STL Import 2 — Remeshing an Imported Mesh[1]

1. *The STL geometry is provided courtesy of Mark Yeoman, Continuum Blue, UK.*

Introduction

The STL file format is one of the standard file formats for 3D printing, and it is also often used as a format for exchanging 3D scan data. STL files contain only the triangulated surface, which we can also call a surface mesh, of a 3D object. The triangles in the file are identified by their normals and vertex coordinates which together form a faceted representation of the object.

COMSOL Multiphysics supports import of an STL file both as a surface mesh and as a geometry with smooth faces. This tutorial series focuses on using available tools to edit imported surface meshes, the different ways of repairing the meshes, and how to generate a volume mesh from the imported surface mesh, either directly or by first creating a geometry with smooth faces from the mesh. Regardless of which method you choose to follow, COMSOL Multiphysics supports a variety of operations, for example:

- Moving, scaling, and rotating the imported mesh
- Combining the imported mesh with parameterized geometry to run parametric sweeps
- Intersecting imported meshes with each other
- Modifying and remeshing the imported mesh
- Generating a tetrahedral mesh in unmeshed domains
- Generating a swept mesh in unmeshed domains
- Creating a boundary layer mesh
- Using curved mesh to represent curved boundaries

Working with the mesh directly can be more robust in case you need to intersect several imported meshes (or intersect the imported mesh with geometric objects of more complex shapes). This can be important to consider when choosing whether to create a geometry or not.

This tutorial, the second in the series, demonstrates a workflow where a simulation mesh is generated by remeshing the imported surface mesh without creating a geometry. The steps from importing the STL file to creating the final mesh are described in detail and include repairing of the imported mesh, combining the imported mesh with a mesh generated to represent a surrounding volume, intersecting the mesh with a plane, remeshing, creating domains, generating the tetrahedral mesh, and finally visualizing the mesh elements using a mesh plot.

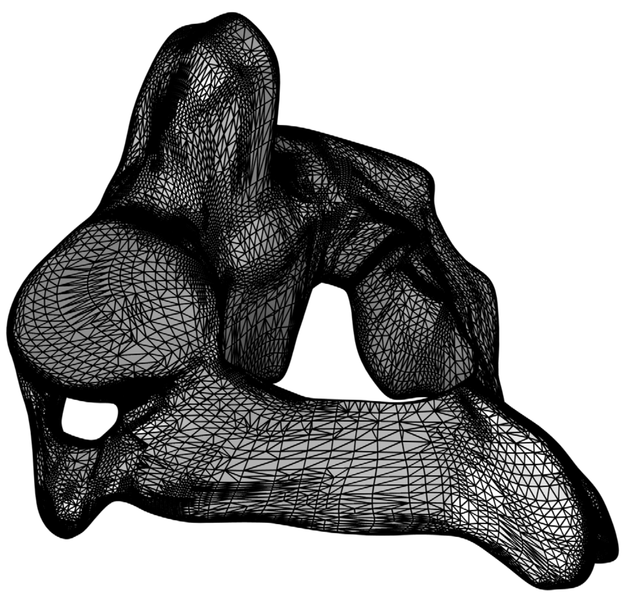
STL Import 1 — Generating a Geometry from an Imported Mesh, the first part of this tutorial series, describes the process of creating a geometry from an imported STL file.

The two tutorials in this series are complementary, and intend to provide a detailed insight into how to work with imported meshes. Apart from arriving at a simulation mesh in two different ways, the tutorials also cover repairing different types of defects, and different ways of visualizing the mesh. Depending on your application and the imported mesh at hand, pick and choose from the tools detailed in the tutorials to arrive at a mesh that suits your needs.

Lastly, it is important to mention that the techniques used in the tutorial series apply to any type of imported surface meshes, such as the formats PLY and 3MF. They also apply when creating a mesh from a Filter or Partition dataset, which you would do when using the results of a simulation as the mesh for a new simulation, for example during a topology optimization study.

Model Definition

Import the STL file of a vertebra geometry shown below.



Follow the instructions in this tutorial to

- Import the STL file
- Rotate the STL mesh and center it around the origin
- Identify and fix small defects in the imported STL mesh

- Import a second mesh for the surrounding volume to be used for simulation
- Intersect and connect the two meshes with planar faces
- Generate a volume mesh for the created domains
- Create a mesh plot to have a look inside the generated mesh

---

Application Library path: `COMSOL_Multiphysics/Meshing_Tutorials/stl_vertebra_mesh_import`

Modeling Instructions

From the **File** menu, choose **New**.

NEW

In the **New** window, click **Model Wizard**.

MODEL WIZARD

1 In the **Model Wizard** window, click **3D**.

2 Click **Done**.

GEOMETRY 1

1 In the **Model Builder** window, under **Component 1 (comp1)** click **Geometry 1**.

2 In the **Settings** window for **Geometry**, locate the **Units** section.

3 From the **Length unit** list, choose **mm**.

This will set the same length unit also for the mesh.

MESH 1

Import 1

1 In the **Mesh** toolbar, click **Import**.

2 In the **Settings** window for **Import**, locate the **Import** section.

3 Click **Browse**.

4 Browse to the model's Application Libraries folder and double-click the file `c2_vertebra.stl`.

Notice that the **Create domains** check box is selected. Domains are needed before you can generate a tetrahedral mesh. The import checks if the imported surface mesh forms any watertight regions, and if so, it forms domains.

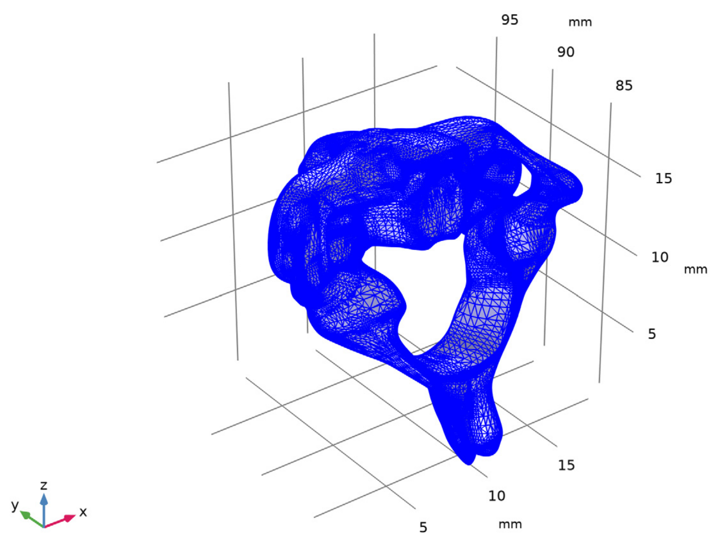
5 From the **Boundary partitioning** list, choose **Minimal**.

