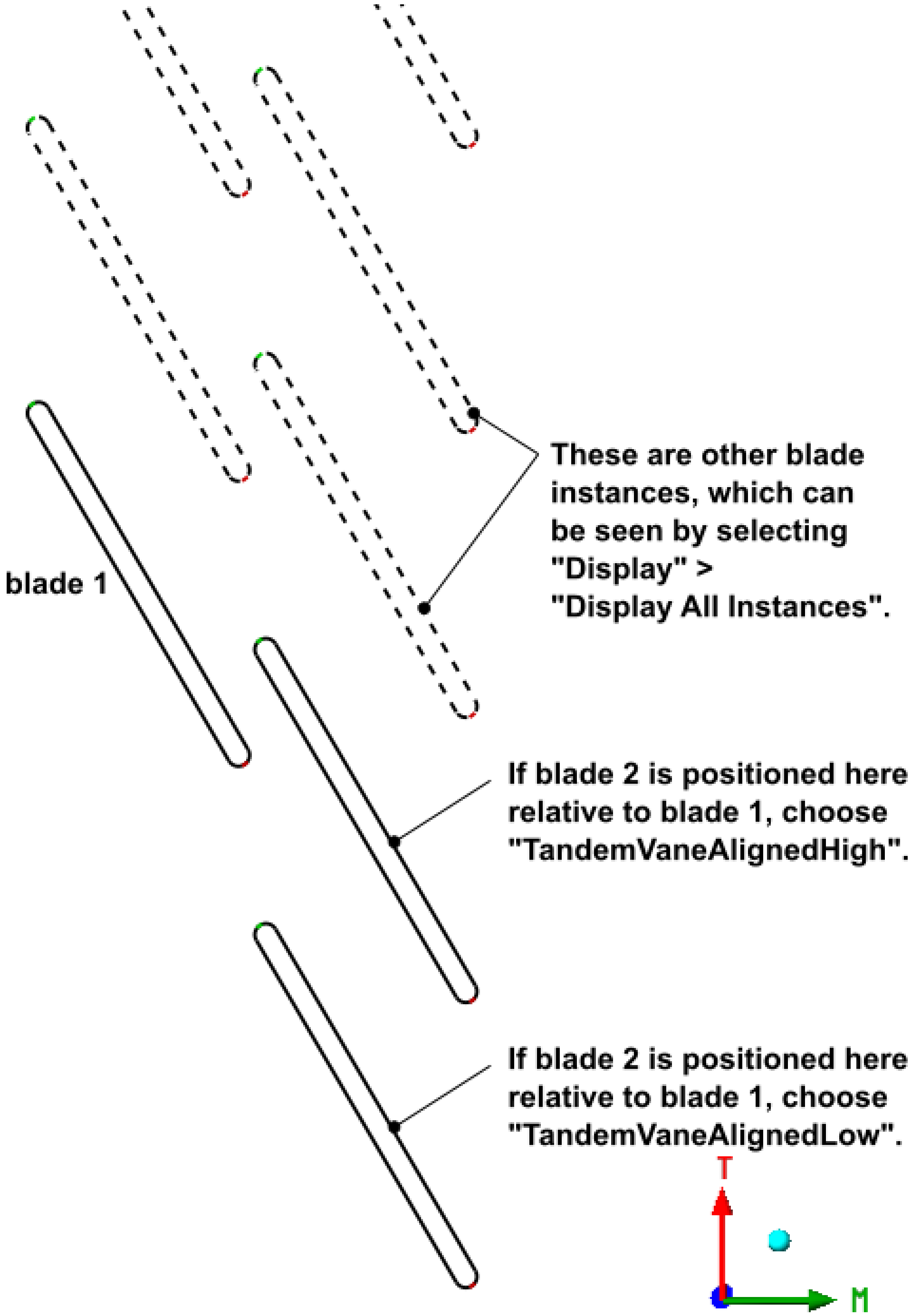
3. Proceed to the next step if Theta decreases towards the trailing edge (towards higher meridional coordinate). Otherwise, use the settings under **Geometry** > **Machine Data** > **Rotation** to reverse the direction of increasing Theta in relation to the geometry.

Figure 10.10: Theta Definition Requirement



4. If Theta increases from the first blade's trailing edge to the second blade's leading edge, select `TandemVaneAlignedHigh`. Otherwise, select `TandemVaneAlignedLow`.

Figure 10.11: Choosing the Tandem Vane Template

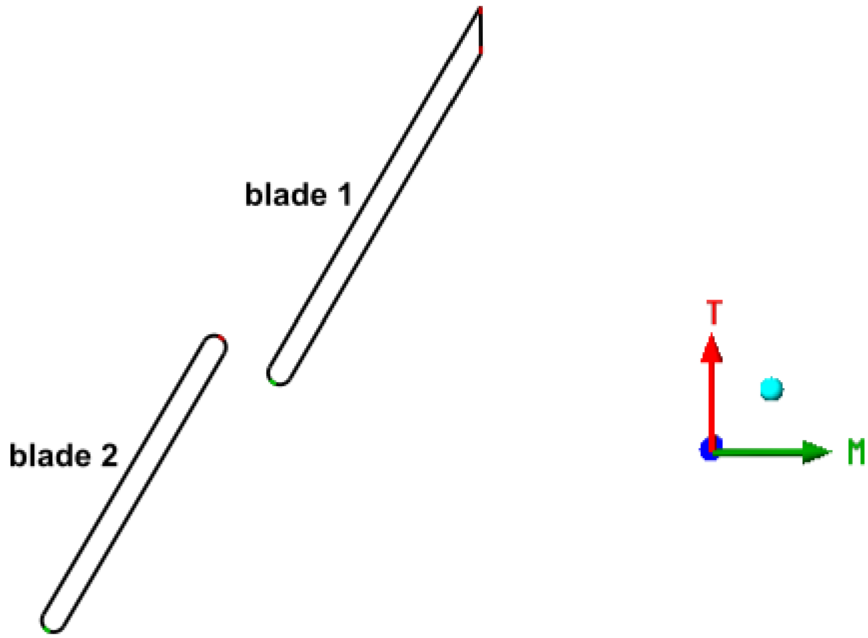


Note:

When running TurboGrid in stand-alone mode (as described in Using the Ansys TurboGrid Launcher in the *TurboGrid Introduction*), it is possible to rotate the second blade along the Theta direction, which can potentially change which template is suitable, as determined by the procedure above.

10.4.5.2. The TandemVanesCustomizedNr1 Template

An example geometry that is suitable for use with the TandemVanesCustomizedNr1 template is shown below:



10.4.6. Advanced Local Refinement Control

When TurboGrid initializes the ATM topology, it builds the topology in pieces and then conglomerates the block structure to minimize the number of blocks in the final topology. Because you can control only the number of elements along a given topology edge and not the distribution, having more blocks in specific areas, for example near the trailing edge, gives you better control of the mesh resolution in these areas. Preserving specific vertices from being conglomerated in these situations will give you better control of the local mesh resolution — the specific vertices will be dependent on the topology pieces used. For a list of the available topology pieces, see Advanced Topology Control (p. 133).

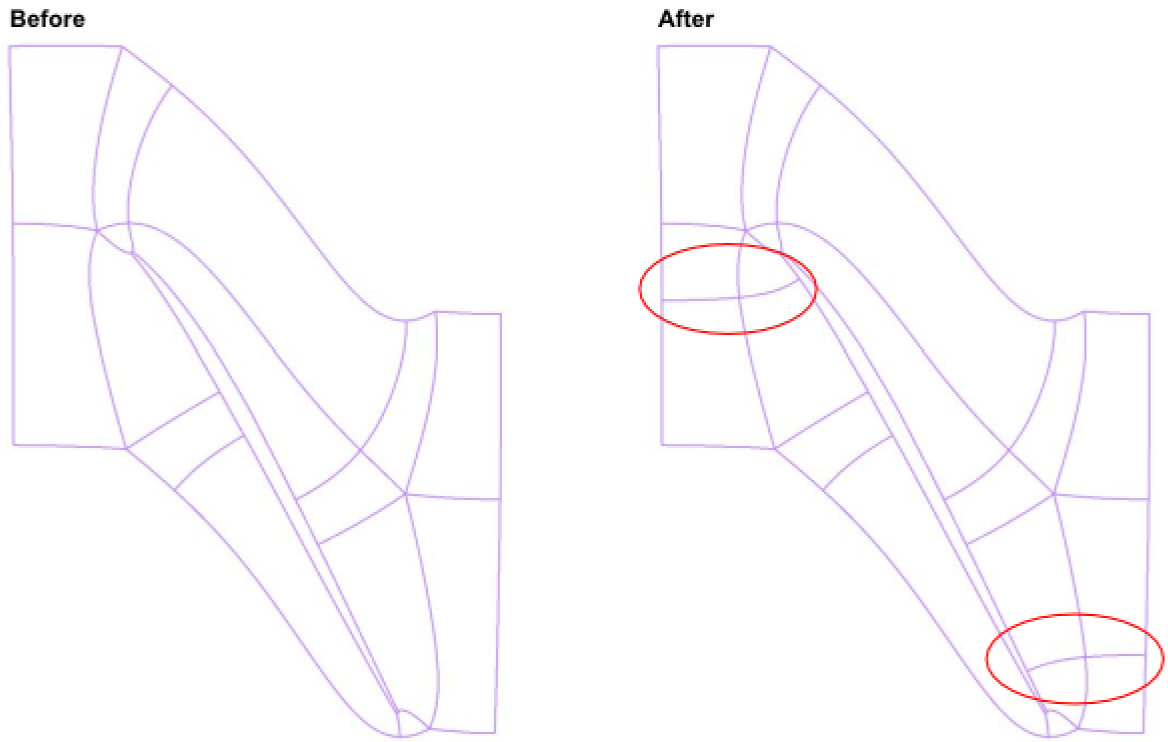
You can retain additional vertices with CCL of the following format: "template name" , "vertex number". This must be repeated for each new vertex number, leading to the form:

```
TOPOLOGY SET:
  Override Conglomeration Retained Vertices List = true
  Conglomeration Retained Vertices List = "template name", "vertex number", "template name", "vertex number", ...
END
```

The following is an example of CCL that provides additional control near the leading and trailing edges for rounded blades:

```
TOPOLOGY SET:
  Override Conglomeration Retained Vertices List = true
  Conglomeration Retained Vertices List = LERoundedSymmetricStar, 0, TERoundedSymmetricStar, 0,\
  LERoundedSymmetricStar, 13, TERoundedSymmetricStar, 13
END
```

The figure below demonstrates the result of applying the example CCL.



You can apply edge splits using `ATM Background Topology Edge Refinement` to influence the passage topology background mesh count. One typical example is to apply edge splits on widely spaced blades. The following CCL is used to apply eight edge splits on edges 8 and 10 of PassageBlock1by4:

```
TOPOLOGY SET:
  ATM Background Topology Edge Refinement = PassageBlock1by4, 8, 8, PassageBlock1by4, 10, 8
END
```

For cases where `ATM Background Topology Edge Refinement` increases the passage topology background mesh count, the final mesh count at the specified edges also goes up. You may want to add an edge split on those edges in order to decrease the final mesh count in the blade-to-blade direction.

10.4.7. Span Location for Controlling Topology

The span is the distance between the hub and shroud, expressed as a fraction between 0 and 1. The span is calculated when m′-Theta coordinates are created. For details, see Transformation Commands and Coordinate Systems (p. 177).

Ansys TurboGrid uses a span parameter for several reasons related to topology, including:

- Obtaining blade metal angles so that Ansys TurboGrid can automatically set topology types after you have specified the topology method: `H/J/C/L-Grid`.
- Calculating an appropriate number of topology blocks between the blade and the periodic interface or interface between blades.

By default, the span location is set to `0.5` (that is, 50% of the span). To adjust the span parameter, called `Span Location`, edit the `Topology Set` object using the **Command Editor** dialog box.

10.4.8. ATM3D (Advanced)

ATM3D is an alternative meshing approach to ATM. Whereas the original ATM meshing process uses a quasi-3D mesh smoothing process, ATM3D uses a full 3D mesh smoothing process. The goal of ATM3D is to ultimately produce a higher quality mesh, particularly for cases that are challenging for ATM.

