Created in COMSOL Multiphysics 6.1

Submarine Cable 7 — Geometry & Mesh 3D

Introduction

The *Inductive Effects 3D* tutorial (the next one in this series), will add a detailed investigation of the cable's electromagnetic properties in 3D. Equations are solved, physical phenomena are analyzed and reflected upon, and advanced postprocessing is included. The tutorial does not show the effort that went into preparing the *geometry* and the *mesh* however, while oftentimes that is what represents a major part of the time spent on large 3D FEM models.

This is especially true for three-phase armored cable models. The geometry is such that — even with the appropriate periodicity conditions in place — extreme aspect ratios will have to be resolved. If one would just try to brute-force it and put in an isotropic free tetrahedral mesh for all domains, the amount of *degrees of freedom* (the DOF count) would quickly rise to thirty million or more. Although iterative solvers may be able to handle that without too much memory consumption, finding an iterative solver configuration that does the job reliably and efficiently is not an easy task.

The best approach it seems, is to use a *direct solver*. But for a direct solver to work smoothly on reasonably cheap hardware — in this case, a desktop machine with 32 GB of RAM and several hundred gigabytes of SSD swap drive capacity — the DOF count will need to be lowered to about 2–4 million. Luckily, direct solvers are more forgiving when it comes to anisotropic meshes (long mesh elements). The challenge is then to find a mesh that resolves the cable geometry and the physics as best as possible, while still keeping the DOF count low enough to have a reasonable solving time on an affordable desktop machine.

A NOTE ON SHORT-PERIODIC MODELS

The mesh constructed here considers a "standard" twisted periodicity (as discussed in [1]), resulting in a periodic length on the order of meters. In addition to this, a "short" twisted periodicity exists that reduces the periodic length to decimeters or even centimeters [2]. The issue of geometrical features with extreme aspect ratios does not seem to hold for *short-periodic* models. Those models, however, come with some challenges of their own, in particular when it comes to *mesh conformity* (for more, see Conforming Meshes).

Furthermore, as the desired complexity increases, the exercise shown here will still be relevant: The short-periodic configuration is able to greatly reduce the model's size, but as more lay lengths are introduced — in a double twisted armor, or a stranded screen, for example — the length required for the periodic condition to match will get longer again, and the challenge of dealing with anisotropic meshes will return. For more on the short-periodic configuration, see the *Inductive Effects 3D* tutorial.

Model Definition

The 3D geometry represents the same cable as the 2D one used in the previous tutorials, see Figure 1. The main difference is the addition of a real twist (not an approximation, as done for the 2.5D models). In practice, the phases and the armor are twisted with different lay lengths, and in opposite directions (to obtain *torsion stability*, see Figure 3). This causes an intricate interaction between the phases, screens, and armor. One that the 2D models cannot capture.

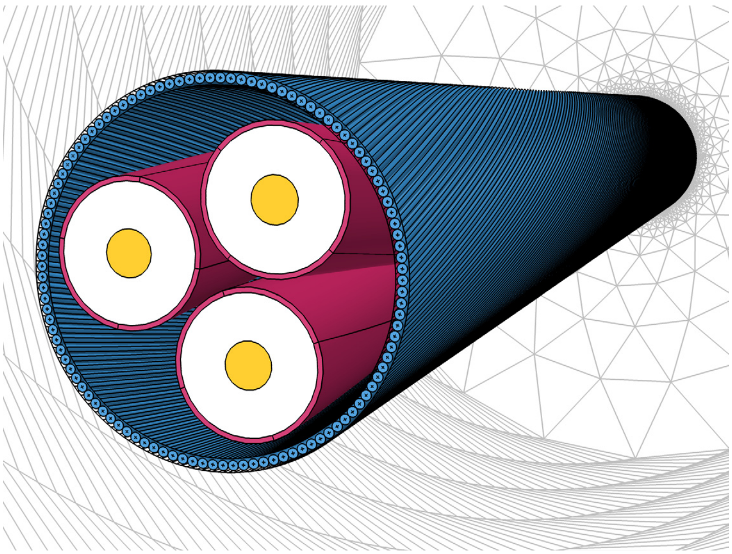


Figure 1: The cable's 3D geometry, including the three phases (yellow), screens (red), the XLPE (white), and the armor (blue).

Depending on the ratio between the phase- and the armor lay length, the true periodic length of the cable may be as long as 30–40 meters, causing the model to become impractically large. An efficient way of dealing with this, is to use a kind of "twisted periodicity" — as discussed in detail in the *Inductive Effects 3D* tutorial. This reduces the length of the geometry to a manageable 1.6 m (known as the cable's *cross pitch*).

In the previous tutorials, you may have noticed that the geometric details within the insulating domains have very little influence on the cable's inductive properties (including the fiber optic cable). To drastically reduce the complexity of the 3D geometry, all insulators have been replaced by a single homogeneous "generic insulator" material. The geometric details that remain, are either necessary because they strongly affect the results (typically the metals), or because they aid meshing and postprocessing.

The resulting structure consists of a central core (the three phases and screens), and the armor (see Figure 2). The core is twisted one way, the armor is twisted the other. To greatly reduce the number of degrees of freedom, the environment of the cable is twisted too, and meshed with the same highly anisotropic, well-structured, *swept mesh* as the armor. A similar reasoning holds for the central domains.

In-between the core and the armor, there is a void region where no swept mesh can be applied. Here, a non-structured *free tetrahedral* mesh is used. The tetrahedrons too, are anisotropic; they are stretched by a factor of six in the longitudinal direction of the cable.

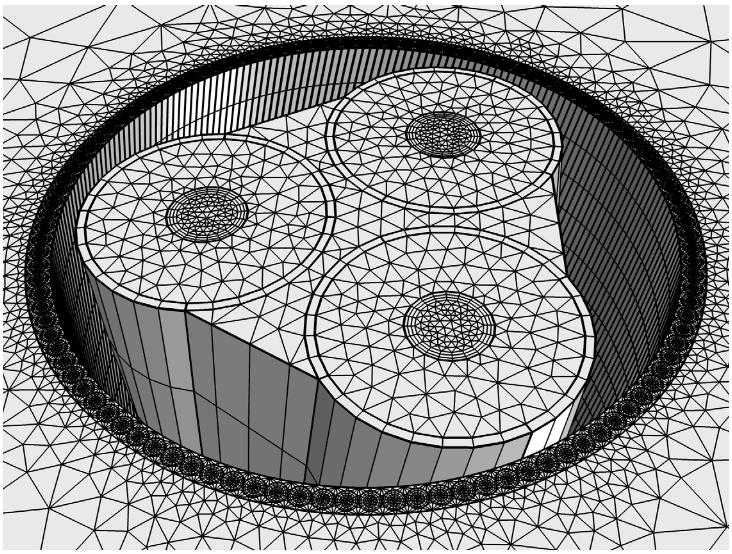


Figure 2: The swept mesh in the phases, screens, and armor. The void region between the screens and the armor will use a free tetrahedral mesh (not shown in this image).

MODELING APPROACH

The modeling instructions in this tutorial are divided in four sections (as listed below), and for each section a reference file has been saved. This will allow you to start somewhere halfway the tutorial (*although going through it from start to finish is recommended*).

- Modeling Instructions (including *Modeling Instructions — Camera Setup*).
- Modeling Instructions — Geometry Sequence.
- Modeling Instructions — Selections.
- Modeling Instructions — Mesh Sequence.

Camera Setup

Before building the geometry and the mesh, the camera setup is prepared. In total, five so-called "views" are created; four orthographic projections, and one with a strong perspective distortion. A **View** is basically a collection of settings that determines how the geometry, the mesh and the plots are shown on the screen — and how they are exported to images, paper, or this *.pdf file. The settings include things like lighting conditions, material colors, transparency, and camera settings.

The aim of this section is twofold: First of all, having the same views as the ones used for the pictures in the tutorial will make it easier for you to check your work. Secondly, mastering the camera settings will come in handy when communicating your work later.

Geometry Sequence

In order to configure the cameras, a very simple dummy geometry was created. This dummy geometry is then removed and replaced by the actual cable geometry. To accommodate a quick building- and testing procedure, the cable geometry is kept short at first, and it does not have a twist. Its main body is constructed using three work planes, two of which are subsequently swept to form the domains. The third one is used to add the *mesh control entities.*

Here, a **Work Plane** is a geometric feature that allows you to draw a 2D geometry (part of the cross section) on a plane in 3D space. The **Sweep** operation is used to create twisted extrusions of these work planes. Finally, the **Mesh Control Entities** are geometric features that have no purpose other than to support the meshing procedure.

Note: For this model, the line between geometry and mesh is blurred in the sense that certain parts of the mesh are completely dictated by-, and therefore, effectively constructed in the geometry sequence.

Selections

Once the parameterized geometry sequence is complete, it is set to a "full 3D twist mode" by modifying the parameter `Nper`. Then, the selections are added.

A **Selection** is basically a stored reference to a number of domains, boundaries, or edges. The model's base selections are *coordinate-based*: Geometric entities are selected based on their location. Other selections are derived from these base selections using operations like the **Union**, **Intersection**, **Complement**, **Adjacent**, and so on. The selections are used when configuring the mesh, and when setting up the physics and postprocessing.

Mesh Sequence

The mesh sequence is added after the selections. Most of the effort is spent on creating a well-defined cross-sectional surface mesh on the bottom periodicity plane (the cable geometry is oriented vertically), taking into account the skin depth — see Boundary Layer Meshes. When this is done, the surface mesh is swept for the central- and outer domains, resulting in the domain mesh shown in Figure 2.

The remaining void is filled with tetrahedra, but not before creating the appropriate boundary meshes: The elements on the swept-tetrahedral interface will need to be converted to create a proper connection between the tetrahedra, the hexahedra, and the prism elements. Furthermore, *mesh conformity* needs to be enforced on the periodicity planes. For the swept meshes conformity comes naturally, but for the free tetrahedral mesh a **Copy Face** operation will have to be added; see Conforming Meshes.

To complete the tutorial, some plots are added to investigate the mesh structure, the low quality elements, and the conformity of the mesh on the periodicity planes.

ON GEOMETRICAL CORRECTION FACTORS

Slant Correction Factors

The geometry consists of a number of helically twisted conductors. The helices repeat themselves after a distance equal to the lay length (see Figure 3). Because of the twist, the periodicity plane does not cut the conductors at right angle. Instead, it creates a *slanted cut*. Due to the slanted cut, the conductors will not show up in the cross section as perfect circles, but more like C-shaped ellipses; stretched in the angular direction — an extreme case of this effect is shown in Figure 4, left.

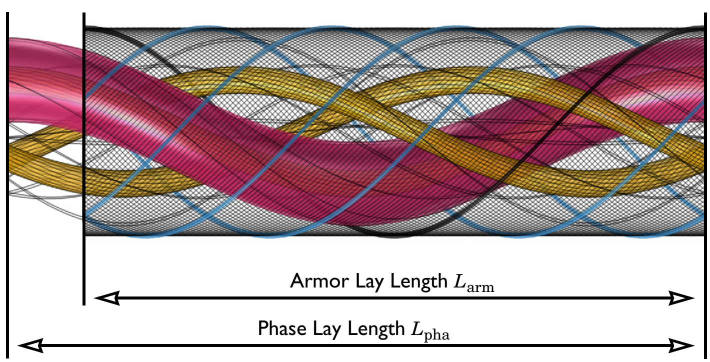


Figure 3: The cable's conductors are helically twisted in accordance with their lay lengths (image scaled by a factor of 5 in the longitudinal direction).