This setting is suitable when importing mesh files that have no obvious boundary partitioning, for example, meshes generated by medical imaging techniques. After the import, the mesh will usually consist of only one boundary that you can partition as needed, using the available tools. Use the **Automatic** or **Detect boundaries** settings when importing meshes that contain planar faces and fillets, which can then be detected to partition the mesh accordingly.

6 Click **Import**.

Information

An information message appears stating that no domains were created for the face of the vertebra.



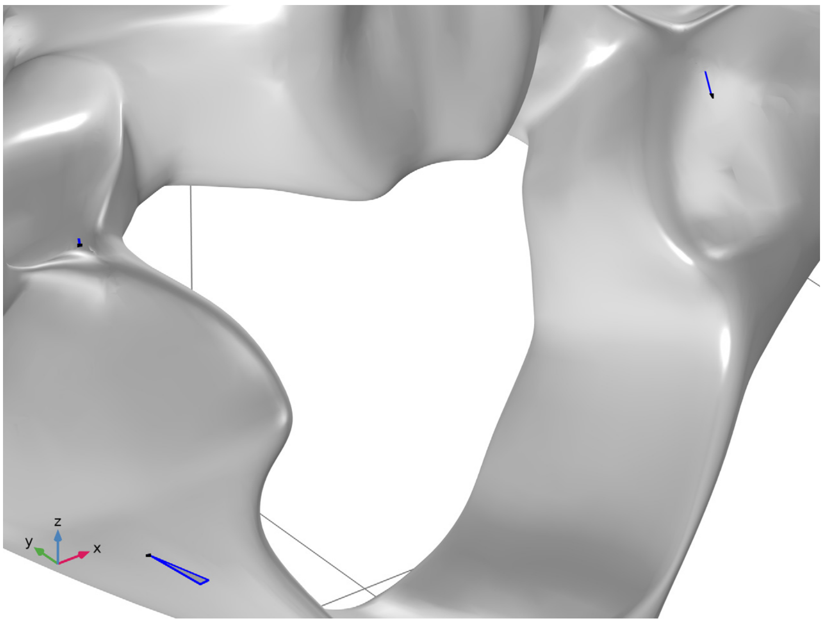
1 In the **Model Builder** window, click **Information**.

2 Expand the **Information** node and click on the **Information** subnode for more details.

Information

This information message indicates that the mesh of the vertebra does not form a watertight component. The edges that appear in the selection for the **Information** node are exterior edges that are adjacent to one or several holes in the mesh. Disable the rendering of the mesh to see them more clearly in the **Graphics** window.

1 Click the **Mesh Rendering** button in the **Graphics** toolbar. Zoom in to see the exact view below.



There are three holes, all located on the same side of the vertebra. You can verify this by going through the list of edges or by turning on **Wireframe Rendering** to make sure there are no other edges hiding behind the faces of the vertebra. If you do, remember to turn off **Wireframe Rendering** before continuing.

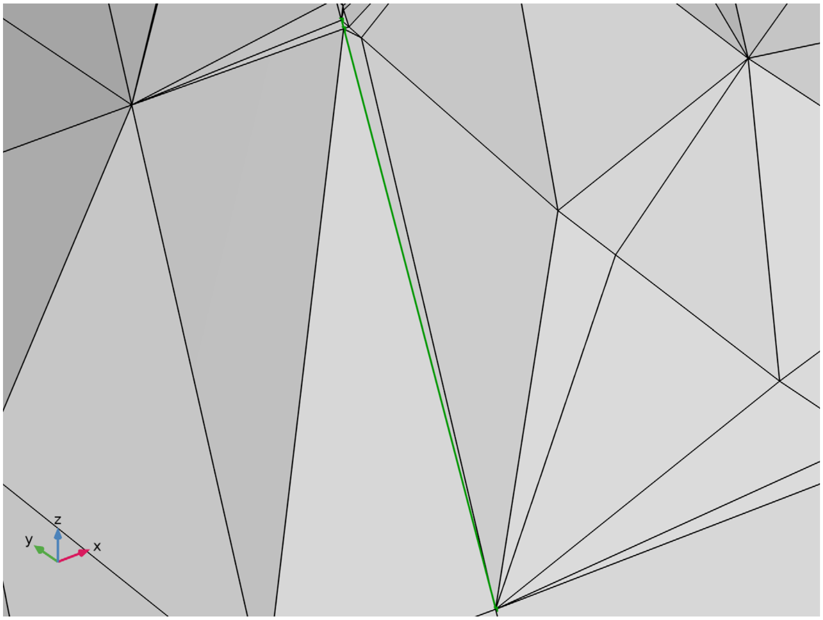
The larger hole at the lower-left is a triangular hole that seems to be caused by a missing mesh element. The two additional holes located in the upper half of the image are both slit-like holes with zero or almost zero areas, as will be clear later when zooming in on the individual holes. Such slits usually result when the vertices of the imported mesh triangles do not match within the specified import tolerance.

Use two different techniques to repair the holes as they require different kinds of treatment. Of the three holes in the image above, first repair the one in the upper right corner of the image. Then, continue with the small hole in the upper-left corner and, lastly, fill in the triangular hole in the lower-left corner.

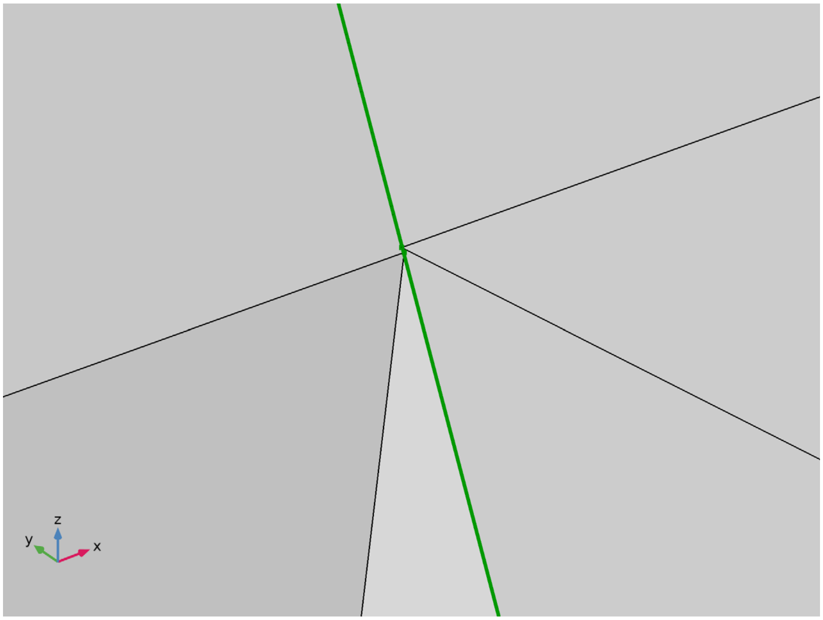
If more than three edges are listed in the **Selection List**, go back to the **Import 1** node and make sure that the **Boundary partitioning** setting is **Minimal**, then reimport the mesh.