The ATM3D meshing approach is an advanced feature, and is accessible only when advanced features are enabled. To enable advanced features, select **Edit** > **Options**, then, in the **Options** dialog box, with `TurboGrid` selected in the tree, select **Enable Advanced Features** and click **OK**.

ATM3D is not active by default. To use it by default, select **Edit** > **Options**, then, in the **Options** dialog box, with `TurboGrid` selected in the tree, select **Use ATM3D Meshing By Default (Advanced)** and click **OK**. This preference takes effect the next time you start TurboGrid.

For a given case, you can activate ATM3D by opening `Topology Set` and selecting **Use ATM3D Mesh Generation (Advanced)**.

Current Limitations of ATM3D

- Geometry: Cut-off blade geometry is not supported: If the geometry has a cut-off blade, ATM3D is turned off regardless of the **Use ATM3D Meshing By Default (Advanced)** preference setting. For details, see Cut-off or square (p. 117).
- Topology: A conformal blade tip is not supported. This option, which is described in Tip Topology Option (p. 134), is not available.
- Layers: Compared to ATM, ATM3D has less need to adjust the number of layers. When using ATM3D, the `Layers` object's **Layers** tab > **Intermediate Layers** > **Insertion Mode** setting can only be set to `Manual - Uniform`. You can only change the number of intermediate layers via the **Count** setting. For details, see Defining Intermediate Layers (p. 161).
- Layers: Unlike the ATM approach, ATM3D does not have a setting to change the type of the spanwise mesh interpolation guide curves that connect the layers. For details, see Spanwise Mesh Interpolation Guide Curves (p. 163).
- Mesh Data: Inserting/editing edge splits is not supported. For details, see Changing the Number of Elements on a Selected Master Topology Edge (p. 159).
- Mesh Data: When using the ATM3D meshing approach, the **Constant First Element Offset** setting is not available. For details, see Constant First Element Offset (p. 148).

Note:

- If you encounter high aspect ratio elements along the leading/trailing edge, or elements with poor angles:
  - Try adjusting the number of layers.

  - Try increasing the value of parameter `Domain Meshing Smoothing Iterations After Remesh`. This parameter is accessed by right-clicking the `3D Mesh` object and selecting **Edit in Command Editor**.

- With ATM3D, layers show neither the refined boundary layer mesh around the blade nor the blade tip mesh. You can, as usual, inspect the mesh using the `3D Mesh` > `Show Mesh` object.

10.5. Mesh Data

To define the mesh properties for the entire mesh, edit the `Mesh Data` object from the object selector or click *Edit Mesh Data* . To define or modify the mesh properties of a blade-specific portion of the mesh, edit the corresponding object stored under the `Mesh Data` object.

Defining the `Mesh Data` object does not create the mesh. The mesh data can only be set after the topology has been defined, but it is recommended that you set the required mesh size before you create the layers (discussed later), because the true mesh nodes are displayed on the topology layers as the topology is adjusted. For details on creating a mesh, see Steps to Create a Mesh (p. 95) and Mesh Command (p. 53).

10.5.1. The Mesh Data Objects

The `Mesh Data` object contains settings that affect the mesh globally. The individual blade mesh data objects (stored under the `Mesh Data` object in the object selector) contain a subset of the settings of the `Mesh Data` object, and affect the mesh for individual blades in the blade set.

10.5.1.1. Mesh Size Tab

The **Mesh Size** tab is applicable to the `Mesh Data` object.

10.5.1.1.1. Lock Mesh Size Check Box

In some situations, for example when you are comparing minor geometry variations, you may want to keep the mesh dimensions the same from one case to the next. This will reduce the influence of the mesh on the CFD results. To lock the mesh dimensions, select the **Lock mesh size** check box. This setting will maintain the current mesh dimensions and prevent any subsequent changes to the global or local mesh controls. The lock status will be preserved from one session to the next if the project or state file is saved. To unlock the mesh, clear the **Lock mesh size** check box.

10.5.1.1.2. Method

The **Method** setting controls the mesh density, and can be set to one of the following:

- `Target Passage Mesh Size`

  The `Target Passage Mesh Size` method sets a target for the number of nodes in the mesh passage. The pre-set options for the **Node Count** setting are `Coarse (20000)`, `Medium`

`(100000)`, and `Fine (250000)`. There is also an option, `Specify`, to specify the target number of nodes. In this mode, if you change the spanwise mesh size or the boundary layer refinement, or make any local refinements to the mesh, Ansys TurboGrid will adjust the mesh size factor (Global Size Factor) automatically to achieve the desired target mesh size.

Ansys TurboGrid attempts to reach the target number of nodes by:

– Adjusting the number of elements placed along each topology block edge. The number of elements can be viewed after pressing **Apply** by changing the method to `Global Size Factor`.

– Adjusting the number of elements from hub (or hub tip) to shroud (or shroud tip) if these have not been explicitly set.

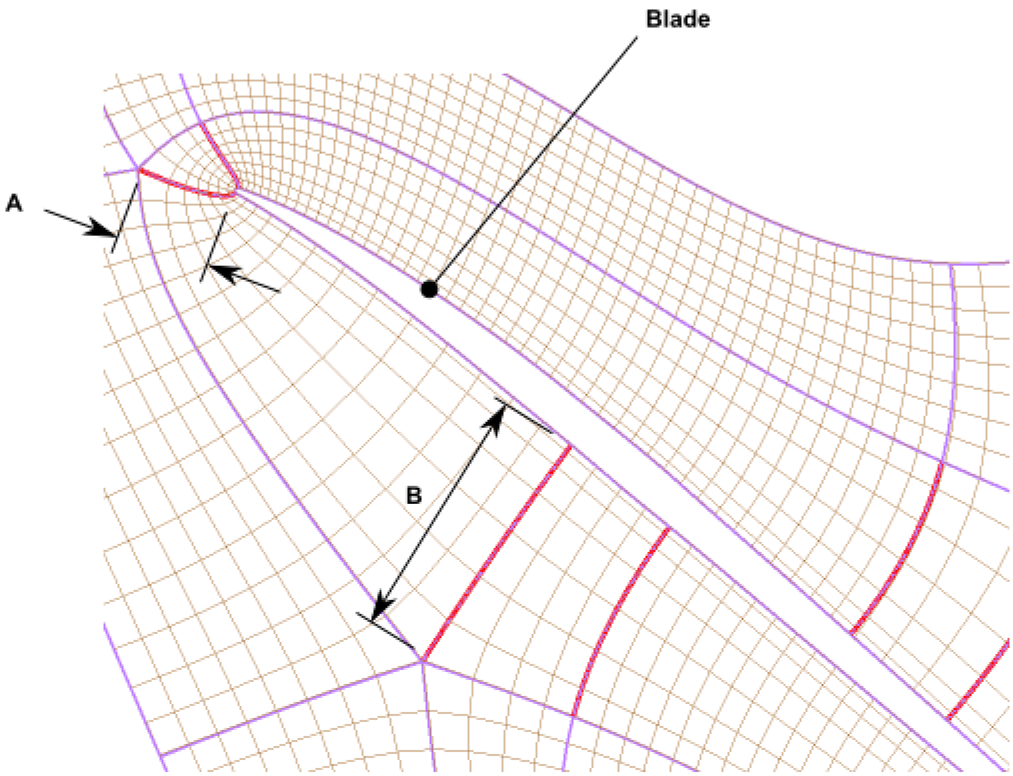
- `Global Size Factor`

  The `Global Size Factor` method defines the overall mesh size. To increase the resolution of the mesh, increase the size factor using the **Size Factor** setting. Note that the change in overall mesh size is not linear. In this mode, if you change the spanwise mesh size or the boundary layer refinement, or make any local edge refinements to the mesh, the Global Size Factor will stay fixed and the overall mesh size may change. This factor, when used with proportional refinement, can be used to scale the mesh size for a mesh refinement study.

10.5.1.1.3. Boundary Layer Refinement Control

In the context of the **Boundary Layer Refinement Control** settings, the boundary layer region is represented by the set of topology blocks along the sides of the blade. The thickness of the boundary layer region varies around the blade profile, as indicated by distances A and B in Figure 10.12: Variations in boundary layer region thickness (p. 147).

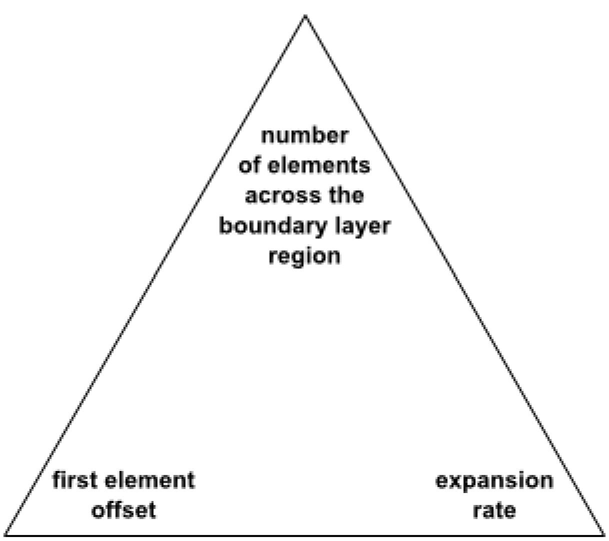
Figure 10.12: Variations in boundary layer region thickness



Within the boundary layer region around the blade, TurboGrid uses various expansion rates to distribute the elements normal to the blade surface to assure a smooth transition to the passage mesh.

The boundary layer region thickness is a function of the variables shown in Figure 10.13: Variables that control the boundary layer distribution (p. 147).

Figure 10.13: Variables that control the boundary layer distribution



The number of elements across the boundary layer region is constant around the blade profile. Because the boundary layer region thickness (see Figure 10.12: Variations in boundary layer region thickness (p. 147)) changes around the blade, either the expansion rate or the first element offset (that is, the height of the first row of elements next to the blade) can be constant, but the other must change. **Constant First Element Offset** (see Constant First Element Offset (p. 148)) controls which aspect is held constant.

You can specify which property of the boundary layer region to control by using the **Boundary Layer Refinement Control** > **Method** setting. The options are:

- `Proportional to Mesh Size`

  This option controls the number of elements across the boundary layer region in proportion to the values specified for **Factor Base** and **Factor Ratio**. This method maintains similar expansion rates when the Global Size Factor is changed. If the blade has a cut-off edge, **Factor Ratio** also controls the number of elements along the cut-off edge.

  Increasing the value of **Factor Base** or **Factor Ratio** increases the number of elements across the boundary layer and along the cut-off edge (if applicable). Conversely, decreasing the value of **Factor Base** or **Factor Ratio** decreases the number of elements. The number of elements across the boundary layer is calculated as Base Count * Global Size Factor * (Factor Base + Factor Ratio * Global Size Factor). The default values of Factor Base and Factor Ratio are 3 and 0 respectively.

- `First Element Offset`

  This option directly controls the height of the first row of elements next to the blade.

- `Edge Refinement Factor`

  This factor is multiplied with the base count to determine the number of edge splits in the boundary layer. The count will not change if the Global Size Factor is modified.

When **Constant First Element Offset** is selected, you can influence the range of expansion rates using the **Target Maximum Expansion Rate** settings, described in Target Maximum Expansion Rate (p. 149).

When the topology is created, Ansys TurboGrid creates a `Boundary Layer Control` object under the `Mesh Data` object in the object selector. The `Boundary Layer Control` object shows information about the boundary layer.

Note:

When **Boundary Layer Refinement Control** > **Method** option is set to `First Element Offset`, the offset value can be controlled either in the `Boundary Layer Control` object or the **Mesh Data** > **Mesh Size** tab.

10.5.1.1.3.1. Constant First Element Offset

The **Constant First Element Offset** setting is available only when using the ATM method; it is not available when using the ATM3D method.

When **Constant First Element Offset** is selected, TurboGrid uses a double-sided node distribution (a non-constant expansion rate) for the boundary layer. In this case, the near-wall expansion rates that are reported in the `Boundary Layer Control` object indicate the approximate minimum and maximum values based on a sample of the layer values.

When **Constant First Element Offset** is not selected (or not available) TurboGrid uses a single sided node distribution for the boundary layer; at any given position around the blade (in the blade-to-blade view), there is a constant expansion rate across the boundary layer. In this case, the user-specified first element offset is applied approximately as an average offset.