When the angle of the cut is close to 90°, an ordinary ellipse suffices as an approximation. In the geometry sequence, the ellipses are created by stretching circles (the **Scale** operator is used to apply an anisotropic scaling). The aspect ratio of the ellipses is set by the *slant correction factor*. The slant correction factor, in turn, can be derived from the aspect ratio between the lay length and the actual helical length of the conductor:

$$F_{\mathrm{arm}} = \frac{\sqrt{(2\pi R_{\mathrm{arm}})^2 + L_{\mathrm{arm}}^2}}{L_{\mathrm{arm}}}, \tag{1}$$

where F_{arm} is the slant correction factor, R_{arm} is the major radius of the helix, and L_{arm} is the lay length. Equation 1 mentions the armor specifically, but the same logic applies to the phases and the screens.

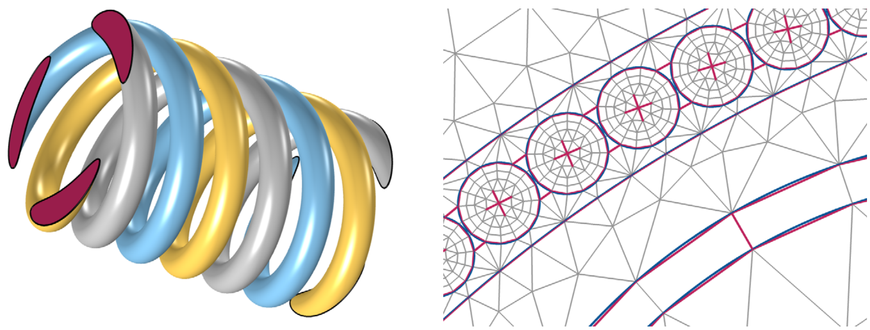


Figure 4: The cross-sectional cut of a helix (left), and the truncation of circles (right).

Truncation Correction Factors

In order to drastically reduce the number of degrees of freedom, the model uses linear elements — see the *Inductive Effects 3D* tutorial, section *Shape Functions and Discretization Order*. One of the implications, is that curved shapes become polygons when the geometry is transformed into a mesh. The effect is illustrated in Figure 4, on the right: The blue lines indicate the actual curved shape of the geometry, the gray lines show the linear elements, and the red lines show how the geometry is being approximated.

The geometry consists almost entirely of scaled, twist-extruded circles. When a circle becomes a polygon, the surface area of the shape reduces slightly — that is; given the radius of the *circumcircle* remains the same (the smallest circle that contains the shape).

For example, the surface area of the circle, the *hexadecagon*, and the *icosagon*, is given by[1]:

1. The hexadecagon has sixteen sides, the icosagon has twenty. Both polygons are *regular polygons*. A regular polygon is a polygon for which all angles and sides are equal.

$$A_c = \pi R_{cc}^2 \qquad A_{p16} = 4R_{cc}^2\sqrt{2-\sqrt{2}} \qquad A_{p20} = 5R_{cc}^2\frac{\sqrt{5}-1}{2}, \tag{2}$$

where R_{cc} is the radius of the circumcircle. In other words, a shape that is drawn as a circle in the geometry, but is subsequently meshed to be come a hexadecagon, loses about $1-4\sqrt{2-\sqrt{2}}/\pi \approx 2.55\%$ of its surface area in the process. For the cable models this effect is significant, because there is a direct relation between the surface area of a conductor, and its resistance (or its *reluctance*, for that matter). For the DC resistance R_{dc} of a conductor with length l, conductivity σ, and cross-sectional surface area A, the relation is given by:

$$R_{dc} = \frac{l}{\sigma A}. \tag{3}$$

So the losses in the cable may be a few percent off, simply because the first-order elements have difficulties representing the geometry accurately.

Typically you would not bother too much about such details because of the following reasons:

- Higher-order elements are used.
- Finer meshes are used.
- The circular cross sections only comprise a small portion of the geometry.
- The kind of physics solved for is not sensitive to this effect in the first place.

In this case, however, the model consists almost entirely of round conductors, and the goal of this exercise is *to represent these conductors (and the corresponding physics) as accurately as possible, with a mesh that produces as little degrees of freedom as possible.*

Note: The main exercise in this tutorial is to represent the conductors (and the corresponding physics) as accurately as possible, with a mesh that produces as little degrees of freedom as possible.

A very inexpensive measure — in terms of computational effort — is to simply enlarge the polygons a bit, so that their surface area is precisely what it should be: This is what the truncation correction factors are used for. They are determined by taking the ratio between the area of the circle and the polygon (note that this ratio does not depend on R_{cc}). The resulting shape has a "radius" that approximates the radius of the circle *on average*, rather than the circle's radius being an upper limit. For more, see the section Modeling Instructions — Geometry Sequence.

MESHING CONSIDERATIONS

Swept Meshes and Tetrahedral Meshes

Both the geometry and the physics show highly anisotropic behavior — that is what makes the 2D models such a good approximation. This results in a serious challenge for the mesh. If the cable would use the same twist for all components (or if there were no twist at all), the solution would be obvious: Use a **Swept** mesh everywhere.

A swept mesh is fast, reliable, and efficient:

- Building the mesh is *fast*, because it only needs to extrude an already existing surface mesh, using a simple, well-defined projection (assuming the geometry supports it).
- It is *reliable* because the procedure is simple and well-determined: The 3D meshing problem is basically reduced to an in-plane 2D mesh, combined with an out-of-plane 1D mesh. The swept mesh will not be sensitive to minor geometric changes.
- It is *efficient* in the sense that it allows for highly anisotropic meshes (for which the aspect ratio can be fine-tuned easily), and because the extruded elements — prisms and hexahedra — produce less degrees of freedom per unit volume than pyramids and tetrahedra do. So it is an effective tool for reducing the DOF count.

All of these advantages are due to the swept mesh being a *structured mesh*. The catch is that the geometry has to support it. The **Free Tetrahedral** mesh on the other hand, is an *unstructured* mesh. Unstructured meshes are formed by filling the volume with a cloud of mesh nodes (taking into account the boundary mesh, the radius of curvature, the size of small openings... and so on) and then trying to optimize those nodes to produce good quality elements — possibly merging elements, or splitting ones.

The unstructured mesh is a generic tool. This is both an advantage, and a disadvantage. The generality makes it difficult to optimize for extreme cases. The aspect ratios in the cable geometry are such that the free tetrahedral mesh would either: 1) Produce far too many degrees of freedom, or 2) have a very hard time finding a mesh of good quality.

One further important drawback is that it is not as deterministic as a structured mesh. A slight modification in the geometry may cause the optimization logic to follow a different path and create poor quality elements in places where they were not before. This is a disadvantage when doing a parametric sweep or an optimization study, for example.

Not that tetrahedral meshes are a bad option in general, they are just not very suitable for cable models. Because of the different twist directions and lay lengths involved, both mesh types will be needed. Section Modeling Instructions — Mesh Sequence contains a fully detailed description of how is this done.

Boundary Layer Meshes

In addition to the geometry, the mesh needs to resolve the physics. Within this context, it is important to note that the current density distribution in the conductors is subject to a skin (and proximity) effect: The current density decays as you go further into the conductor, see Figure 5. A characteristic length scale for this phenomenon is given by the *skin depth* δ. Since the decay is not linear, it seems appropriate to have at least a couple of first order elements to resolve the effect.

A suitable meshing tool for this kind of field shape, is the built-in **Boundary Layers** feature. It is supported in 2D, 2D axial symmetry, and 3D geometries. The domain that receives the boundary layers, is typically already equipped with a volumetric mesh. This mesh is then pushed further into the domain and several layers of thin mesh elements are squeezed in-between. Afterward, the interior mesh is smoothed to create a transition zone between the flat boundary layers and the more isotropic interior mesh.

Using the built-in boundary layer functionality for the cable would mean that the domains should first be meshed (resulting in a large number of elements). Subsequently, this large volumetric mesh would have to be squeezed and smoothed. This works, but a more robust and efficient way is to include the boundary layers in the surface mesh on the periodicity plane, before sweeping it to form the volumetric mesh. That requires a boundary layer mesh within a 2D surface mesh in 3D, however. This functionality is on its way, but has not been released yet (at the time of writing; 2019).

This is of no concern however — *on the contrary* — it provides a nice opportunity to investigate a workaround. This workaround consists of drawing polygons on the periodicity plane, to allow a **Mapped** mesh to take the place of the surface boundary layer mesh. The number, thickness, and distribution of the boundary layers is dictated entirely in the geometry. In essence, the geometry is used to “draw” the mesh.

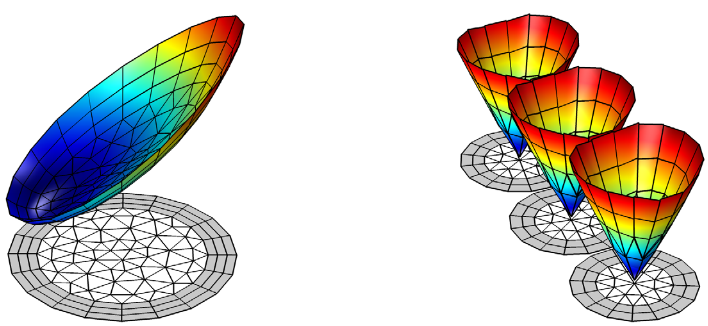


Figure 5: The skin- and proximity effect in the phase (left), and the armor wires (right). The boundary layers are indicated in gray.

Note: When it comes to geometry handling and meshing, there are typically lots of ways to arrive at the same end result, and the line between “workaround” and “native solution” may be blurry: Even when the surface boundary layer mesh in 3D is supported, constructing the boundary layers in the geometry may still be a useful alternative.

Meshing the Gaps

In addition to the boundary layers in the conductors, thin hexahedral mesh elements are added to the insulating gaps between the armor wires — so the armor wires do not actually touch (see Figure 4, right). This is not at all an obvious solution. Alternatively, the wires could touch each other. They could share a *single edge*, or they could share a *boundary*.

As soon as they touch, however, currents will start to flow from one armor wire to another. If they share an edge, there is very little that can be done to form (and fine-tune) an electric barrier. If they share a boundary, **Electric Insulation** can be applied to that boundary if the appropriate set of partial differential equations is used[2] (currents will still be leaking through the edges adjacent to that boundary though). Another issue is the *reluctance*. The gaps will not only have to be electric insulators, but magnetic insulators as well. A thin **Thin Low Permeability Gap** boundary feature can be used for this, but it will suffer from similar issues.

By far the easiest and computationally cheapest way to deal with the gaps is to put in a single, linear, hexahedral element. This comes with a small warning though: *the geometry sequence needs to make sure the wires do not touch* (even when changing its parameters). In addition to this, the resistance and permeability of the gap may need a bit of fine tuning to compensate for the inability of the mesh to resolve certain details in the gaps.

Conforming Meshes

A mesh is said to be *conforming* when the intersection between any two elements in the mesh is a subelement (mesh face, mesh edge, or mesh vertex) of both, or nothing[3]. In practice, this means that when a mesh element is adjacent to an interior boundary, there should be a mesh element on the other side of that boundary sharing the same face, edges, and mesh nodes.

2. Applying electric insulation to an interior boundary in an inductive model, would require the AV-formulation. This formulation is available in the AC/DC Module, as the *Magnetic and Electric Fields* interface. The cable model uses the A-formulation however, available as the *Magnetic Fields* interface. For more on the used formulation, see the section *Theoretical Basis* in the *Inductive Effects 3D* tutorial.