2 In the **Model Builder** window, expand the **Information** node, then click **Information**.
3 In the **Settings** window for **Information**, locate the **Geometric Entity Selection** section.
4 In the list, select **3** (the edge located in the upper right corner of the previous image).
5 Click the **Zoom to Selection** button in the **Graphics** toolbar.

6 Click the **Mesh Rendering** button in the **Graphics** toolbar.



The selected edge is next to a very narrow slit that has the same number of edge elements on both sides. Zoom in further on the upper center of the image above, a bit down from the end of the slit where the shorter edge elements are located.



The corners of the triangular elements do not match since the corresponding mesh vertices were not merged during the import. To fix this, reimport the file using a larger tolerance.

Import 1

1 In the **Model Builder** window, under **Component 1 (comp1)>Mesh 1** click **Import 1**.

2 In the **Settings** window for **Import**, locate the **Import** section.

3 From the **Repair tolerance** list, choose **Absolute**.

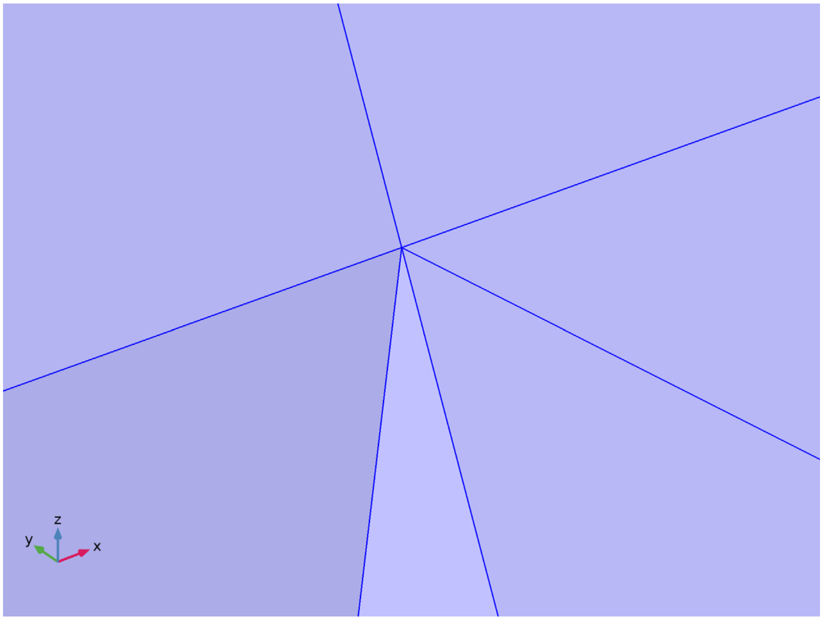
4 In the **Absolute tolerance** text field, type `1e-4[mm]`.

Also, under the **Boundary Selections** section, notice that a selection of the surfaces of the vertebra has been automatically created during the import. We will use this selection later to easily select the surface of the vertebra in operations and size attributes. Rename it to something more descriptive.

5 Locate the **Boundary Selections** section. In the table, enter the following settings:

Name	**SOLID section in file**
`Vertebra boundary`	COMSOL mesh mesh1

6 Locate the **Import** section. Click **Import**.



The mesh vertices are now merged, thereby eliminating the hole and the edge adjacent to it. You can easily verify this by inspecting the number of edges listed in the **Information** node or by turning off the mesh rendering temporarily.

7 Click the **Zoom Extents** button in the **Graphics** toolbar.

Transform 1

Before repairing the rest of the holes, move the mesh to the origin. You can easily move, rotate, and scale an imported mesh by adding a **Transform** subnode to an **Import** node. Add several **Transform** subnodes to rotate an imported mesh around more than one axis.

1 In the **Mesh** toolbar, click **More Attributes** and choose **Transform**.

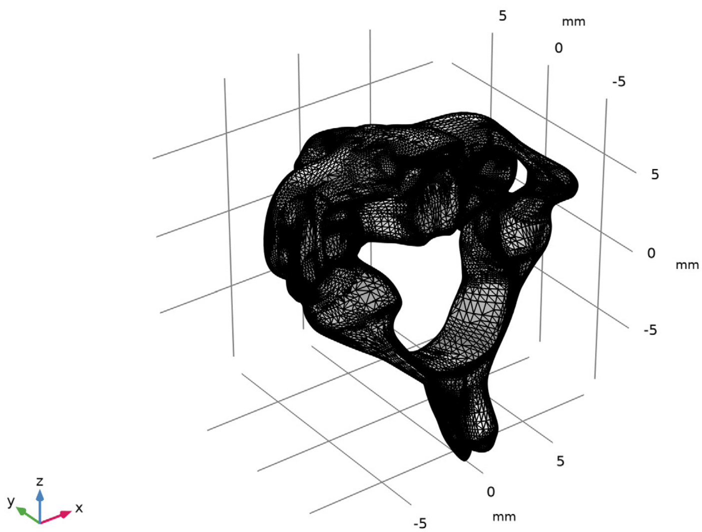
2 In the **Settings** window for **Transform**, locate the **Displacement** section.

3 In the **x** text field, type `-10[mm]`.

4 In the **y** text field, type `-90[mm]`.

5 In the **z** text field, type `-10[mm]`.

6 Click **Build Selected**.



Information

1 In the **Model Builder** window, expand the **Component 1 (comp1)>Mesh 1>Import 1>Information** node, then click **Information**.

2 In the **Settings** window for **Information**, locate the **Geometric Entity Selection** section.

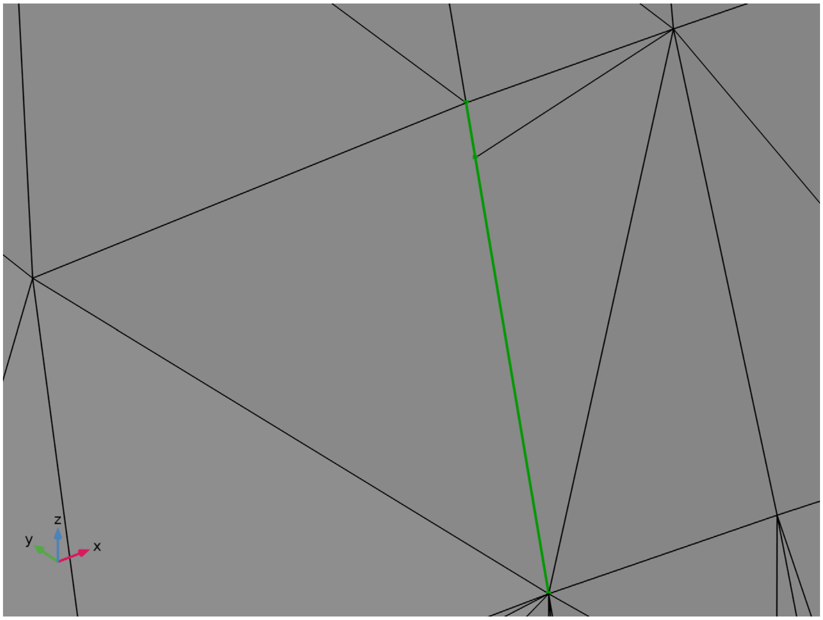
3 Click **Zoom to Selection** to zoom in on the two remaining holes.