Note:

Mesh element aspect ratios (spanwise length to wall offset length) on the blade surface at the leading and trailing edges might be higher with **Constant First Element Offset** not selected (or not available) compared to with **Constant First Element Offset** selected. With ATM3D, you can reduce these aspect ratios by increasing the spanwise mesh resolution or by increasing the boundary layer first element offset. If you see solver convergence issues related to high aspect ratios, try running a double precision solver.

10.5.1.1.3.2. Cutoff Edge To Boundary Layer

The **Cutoff Edge To Boundary Layer** setting is available only when **Constant First Element Offset** is not selected.

You can adjust **Cutoff Edge To Boundary Layer** > **Factor** to change the expansion rate used near the cut-off edge.

10.5.1.1.3.3. Cutoff Edge Split Factor

The **Cutoff Edge Split Factor** setting is available only when **Constant First Element Offset** is selected.

In cases where the leading edge and/or trailing edge is cut-off, you can specify a factor for each cut-off edge to control the number of elements along that edge (as viewed in any given layer).

10.5.1.1.3.4. Target Maximum Expansion Rate

The **Target Maximum Expansion Rate** setting is available only when **Constant First Element Offset** is selected.

Selecting **Target Maximum Expansion Rate** enables the specification of a target maximum expansion rate. TurboGrid attempts to prevent the expansion rate (at any place around the blade profile) from exceeding the specified maximum in different ways, depending on the **Boundary Layer Refinement Control** > **Method** option:

- `Proportional to Mesh Size`

  The number of elements across the boundary layer is fixed. TurboGrid may increase the first element offset in order to reduce the maximum expansion rate as appropriate.

- `First Element Offset`

The first element offset is fixed. TurboGrid may increase the number of elements across the boundary layer in order to reduce the maximum expansion rate as appropriate.

Note:

Target Maximum Expansion Rate is available only when **Constant First Element Offset** is selected. Furthermore, **Constant First Element Offset** is not applicable (effectively not selected) when using the ATM3D meshing approach, which makes **Target Maximum Expansion Rate** unavailable.

If reducing the maximum expansion rate to the specified maximum results in a sub-unity expansion rate (that is, a contraction rate) elsewhere, specifically such that the expansion rate is, at some place, less than the inverse of the specified maximum, then TurboGrid may balance the amounts by which:

- The maximum expansion rate is above the maximum
- The minimum expansion rate is below the inverse of the maximum

If the balance involves exceeding the rate by more than 15%, the problem will be indicated in the `Boundary Layer Control` object (which is under the `Mesh Data` object in the object selector).

10.5.1.1.3.5. Near Wall Element Size Specification

The **Near Wall Element Size Specification** setting controls the method by which the near-wall node spacing is specified on the **Passage** and **Hub/Shroud Tip** tabs. The near-wall node spacing is the distance between a wall (for example, hub, shroud, or blade) and the first layer of nodes from the wall.

The available **Method** options for calculating the near wall spacing are:

- `y+`

  For details, see Y Plus (p. 150).

- `Absolute`

  For details, see Absolute (p. 151).

10.5.1.1.3.5.1. Y Plus

The y^+ method allows you to set the near wall spacing, Δy, in accordance with a target value of y^+. The target value of y^+ may then be specified on the **Passage** and **Hub/Shroud Tip** tabs, as applicable (that is, when a near wall size is required by the specified distribution method).

The following formula relates the near wall spacing to y^+:

$$\Delta y = L \Delta y^+ \sqrt{80}\, Re_x^{1/14} \frac{1}{Re_L} \tag{10.1}$$

where L is the blade chord, Δy^+ is the specified target y^+ value, Re_x is the Reynolds number based on the distance along the chord (measured from the leading edge), and Re_L is the

Reynolds number based on chord length. Ansys TurboGrid approximates L as the algebraic average of the chord lengths of each blade profile in the blade file. You must specify Re_L. Ansys TurboGrid approximates Re_x as being equal to the specified value of Re_L.

Note:

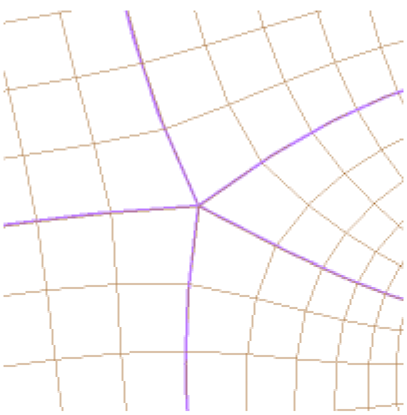
If you specify a near wall size (on the **Passage** or **Hub/Shroud Tip** tab) when the near wall method is not y^+, you can switch the method to y^+ (at least temporarily) to see the estimated value of y^+ as a setting on the **Passage** or **Hub/Shroud Tip** tab.

10.5.1.1.3.5.2. Absolute

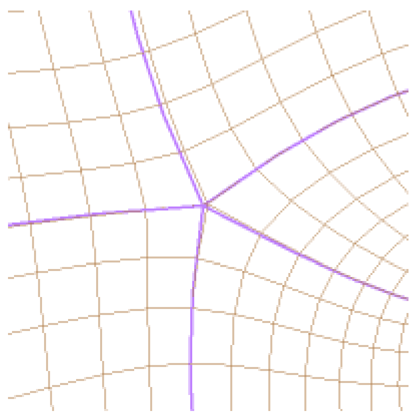
The `Absolute` method allows you to set the near wall spacing directly. Such a specification (that is, on the **Passage** or **Hub/Shroud Tip** tab) requires a dimensional value with units of distance.

10.5.1.1.4. Five-Edge Vertex Mesh Size Reduction

An ATM-based mesh may contain one or more vertices that each join exactly five edges. At these locations, the mesh generation process will tend to pull the mesh away from the five-edge vertex to make the surrounding element edges more orthogonal, resulting in relatively larger elements surrounding the vertex than the rest of the mesh in their vicinity. The **Five-Edge Vertex Mesh Size Reduction** > **Factor** setting, when set to a value less than 1.0, will reduce this tendency, trading off between mesh orthogonality for sizing.



Factor = 1.0



Factor = 0.5

10.5.1.1.5. Inlet Domain and Outlet Domain Check Boxes

The **Inlet Domain** and **Outlet domain** check boxes determine whether or not the inlet and outlet domains are to be generated as part of the mesh. Settings that affect these grid regions are found on the **Inlet/Outlet** tab.

10.5.1.2. Passage Tab

The **Passage** tab is applicable to the `Mesh Data` object, and is used to specify distribution settings. Distribution settings are described generically in Distribution Settings in General (p. 154).

10.5.1.2.1. Spanwise Blade Distribution Parameters

Use the **Spanwise Blade Distribution Parameters** section to control the distribution of mesh elements in the spanwise direction along the blade.

The **Method** setting and its associated settings are described in Distribution Settings in General (p. 154).

10.5.1.3. Hub Tip and Shroud Tip Tabs

The **Hub Tip** and **Shroud Tip** tabs of the `Mesh Data` object are available when their respective geometry objects (Hub Tip and Shroud Tip) exist. For details on defining those geometry objects, see The Hub Tip and Shroud Tip Objects (p. 120).

The **Hub Tip** and **Shroud Tip** tabs are used to specify distribution settings.

10.5.1.3.1. Hub Tip Distribution Parameters and Shroud Tip Distribution Parameters

Use these settings to control the distribution of mesh elements in the spanwise direction across the tip gap.

Set Method to one of the following:

- `Match Expansion at Blade Tip`
- `Element Count and Size`
- `Uniform`

These methods are described in Distribution Settings in General (p. 154).

10.5.1.3.2. Blade Tip Settings

The settings under **Blade Tip** (for cases with a single blade per blade set) or **Apply Blade Tip Parameters To All Blades** (for cases with multiple blades per blade set) control details of the mesh elements on the blade tip (and throughout the blade tip region of the mesh).

The settings that are available depend on the type of leading edge and the type of trailing edge:

10.5.1.3.2.1. Blades with two rounded edges

The first element height in the tip mesh is matched with the first element height in the boundary layer region (outside the blade). The match may not be exact in rounded leading edge or trailing edge regions.

In the tip mesh, the mesh elements change height across the blade thickness according to locally applied near-wall expansion rates. These expansion rates are limited at the thickest part of the blade by a target maximum rate, which is determined by the first applicable method of the following methods:

- If you select **Override Target Maximum Expansion Rate**, the corresponding specified rate is used.
- If you select **Target Maximum Expansion Rate** (on the **Mesh Size** tab), the corresponding specified rate is used.
- The maximum achieved expansion rate in the boundary layer region (outside the blade) is used.

The expansion rate influences the number of mesh elements across the blade tip.

After the topology has been generated, TurboGrid reports the minimum and maximum achieved expansion rates, and the number of mesh elements across the blade tip, in the **Hub Tip** and **Shroud Tip** tabs.

10.5.1.3.2.2. Blades with one rounded edge and one cut-off edge

The distribution of elements across the blade tip mesh is governed by the distribution of elements across the cut-off edge. The latter is controlled indirectly by the **Cutoff Edge Split Factor** setting (on the **Mesh Size** tab) which controls the number of elements across the cut-off edge. For details, see Cutoff Edge Split Factor (p. 149).

10.5.1.3.2.3. Blades with two cut-off edges

The **Tip Centerline Location** setting is available when the following conditions are met:

- The corresponding blade tip, hub or shroud, exists.
- The tip topology is set to `H-Grid Not Matching`.
- The corresponding edge of the blade, leading or trailing, is cut-off.

The setting controls how the blade centerline behaves at the cut-off end. The options are:

- `Automatic`

  Effectively selects one of the other two methods automatically, based on the geometry. If one of the corners of the cut-off edge is at a sufficiently acute angle (default is 45°), then the `Corner` option is selected, otherwise the `Middle` option is selected. The threshold angle can be set by defining the parameter `GGI Tip Angle To Switch Mean Line Into Cut Off Corner` (to an angular value; default unit is `[degree]`) in the CCL for the blade-specific mesh data object.

- `Middle`

  The centerline meets the cut-off blade edge in the middle (between the corners).

- `Corner`

The centerline meets the corner of the cut-off blade edge.

10.5.1.4. Distribution Settings in General

The distribution of elements along the blade (in the spanwise direction) is controlled by one of these methods:

- `Element Count and Size`

  To use the `Element Count and Size` method, specify the number of elements that span the pertinent extent and whether the distribution is uniform or non-uniform. If the distribution is non-uniform, you must also specify the number of uniformly-distributed elements (**Const Elements**) and the element s ize next to the pertinent wall(s).

- `Boundary Layer`

  To use the `Boundary Layer` method (not applicable for O-Grid settings), specify distribution parameters for three sections: boundary layer at the hub, boundary layer at the tip/shroud, and the section in between. **Layer Offset** means the thickness of the boundary layer and **Wall Offset** means the thickness of the first element next to the wall.

- `Expansion Rate`

  To use the `Expansion Rate` method, specify an expansion rate and the element size next to the blade wall.

- `Uniform`

  To use the `Uniform` method, specify the number of elements. Each element is the same size.

- `Proportional` (the only option when using a meridional splitter)

  This is the default method when ATM topology is used. With this method, Ansys TurboGrid automatically computes the number and distribution of elements in the spanwise direction so that the near wall element heights on the hub and shroud match the maximum near wall element height on the blade. Ansys TurboGrid adjusts the spanwise mesh count so that the average element aspect ratio (spanwise height divided by streamwise length) near the midspan is approximately equal to the blade aspect ratio divided by the Proportional **Factor**. Increasing or decreasing the Proportional **Factor** increases or decreases the mesh resolution in the spanwise direction respectively.

  When there is a meridional splitter, the **Passage Boundary Layer Refinement Control** > **Factor** setting is available, and scales the near wall element heights on the hub, meridional splitter, and shroud surfaces.

Note:

The **Const Elements** value cannot be negative and must be at least 2 less than the **# of Elements** value for that region.

The distribution of elements across a tip gap (in the spanwise direction) is controlled by one of these methods:

- `Match Expansion at Blade Tip`

  The `Match Expansion at Blade Tip` method is default. This method will apply the same expansion factor as that used along the blade span, and will match the size of the elements on each side of the tip clearance gap with the size of the first element along the blade. An appropriate number of elements is then derived.