3. For more, see the *COMSOL Multiphysics Reference Manual*.

Having a conforming mesh is important to get a numerically stable, accurate model. Without it, the solution will have to be interpolated to get from one mesh to another. This causes numerical noise on the mesh interface, and electric current may not be conserved: Ampère's law — the basic physical law that the cable model solves for — will be violated.

For all interior boundaries of the cable, COMSOL produces conforming meshes naturally. For the exterior boundaries that make up the periodicity plane this is a different matter however; special care needs to be taken to make sure they have conforming meshes as well. After all, as far as the physics is concerned these boundaries are interior boundaries too, while for the mesh sequence this is not at all obvious.

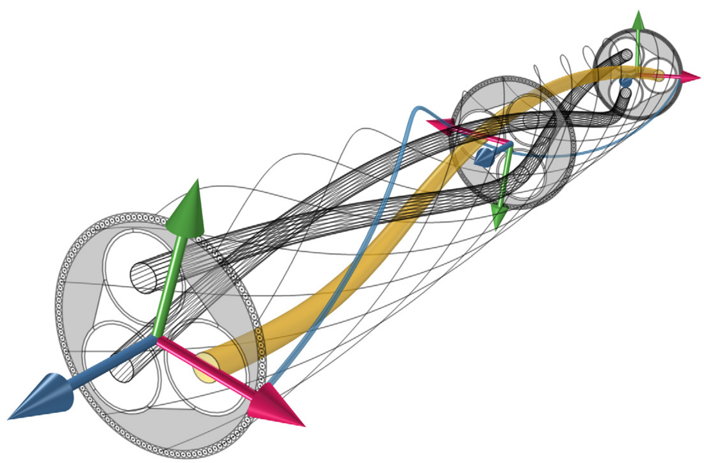


Figure 6: The orientation of the periodicity plane changes as you progress along the cable. The parts of the plane that lie adjacent to the tetrahedral mesh are indicated in gray.

What complicates matters further, is that the cable's periodicity is not a simple translation. Rather, it involves both a translation and a rotation (see Figure 6, or section *Twisted Periodicity* in the *Inductive Effects 3D* tutorial). Luckily, for the swept mesh the required conformity comes naturally: As the swept mesh follows its helical path, it automatically projects the correct surface mesh from the source to the destination boundaries.

For the tetrahedral mesh, two measures are taken. Before adding the tetrahedra, a **Copy Face** operator is used to map the boundary mesh from source to destination. The operator uses *boundary similarity*. In order to ensure a unique mapping, the three-fold symmetry needs to be broken (otherwise, the mesh might be 120° off). This is done in the geometry sequence by adding an extra edge to the exterior face of one of the screens.

A NOTE OF CAUTION ON MESH CONVERGENCE

Ideally, the mesh should be refined to the point where further modification of the mesh has no influence on the solution (typically investigated using a *mesh convergence study*). This is when the "true solution" emerges. For any mesh much coarser than this, the solution is partially dictated by the mesh. This raises the question:

What if we get the solution we expect, simply because we have optimized the mesh for the solution we expect?

In case of doubt, you will have to refine the mesh to find the true solution. For the cable, you could do a mesh convergence study in 2D and use the acquired statistics in 3D. Alternatively, you can model one phase conductor and a couple of armor wires with an extremely fine mesh and investigate if the results in those differ significantly from their coarse-meshed counterparts. Once the general shape of the solution is known, there is nothing wrong with constructing a mesh that is optimized for it.

References

1. J.C. del-Pino-López, M. Hatlo, and P. Cruz-Romero, "On Simplified 3D Finite Element Simulations of Three-Core Armored Power Cables," *Energies* 2018, 11, 3081.

2. D. de Vries, "3D Cable Modeling in COMSOL Multiphysics®" *IEEE Spectrum*, 2020, https://spectrum.ieee.org/webinar/3d_cable_modeling_in_comsol_multiphysics.

Application Library path: `ACDC_Module/Tutorials,_Cables/submarine_cable_07_geom_mesh_3d`

Modeling Instructions

This tutorial will focus almost entirely on the geometry and mesh of a *3D twist model*. No physics is added and no equations are solved. Even so, it forms an important part of the series. The instructions on the following pages are organized in four sections:

- Modeling Instructions (including *Modeling Instructions — Camera Setup*).
- Modeling Instructions — Geometry Sequence.
- Modeling Instructions — Selections.
- Modeling Instructions — Mesh Sequence.

These sections represent the four stages, as described in section Modeling Approach. In-between these sections the model is saved and reopened. This will allow you to get back on track when you made a mistake, or to skip parts on purpose — *although going through the instructions from start to finish is recommended*.

If anything seems out of order, please retrace your steps. The reference files — available in the model's Application Libraries folder — can help you out. You can compare them directly to your current model by means of the **Compare** option on the **Developer** toolbar. When doing so however, remember that the camera settings stored in your model depend on your COMSOL Desktop configuration: Your file will probably deviate a bit from the reference file.

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**.

ROOT

1 From the **File** menu, choose **Save As**.

2 Browse to a suitable folder and type the filename `submarine_cable_07_a_geom_mesh_3d.mph`.

GLOBAL DEFINITIONS

This model uses a subset of the parameters already defined for the other tutorials. Additionally, there is a set of geometric parameters defined specifically for the 3D geometry. In order to gain access to them, you can load them all.

Geometric Parameters 1

1 In the **Model Builder** window, under **Global Definitions** click **Parameters 1**.

2 In the **Settings** window for **Parameters**, type `Geometric Parameters 1` in the **Label** text field.

3 Locate the **Parameters** section. Click **Load from File**.

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

Geometric Parameters 2

1 In the **Home** toolbar, click **Parameters** and choose **Add>Parameters**.

2 In the **Settings** window for **Parameters**, type `Geometric Parameters 2` in the **Label** text field.

3 Locate the **Parameters** section. Click **Load from File**.

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

Geometric Parameters 3

1 In the **Home** toolbar, click **Parameters** and choose **Add>Parameters**.

2 In the **Settings** window for **Parameters**, type `Geometric Parameters 3` in the **Label** text field.

3 Locate the **Parameters** section. Click **Load from File**.

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

Modeling Instructions — Camera Setup

This section about cameras is meaningful for two reasons: First of all, having the same camera settings as the ones used for the pictures in the tutorial will make it easier for you to check your work. Secondly, mastering the camera settings will come in handy when communicating your work later. The cable geometry is suitable as a demonstration, because of its high aspect ratio. You can either scale it down when making an *orthographic projection* (in order to provide a good overview), or you can deliberately create a strong *perspective distortion* for marketing purposes.

GEOMETRY 1

The first geometry will be a very simple dummy geometry. It has the same dimensions as the cable, but builds and renders instantly. Its sole purpose is to help you configure the cameras correctly.

Cylinder 1 (cyl1)

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

2 In the **Settings** window for **Cylinder**, locate the **Size and Shape** section.

3 In the **Radius** text field, type `Darm/2+2*Tarm/3`.

4 In the **Height** text field, type `Lsec`.

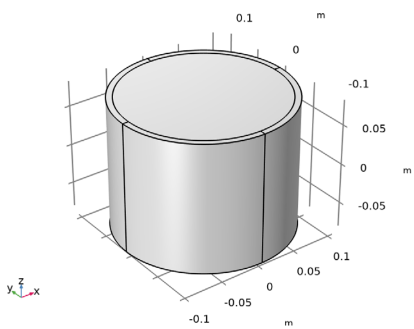
5 Locate the **Position** section. In the **z** text field, type `-Lsec/2`.

6 Click to expand the **Layers** section. In the table, enter the following settings:

Layer name	Thickness (m)
`Layer 1`	`4*Tarm/3`

7 Click **Build Selected**.

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



The cable looks rather short at this point. The parameter `Nper` is set to `1/10`, which means the geometry is only one-tenth of its actual length “`CPcab`”.

In this tutorial (and the next one) we will switch between short and long geometries, because the building-, meshing-, and rendering procedures can be a bit time consuming for the fully detailed twist geometry. The strategy is to build a short fully parameterized cable first, and then — when everything seems to be in order — crank up the parameter `Nper` to a value of `1`.

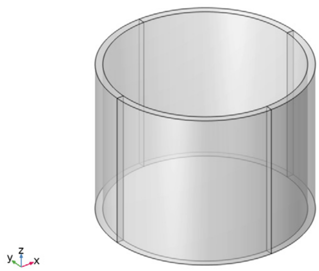
Proceed by configuring your first **View**.

9 Click the **Show Grid** button in the **Graphics** toolbar once (to hide the grid).

10 Click the **Transparency** button in the **Graphics** toolbar.

11 Click the **Orthographic Projection** button in the **Graphics** toolbar.

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



DEFINITIONS

View 1 (Orthographic)

1 In the **Model Builder** window, expand the **Component 1 (comp1)>Definitions** node, then click **View 1**.

2 In the **Settings** window for **View**, type `View 1 (Orthographic)` in the **Label** text field.

3 Locate the **View** section. Select the **Lock camera** check box.

4 Locate the **Colors** section. Select the **Show material color and texture** check box.

Having the camera locked is convenient — *especially if it took you some effort to find a camera position you like*. With the camera locked, you can pan and zoom all you want without affecting the stored settings. When you switch between views or update the camera, the camera will be restored to its original state. The material colors will not show up just yet by the way, they will become visible as you add materials in the next tutorial.

Camera

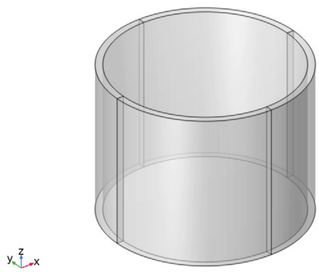
1 In the **Model Builder** window, expand the **View 1 (Orthographic)** node, then click **Camera**.

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

3 From the **View scale** list, choose **Manual**.

4 In the **z scale** text field, type `1/(10*Nper)`.

5 Click **Update**.



Changing the scale should have no effect. This is because `1/(10*Nper)` evaluates to "`1`". Now, if you would change the parameter `Nper` itself to "`1`" and rebuild your geometry, the cable would become ten times as long. At the same time, `1/(10*Nper)` would evaluate to `1/10`. This causes the camera to scale the geometry such that its on-screen aspect ratio is still the same. Consequently, this parameterized camera will always give a good overview regardless whether the cable is short or long.

Next, let us add a block. The block will help you to distinguish left from right and top from bottom. Additionally, it will show you how the scaling of the camera affects the picture (because its height does not depend on `Nper`).

GEOMETRY 1

Block 1 (blk1)

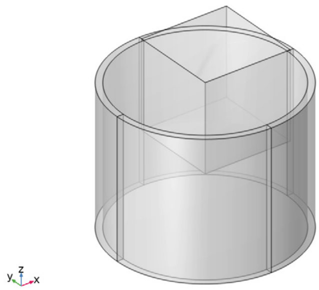
1 In the **Geometry** toolbar, click **Block**.

2 In the **Width**, **Depth**, and **Height** text fields, type `Darm/2`.

3 In the **Settings** window for **Block**, locate the **Position** section.

4 In the **z** text field, type `Lsec/2-Darm/2`.

5 Click **Build Selected**.



The block looks like a cube. This is a good thing. It means the camera applies no anisotropic scaling at this point and the cable is indeed as short as it looks like.

Proceed by adding orthographic *top* and *bottom* views that focus on the cable's cross section (much like the 2D models do).