Next, repair the other slit in the mesh, which is the smaller of the two remaining holes.

4 In the list, select **2**.

5 Click **Zoom to Selection**.

6 Zoom out a bit to see the triangle elements on both sides of the slit, as shown below.



On one side of the slit (on the left side of the green edge in the figure), there is one triangle element, whereas on the other side there are two elements. Since the import functionality cannot partition elements or add new ones, the mesh edges on the two sides of the slit could not be merged even with the larger tolerance.

Fill Holes 1

To repair the slit, we will use the **Fill Holes** operation, which can introduce new mesh edges in order to merge edges. The **Fill Holes** operation searches for holes within a selection of boundaries and will automatically fill all holes smaller than a specified tolerance. We can therefore use this operation to also repair the third hole.

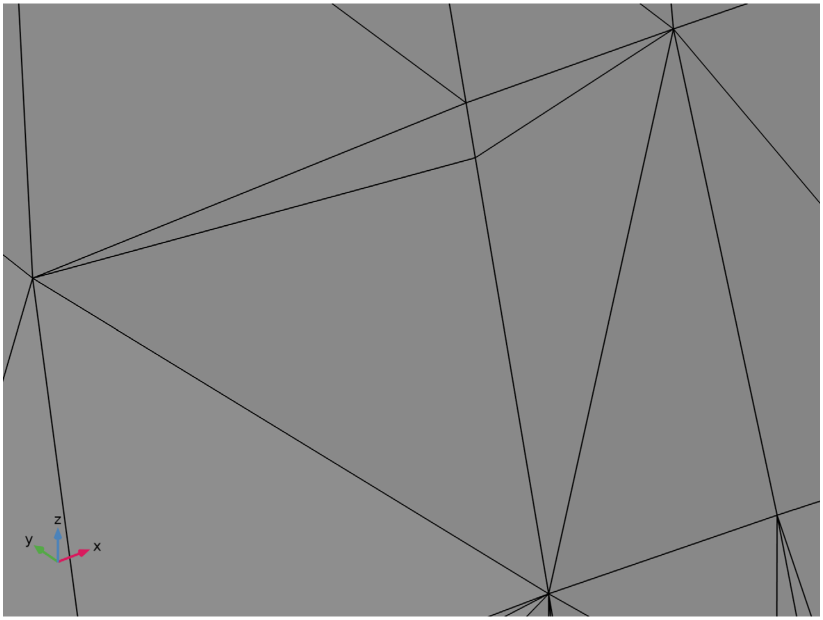
1 In the **Mesh** toolbar, click **Cleanup and Repair** and choose **Fill Holes**.

2 In the **Settings** window for **Fill Holes**, locate the **Boundary Selection** section.

3 From the **Selection** list, choose **Vertebra boundary** (the boundary of the vertebra).

Notice that the **Create Domains** check box is automatically selected, which means that once all holes are filled and the surface mesh forms a watertight boundary, a domain will be created.

4 Click **Build Selected**.



The slit has been repaired by introducing a mesh edge on the left side of the slit, merging the two sides of the hole, and removing the edge.

Information

Another information message indicates that there is still a hole in the surface mesh as no domain could be created. Zoom in on the hole by following the steps below.

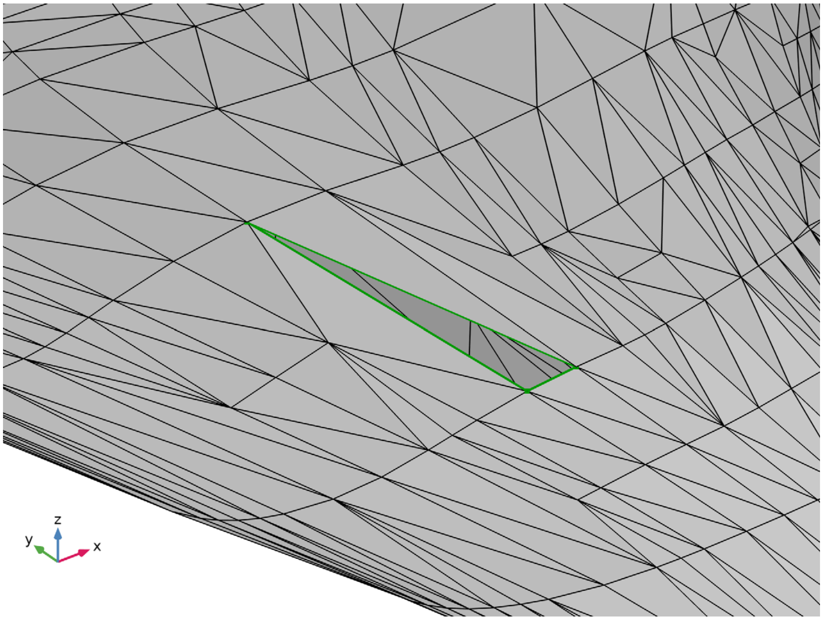
Information

1 In the **Model Builder** window, expand the **Component 1 (comp1)>Mesh 1>Fill Holes 1> Information** node, then click **Information**.

2 In the **Settings** window for **Information**, locate the **Geometric Entity Selection** section.

3 In the list, select **1**.

4 Click **Zoom to Selection**.



The original triangular hole remains since it is larger than the automatically determined tolerance for **Fill Holes**. Measure the perimeter of the remaining hole to obtain an estimate for the tolerance value to use with the **Fill Holes** feature. The edge should already be selected, but select it again if needed.

5 In the **Mesh** toolbar, click **Measure**.

The length of the edge is reported in the **Messages** window. It is about 2.2 mm long.

Next, go back to the **Settings** window for **Fill Holes**.

Fill Holes 1

1 In the **Model Builder** window, under **Component 1 (comp1)>Mesh 1** click **Fill Holes 1**.

2 In the **Settings** window for **Fill Holes**, locate the **Fill Holes** section.

3 From the **Maximum hole perimeter** list, choose **Manual**.

4 In the **Perimeter** text field, type `2.3[mm]`. The value you enter here should be slightly larger than the measured hole perimeter.

5 Click **Build Selected**.

This concludes the repair of the imported mesh and the **Messages** window reports that there is now one domain. Continue with the steps below to learn how to combine the imported mesh with a block created in the same component, intersect the mesh with a plane to delete a portion of the mesh, and generate a volume mesh both inside and outside the vertebra. The mesh of the block could just as well be imported from file, if available.