- `Element Count and Size`

  (described above)

- `Uniform`

  (described above)

When using the `End Ratio` method, the element count and wall size(s) are updated accordingly in the **Element Count and Size** settings. (When two wall sizes are adjusted, they are adjusted to the same value.) When using the `Element Count and Size` method, the **End Ratio** setting is updated accordingly.

In the case of an O-Grid distribution, the **Blade End Ratio** is the size ratio of the passage mesh element nearest the O-Grid to the O-Grid element nearest the blade surface. A special feature of the distribution settings for O-Grids is that, if **# of Elements** is set to `0`, the number of elements will be calculated automatically so that the best possible matching of element sizes at the interface between the O-Grid and the passage is ensured.

Size of Elements Next to Wall (Normalized) is normalized based on the total distance along the blade (not counting any tip clearance gaps). For example, on the **Hub Tip** tab, a value of `0.05` for **Size of Elements Next to Wall (Normalized)** > **Hub** represents an element size of five percent of the distance along the blade, applied on the elements next to the hub, and a value of 0.05 for the **Size of Elements Next to Wall (Normalized)** > **Tip** represents an element size of five percent of the distance along the blade, applied on the elements next to the hub end of the blade (both in the tip clearance gap and on the blade).

Note:

The **Size of Elements Next to Wall (Normalized)** value must be less than 1 (unity) divided by the number of elements for that region. If it is not, Ansys TurboGrid decreases it to meet this criterion.

To increase the quality of the mesh, try to minimize drastic changes in element size. Wherever possible, attempt to have gradual increases and decreases in element size in all directions.

10.5.1.5. Inlet/Outlet Tab

The geometries of the inlet and outlet domains of the mesh are controlled by the hub and shroud curves, and the `Inlet` and `Outlet` geometry objects. For details, see The Hub and Shroud Objects (p. 108) and The Inlet and Outlet Objects (p. 122).

The **Inlet/Outlet** tab contains settings, as applicable, that affect the mesh in the inlet and outlet domains. The **Inlet Domain** and **Outlet domain** check boxes on the **Mesh Size** tab of the `Mesh`

`Data` object determine whether or not the inlet and outlet domains are to be generated as part of the mesh.

The nodes of the inlet and outlet domains match one-to-one with the nodes of the passage domain where they meet at the interfaces. The rest of the mesh in the inlet or outlet domain is then as near to being isotropic and orthogonal as possible.

10.5.1.5.1. Inlet Domain and Outlet Domain Settings

Mesh Type can be set to one of the following options:

- `H-Grid` (default)

  If you have any hub and/or shroud regions defined (see Hub Regions and Shroud Regions Tab (p. 110)), and the applicable domain (**Inlet Domain** or **Outlet Domain**) is selected on the **Mesh Size** tab, then setting **Mesh Type** to `H-Grid` makes available the **Limit Aspect Ratio** and **Inlet Multi Segment Enabled**/**Outlet Multi Segment Enabled** settings (described below).

- `H-Grid in Parametric Space`

  In this case, the resulting mesh tries to follow a parametric space created by an elliptic smoothing method. This type of mesh is particularly suitable for return channels.

Both options specify an H-Grid type mesh in the inlet/outlet domain that preserves the boundary layer mesh resolution at the hub and shroud and automatically matches the element size at the interface between the inlet/outlet domain and passage domain.

The streamwise distribution of elements is non-uniform and uses a constant expansion rate.

Define By can be set to one of the following options:

- `Target Expansion Rate` (default)

  In this case, the number of elements in the inlet/outlet domain is determined by matching the element size at the passage interface and expanding the size at a prescribed rate for each successive element. The default expansion rate (element size ratio) is 1.2

- `Target Number of Elements`

  The **# of Elements** setting controls the target number of elements in the inlet/outlet domain in the streamwise direction. The expansion rate is determined by the target number of elements, the streamwise length of the inlet/outlet domain, and the passage element size.

Important:

Clicking the **Apply** button does not create the mesh, it only saves the `Mesh Data` object and updates the 2D mesh previews. To create the mesh, select **Insert** > **Mesh** from the main menu or click *Create Mesh* .

If you have any hub and/or shroud regions defined (see Hub Regions and Shroud Regions Tab (p. 110)), and the applicable domain (**Inlet Domain** or **Outlet Domain**) is selected on the **Mesh Size** tab, and **Mesh Type** is set to `H-Grid`, then the **Limit Aspect Ratio** option is available.

The **Limit Aspect Ratio** option enables you to specify a maximum target aspect ratio for mesh elements in the inlet/outlet domain by setting a value for **Request Max AR** value. TurboGrid may increase the element count in the streamwise direction in an attempt to reduce the maximum aspect ratio to (or below) the target value. The existing maximum aspect ratio is shown in the object editor beside the label **Current Max AR**. Note that the number of elements in the spanwise direction is controlled by the passage mesh.

If you have any hub and/or shroud regions defined (see Hub Regions and Shroud Regions Tab (p. 110)), and the applicable domain (**Inlet Domain** or **Outlet Domain**) is selected on the **Mesh Size** tab, and **Mesh Type** is set to `H-Grid`, then the **Inlet Multi Segment Enabled**/**Outlet Multi Segment Enabled** option is available, and is selected by default. When this option is selected:

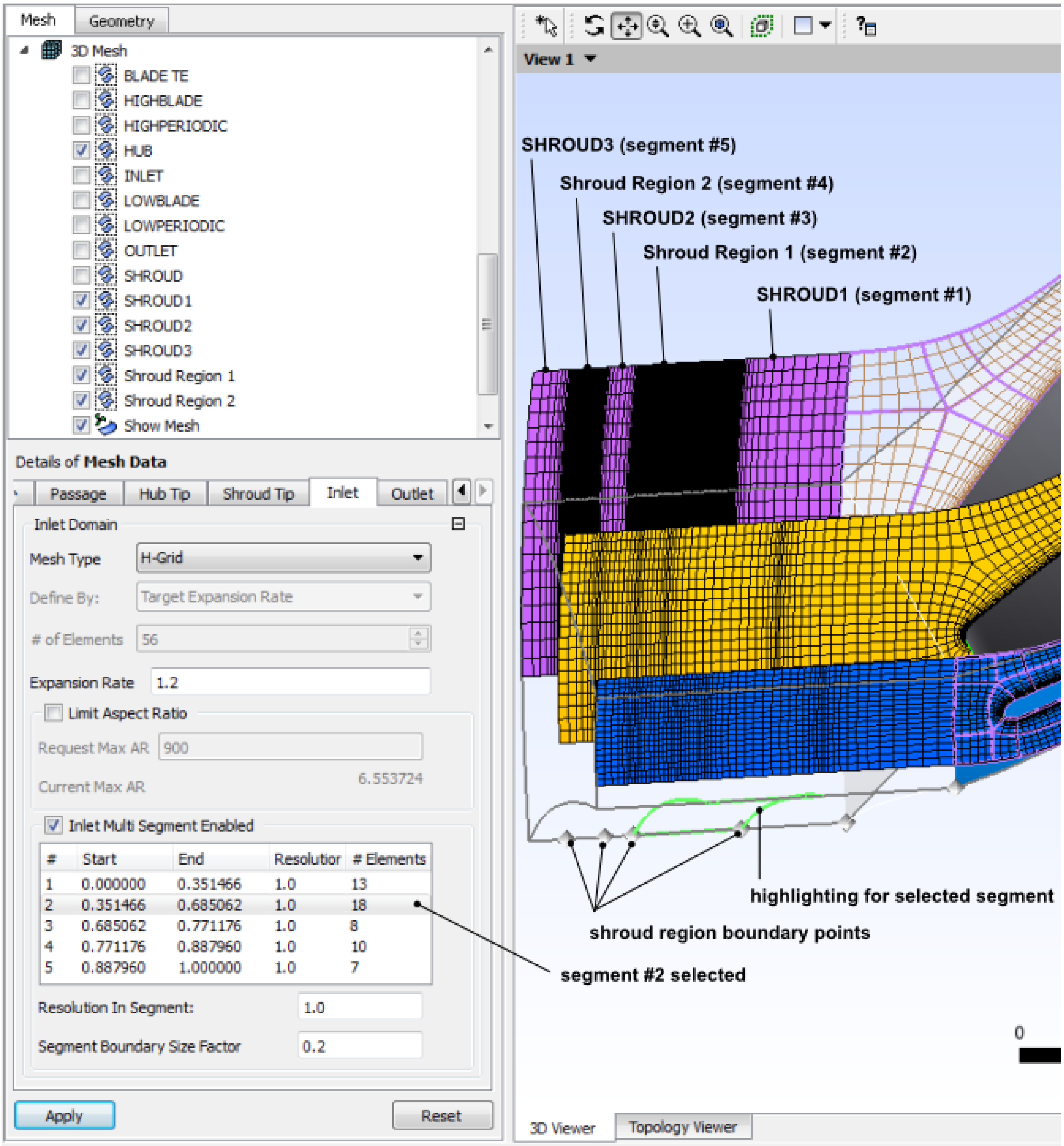
- Any defined hub/shroud regions are enabled (affecting the mesh and surface regions in the output), and
- Settings for controlling the mesh distribution within the hub/shroud regions become available.

  If you select a segment (a defined hub/shroud region or a numbered surface region that is beside a defined hub/shroud region) in the list, the segment is highlighted in the **3D Viewer** and you can then change the resolution via the **Resolution in Segment** setting. The specified value directly scales the number of elements (**# Elements**) along the segment (in the streamwise direction). Note that the minimum number of elements along any segment is 4.

  The **Segment Boundary Size Factor** setting directly scales the element thickness (as measured in the streamwise direction) on all segment boundaries except the outermost segment boundaries. Note that this setting indirectly affects the number of elements along each and every segment.

Figure 10.14: Inlet Domain Segments (p. 158) shows an example of a `3D Mesh` object with 2D regions that correspond with the segments inside the inlet domain.

Figure 10.14: Inlet Domain Segments



In this example, there are:

- Two defined shroud regions in the inlet domain: "Shroud Region 1" and "Shroud Region 2". These regions can be renamed and otherwise redefined in the `Shroud` geometry object settings.
- Three other regions on the shroud in the inlet domain: "SHROUD1", "SHROUD2", and "SHROUD3". These regions are automatically generated in order to cover the rest of the shroud within the inlet domain.

- No defined hub regions. (Hub regions could potentially be added.)

10.5.1.6. Mesh Around Blade Tab

The **Mesh Around Blade** tab is applicable to the blade-specific mesh data objects.

The **Size of Elements Next to Wall (Normalized)** setting is described in Distribution Settings in General (p. 154).

The **Hub Tip**, and **Shroud Tip** settings are for controlling the number of elements across the hub tip gap or shroud tip gap. They are similar to the settings on the **Hub Tip** and **Shroud Tip** tabs for the `Mesh Data` object. For details, see Hub Tip and Shroud Tip Tabs (p. 152).

10.5.1.6.1. Tip Centerline Location

For details, see Blades with two cut-off edges (p. 153).

10.5.2. Changing the Number of Elements on a Selected Master Topology Edge

You can control the number of elements along any master topology edge that is outside of the boundary layer region by right-clicking the edge in the viewer and selecting one of the following commands:

- **Insert Edge Split Control**

  This option inserts an `Edge Split` object under the `Mesh Data` object, provided that the applicable master topology edge is outside, or along the outer perimeter of, the boundary layer region. You can then edit the `Edge Split` object to adjust the edge split factor.

  Note:

  `Edge Split` objects are not available when using the ATM3D meshing approach.

- **Increase Edge Refinement** and **Decrease Edge Refinement**

  These options enable you to choose a preset amount by which to increase or decrease the edge split factor. An `Edge Split` object is automatically added for the selected edge if one has not been created already.

The `Edge Split` object displays the base and actual block edge split counts for the applicable master topology edge. You can remove the `Edge Split` object by right-clicking it in the object selector and selecting **Delete**.