DEFINITIONS

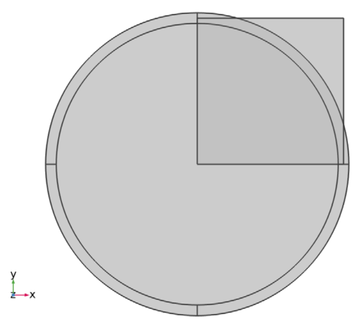
View 2 (Orthographic, Top)

1 In the **Model Builder** window, right-click **View 1 (Orthographic)** and choose **Duplicate**.

2 In the **Settings** window for **View**, type `View 2 (Orthographic, Top)` in the **Label** text field.

3 Locate the **View** section. Clear the **Lock camera** check box.

4 Click the **Go to XY View** button in the **Graphics** toolbar.



5 Select the **Lock camera** check box.

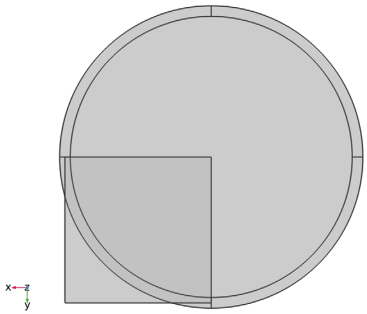
View 3 (Orthographic, Bottom)

1 Right-click **View 2 (Orthographic, Top)** and choose **Duplicate**.

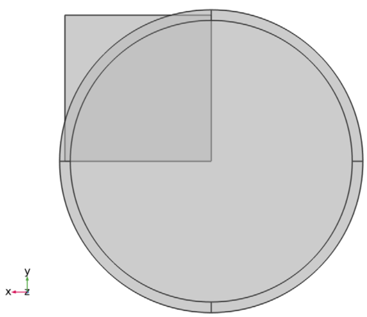
2 In the **Settings** window for **View**, type `View 3 (Orthographic, Bottom)` in the **Label** text field.

3 Locate the **View** section. Clear the **Lock camera** check box.

4 Click the **Go to XY View** button in the **Graphics** toolbar.



5 Click the **Go to XY View** button in the **Graphics** toolbar again.



6 Select the **Lock camera** check box.

View 4 (Orthographic, Side)

1 Right-click **View 3 (Orthographic, Bottom)** and choose **Duplicate**.

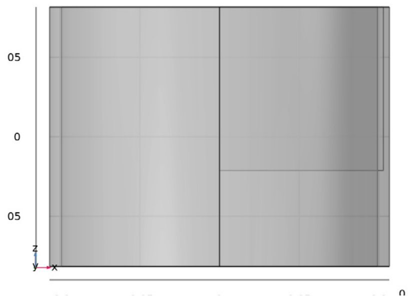
2 In the **Settings** window for **View**, type `View 4 (Orthographic, Side)` in the **Label** text field.

For the *side* view, the grid is temporarily enabled to force the default **XZ View** to zoom out a little bit more:

3 Locate the **View** section. Select the **Show grid** check box.

4 Clear the **Lock camera** check box.

5 Click the **Go to XZ View** button in the **Graphics** toolbar.



6 Clear the **Show grid** check box.

7 Select the **Lock camera** check box.

Finally, let us add a view with a strong perspective distortion. In the following, the optimal values — in particular those for the **Zoom angle**, **Position**, and **View Offset** — will depend on the aspect ratio of your graphics window, *and your personal preferences of course*. The used parameters however (`Dcab`, `Lsec`, and `Nper`), should be copied as-is to achieve some form of automation.

For example; regardless the values of the numbers, it is good to keep the *z*-position of the camera a function of the cable's length, and the *x,y*-position a function of the cable's main diameter. That way, the camera will move appropriately when you change the geometry. In case you are wondering: The numbers used here should work for anything between a 4:3 and 16:9 aspect ratio.

View 5 (Perspective)

1 Right-click **View 4 (Orthographic, Side)** and choose **Duplicate**.

2 In the **Settings** window for **View**, type `View 5 (Perspective)` in the **Label** text field.

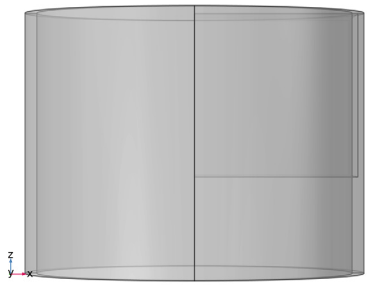
Camera

1 In the **Model Builder** window, expand the **View 5 (Perspective)** node, then click **Camera**.

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

3 From the **Projection** list, choose **Perspective**.

4 Click **Update**.



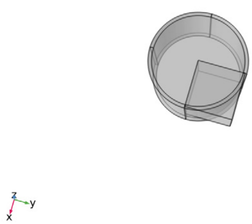
5 In the **Zoom angle** text field, type `36`.

6 Locate the **Position** section. In the **x** text field, type `1.4*Dcab`.

7 In the **y** text field, type `0.4*Dcab`.

8 In the **z** text field, type `Lsec/(2*Nper)+2.8*Dcab`.

9 Click **Update**.

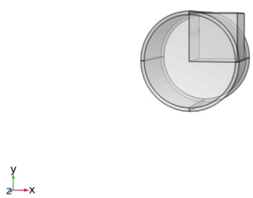


10 Locate the **Up Vector** section. In the **x** text field, type `0`.

11 In the **y** text field, type `1`.

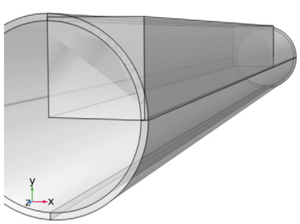
12 In the **z** text field, type `0`.

13 Click **Update**.



14 Locate the **Camera** section. In the **z scale** text field, type `1/Nper`.

15 Click **Update**.

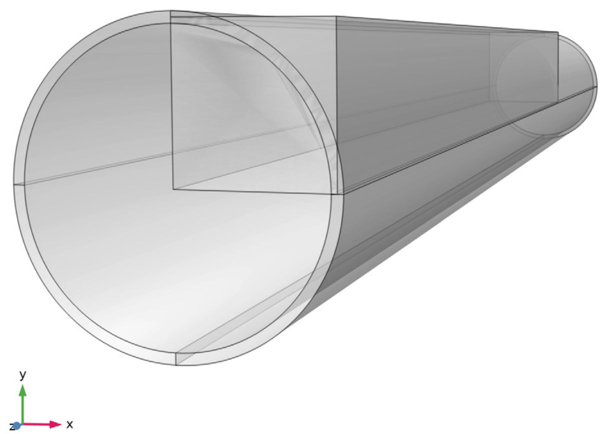


Notice that the four *orthographic projection* cameras use the value `1/(10*Nper)` here. They will therefore always show a short cable, regardless whether `Nper` is set to `1/10`, or `1`. Instead, **View 5** will always show a long cable (ten times longer). The scaling is given away by the cube: It looks more like a bar this time.

16 Locate the **View Offset** section. In the **x** text field, type `0.8*Dcab`.

17 In the **y** text field, type `0.8*Dcab`.

18 Click **Update**.



The extra margin on the right hand side is intentional. It will give some space for the color legend when plotting.

Next, proceed by checking whether the parameterization works as intended.

GLOBAL DEFINITIONS

Geometric Parameters 3

1 In the **Model Builder** window, under **Global Definitions** click **Geometric Parameters 3**.

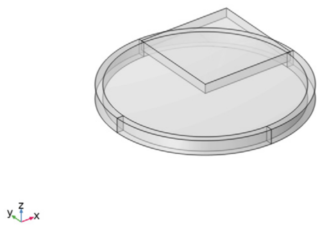
2 In the **Settings** window for **Parameters**, locate the **Parameters** section.

3 In the table, set the parameter `Nper` to `1`.

GEOMETRY 1

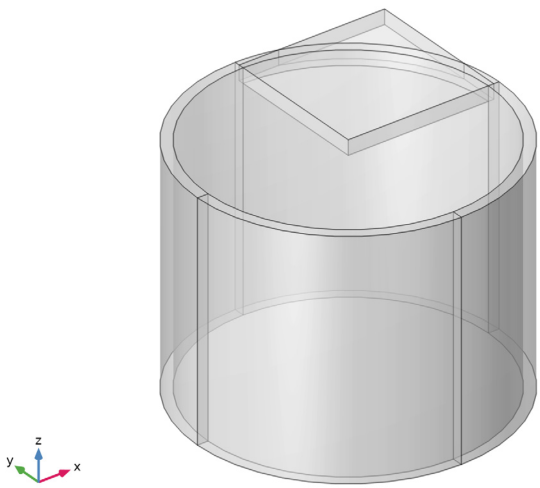
1 In the **Model Builder** window, under **Component 1 (comp1)** click **Geometry 1**.

2 Click the **Go to View 1 (Orthographic)** button in the **Graphics** toolbar.



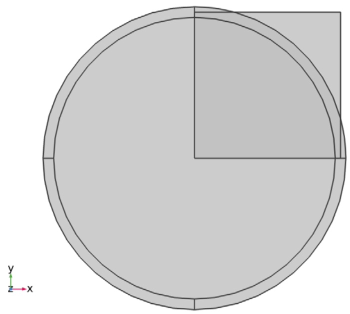
At first, the geometry looks compacted, because the camera is using the latest parameter values while the geometry has not yet been rebuilt.

3 In the **Geometry** toolbar, click **Build All**.

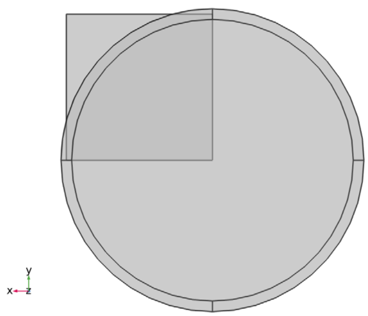


Notice the block is flat now, giving away the scaling of 1/10.

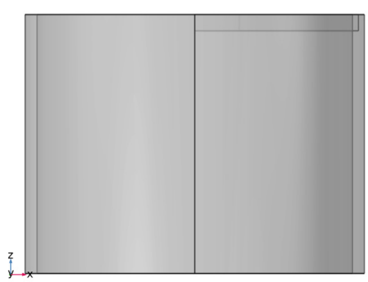
4 In the **Settings** window for **Geometry**, in the **Graphics** window toolbar, click next to **Go to Default View**, then choose **Go to View 2 (Orthographic, Top)**.



5 In the **Graphics** window toolbar, click next to **Go to Default View**, then choose **Go to View 3 (Orthographic, Bottom)**.



6 In the **Graphics** window toolbar, click next to **Go to Default View**, then choose **Go to View 4 (Orthographic, Side)**.



7 In the **Graphics** window toolbar, click next to **Go to Default View**, then choose **Go to View 5 (Perspective)**.

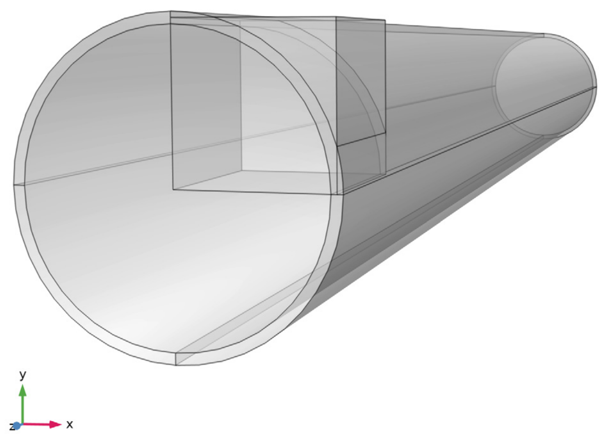
If the camera is zoomed in too much (or too little), it could be that it needs to be refreshed after the parameter change. This can be done as follows:

DEFINITIONS

Camera

1 In the **Model Builder** window, under **Component 1 (comp1)>Definitions> View 5 (Perspective)** click **Camera**.

2 In the **Settings** window for **Camera**, click **Update**.



This time, **View 5** shows a cube. No scaling is applied any more, and the cable is indeed as long as it looks like.