GEOMETRY 1

Block 1 (blk1)

1 In the **Geometry** toolbar, click **Block**.

2 In the **Combine Geometry with Mesh** dialog box, , you get the question if you want to combine this geometry with the mesh in **Mesh 1**. If you don't want to do this or don't want to decide at this point, you can just click **Cancel** and decide what to do with the geometry at a later stage. In this tutorial, we want to combine the geometry of the block with the mesh of the vertebra, so let us look at the options of how to do this.

   - Add an **Import** node to **Mesh 1**. This will import a surface mesh of the geometry to the meshing sequence, where the two will be combined. This is what we intend to do in this tutorial and the option we will use.
   - Add a new meshing sequence which is conforming to the geometry and an **Import** node in **Mesh 1** where the mesh of the geometry is imported. Use this option if you want full control over the mesh you import into **Mesh 1**.

3 Click **OK** to confirm the first option.

   Look to verify that an **Import 2** node has been added last in the **Mesh 1** sequence. However, before going there, finish building the geometry by choosing the size and position of the block such that it will contain the vertebra mesh.

4 In the **Settings** window for **Block**, locate the **Size and Shape** section.

5 In the **Width** text field, type `30[mm]`.

6 In the **Depth** text field, type `20[mm]`.

7 In the **Height** text field, type `25[mm]`.

8 Locate the **Position** section. From the **Base** list, choose **Center**.

9 Click **Build Selected**.

10 Click the **Zoom Extents** button in the **Graphics** toolbar.

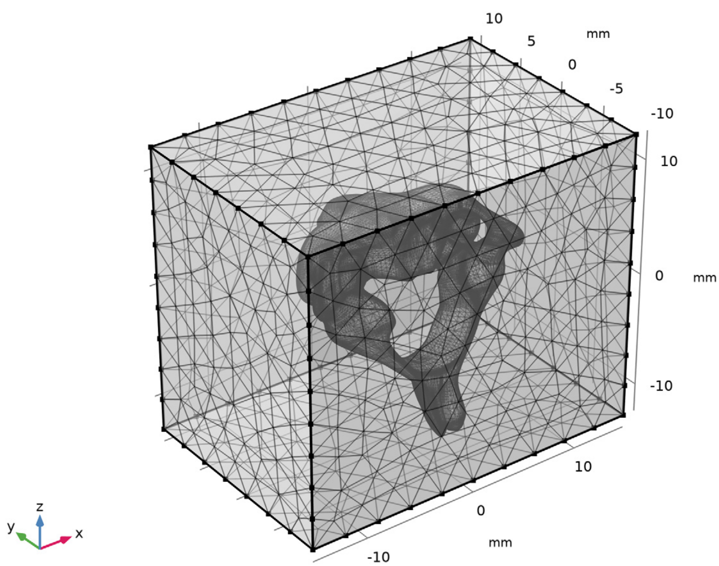
MESH 1

Now, go back to **Mesh 1** and import a surface mesh of the built block into the meshing sequence.

Import 2

The settings are already set up to import a surface mesh of the finalized geometry in **Geometry 1**. The mesh will be built by the **Free Triangular** operation of, in this case, **Normal** size, but there is also the option to import a visualization mesh of the geometry. The option **Resolve narrow domain regions** will decrease the mesh size where the boundaries are close to a narrow region, similar to what the boundary mesh would look like if you generate a domain mesh. For the block, there are no narrow regions, so you will just get a triangle mesh with similar size on all boundaries.

1 In the **Model Builder** window, under **Component 1 (comp1)>Mesh 1** right-click **Import 2** and choose **Build Selected** to import a **Normal** sized surface mesh of the block.
2 Click the **Transparency** button in the **Graphics** toolbar, to see the vertebra inside the block.



3 Rotate the mesh in the **Graphics** window to check that the vertebra is fully contained in the block, the two meshes should not intersect. Click the **Go to Default view** button.

   You can also use mesh **Statistics** to check that only triangle elements were imported for the block.

4 In the **Model Builder** window, right-click **Mesh 1** and choose **Statistics**.

The **Statistics** state that this is a mesh with unmeshed domains, and it only lists **Triangles**, **Edge elements**, and **Vertex elements**. If the meshes intersect, check and make sure that the settings for **Block 1** are correct, then reimport the mesh of the block.

5 In the **Statistics** window for **Mesh**, locate the **Geometric Entity Selection** section.

6 From the **Geometric entity level** list, choose **Boundary**.

7 From the **Selection** list, choose **All boundaries**.

In the **Statistics** window, we can now also see that the **Minimum element quality** of the surface mesh is rather low (about 1.7e-4). Furthermore, the **Element quality histogram** reveals that a large portion of the elements have low quality. You will remesh the vertebra further ahead to get a higher element quality.

Intersect with Plane 1

Continue with intersecting the mesh of the block and vertebra with an assumed symmetry plane.

1 In the **Mesh** toolbar, click **Booleans and Partitions** and choose **Intersect with Plane**.

2 In the **Settings** window for **Intersect with Plane**, locate the **Plane Definition** section.

3 From the **Plane** list, choose **yz-plane**.

4 Locate the **Cleanup** section. From the **Repair tolerance** list, choose **Absolute**.

The **Absolute tolerance** is set to 0.01 mm by default, which is the value we will use here. The **Automatic** setting uses a higher tolerance (0.09 mm), which is something you can see if you first build the operation with the **Automatic** setting, and then switch to **Absolute**.

5 Click **Build Selected**.

You should now see in the **Graphics** window that the mesh of the vertebra and the block have been partitioned by the plane. New mesh surfaces that connect the vertebra and the block along the plane have also been created inside the block. This is controlled by the **Create intersection faces** check box. When selected, the operation creates planar faces inside closed edge loops generated by the intersection.

Intersecting a mesh can introduce small and sliver mesh elements. The cleanup part of the operation collapses these, but in doing so, it may change the shape of the mesh that is intersected to ensure a planar intersection face. A lower tolerance will preserve the original shape of the mesh better, but will also keep more of the small and sliver elements introduced by the intersection. Here, the lower tolerance helps to keep the original shape better and avoid some small faces. The small faces result when parts of the vertebra that are very close to the intersecting plane are collapsed onto the plane.