10.6. Layers

A layer shows the topology projected onto a given span. By default, and as a minimum, there are two layers present: one at the hub and one at the shroud. In many cases, additional layers are required to improve mesh quality. The addition of layers improves the 3D mesh by adapting the topology to the local geometry before mesh generation. Creating (and adjusting if necessary) additional layers enhances the quality of the mesh by creating a curve for the mesh to follow between the hub and the shroud. The more complex the blade shape-change from hub to shroud, the more layers are required.

10.6.1. Adding Layers

Layers can be added from the editor for the `Layers` object (see The Layers Object (p. 160)), or by right-clicking a layer object in the selector, and selecting the appropriate command from the **Insert** submenu of the shortcut menu. Layers can also be added automatically at the time of mesh creation (default), depending on a setting found in the editor for the `Layers` object.

10.6.2. Deleting Layers

The object selector and the editor for the `Layers` object (The Layers Object (p. 160)) each have an icon to delete layers () as well as a shortcut menu option for deleting layers.

10.6.3. Editing the Settings of Layers

The editor for the `Layers` object (The Layers Object (p. 160)) and the editor for layer objects (Layer Objects (p. 164)) are used to control various properties of layers.

10.6.4. Layer Visibility

There are visibility check boxes next to each layer listed in the object selector. By right-clicking a layer object, a shortcut menu will appear, allowing the following visibility options as applicable:

- `Show`

  Makes the layer visible.

- `Show + Hide All Siblings`

  Makes the layer visible and turns off the visibility of all other layers.

- `Hide`

  Turns off the visibility of the layer.

The visibility of a layer can also be controlled by the visibility settings in the individual layer objects.

To change which parts of a layer are visible, select the layer(s) from the object selector, then right-click the selection and select an appropriate rendering option from the shortcut menu. For details, see Master Topology Visibility (p. 164), Topology Visibility (p. 164), and Refined Mesh Visibility (p. 164).

10.6.5. The Layers Object

The `Layers` object is used to control the individual layer objects. You can access the Layers objects in the object selector.

10.6.5.1. Layers Tab

A number of possible operations are available for a layer by clicking it in the list box and then selecting one of the icons on the right of the list box, or by right-clicking the object and then selecting an operation from the shortcut menu.

10.6.5.1.1. Defining Intermediate Layers

To ensure that the mesh is adequately guided in the spanwise direction, a mesh might require more layers than just a Hub layer and a Shroud layer. You can set up intermediate layers in different ways, as controlled by the **Insertion Mode** option:

- `Manual - Uniform`

  TurboGrid creates the specified number (**Count**) of layers and positions them uniformly in the spanwise direction from hub to shroud.

  Existing layers are not preserved when you apply the `Manual Uniform` option.

- `Automatic - Adaptive` (default for the ATM meshing approach; unavailable for the ATM3D meshing approach)

  The `Automatic - Adaptive` option is similar to the `Manual` option (described below).

  The only difference is that TurboGrid automatically adds its "recommended number of additional Layers" (as displayed in the **Layers** object editor) upon topology generation or 3D mesh generation. The recommended number may be zero, in which case no new layers are added. Whenever new layers are added, a notification appears in the status bar in the lower-left corner.

  Note:

  The recommended number of layers is a recommendation based on blade shape, size of the topology around the blade, proximity of inlet/outlet, and so on. As such the recommendation is not always reliable.

- `Manual` (unavailable for the ATM3D meshing approach)

  You can control the number and positioning of intermediate layers.

  Existing layers are preserved when you switch to the `Manual` option. This enables you to, for example, make adjustments after using the `Manual - Uniform` option.

  You can add one layer at a time either:

  – wherever TurboGrid chooses (right-click any Layer in the list and select **New Layer** or click *New Layer* ), or

  – "after" (on the Shroud side of) any particular existing layer (right-click a layer in the list and select **Insert Layer After**).

You can add the recommended number of layers all at once, letting TurboGrid choose the layer positions (right-click a layer in the list and select **Auto Add Layers** or click *Auto Add Layers*).

You can select one or (using **Shift** or **Ctrl**) more layers, then delete them (right-click the layer or selection of layers and select **Delete Selected Layer(s)** or click **Delete Selected Layer(s)**).

Note:

When the `Manual` option is applied, new layers are not generated upon topology generation or 3D mesh generation. If you want TurboGrid to automatically add its "recommended number of additional Layers" (as displayed in the **Layers** object editor) upon topology generation or 3D mesh generation, switch to the `Automatic - Adaptive` option.

10.6.5.1.2. Count

(applies when **Insertion Mode** is set to `Manual - Uniform`)

The number of intermediate layers to be generated and placed uniformly in the spanwise direction between the Hub and Shroud layers is specified in the **Count** box.

10.6.5.1.3. New Layer

(applies when **Insertion Mode** is not set to `Manual - Uniform`)

Clicking *New Layer* (or right-clicking a layer and selecting **New Layer**) creates a new layer at the most suitable span location as determined by TurboGrid.

10.6.5.1.4. Delete Selected Layers

(applies when **Insertion Mode** is not set to `Manual - Uniform`)

Clicking *Delete Selected Layer(s)* removes the selected layer(s). Hold **Shift** or **Ctrl** to select multiple layers to delete. You cannot delete the hub or the shroud layers.

10.6.5.1.5. Inserting Layers Automatically

(applies when **Insertion Mode** is not set to `Manual - Uniform`)

Selecting **Insert** > **Layer Automatically** from the shortcut menu for any layer object will cause Ansys TurboGrid to insert one layer automatically using built-in heuristics to determine an appropriate location. The position of the inserted layer does not depend on which layer object you right-click to access the shortcut menu.

Clicking *Auto Add Layers* in the editor for the `Layers` object, or selecting **Insert** > **Layers Automatically** from the shortcut menu for a layer object, causes Ansys TurboGrid to insert layers automatically, using built-in heuristics to determine an appropriate number of layers to insert, and the location of each layer. The number of layers that Ansys TurboGrid proposes to add is shown in the editor for the `Layers` object.

To have layers inserted automatically upon topology generation or 3D mesh generation, set **Insertion Mode** to `Automatic - Adaptive`.

10.6.5.1.6. Inserting Layer After Selected Layer

(applies when **Insertion Mode** is not set to `Manual - Uniform`)

The **Insert Layer After** and **Insert** > **Layer After** shortcut menu items cause a new layer to be created at the span halfway between the selected layer and the next layer. For example, if the only existing layers are on the hub and the shroud, and **Insert** > **Layer After** is selected from the hub, Ansys TurboGrid inserts the new layer at a span of 0.5.

10.6.5.1.7. Span Location

(applies when **Insertion Mode** is not set to `Manual - Uniform`)

The span location of a layer can be modified in the **Span Location** box.

10.6.5.1.8. Spanwise Mesh Interpolation Guide Curves

Topological guide curves are constructed in the spanwise direction. Such curves pass through corresponding topology points on each layer. The curves can be constructed using line segments (piece-wise linear) or curves (Bspline curves). The **Spanwise Mesh Interpolation Guide Curves** > **Type** setting, which is available when not using ATM3D, controls the type of curve, and has the following options:

- `Automatic`

  Chooses one of the other options automatically. TurboGrid chooses `Bspline` unless either of the following conditions are met, in which case it chooses `Piece-wise linear`:

  – The blade is ruled in the spanwise direction.

  – In the `Inlet/Outlet` object, **Interface Specification Method** > **Points** is selected and more than two inlet/outlet points are defined in the list of points.

- `Piece-wise linear`

- `Bspline`

  This option may produce adverse effects in cases with many layers.

Note:

If you have negative mesh element volumes near the leading or trailing edge of the blade, switching the curve type might correct the problem. If the problem is not corrected by switching the curve type then it is recommended that you add more layers and use the `Piece-wise linear` option.

10.6.6. Layer Objects

You can access the layer objects under the `Layers` object in the object selector.

10.6.6.1. Data Tab

10.6.6.1.1. Master Topology Visibility

Clicking in the box will toggle the visibility of the master topology. The master topology is shown as violet line segments.

You can also change the **Master Topology Visibility** setting from the shortcut menu after selecting and right-clicking the layer(s) in the object selector, or right-clicking a layer in the viewer.

10.6.6.1.2. Topology Visibility

Clicking in the box will toggle the visibility of the topology. The topology is shown as yellow line segments.

You can also change the **Topology Visibility** setting from the shortcut menu after selecting and right-clicking the layer(s) in the object selector, or right-clicking a layer in the viewer.

10.6.6.1.3. Refined Mesh Visibility

Selecting **Refined Mesh Visibility** causes the refined mesh to appear in the viewer. On the first and last layers, the refined mesh shows the 3D mesh. On intermediate layers, the refined mesh shows an approximation of the 3D mesh.

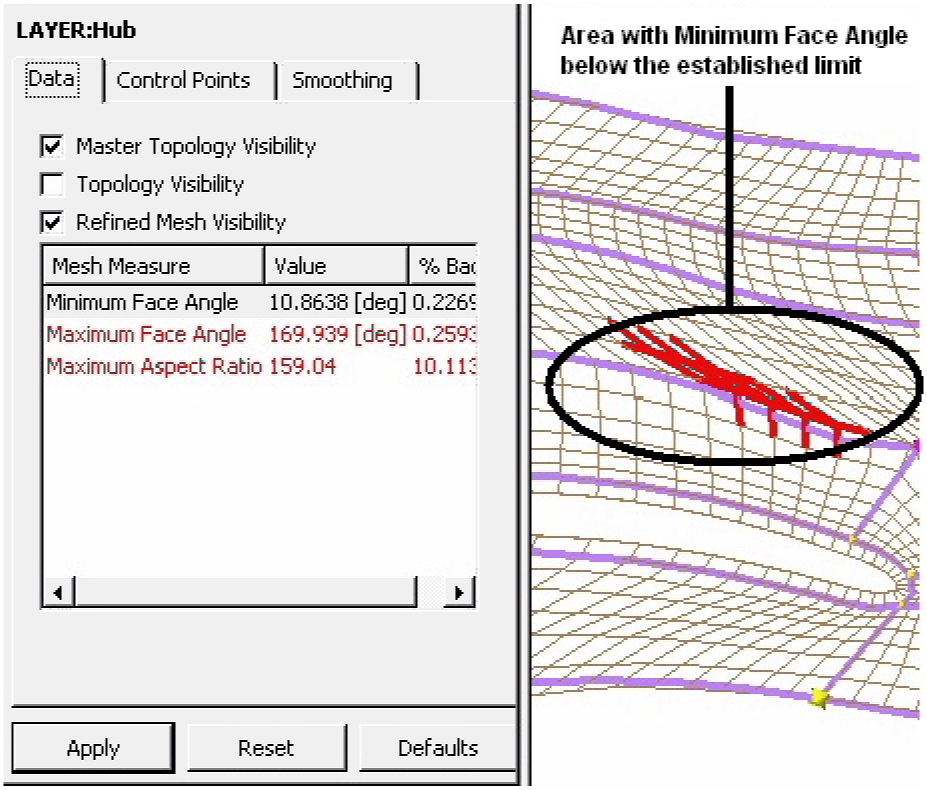
You can also change the **Refined Mesh Visibility** setting from the shortcut menu after selecting and right-clicking the layer(s) in the object selector, or right-clicking a layer in the viewer.

When **Refined Mesh Visibility** is selected, the following refined-mesh quality statistics—Mesh Measure—are available for viewing:

- `Minimum Face Angle`
- `Maximum Face Angle`
- `Maximum Aspect Ratio`

To display problem areas in the refined mesh, double-click one of the **Mesh Measure** statistics in the list. Figure 10.15: Refined Mesh Showing Areas of Unacceptable Minimum Face Angle (p. 165) shows an example of problem areas, shown with thick red lines, for the `Minimum Face Angle` statistic.

Figure 10.15: Refined Mesh Showing Areas of Unacceptable Minimum Face Angle



The `Mesh Limits` object holds the criteria that determine if the **Mesh Measure** statistics are acceptable.

If, for a given layer, **Refined Mesh Visibility** is turned on and either the `Minimum Face Angle` or `Maximum Face Angle` is outside the applicable limit, the layer will be listed in red text in the selector tree.

10.7. 3D Mesh

The 3D mesh generated by Ansys TurboGrid has objects associated with it. These objects are useful for generating and examining the mesh.

10.7.1. The 3D Mesh Object

You can open the `3D Mesh` object editor to see node and element counts for the passage, inlet, outlet, and the whole mesh.