You have now finished configuring the cameras. Save the resulting file, so that you can use it as a basis for the next part.

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

Now things will get more interesting. The parameter `Nper` will be restored to its initial value. Then, the dummy geometry will be replaced by the real 3D cable geometry (still in its short, extruded form). Finally, `Nper` will be put back to "`1`" and the geometry will go into *full 3D twist mode* — covering a distance equal to the cable's cross pitch.

ROOT

You can either continue where you left off, or start by opening a reference file from the Application Libraries folder. Either way, it is convenient to resave the file under a new name.

1 From the **File** menu, choose **Open**.
2 Browse to the model's Application Libraries folder and double-click the file `submarine_cable_07_a_geom_mesh_3d.mph`.
3 From the **File** menu, choose **Save As**.
4 Browse to a suitable folder and type the filename `submarine_cable_07_b_geom_mesh_3d.mph`.

GLOBAL DEFINITIONS

Let us proceed by modifying `Nper`, and adding some new parameters for the electromagnetic properties of the cable.

Geometric Parameters 3

1 In the **Model Builder** window, under **Global Definitions** click **Geometric Parameters 3**.
2 In the **Settings** window for **Parameters**, locate the **Parameters** section.
3 In the table, set the parameter `Nper` back to `1/10`.

   Before adding one more set of parameters, notice that the model already contains three of them. These have been added in the previous section, and will become more relevant shortly.

The table in **Geometric Parameters 1** contains all the dimensions needed for a detailed description of the cable's cross section. These have been used extensively in the 2D models (and a small subset will be used here). The values in **Geometric Parameters 2** are used in the 2D models as well. They are not used in the geometry directly, but are nevertheless considered to be "geometric parameters". They typically relate to the length of the cross bonding sections or the conductor packing density. For more on this, see the *Inductive Effects*, *Bonding Inductive*, and *Bonding Capacitive* tutorials.

The third list of parameters is of special importance for 3D. More specifically; it relates to the twist. The parameters `LLpha`, `LLarm`, and `CPcab` specify the lay lengths and cross pitch respectively — as explained in the section *On Lay Length and Cross Pitch* in the *Inductive Effects 3D* tutorial. Furthermore, `Nper` sets how many cross pitch periods should be included in the model, and `Tenab` is a boolean value that decides whether the twist should be enabled.

Finally, `Lsec` sets the actual length of the geometry, and `Tsec` sets its twist angle. The last four are the slant- and truncation correction factors, as described in sections Slant Correction Factors, and Truncation Correction Factors.

Hint: If you press Ctrl+F and search for the strings "Nper", "Tenab", or "Lsec" in the model, you can see where these parameters will have an effect.

Electromagnetic Parameters

1 In the **Home** toolbar, click Pi **Parameters** and choose **Add>Parameters**.

2 In the **Settings** window for **Parameters**, type `Electromagnetic Parameters` in the **Label** text field.

3 Locate the **Parameters** section. Click **Load from File**.

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

The electromagnetic parameters are the same as those used in the previous tutorials. The main reason they have been loaded here already (as opposed to postponing them to the next tutorial), is that the skin depth δ will come in handy when deciding what mesh element size to choose. Since the model will be solved in the frequency domain, the skin depth in the conductors gives a fair approximation of the element size required to resolve the current density distribution.

Strictly speaking, the skin depth is given by $\delta = 1/\mathrm{Re}[(j\omega\mu(\sigma + j\omega\varepsilon))^{1/2}]$, which reduces to $\delta = 1/\mathrm{Re}[(j\omega\mu\sigma)^{1/2}]$ when displacement currents are negligible compared to conduction currents. In literature this is typically simplified further, to $\sqrt{(2/(\omega\mu\sigma))}$. We will not do so here however, as μ_{arm} contains both real and imaginary parts.

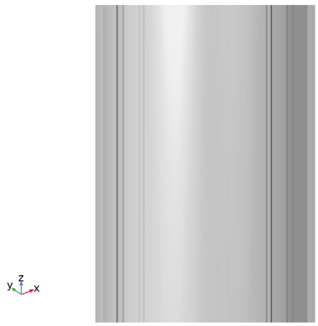
Please remember that these expressions are strictly speaking valid for infinite flat conductors only. For round conductors the current density does not follow an exponential decay, but instead can be described by the Bessel function of the first kind: $J_0(r)$. As an approximation, the values for `Dscup`, `Dspbs`, and `Dsarm` should still suffice however. The `min()` operator is used as a precaution to set an upper limit for the skin depth — otherwise, the mesh elements might not fit in the conductor any more. For more on electromagnetic theory and material properties, see the *Inductive Effects 3D* tutorial.

Now that `Nper` has been set, it is time to rebuild the dummy geometry. Proceed by switching views.

GEOMETRY 1

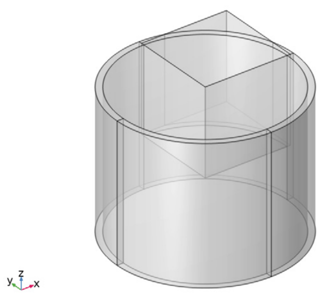
1 In the **Model Builder** window, under **Component 1 (comp1)** click **Geometry 1**.

2 Click the **Go to View 1 (Orthographic)** button in the **Graphics** toolbar.



At first, the geometry looks stretched. This is because the camera is using the latest parameter values while the geometry has not yet been rebuilt.

3 In the **Home** toolbar, click **Build All**.



The camera scaling for **View 1** has been restored to unity (this is given away by the cube), and the dummy cable is indeed as short as it looks like.

4 In the **Model Builder** window, expand the **Geometry 1** node.

Block 1 (blk1), Cylinder 1 (cyl1)

1 In the **Model Builder** window, under **Component 1 (comp1)>Geometry 1**, Ctrl-click to select **Cylinder 1 (cyl1)** and **Block 1 (blk1)**.

2 Right-click and choose **Delete**.

GEOMETRY I

Before we start building the cable geometry, let us reflect on how the COMSOL geometry sequence works. Although COMSOL supports a 2D sketch mode nowadays (that behaves more like your typical CAD tool), most COMSOL geometries are still built as a sequence of operations. The main difference when compared to most 3D drawing and sketch tools, is that the focus is not on the end result. The focus is on *how you get there*.

Take deletion of a boundary or edge for example. Instead of clicking the edge in the graphics window with some kind of "deletion tool", you will have to add a **Delete** operation to the geometry sequence and tell it to delete that particular edge for you. At first, you might perceive this as cumbersome. There is one very big advantage however; you can "program" the sequence of operations leading to the end result. This allows for easy parameterization.

Furthermore, the geometry sequence is automatically your editing history. If you change your mind and need to make changes that affect the geometry at an early stage, you can just do so (instead of having to modify its final state). From that viewpoint, you should look at the geometry sequence as a *building script*, rather than a live editing tool. This also implies you can test certain operations individually, before putting them together to create something more sophisticated.

As pointed out in the previous section, the cable geometry takes more time to build and render in its long twisted state. The strategy therefore, is to build and test a short fully parameterized cable first, and then — when everything seems to be in order — let it switch to its true form. This will save you a lot of time.

Proceed by adding a ring of polygons for the **Cable Armor, Centerline** selection.

Polygon 1 (pol1)

1 In the **Geometry** toolbar, click **More Primitives** and choose **Polygon**.

2 In the **Settings** window for **Polygon**, locate the **Coordinates** section.

3 From the **Data source** list, choose **Vectors**.

4 In the **x** text field, type `Darm/2`.

5 In the **y** text field, type `0`.

6 In the **z** text field, type `-(Lsec/2+Tarm*{0,10})`.

Note that for longer expressions like this one, *the easiest way to go, is to copy-paste them directly from this *.pdf file to COMSOL.*

7 Click **Build Selected**.

Here, you have used a “vector syntax” to describe the points of the polygon. The expression {0,10} basically means; do this for 0, and 10. So -(Lsec/2+Tarm*{0,10}) evaluates to -Lsec/2, and -Lsec/2-10*Tarm. It is a line with two vertices, both of them having x,y-coordinates equal to $D_{\mathrm{arm}}/2,0$. The first vertex is positioned at z-coordinate $-L_{\mathrm{sec}}/2$, making it touch the bottom periodicity plane. The second one is $10T_{\mathrm{arm}}$ further down.

Rotate 1 (rot1)

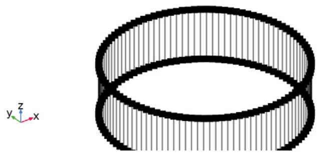
1 In the **Geometry** toolbar, click **Transforms** and choose **Rotate**.

2 Select the object **pol1** only.

3 In the **Settings** window for **Rotate**, locate the **Rotation** section.

4 In the **Angle** text field, type 360[deg]*range(1/Narm,1/Narm,1).

5 Click **Build Selected**.



The `range()` operator is used for creating a list of equally spaced values. The expression `range(1/Narm,1/Narm,1)` should be interpreted as `range(start,step,end)`, and evaluates to (in vector syntax): `{1/Narm, 2/Narm, 3/Narm, ..., 1}`. As a result, the **Rotate** operator creates N_{arm} equally spaced copies of **Polygon 1 (pol1)**, forming a full circle. There is a whole list of operators available, see the section *Operators, Functions, and Constants* in the COMSOL Multiphysics Reference Manual.

This ring does not form an integral part of the cable or anything, it is merely there to accommodate the selections for the armor (see section Modeling Instructions — Selections). The main body of the cable will be constructed by drawing parts of the cross section on **Work Planes** in 3D, and then sweeping those planes along a polygon. The polygon then serves as a "spine": During the sweep, the cross section may twist around it. By using a parameterized expression, the twist can be enabled or disabled.

Start by adding a work plane for the phases and screens.

Phases and Screens

1 In the **Geometry** toolbar, click **Work Plane**.

2 In the **Settings** window for **Work Plane**, type `Phases and Screens` in the **Label** text field.

3 Locate the **Plane Definition** section. In the **z-coordinate** text field, type `-Lsec/2`.

4 Locate the **Selections of Resulting Entities** section. Select the **Resulting objects selection** check box.

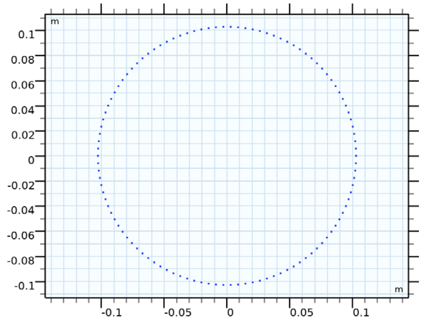
5 From the **Show in physics** list, choose **Off**.

The combination of these last two settings will make the work plane available for input selections of subsequent operations, while hiding it outside the geometry sequence — it will not show up as a boundary selection in the physics, for example.