Delete Entities 1

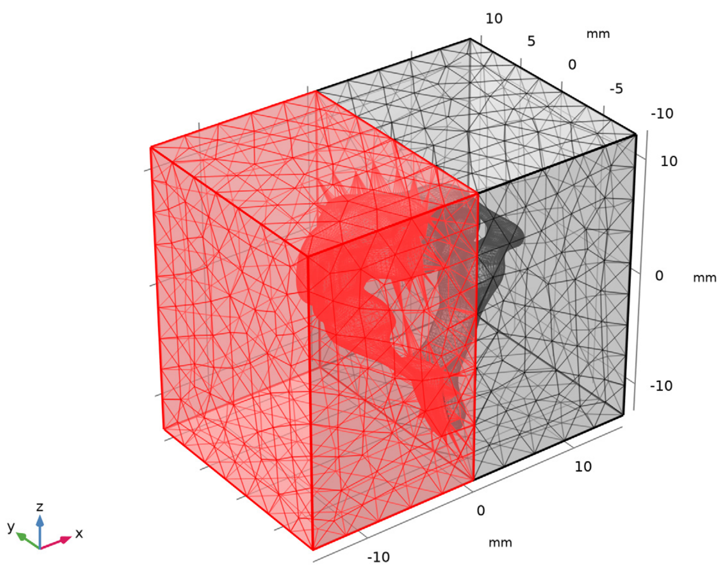
Next, delete half of the mesh to make use of the symmetry plane that was introduced above.

1 In the **Mesh** toolbar, click **Delete Entities**.
2 Go to the **Home** tab.
3 In the **Home** toolbar, click **Windows** and choose **Selection List**.

Keep the **Selection List** open as you will use it throughout the tutorial.

SELECTION LIST

1 Go to the **Selection List** window.
2 In the list, choose **1** and **4**.



3 Click **Add to Active Selection for Delete Entities 1** in the window toolbar.

MESH 1

Delete Entities 1

1 In the **Model Builder** window, click **Delete Entities 1**.

2 In the **Settings** window for **Delete Entities**, click **Build Selected** to delete the part of the vertebra and block on one side of the symmetry plane.

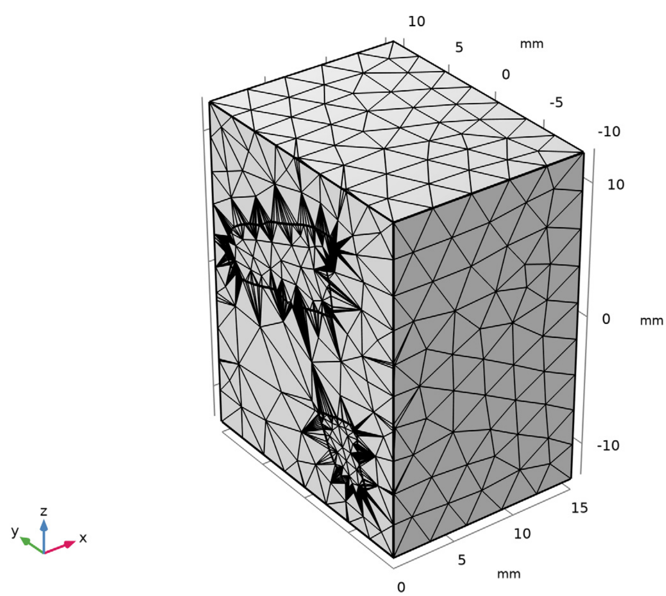
3 Click the **Go to Default View** button in the **Graphics** toolbar.

4 In the **Graphics** window toolbar, click ▾ next to **Select Domains**, then choose **Select Boundaries**.

5 In the **Selection list**, confirm that you have the expected nine boundaries; six for the block and three for the vertebra. If there are more than nine boundaries in the list, check the tolerance setting for the **Intersect with plane** operation.

After intersecting a mesh with a plane, remeshing is usually needed to eliminate small elements that often result on the faces intersected by the plane. Also, the mesh on intersection faces is always generated as coarsely as possible and will typically need to be refined. Remeshing the faces will result in mesh elements of more uniform size, and will also result in triangles that are closer to the wanted equilateral shape.

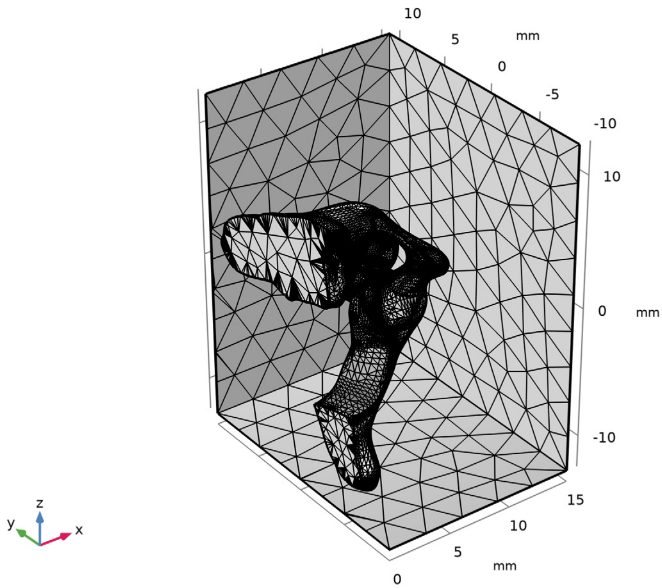
6 Click the **Transparency** button in the **Graphics** toolbar, as this makes it easier to see the mesh on the outside of the block. Also, if you have turned off **Mesh Rendering** earlier, turn it on again.



7 In the **Model Builder** window, click **Mesh 1**.

8 Click the **Click and Hide** button in the **Graphics** toolbar.

9 Hide boundaries 2,6, and 8 by clicking on them in the graphics to arrive at the image below.



10 Click the **Click and Hide** button in the **Graphics** toolbar again to deactivate the functionality and avoid hiding more boundaries.

Next, you will remesh the faces with the **Free Triangular** operation to generate a finer mesh and improve element quality. This may be needed when the quality of an imported mesh is not suitable for the simulation at hand or, as discussed earlier, to improve the mesh on the intersection faces and close to the intersection edges.

Free Triangular 1

1 In the **Mesh** toolbar, click **Boundary** and choose **Free Triangular**.

2 Select Boundaries 2–9 only. This is most easily done from the **Selection List**, as some of the boundaries are hidden.

The mesh on the far end of the block (Boundary 1) has not been modified, and is of good quality. Therefore, there is no need to remesh it.

3 In the **Settings** window for **Free Triangular**, click to expand the **Mesh Preprocessing** section.

4 In the **Relative simplification tolerance** text field, type `0.001`. The lowered tolerance allows for a closer representation of the curved parts of the vertebra faces with a smaller radius.

Size 1

1 In the **Model Builder** window, right-click **Free Triangular 1** and choose **Size**.

This adds a size attribute, **Size 1**, to the **Free Triangular** operation. Use it to set a smaller element size on the boundary of the vertebra. The first subnode specifies the default element size for the operation, which applies to all boundaries, unless it is overridden by other size settings.