By default, the `3D Mesh` object is initially unsuspended, so mesh generation happens automatically whenever the `Layers` object is (re)processed.

To reduce the update delay for large meshes, you can suspend the `3D Mesh` object (right-click it in the **Mesh** workspace tree and select **Suspend Object Updates**). By suspending the `3D Mesh` object, you prevent it from being updated in response to changes to other objects.

If the `3D Mesh` object is suspended, you can generate the 3D mesh in any of the following ways:

- Unsuspend the `3D Mesh` object by right-clicking it in the **Mesh** workspace tree and clearing **Suspend Object Updates**.
- Click **Insert** > **Mesh** from the main menu.
- Click the *Mesh*  icon.
- Click the **Generate** button in the `3D Mesh` object editor.

  The **Generate** button is enabled whenever it is possible to generate (or refresh) a 3D mesh.

Note:

The **Undo** command cannot properly undo a suspension change (for example, to the `Topology Set` object); attempting to undo a suspension change can lead to an inconsistent state.

10.7.2. Surface Group and Turbo Surface Objects

The `3D Mesh` branch contains objects that display the mesh on geometry surfaces and turbo surfaces. Of these display objects, only the turbo surface object `Show Mesh` is visible by default. To view another surface, select the visibility check box for that object.

The mesh can be displayed only after it has been created. Changes made to the `Geometry`, `Topology`, or `Mesh Data` objects will cause the mesh, and the surface objects found under the `3D Mesh` object, to be deleted.

10.7.2.1. 3D Mesh Turbo Surfaces

The turbo surface is a flexible object that allows you to closely examine every area of the mesh. One turbo surface exists in the `3D Mesh` branch of the object selector after you create a mesh. You can create other turbo surfaces. For details, see Turbo Surface Command (p. 59).

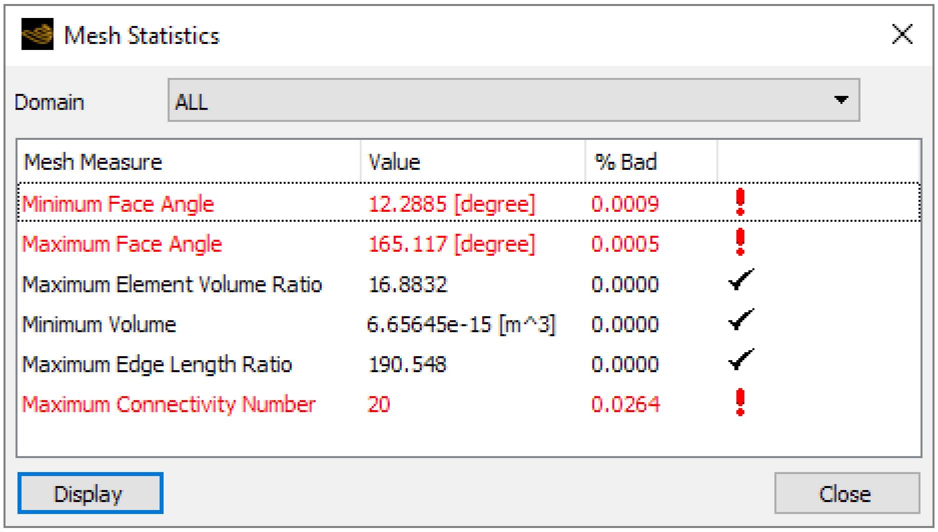
10.7.2.2. 3D Mesh Surface Groups

A collection of surface groups exists in the 3D Mesh branch of the object selector after you create a mesh. For details, see Surface Groups (p. 64).

10.8. Mesh Analysis

The `Mesh Analysis` branch offers a variety of tools to analyze the quality of the mesh. The mesh can be analyzed only after it has been created. If changes are made to the `Geometry`, `Topology`, `Layers`, or `Mesh Data` objects, the mesh cannot be analyzed again until it has been recreated. For details, see Mesh Command (p. 53).

10.8.1. Mesh Statistics



The `Mesh Statistics` object provides details about the quality of the current mesh. The `Mesh Statistics` object cannot be viewed until the mesh is created. For details, see Steps to Create a Mesh (p. 95). To view the mesh statistics, double-click the `Mesh Analysis` object or `Mesh Statistics` object in the object selector.

The `Mesh Statistics` object does not open in the object editor like the other objects in the object selector. The `Mesh Statistics` object is in a window of its own so it can stay open while other objects are edited in the object editor. This allows for constant monitoring of the statistics.

The available mesh statistics include `Minimum Face Angle`, `Maximum Face Angle`, `Element Volume Ratio`, `Edge Length Ratio` and `Connectivity Number`. Statistics displayed in red are outside the limits defined in the `Mesh Analysis` > `Mesh Limits` object. For details, see Mesh Limits (p. 167). To display the regions where the mesh statistics are outside the limits, select the mesh measure, then click **Display**. Alternatively, double-click the mesh analysis variable in the list.

Each calculated mesh measure is added to the list of available variables for creating new plots (for example, a contour plot).

10.8.2. Mesh Limits

The `Mesh Limits` object defines the acceptable values for the mesh analysis variables. These variables are described here. The default limits are generally good and should highlight any problem areas of the mesh.

10.8.2.1. Maximum Face Angle

This is the greatest face angle for all faces that touch the node. For each face, the angle between the two edges of the face that touch the node is calculated. The largest angle from all faces is returned. The maximum face angle can be considered to be a measure of skew.

10.8.2.2. Minimum Face Angle

This is the smallest face angle for all faces that touch the node.

10.8.2.3. Connectivity Number

`Connectivity Number` is the number of elements that touch a node. This variable is the maximum connectivity number on any element. For an unstructured solver, this value is not important. However, high connectivity numbers in much of the mesh can have an adverse effect on the speed of a structured solver.

10.8.2.4. Element Volume Ratio

`Element Volume Ratio` is defined as the ratio of the maximum volume of an element that touches a node, to the minimum volume of an element that touches a node. The value returned can be used as a measure of the local expansion factor.

10.8.2.5. Minimum Volume

`Minimum Volume` is used to ensure that no negative volumes are created within the passage. The value given is that of the minimum volume of an element touching any of the nodes.

10.8.2.6. Edge Length Ratio

This is a ratio of the longest edge of a face divided by the shortest edge of the face. For each face,

$$\frac{max(l_1, l_2)}{min(l_1, l_2)}$$

is calculated for the two edges of the face that touch the node. The largest ratio is returned. The edge length ratio can be considered to be a measure of aspect ratio.

10.8.3. Mesh Statistics Parameters - Order Of Importance

Generally, the mesh statistics can be ranked as follows (most important to least important)

1. `Minimum Volume` - these MUST be fixed before the mesh would be usable.
2. `Maximum Face Angle`/`Minimum Face Angle` - these should be improved until they fall within the constraints (minimum of 15° and maximum of 165°), if possible. Values close to, but just outside the constraints may still be acceptable for your simulation.
3. `Edge Length Ratio` - this can often be fixed by increasing the number of elements from hub to shroud. The default limit is 100, so values close to this will normally be acceptable.
4. `Element Volume Ratio` - depending on the mesh, it may not be possible to satisfy this constraint.
5. `Connectivity Number` - may or may not be pertinent depending on the type of solver used. For details, see Connectivity Number (p. 168).

10.8.4. Volume

The `Mesh Analysis` > `Show Limits` volume object provides a convenient way to examine the mesh. It functions exactly the same as the volume object in the create menu, except that the only variables available are the mesh statistics variables. For details, see Volume Command (p. 59).

Note:

The visibility of the volume object is off by default.

10.9. User Defined Objects

The `User Defined` branch in the object selector is initially empty. Any new objects created using the **Insert** menu will be displayed in this branch. Examples include points and legends. For details, see Insert Menu (p. 53).

10.10. Default Instance Transform

The default instance transform is initially applied to all objects for which instance transforms are possible. As a result, editing the definition of the default instance transform causes all such plots and objects to be transformed. Additional instance transform objects can be created. For details, see Instance Transform Command (p. 67). A different instance transform can be applied to an object using its **Render** tab. For details, see Applying Instance Transforms to Objects (p. 24).

10.11. Shortcut Menu Commands

You can perform many actions from shortcut menus; simply right-click one of the following:

- The viewer background (that is, a blank area in the viewer)
- An object in the viewer
- An object listed in the object selector (in the **Mesh** or **Geometry** workspace)
- An object listed in the object editor (in the **Mesh** or **Geometry** workspace)

The following sections describe all of the shortcut menu commands that can appear:

10.11.1. Assign [Family | Families | Geometry Only] To Topology Instance

Each of these commands associates the applicable CAD entity with a topological feature. For details, see Associating CAD Objects in TurboGrid's Geometry Workspace (p. 180).

10.11.2. Auto Add Layers and Insert Layers Automatically Commands

For details, see Inserting Layers Automatically (p. 162).

10.11.3. Clear Selected Entities Command

This command resets the settings for the topological entities (such as *LeadingCutoffSurface* and *LeadingEdge1*) that you have selected (using **Ctrl** or **Shift** while clicking). In particular, CAD geometry is unassigned.

10.11.4. Color Command

The **Color** command provides a subset of the functionality offered by the **Color** tab. The latter is available by selecting an eligible object from the object selector, then clicking *Rendering* . For details, see Color Tab (p. 21).

10.11.5. Create Entity and Assign Family

This command creates a topological feature for the object that is selected via the submenu. This command then associates the applicable CAD family with that new topological feature. For details, see Associating CAD Objects in TurboGrid's Geometry Workspace (p. 180).

10.11.6. Create Mesh Command

The **Create Mesh** command is available when the topology has been created. It is the same as selecting **Insert** > **Mesh**. For details, see Mesh Command (p. 53).

10.11.7. Create New View Command

Adds a new view based on the current state of the viewer.

10.11.8. Delete New View Command

Deletes an existing user-defined view.

10.11.9. Delete Secondary Passage Command

The **Delete Secondary Passage** command deletes a secondary flow path object from the `Topological Entity Instances` > `Secondary Flow Paths` object in the **Geometry** workspace.

10.11.10. Delete Selected Entity Command

This command deletes the selected entity.

10.11.11. Edit Command

The **Edit** command opens the object editor for the selected object.

10.11.12. Edit Blade Topology Command

The **Edit Blade Topology** command opens the blade settings in the object editor.

10.11.13. Edit in Command Editor Command

The **Edit in Command Editor** command opens the **Command Editor** dialog box for the selected object. For details, see Command Editor Command (p. 90).

10.11.14. Edit Render Properties Command and Edit Render Properties For All Entities Command

The **Edit Render Properties** command enables you to adjust color settings for the selected object.

The **Edit Render Properties For All Entities** command enables you to adjust color settings for all entities that belong to a blade.

10.11.15. Fit View Command

The **Fit View** command centers the view on the displayed objects and sets the zoom to an appropriate level for viewing all of the visible objects. Only the active viewport is affected by this command.

Note:

Objects that have their visibility check box cleared do not necessarily fit into the viewer. Only visible objects are considered.

10.11.16. Hide Command

The **Hide** command turns off the visibility of the selected object(s).

To turn the visibility back on for an object in the **Mesh** workspace, find the object in the object selector and select its check box, or right-click the object in the object selector and select **Show**, or use the appropriate **Show Object** command available in the shortcut menu that appears when right-clicking the viewer background.

To turn the visibility back on for an object in the **Geometry** workspace, find the object in the object selector and select its check box, or right-click the object in the object selector and select **Show**.

10.11.17. Insert Blade Command (Mesh Workspace)

The **Insert** > **Blade** command adds a blade to the blade set. After selecting this command, Ansys TurboGrid prompts you for a name for the blade, then generates the new blade object. If you already have a blade loaded, most settings for the new blade will default to those from the first blade. Simply change the **File Name** setting to indicate the file for the added blade and click **Apply**.

10.11.18. Insert Blade Command (Geometry Workspace)

The **Insert Blade** command adds a blade to the `Blade Set` group of topological entity instances. After selecting this command, Ansys TurboGrid prompts you for a name for the blade, the leading edge type, and the trailing edge type, then generates a new blade object.

10.11.19. Insert Hub Tip Line

The **Insert Hub Tip Line** command enables you to select a hub tip line from the CAD geometry. This command is offered only when the hub curve has not been defined.