Phases and Screens (wp1)>Plane Geometry

1 In the **Model Builder** window, click **Plane Geometry**.

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



The phase conductors will be drawn as *icosagons* (twenty-sided regular polygons), instead of circles. The reasoning is that, as this geometry will be used together with linear elements (see the *Inductive Effects 3D* tutorial), the circular shapes of the phases will turn into polygons anyway. By adding them as polygons in the geometry already, you can precisely dictate what kind of polygon it will be, and you can compensate for truncation effects.

Phases and Screens (wp1)>Polygon 1 (pol1)

1 In the **Work Plane** toolbar, click **Polygon**.

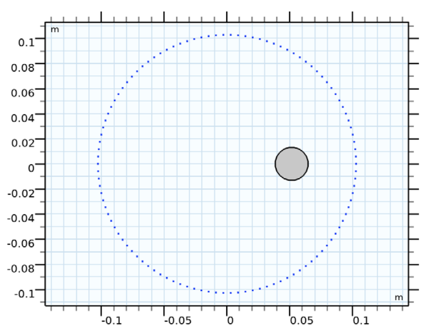
2 In the **Settings** window for **Polygon**, locate the **Coordinates** section.

3 From the **Data source** list, choose **Vectors**.

4 In the **xw** text field, type `Dpha/sqrt(3)+cos(360[deg]*range(1/40,1/20,1-1/40))*TCFp20*Dcon/2`.

5 In the **yw** text field, type `sin(360[deg]*range(1/40,1/20,1-1/40))*TCFp20*Dcon/2`.

6 Click **Build Selected**.



Here, the expression `360[deg]*range(1/40,1/20,1-1/40)` defines 20 equidistant angles. These angles are then fed into the `cos()` and `sin()` operators, to get 20 equidistant points on the unit circle. After multiplication by `Dcon/2` (the main conductor radius), the polygon has the appropriate shape and size. Finally, `Dpha/sqrt(3)` is added to all x-coordinates to move the shape to the desired location.

The multiplication by `TCFp20` is a compensation for the truncation that occurs when going from circle to icosagon. The surface area of a circle is given by πr^2. For an icosagon it is $5(\sqrt{5}-1)r^2/2$ instead, where r refers to the radius of the *circumcircle*. The truncation correction factor sees to it that the icosagon has the correct surface area (see section Truncation Correction Factors).

Phases and Screens (wp1)>Circle 1 (c1)

1 In the **Work Plane** toolbar, click **Circle**.

2 In the **Settings** window for **Circle**, locate the **Object Type** section.

3 From the **Type** list, choose **Curve**.

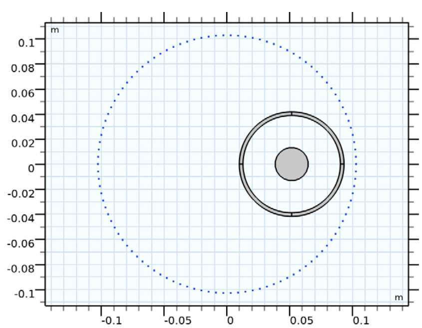
4 Locate the **Size and Shape** section. In the **Radius** text field, type `Dins/2+Tpbs`.

5 Locate the **Position** section. In the **xw** text field, type `Dpha/sqrt(3)`.

6 Click to expand the **Layers** section. In the table, enter the following settings:

Layer name	Thickness (m)
Layer 1	Tpbs

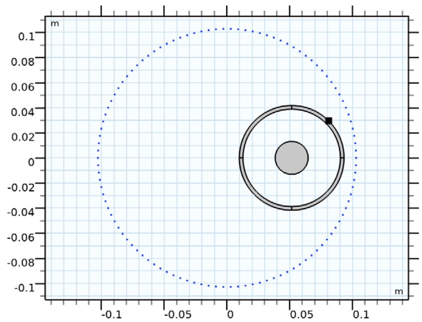
7 Click **Build Selected**.



For the screens, truncation should not be that much of an issue. First of all, the number of sides of the screen polygons will be larger — 32, with the current mesh settings — resulting in a smaller truncation error. Secondly, the cross section of the screen looks like a ring rather than a solid circle. So any truncation that occurs on the outer perimeter, will occur on the inner perimeter too (and more or less to the same degree). Therefore, drawing the screens as hollow circles should have little effect on the model's accuracy.

Phases and Screens (wp1)>Point 1 (pt1)

1 In the **Work Plane** toolbar, click **Point**.
2 In the **Settings** window for **Point**, locate the **Point** section.
3 In the **xw** text field, type `Dpha/sqrt(3)+(Dins/2+Tpbs)/sqrt(2)`.
4 In the **yw** text field, type `(Dins/2+Tpbs)/sqrt(2)`.
5 Click **Build Selected**.



This single point may not look like much, but it serves a crucial purpose. One of the corner stones of the models discussed in the *Inductive Effects 3D* tutorial, is the **Periodic Condition**. And for the periodic condition to work reliably, you will need *conforming meshes* (see section Conforming Meshes).

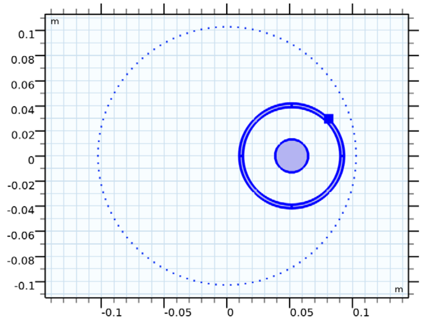
This point has been put in such a location — on the outer perimeter of the screen, at an angle of 45° — that it breaks not only the three-fold rotational symmetry of the phases and screens, but also any kind of mirror symmetry. As a consequence, when the **Copy Face 3** mesh operation performs its topology analysis to determine how the mesh should be copied from source to destination, it will encounter a shape with no symmetries and therefore, no ambiguity in the required mapping.

In short; there will be only one unique way in which the boundary mesh can be copied and it will be the right one. We will get back to this, in section Modeling Instructions — Mesh Sequence. Next, will be a *slant correction*.

Phases and Screens (wp1)>Scale 1 (sca1)

1 In the **Work Plane** toolbar, click **Transforms** and choose **Scale**.

2 Click in the **Graphics** window and then press Ctrl+A to select all objects.

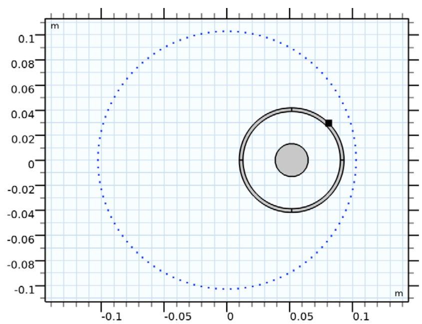


3 In the **Settings** window for **Scale**, locate the **Scale Factor** section.

4 From the **Scaling** list, choose **Anisotropic**.

5 In the **yw** text field, type `SCFpha`.

6 Click **Build Selected**.

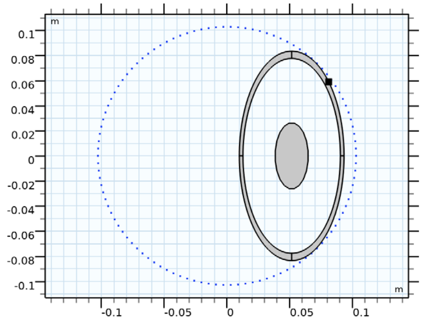


It looks like nothing is happening. That is because `SCFpha` evaluates to “`1`” at this point. The definition for `SCFpha` in **Geometric Parameters 3** contains an `if()` statement that explicitly checks whether there is a twist. When the cable is in its short extruded state there is no twist, and therefore, no slanted cut.

Still, you probably would like to check whether this part of your geometry sequence works as intended. Add a multiplication by 2 temporarily, to check that the anisotropic scaling works.

7 In the **yw** text field, type `2*SCFpha`.

8 Click **Build Selected**.

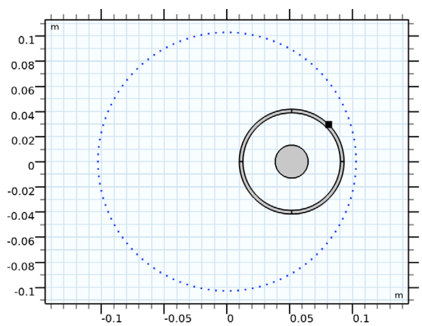


Note that the slant correction is only valid within certain bounds. A factor of 2 means your phase conductor and screen are cut at an angle of 30°. At that angle, the cross-sectional cut of the helical conductors should start to look like a “C-shape” rather than an ellipse. You can see this, because the screen is starting to intersect with the armor (for more on this, see section Slant Correction Factors).

When the twist becomes active, `SCFpha` will obtain a value of 1.0044, well within reasonable bounds. Restore the expression for the anisotropic scaling before proceeding with the other two phases.

9 In the **yw** text field, type `SCFpha`.

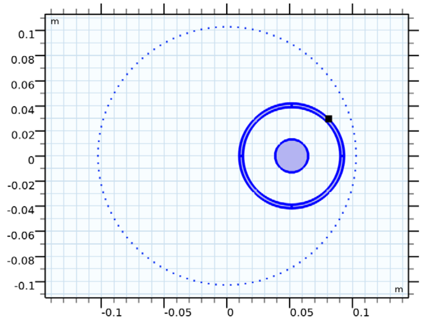
10 Click **Build Selected**.



Phases and Screens (wp1)>Rotate 1 (rot1)

1 In the **Work Plane** toolbar, click **Transforms** and choose **Rotate**.

2 Select the objects **sca1(1)** and **sca1(2)** only.

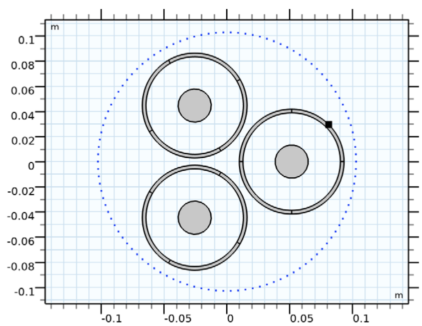


Make sure to not select the point; just the phase and screen. If you would select the point along with it, the result would still have three-fold rotational symmetry.

3 In the **Settings** window for **Rotate**, locate the **Rotation** section.

4 In the **Angle** text field, type `360[deg]*{1/3,2/3,1}`.

5 Click **Build Selected**.



The used vector syntax causes two duplicates to appear, at angles 120°, and 240°. You could have achieved the same effect with other expressions, including (but not limited to): `2*pi[rad]*range(1/3,1/3,1)`, or `{120[deg],240[deg],360[deg]}`, or space-separated: `0 120 240`.

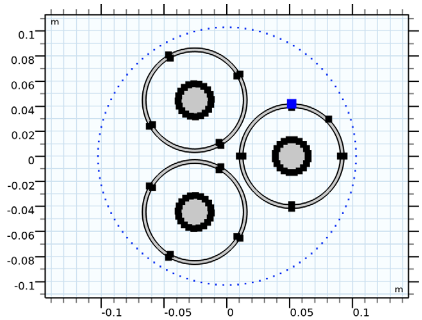
Together with the screens, the three phases form a "central core". This core is twisted according to the phase's lay length `LLpha`, and will be meshed using a **Swept** mesh. The armor (and the exterior of the cable) is twisted in the opposite direction and meshed with a swept mesh as well. In-between the core and the armor there is a void region where no swept mesh can be used: The helical swept meshes will have to be glued together with a **Free Tetrahedral** mesh (see section Swept Meshes and Tetrahedral Meshes).