2 In the **Settings** window for **Size**, locate the **Geometric Entity Selection** section.

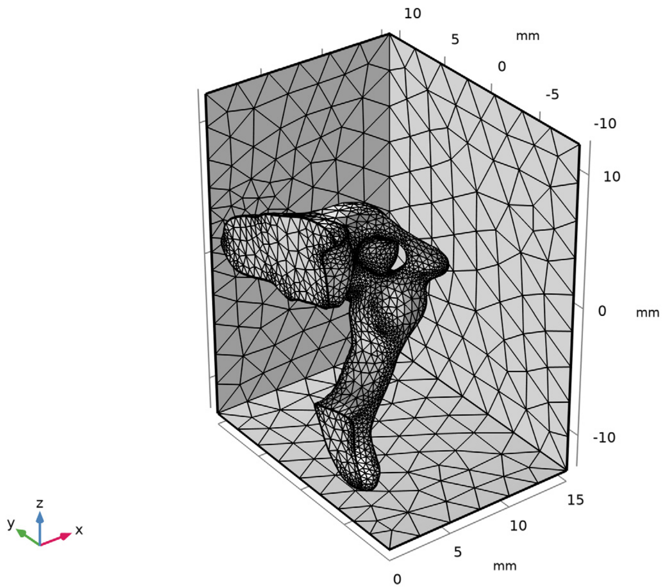
3 From the **Selection** list, choose **Vertebra boundary** to set a finer mesh size on the face of the vertebra.

4 Locate the **Element Size** section. From the **Predefined** list, choose **Fine**.

It is possible to change the individual mesh size parameters to customize the mesh size even further. This is described in the meshing tutorial *Adjusting the Element Size for the Unstructured Mesh Generator*, available in the Application Libraries.

5 Click **Build Selected**.

The mesh now consists of approximately 6500 triangles.



When generating a new mesh with the **Free Triangular** operation, smooth surfaces are first created in the background, based on the original mesh. These surfaces are then used when the new mesh is generated. See the discussion in the section *Comparing the Meshed Geometry with the STL Mesh* in the tutorial *STL Import 1 — Generating a Geometry from an Imported Mesh* for more information on how the **Relative simplification tolerance** influences the shape of the faces to remesh.

Another alternative is to modify the mesh using the **Adapt** operation with an absolute size expression. For an imported linear mesh, the **Adapt** operation places new or moved mesh vertices on a curved representation of the mesh, where the curvature is derived from the linear mesh.

6 On the **Mesh** toolbar, click the **Statistics** button.

In the **Messages** window, you can verify that the Minimum element quality is now much improved (0.4) and the high average quality (0.8) indicates that most of the triangle elements have good quality.

Free Tetrahedral 1

Now, fill the domains with a volume mesh.

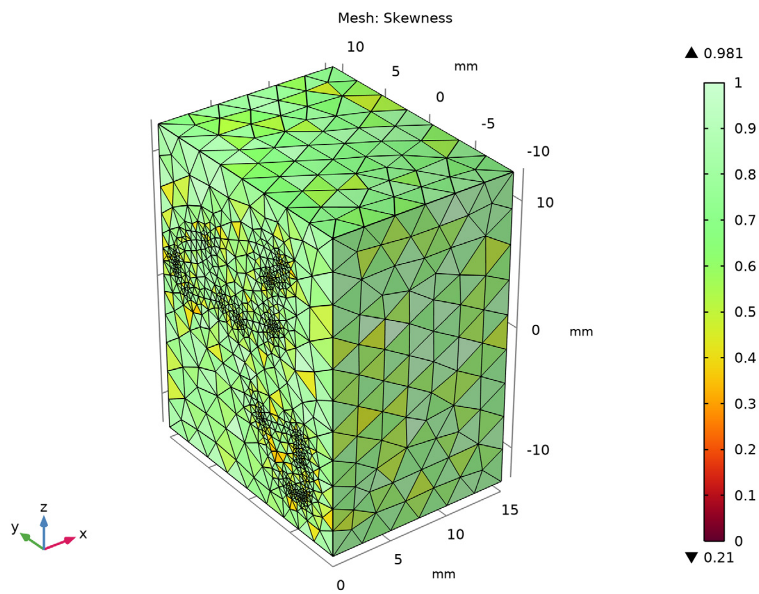
1 In the **Mesh** toolbar, click **Free Tetrahedral**.

2 In the **Settings** window for **Free Tetrahedral**, click **Build Selected**.

The **Messages** window reports that the mesh now contains domain (tetrahedral) elements. If you check the **Selection List** window, you can also see that the two domains listed there are marked as '(selected, meshed)' and no unmeshed domains are left. This means that the mesh is now ready and can be used to set up materials and physics for a 3D simulation in the software.

Before concluding the tutorial, let us take a closer look at how to create and customize a mesh plot for a visual inspection of the mesh.

3 In the **Mesh** toolbar, click **Plot**.



The default mesh plot shows the quality of the mesh elements. The light green color indicates a good quality mesh (quality close to 1).

RESULTS

Mesh 1

Follow the steps below to create a plot of the volume elements colored according to the domain they belong to and apply filtering to see the elements inside the block and vertebra.

1. In the **Settings** window for **Mesh**, locate the **Coloring and Style** section.
2. From the **Element color** list, choose **White** to color the mesh elements white.
3. Click to expand the **Element Filter** section. Select the **Enable filter** check box.
4. In the **Expression** text field, type `x>1[mm]` to visualize elements with at least one vertex with x-coordinate higher than 1 mm.
5. In the **Mesh Plot 1** toolbar, click **Plot**.

Next, add a **Selection** to the plot node and select the vertebra domain.

Selection 1

1. Right-click **Mesh 1** and choose **Selection**.
2. Select Domain 2 only (the domain of the vertebra).
3. In the **Mesh Plot 1** toolbar, click **Plot**.

Now, duplicate the **Mesh 1** plot node and make some changes to color the surrounding elements in a light blue color.

Mesh 2

Right-click **Mesh 1** and choose **Duplicate**.

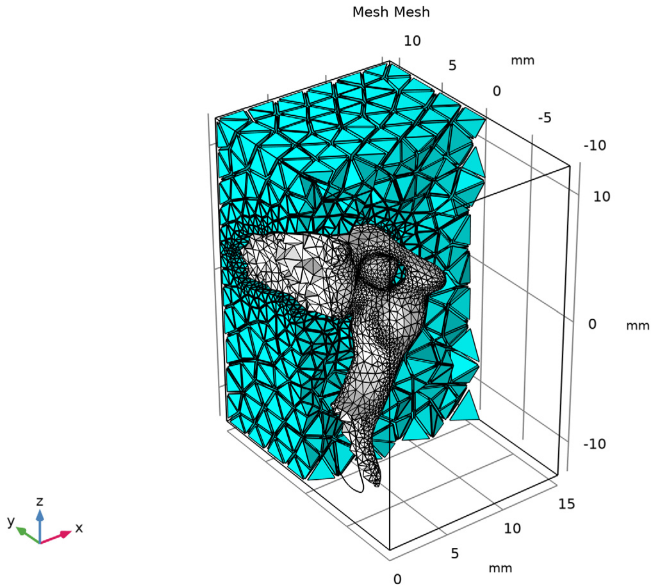
Selection 1

1. In the **Model Builder** window, expand the **Mesh 2** node, then click **Selection 1**.
2. In the **Settings** window for **Selection**, locate the **Selection** section.
3. Click **Clear Selection**.
4. Select Domain 1 only (the surrounding block).