10.11.20. Insert Layer After and Insert Layer After Selected Layer Commands

For details, see Inserting Layer After Selected Layer (p. 163).

10.11.21. Insert Layer Automatically Command

For details, see Inserting Layers Automatically (p. 162).

10.11.22. Insert Secondary Passage Command

The **Insert Secondary Passage** command enables you to define the geometry of a secondary flow path by specifying the constituent boundary entities. For details, see Associating CAD Objects in TurboGrid's Geometry Workspace (p. 180).

10.11.23. Insert Secondary Passage Boundary Command

The **Insert Secondary Passage Boundary** command enables you to add a boundary entity to a secondary flow path object. For details, see Associating CAD Objects in TurboGrid's Geometry Workspace (p. 180).

10.11.24. Insert Shroud Tip Line

The **Insert Shroud Tip Line** command enables you to select a shroud tip line from the CAD geometry. This command is offered only when the shroud curve has not been defined.

10.11.25. Insert USER DEFINED Object Command

The **Insert** > **USER DEFINED Object** command creates a new object. It functions like the **Insert** menu. For details, see Insert Menu (p. 53).

10.11.26. Insert Edge Split Control Command

The **Insert Edge Split Control** command is available by right-clicking (on a visible layer, in the viewer when in the **Mesh** workspace) a master topology line, a topology line, or a refined mesh line. This command creates/edits an edge split control object for the nearest master topology segment. Edge split control objects appear in the object selector under `Mesh Data`. For details, see Changing the Number of Elements on a Selected Master Topology Edge (p. 159).

10.11.27. Predefined Camera Commands

The **Predefined Camera** commands allow you to set the viewing angle according to one of the built-in presets. These commands also set the zoom level automatically after adjusting the viewing angle.

10.11.28. Save Picture Command

The **Save Picture** command is available by right-clicking the background. It is the same as selecting **File** > **Save Picture**.

10.11.29. Projection Commands

10.11.29.1. Orthographic Command

The **Orthographic** command sets an orthographic view.

10.11.29.2. Perspective Command

The **Perspective** command sets a view with a fixed amount of perspective.

10.11.30. Render Properties Edit Options Command

The **Render (Properties)** > **Edit Options** command enables you to adjust color and render settings for the selected object. For details, see Color Tab (p. 21) and Render Tab (p. 23).

10.11.31. Render Properties Show Curves Command

The **Render (Properties)** > **Show Curves** command is available by right-clicking an object (for example, the hub, shroud or blade surface). Use this command to view the raw curve file data instead of the surface.

10.11.32. Render Properties Show Surfaces Command

The **Render (Properties)** > **Show Surfaces** command is available by right-clicking the hub or shroud curve, or one of the blade curves. Use this command to view the surface instead of the raw curve file data.

Note:

The blade surface and resulting mesh can be adversely affected when a blade shroud tip is defined by the second-last profile and the last profile has a shape that is inconsistent with where the blade would exist if the blade were extended in the spanwise direction while ignoring the last profile. To avoid this problem, you can either ensure that the last profile is consistent with the second last profile, or manually remove the last profile from the curve file. A similar statement applies for the hub end of a blade having a hub tip.

10.11.33. Render Properties Topology and Refined Mesh Visibility Commands

The **Render (Properties)** > **Master Topology Visibility**, **Topology Visibility**, and **Refined Mesh Visibility** commands toggle the visibility of the corresponding items for the selected layer(s).

10.11.34. Show Object and Show Commands

The **Show Object** commands are available by right-clicking the background in the viewer when in the **Mesh** workspace. These commands turn on the visibility of any geometry, volume, or mesh data object.

The **Show** command turns on the visibility of the object(s) selected in the object selector.

As an alternative, you can select the check box next to an item in the object selector to show that item in the viewer.

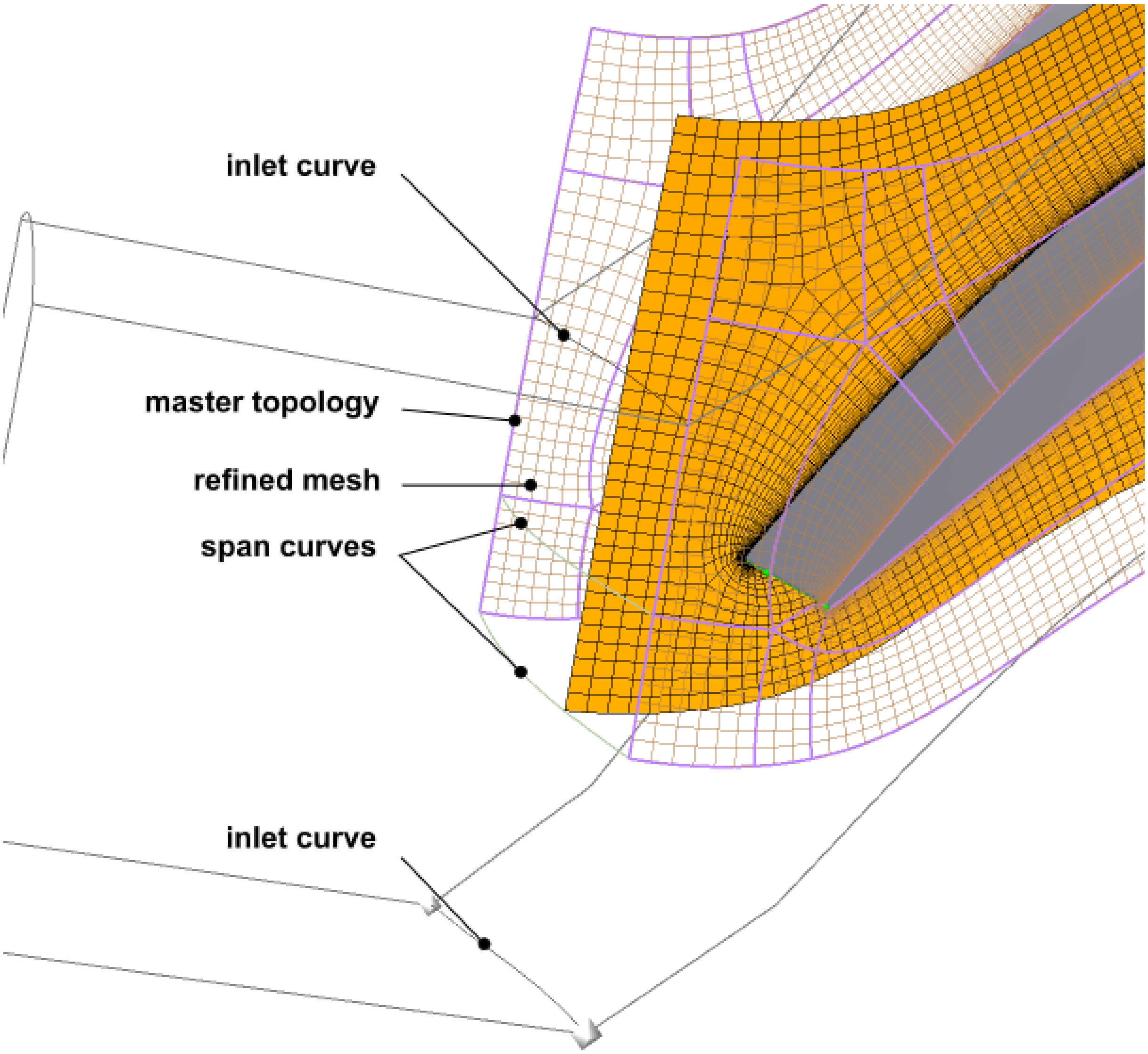
10.11.35. Show and Hide All Siblings Command

The **Show + Hide All Siblings** command turns on the visibility of the object selected in the object selector, and turns off the visibility of every sibling object in the tree.

10.11.36. Span Curve Visibility Commands

The **Span Curve Visibility** commands are available by right-clicking (on a visible layer, in the viewer when in the **Mesh** workspace) a master topology line, a topology line, or a refined mesh line. These commands control the visibility of the span curves. A span curve is a line that passes through corresponding master topology nodes across layers. Span curves are used to construct the 3D mesh. A span curve that is sharply curved can indicate a potential problem with mesh quality.

Figure 10.16: Span Curves



The **Toggle span curve visibility** command toggles the visibility of the span curve that passes through the nearest node of the master topology.

The **Turn on all span curves** and **Turn off all span curves** commands control the visibility of all span curves at once.

10.11.37. Suspend Object Updates Command

You can save time by appropriately suspending objects such as the `Topology Set` and `3D Mesh` objects. The **Suspend Object Updates** command toggles the state of suspension of an object. By suspending an object, you prevent it (and objects that depend on it) from being updated in response to changes to other objects.

For example, the `Topology Set` object depends on the `Geometry` object, and, when not suspended, will be reprocessed each time the geometry is changed. If you want to make several changes to the geometry, you can skip the intermediate processing of the topology by suspending the `Topology Set` object. Note that the `Topology Set` object is initially suspended when you start a new case.

To process a suspended object, unsuspend it.

An icon overlay indicates whether or not an object is suspended. Icon overlays are illustrated in Table 1.1: Icon Overlays and Text Styles (p. 18).

Note:

The **Undo** command cannot properly undo a suspension change (for example, to the `Topology Set` object); attempting to undo a suspension change can lead to an inconsistent state.

10.11.38. Toggle Axis Visibility Command

The **Toggle Axis Visibility** command is available by right-clicking the background. It controls the visibility of the axis orientation indicator that appears in the lower-right corner of the viewer.

10.11.39. Toggle Ruler Visibility Command

The **Toggle Ruler Visibility** command is available by right-clicking the background. It controls the visibility of the ruler that appears near the lower edge of the viewer.

10.11.40. Transformation Commands and Coordinate Systems

The **Transformation** commands are available by right-clicking the background.

You can use different coordinate systems to display various views of the geometry and mesh objects in the viewer window. This is useful when closely examining every aspect of the mesh before saving it for use in a CFD solver.

10.11.40.1. Cartesian

The Cartesian coordinate system is used by default in the viewer window when there is only one viewport. The 3 axis coordinates for the Cartesian coordinate system are X, Y and Z.

10.11.40.2. Blade-to-Blade

The blade-to-blade coordinate system is used to display the geometry in the m' - θ conformal space, which is familiar to blade designers. The m' - θ coordinate space is angle-preserving and minimizes the effect of changing radius on viewing and manipulation. By utilizing the blade-to-blade coordinate system, a wide variety of machine types, from axial to radial, can be treated similarly.

m' is defined in Meridional (p. 178).

Note:

Due to the fact that m' is ill-conditioned at $r=0$, you can expect to see different behavior in cases for which the geometry extends to the machine axis.

10.11.40.3. Meridional

The meridional coordinate system is one transformation used by blade designers. It is useful for viewing the flow path as well as the upstream and downstream extents of the mesh. The three axis coordinates for the meridional coordinate system are A (axial), R (radial) and θ (Theta). The viewer shows only the A and R coordinates.

Variable	Description
a	Axial location
r	Radius or radial location
s	Normalized distance along meridional curve (for example, from 0 to 1)
S	The particular value of s that corresponds to the point location for which m and m' are to be computed
m	Meridional coordinate (distance along meridional curve)
m'	Normalized meridional coordinate (radius normalized distance along meridional curve)

$$m=\int_0^S \sqrt{\left(\frac{dr}{ds}\right)^2+\left(\frac{da}{ds}\right)^2}\ ds$$

$$m'=\int_0^S \left[\frac{\sqrt{\left(\frac{dr}{ds}\right)^2+\left(\frac{da}{ds}\right)^2}}{r}\right] ds$$

10.11.40.4. 3D Turbo

The three axis coordinates for the 3D Turbo Coordinate System are M (m'), T (Theta) and S (span).

10.11.41. Unassign Selected Geometries Command

Unassigns the applicable geometries.