The resulting tetrahedra will not be of a particularly good quality though, and they will introduce more *degrees of freedom* (DOFs) per unit volume than the prisms and hexahedra in the swept meshes will. It is therefore advantageous to include the space between the phases and screens in the central swept mesh. For this, you need a few interior boundaries. You can connect the screens with three line segments.

Phases and Screens (wp1)>Line Segment 1 (ls1)

1 In the **Work Plane** toolbar, click **More Primitives** and choose **Line Segment**.

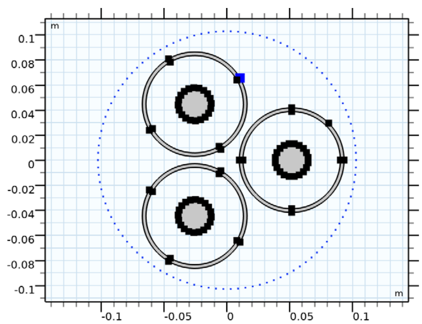
2 On the object **rot1(3)**, select Point 6 only.



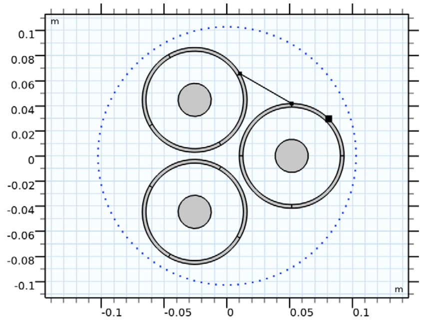
3 In the **Settings** window for **Line Segment**, locate the **Endpoint** section.

4 Find the **End vertex** subsection. Click to select the **Activate Selection** toggle button.

5 On the object **rot1(1)**, select Point 3 only.



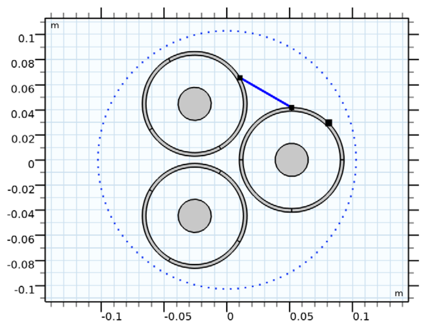
6 Click **Build Selected**.



Phases and Screens (wp1)>Rotate 2 (rot2)

1 In the **Work Plane** toolbar, click **Transforms** and choose **Rotate**.

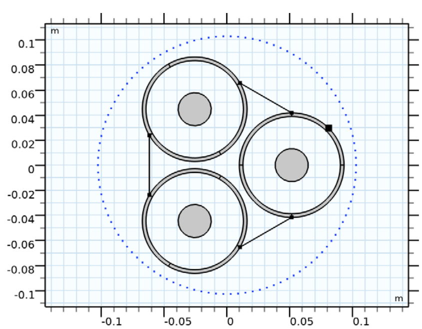
2 Select the object **ls1** only.



3 In the **Settings** window for **Rotate**, locate the **Rotation** section.

4 In the **Angle** text field, type `360[deg]*{1/3,2/3,1}`.

5 Click **Build Selected**.

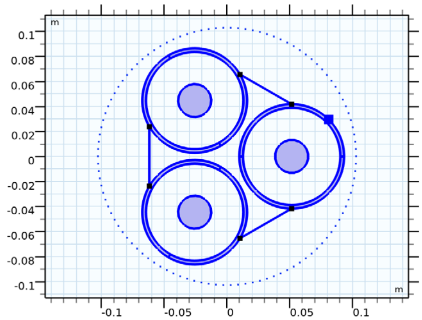


Now that you have closed off the central area, you can unify the core to one solid object:

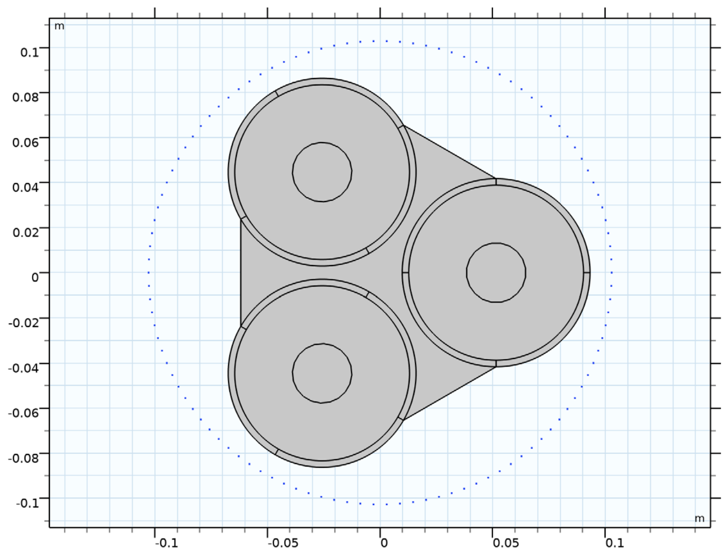
Phases and Screens (wp1)>Convert to Solid 1 (csol1)

1 In the **Work Plane** toolbar, click **Conversions** and choose **Convert to Solid**.

2 Click in the **Graphics** window and then press Ctrl+A to select all objects.



3 In the **Settings** window for **Convert to Solid**, click **Build Selected**.

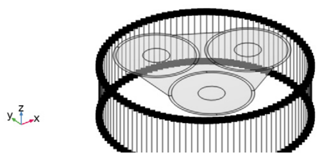


The resulting object will be extruded to become the central part of the cable. You have now finished your efforts on the first work plane. Let us proceed to the next one.

Phases and Screens (wp1)

1 In the **Model Builder** window, under **Component 1 (comp1)>Geometry 1** click **Phases and Screens (wp1)**.

2 In the **Settings** window for **Work Plane**, click **Build Selected**.



3 In the **Model Builder** window, collapse the **Phases and Screens (wp1)** node.

Cable Armor and Sea Bed

1 In the **Geometry** toolbar, click **Work Plane**.

2 In the **Settings** window for **Work Plane**, type `Cable Armor and Sea Bed` in the **Label** text field.

3 Locate the **Plane Definition** section. In the **z-coordinate** text field, type `-Lsec/2`.

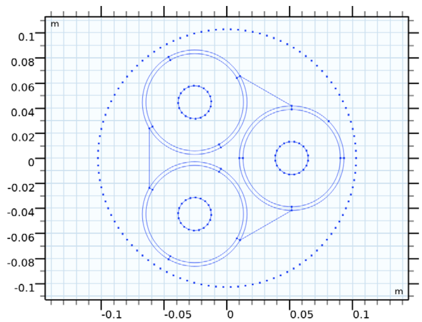
4 Locate the **Selections of Resulting Entities** section. Select the **Resulting objects selection** check box.

5 From the **Show in physics** list, choose **Off**.

Cable Armor and Sea Bed (wp2)>Plane Geometry

1 In the **Model Builder** window, click **Plane Geometry**.

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



To start with, you can add the armor wire centerline. This centerline will come in handy when creating the armor selections later on (and when evaluating the net longitudinal current density in the next tutorial). In addition to this, the armor mesh should preferably have a nice regular shape to better resolve the expected eddies. To this end, let us create a small "plus mark" at the center of the first armor wire.

Cable Armor and Sea Bed (wp2)>Polygon 1 (pol1)

1 In the **Work Plane** toolbar, click **Polygon**.

2 In the **Settings** window for **Polygon**, locate the **Object Type** section.

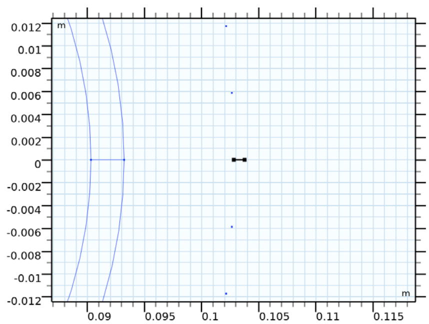
3 From the **Type** list, choose **Open curve**.

4 Locate the **Coordinates** section. From the **Data source** list, choose **Vectors**.

5 In the **xw** text field, type `Darm/2+TCFp16*Tarm/6*{0,1}`.

6 In the **yw** text field, type `0`.

7 Click **Build Selected**.

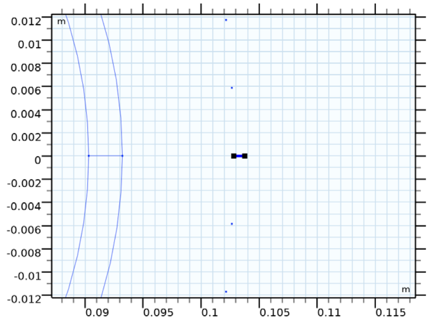


The first vertex of this polygon coincides with the center of the armor wire (at *x*,*y*-coordinate $D_{\mathrm{arm}}/2,0$), and the second one is `TCFp16*Tarm/6` further to the right. As the *hexadecagon* that is to be used for the armor wire will have a circumradius of `TCFp16*Tarm/2`, the plus mark will cover one third of the polygon radius. This will, more or less, dictate the mesh element size at the wire center.

Cable Armor and Sea Bed (wp2)>Rotate 1 (rot1)

1 In the **Work Plane** toolbar, click **Transforms** and choose **Rotate**.

2 Select the object **pol1** only.

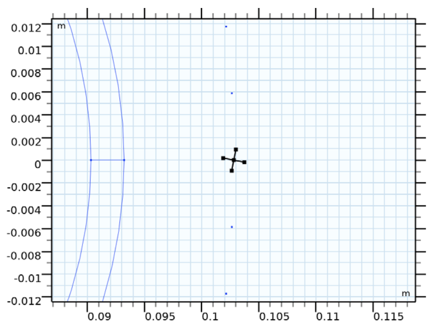


3 In the **Settings** window for **Rotate**, locate the **Rotation** section.

4 In the **Angle** text field, type `360[deg]*range(7/32,1/4,1-1/32)`.

5 Locate the **Center of Rotation** section. In the **xw** text field, type `Darm/2`.

6 Click **Build Selected**.



The plus is put at a slight angle, so that it will coincide with the spokes of the polygon.

Cable Armor and Sea Bed (wp2)>Polygon 2 (pol2)

1 In the **Work Plane** toolbar, click **Polygon**.

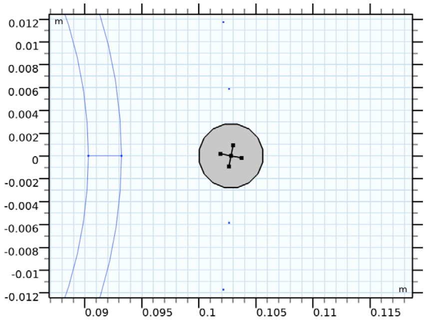
2 In the **Settings** window for **Polygon**, locate the **Coordinates** section.

3 From the **Data source** list, choose **Vectors**.

4 In the **xw** text field, type `Darm/2+cos(360[deg]*range(1/32,1/16,1-1/32))*TCFp16*Tarm/2`.

5 In the **yw** text field, type `sin(360[deg]*range(1/32,1/16,1-1/32))*TCFp16*Tarm/2`.

6 Click **Build Selected**.



Notice that this polygon is basically a small copy of **Polygon 1 (pol1)** from the first work plane. The syntax in the expressions is the same.