Mesh 2

1. In the **Model Builder** window, click **Mesh 2**.
2. In the **Settings** window for **Mesh**, locate the **Coloring and Style** section.
3. From the **Element color** list, choose **Cyan**.
4. Locate the **Element Filter** section. In the **Expression** text field, type `y>1[mm]`.
5. Click to expand the **Shrink Elements** section. In the **Element scale factor** text field, type `0.8`.

6 In the **Mesh Plot 1** toolbar, click **Plot**.



Set the **Element scale factor** to a value smaller than 1 to inspect how individual mesh elements are connected, as you can then see the parts of the surrounding elements that would otherwise be hidden behind the first layer of elements in the view. You can enter a value larger than 1 to increase the size of the elements in the plot. Use this when you need to inspect the shape of individual elements that are really small.

When the mesh conforms with the geometry in the model, the elements are curved to match the boundaries of the geometry according to the specified geometry shape function. If there is no geometry, as is the case for imported meshes, the curved elements are generated in one of two ways: For imported linear elements, as this STL file of the vertebra, the software estimates the shape of surfaces and curves from the linear elements to place higher-order nodes. Else, for mesh files that contain second-order elements, as some NASTRAN and COMSOL Multiphysics files do, the software extracts this information to curve the boundary elements and place higher-order nodes accordingly. A curved, second-order representation of the geometry shape is automatically used when solving for most physics interfaces. On the other hand, CFD problems are typically solved using a linear geometry shape function.

Follow the remaining few steps to visualize the curved elements and higher-order nodes on the boundary of the vertebra.

Mesh Plot 1

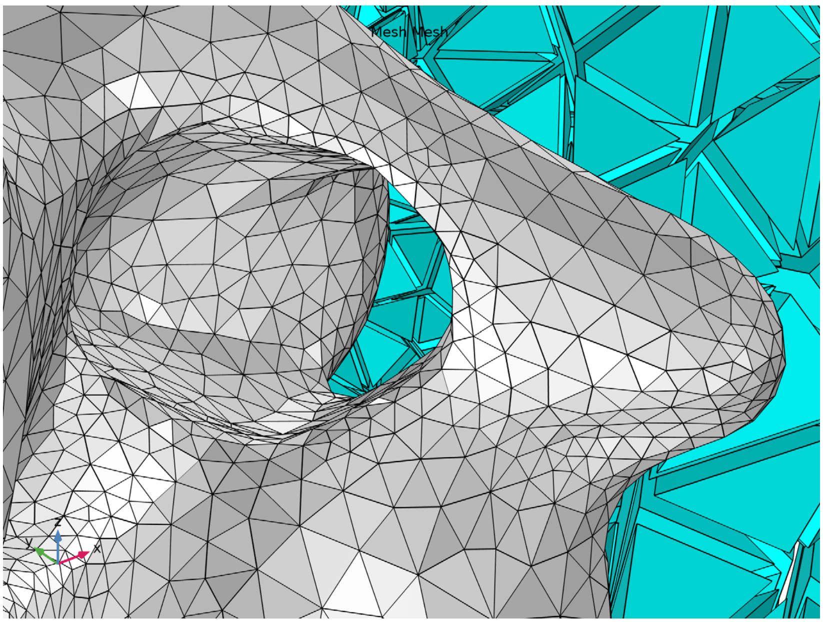
1 In the **Model Builder** window, click **Mesh Plot 1**.

2 In the **Settings** window for **3D Plot Group**, locate the **Plot Settings** section.

3 Clear the **Plot dataset edges** check box.

4 In the **Mesh Plot 1** toolbar, click **Plot** to hide the black edges of the block and vertebra. You can turn them on again later if you want to see them.

5 Zoom in on a part of the vertebra boundary that has higher curvature.



This is what the mesh looks like with a linear representation of the faces.

Now, change the **Geometry shape function** to plot second-order elements.

Mesh 1

1 In the **Model Builder** window, expand the **Results>Datasets** node, then click **Mesh 1**.

2 In the **Settings** window for **Mesh**, locate the **Mesh** section.

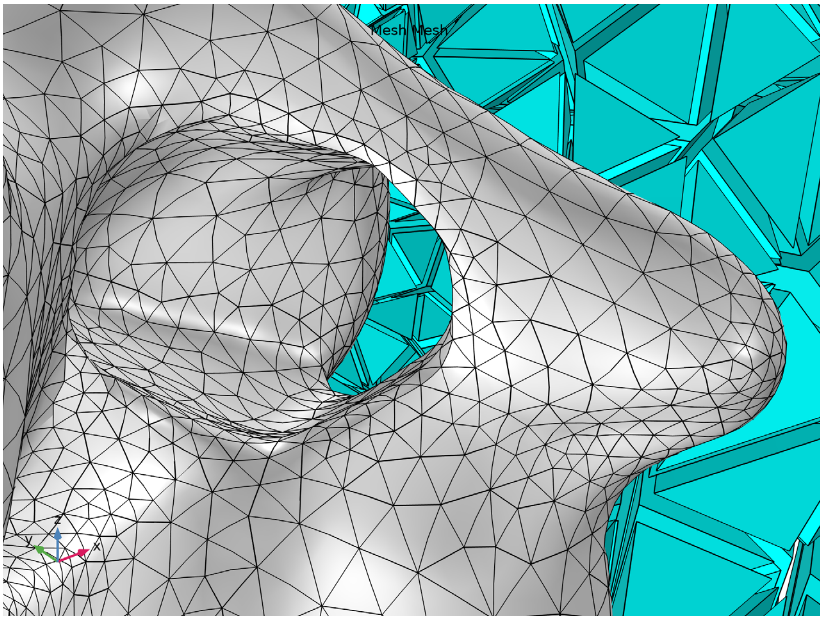
3 From the **Geometry shape function** list, choose **Quadratic Lagrange**.

Note that this is just a plot setting and does not influence the **Geometry shape function** used when solving.

Mesh 1

1 In the **Model Builder** window, under **Results>Mesh Plot 1** click **Mesh 1**.

2 This updates the plot.



The faces are now represented with second-order elements, which results in a smoother shape.

Lastly, let us add the node points of the second-order elements to the plot.

3 In the **Settings** window for **Mesh**, locate the **Coloring and Style** section.

4 From the **Node points** list, choose **Geometry shape function**.

5 In the **Mesh Plot 1** toolbar, click **Plot**.