10.11.42. Viewer Options Command

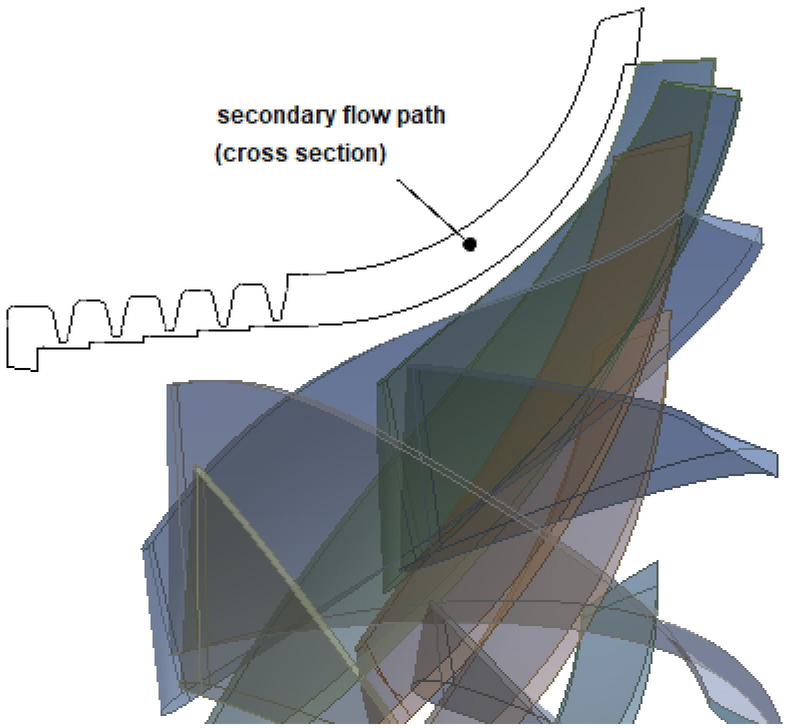
The **Viewer Options** command is available by right-clicking the background. It is the same as selecting **Edit** > **Options** and then navigating to **TurboGrid** > **Viewer**. For details, see Viewer (p. 44).

Chapter 11: Secondary Flow Paths

Some turbomachinery designs have secondary flow paths that are outside the main passage.

TurboGrid enables you to add one or more secondary flow paths to a case.

Figure 11.1: Secondary Flow Path for a Compressor



Each secondary flow path is axisymmetric, located outside the main flow passage, and has one or more interfaces with either the hub or shroud.

The following topics are discussed:

11.1. Creating Geometry for Input to TurboGrid

The shape of a secondary flow path is mainly defined by a curvilinear closed loop on an axial-radial plane. The loop is made up of several segments, known here as boundary entities, that are connected end-to-end.

The following topics are discussed:

11.1.1. Source of Geometry

In order to be able to create secondary flow paths in TurboGrid, the geometry must be defined from a CAD source. TurboGrid has no support for creating secondary flow paths for geometry that is defined from profile points. For details on defining geometry in TurboGrid using these different sources of geometric data, see Geometry (p. 97).

Note that BladeEditor can serve as a source of CAD geometry for TurboGrid. In particular, native BladeEditor blades (blades that were produced by the Blade or Splitter feature, without subsequent modification to the blade or splitter) with added 2D sketches of the secondary flow path(s) can be sent to TurboGrid as CAD data by making a cell connection in Workbench from the Geometry cell (that represents BladeEditor) to the Turbo Mesh cell (that represents TurboGrid).

To prepare the geometry in BladeEditor:

1. Create line bodies for the sketches: at least one line body for each boundary entity that will be made in TurboGrid.
2. Create Attributes (**Tools** > **Attribute**): as many as one for each line body; at least one for each boundary entity that will be made in TurboGrid.

   The `Attribute Name` property values (that is, the attribute text names) must be prefixed with "`NamedSelection:`". The remainder of the name is what will appear in downstream applications. Note: Unicode is not supported for names in TurboGrid. Set the attribute `Geometry` property by selecting edges from the defined wire bodies.

11.1.2. Interfaces with the Main Passage

Any given secondary flow path is permitted to have one or more interfaces with the main passage's hub and shroud boundaries.

At each interface, the curvilinear closed loop that defines the secondary flow path must be near or cross either the hub or shroud. The curvilinear closed loop is adjusted as needed so that it follows the hub or shroud along the interface.

11.2. Specifying Secondary Flow Path Geometry in TurboGrid's Geometry Workspace

Your input is required in order for TurboGrid to be able to recognize, within a given loaded CAD geometry, each secondary flow path and its various boundary entities (walls and hub or shroud interfaces).

You can provide this input through TurboGrid's **Geometry** workspace.

The following topics are discussed:

11.2.1. Associating CAD Objects in TurboGrid's Geometry Workspace

This section describes how to set up secondary flow path geometry in TurboGrid using the **Geometry** workspace.

With the CAD geometry already loaded into TurboGrid, you can specify the geometry for each secondary flow path as follows:

1. In the **Geometry** workspace tree, right-click `Topological Entity Instances` > `Secondary Flow Paths` and select **Insert Secondary Passage**.

   The **Create New Secondary Passage Topology** dialog box appears.

2. Enter a name for the secondary flow path and click **OK**.

   A new, incomplete, secondary flow path object appears in the tree under `Secondary Flow Paths`. This object requires boundary entity definitions.

3. Define boundary entities to complete the secondary flow path object using one or both of the following procedures:

   Create a boundary entity and then assign it one or more CAD entities

   1. Right-click the new secondary flow path object and select **Insert Secondary Passage Boundary**.

      The **Create New Secondary Passage Boundary** dialog box appears.

   2. Enter a name for a new boundary entity for the secondary flow path object and click **OK**.

      A new, incomplete, boundary entity appears in the tree under the secondary flow path object that is being defined. This boundary entity requires an association with a curvilinear line selected from the `CAD Families` branch.

   3. Right-click one CAD entity, or a selection of CAD entities, and select **Assign [Family | Families | Geometry Only] To Topology Instance** > **[Name of the secondary flow path object that has an incomplete boundary entity]** > **[Name of the incomplete boundary entity that requires an association with the CAD entity/entities]**.

   Create a boundary entity directly from a CAD entity

   1. Right-click a CAD family (a high-level item listed directly under `CAD Families`) and select **Create Entity and Assign Family** > **[Name of the secondary flow path object that requires a new boundary entity]**.

      The **Create New Secondary Passage Boundary** dialog box appears.

   2. Enter a name for a new boundary entity for the secondary flow path object that is being defined and click **OK**.

A new, complete, boundary entity is created and appears in the tree under the secondary flow path object that you chose in the previous step. This boundary entity is associated with the CAD family that you right-clicked in the previous step.

Note:

If you assign a CAD family to a boundary entity, the entire family is associated with the boundary entity. Any upstream geometry updates to the family will be propagated to the assigned topological entity. If you right-click on a CAD sub-entity (curve or surface) and assign it to a boundary entity, the family association will be lost and only the specified sub-entities will be maintained during upstream geometry updates.

This also applies for all other TurboGrid CAD topological entities.

11.3. Specifying Secondary Flow Path Details in TurboGrid's Mesh Workspace

With the geometry loaded and with the secondary flow paths recognized by TurboGrid (as reflected in the **Geometry** workspace), you can use the **Mesh** workspace to specify meshing details of the secondary flow paths (as well as the main passage).

In the **Mesh** workspace tree, the secondary flow path objects are listed under three locations, as detailed in the following sections:

11.3.1. Geometry Branch

11.3.2. Mesh Data Branch

11.3.3. 3D Mesh Branch

11.3.1. Geometry Branch

The secondary flow path objects listed under `Geometry` > `Secondary Flow Paths` provide the following functions:

- You can toggle the visibility of the geometry of a secondary flow path, or any of its boundaries, by toggling the corresponding visibility check box.
- You can edit any boundary of a secondary flow path in order to change its type via the **Boundary Type** setting. The available types are: `Wall`, `Hub Interface`, `Shroud Interface`. The type specification affects meshing details near the boundary.

  A type setting of `Hub Interface` or `Shroud Interface` makes the **Automatically Manage Opening Region** setting available. Selecting the latter causes the hub or shroud (as applicable) to be divided into regions; in particular, one of the regions is made to match (line up with) the interface. The resulting passage mesh is therefore able to have nodes that are aligned with the interface edges, facilitating mesh refinement within the passage mesh near the interface edges. The regions that are created in this process can be reviewed in the object editor for `Geometry` > `Hub` on the **Hub Regions** tab, or the corresponding tab for the shroud, as applicable.

Note that, when using the **Automatically Manage Opening Region** option, the interface must be fully within the inlet or outlet block.

11.3.2. Mesh Data Branch

The secondary flow path objects listed under `Mesh Data` provide the following functions:

- The object editor for a secondary flow path object provides controls that affect the mesh within the bulk of the secondary passage.
- The object editor for a given boundary of a secondary flow path object provides similar settings that affect the mesh near the boundary. These settings enable you to override, for the applicable boundary, the default settings that are established for the secondary flow path (parent object).

11.3.2.1. Setting Descriptions for the Secondary Flow Path Object

- **Triangle Refinement Factor**

  As this value increases, the element size in the mesh interior decreases.

- **Default Boundary Edge Refinement Factor**

  As this value increases, the node spacing along the boundary decreases.

- **Inflation Options**
  - **Default Number of Boundary Layers**

    The number of rows of hexahedral elements in the inflation layer.

  - **Default Inflation Growth Ratio**

    Height ratio between a row of mesh elements in the inflation layer and the adjacent row closer to the boundary.

- **Advanced Options**
  - **Default Last Layer Height Ratio**

    This is the length scale ratio between the boundary layer mesh elements and the interior mesh elements where the boundary layer elements meet the interior elements.

  - **Default Maximum Feature Angle** and **Default Curvature Angle**

    Decreasing the **Maximum Feature Angle** reduces the "detection" of boundary curve details.

    Increasing the **Curvature Angle** reduces the number of elements along a curved section of the boundary.

    Note that the **Maximum Feature Angle** and **Curvature Angle** values must sum to a value greater than 180 degrees.

  - **Default Cells Per Gap (Proximity Refinement)**

This parameter controls the mesh proximity. Increasing this value increases boundary and interior mesh refinement in narrow gaps.

- **3D Options** (available only for the secondary flow path object, not a boundary object)

  A secondary flow path is typically modeled as a sector of the full 360° geometry. The appropriate pitch (angular extent about the rotation axis) of the sector, and number of elements across the pitch, depend on details of the intended CFD simulation:

  – If you intend to connect the main passage to the secondary flow path using a mixing plane interface (to model swirling flow with no circumferential variation) then the secondary flow path can be modeled using a thin, almost 2D, mesh that has a minimum of 2 layers of mesh elements across the pitch.

  – If you want to model circumferential flow variations in the secondary flow path (for example using a `Frozen Rotor` or `Transient Rotor-Stator` frame change model in CFX) then the secondary flow path should have an appropriate pitch.

  The settings under **3D Options** are:

  – **Number of Extrusions**

    Specify the number of rows of elements to generate (minimum 2) in the circumferential direction.

  – **Theta Extrusion Angle**

    Specify the sector angle (pitch) for the secondary flow path mesh.

  – **Theta Extrusion Base**

    Specify the starting value of Theta for the mesh extrusion.

11.3.2.2. Setting Descriptions for the Secondary Flow Path's Boundary Object

- **Inflation Zone**

  This setting controls whether or not inflation is applied at the specified boundary. A value of `Default` applies inflation if the boundary is a wall; it applies no inflation if the boundary is an interface.

- **First Layer Height**

  A read-only display of the average height of the row of mesh elements next to the boundary.

  Note:

  There is no setting to directly control the first element height in an inflation region. However, you can see the average height in the object editor for a boundary. To increase/decrease the height, decrease/increase the number of boundary layers. Other mesh factors affect the height, including the **Global Size Factor** (see Method (p. 145)).

- **Use Override Boundary Settings (available only for a boundary object, not the secondary flow path object)**

  Select this option to override defaults set for the secondary flow path object:

- **Freeze Edges**

  Try changing this option if there is a mesh quality problem at a transition between a boundary with an inflation layer and a boundary without an inflation layer.

11.3.3. 3D Mesh Branch

The secondary flow path objects listed under `3D Mesh` provide the following functions:

- You can toggle the visibility of the mesh associated with a secondary flow path, or any of its boundaries, by toggling the corresponding visibility check box.
- You can edit any boundary of a secondary flow path in order to change its color and/or rendering settings.

For details, see Surface Groups (p. 64).