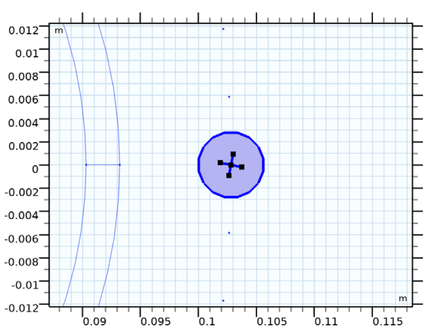
The number of sides is smaller though, causing a larger truncation correction factor: 1.013. If you would omit this factor in the expression for the polygon, the surface area would be 1.3% smaller than $\pi(T_{\mathrm{arm}}/2)^2$, the area of the circle that it approximates. Likewise, if you would draw real circles in your geometry — *with the correct surface area* — and subsequently mesh them to become hexadecagons (to be used together with first-order elements), you would get the same effect.

In most cases one would not bother about such details, but since the cable armor is of particular interest — and since the armor wire's resistive (and "reluctive") properties scale with its cross-sectional surface area — it is good to have a correction here. Besides, the improvement in accuracy virtually comes for free. It does not require finer meshes or stricter solver settings. A very similar reasoning holds for `SCFarm`, the slant correction factor:

Cable Armor and Sea Bed (wp2)>Scale 1 (sca1)

1 In the **Work Plane** toolbar, click **Transforms** and choose **Scale**.

2 Click in the **Graphics** window and then press Ctrl+A to select all objects.

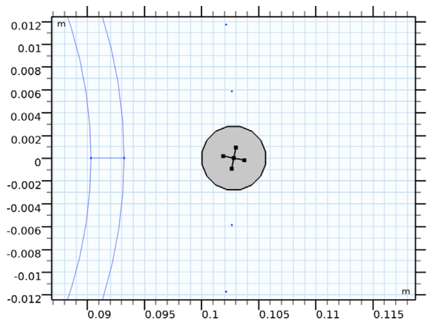


3 In the **Settings** window for **Scale**, locate the **Scale Factor** section.

4 From the **Scaling** list, choose **Anisotropic**.

5 In the **yw** text field, type `SCFarm`.

6 Click **Build Selected**.



Like before, you can add a multiplication by 2 temporarily, to check that the anisotropic scaling works as intended — not included in the following instructions, but *left as an exercise to the reader*.

Next, add two more polygons to separate the armor wires from the rest of the geometry.

Cable Armor and Sea Bed (wp2)>Polygon 3 (pol3)

1 In the **Work Plane** toolbar, click **Polygon**.

2 In the **Settings** window for **Polygon**, locate the **Object Type** section.

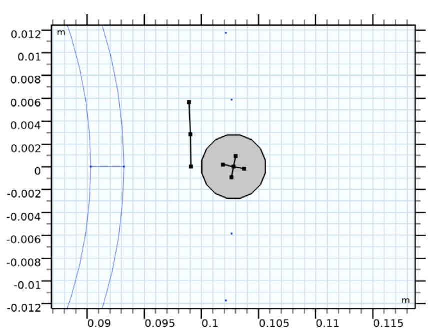
3 From the **Type** list, choose **Open curve**.

4 Locate the **Coordinates** section. From the **Data source** list, choose **Vectors**.

5 In the **xw** text field, type `cos(360[deg]/Narm*{0,1/2,1})*(Darm/2-2*Tarm/3)`.

6 In the **yw** text field, type `sin(360[deg]/Narm*{0,1/2,1})*(Darm/2-2*Tarm/3)`.

7 Click **Build Selected**.



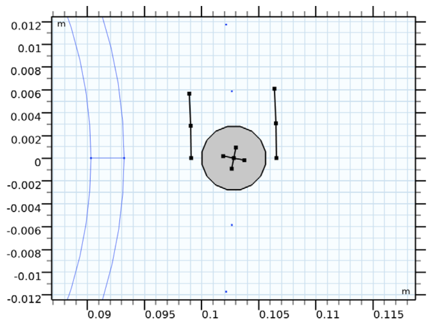
Cable Armor and Sea Bed (wp2)>Polygon 4 (pol4)

1 Right-click **Component 1 (comp1)>Geometry 1>Cable Armor and Sea Bed (wp2)>Plane Geometry>Polygon 3 (pol3)** and choose **Duplicate**.

2 In the **Settings** window for **Polygon**, locate the **Coordinates** section.

3 In the **xw** and **yw** text fields, update the expression. Type `...*(Darm/2+2*Tarm/3)`, that is; replace “`-2*Tarm/3`” with “`+2*Tarm/3`” to increase the radius.

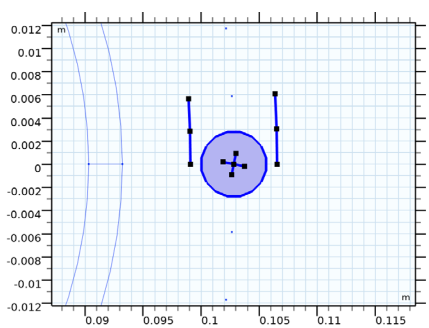
4 In the **Settings** window for **Polygon**, click **Build Selected**.



Cable Armor and Sea Bed (wp2)>Rotate 2 (rot2)

1 In the **Work Plane** toolbar, click **Transforms** and choose **Rotate**.

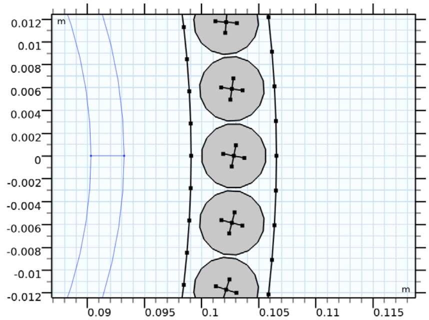
2 Click in the **Graphics** window and then press Ctrl+A to select all objects.



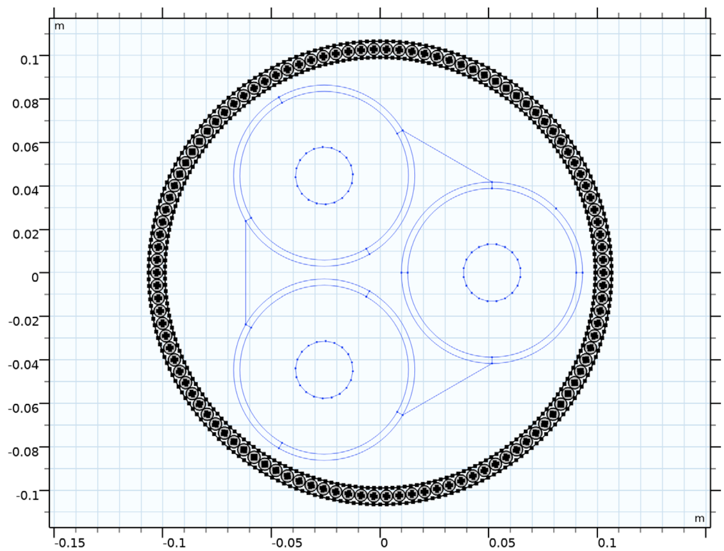
3 In the **Settings** window for **Rotate**, locate the **Rotation** section.

4 In the **Angle** text field, type `360[deg]*range(1/Narm,1/Narm,1)`.

5 Click **Build Selected**.



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



The armor wires are now sandwiched between two circles. The main purpose of these circles is to help out with the mesh. Both the inner- and outer circle will force a rapid growth in element size. In addition to this, the inner circle will serve as an interface between the outer swept mesh and the tetrahedra.

Cable Armor and Sea Bed (wp2)>Circle 1 (c1)

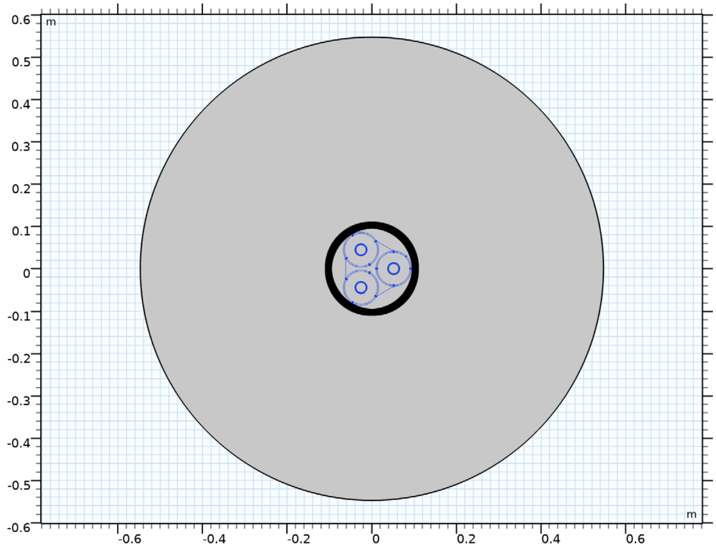
1 In the **Work Plane** toolbar, click **Circle**.

2 In the **Settings** window for **Circle**, locate the **Size and Shape** section.

3 In the **Radius** text field, type `5*Dcab/2`.

4 Click **Build Selected**.

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



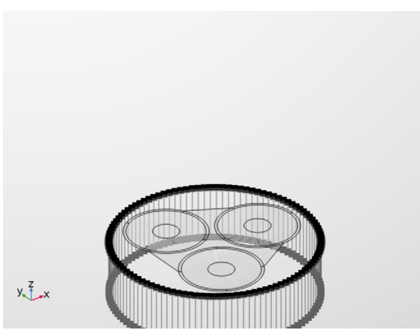
This circle represents the direct environment of the cable — magnetic insulation will be applied to its exterior boundaries. Its size has been determined by sweeping over (that is; *varying*) the radius in 2D models, and checking when the influence of the magnetic insulation on the cable's lumped quantities becomes "negligible".

You have now finished your efforts on the second work plane. Proceed to the last one.

Cable Armor and Sea Bed (wp2)

1 In the **Model Builder** window, under **Component 1 (comp1)>Geometry 1** click **Cable Armor and Sea Bed (wp2)**.

2 In the **Settings** window for **Work Plane**, click **Build Selected**.



3 In the **Model Builder** window, collapse the **Cable Armor and Sea Bed (wp2)** node.

Mesh Control Entities

1 In the **Geometry** toolbar, click **Work Plane**.

2 In the **Settings** window for **Work Plane**, type `Mesh Control Entities` in the **Label** text field.

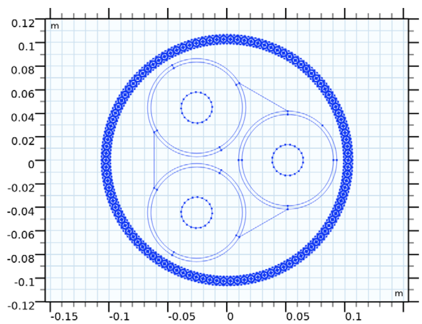
3 Locate the **Plane Definition** section. In the **z-coordinate** text field, type `-Lsec/2`.

Mesh Control Entities (wp3)>Plane Geometry

1 In the **Model Builder** window, click **Plane Geometry**.

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

3 Click the **Zoom In** button in the **Graphics** toolbar, twice.



This work plane is different from the other two. It will not we "swept" nor extruded; it will not add boundaries or domains to the geometry. Its sole purpose is to accommodate the construction of a precisely defined, highly optimized surface mesh (before it is swept over the rest of the geometry using swept meshes).

Three geometric features are added. The first one consist of a number or polygons and line segments that allows for a manually constructed boundary layer mesh in the phases (using a mapped mesh, see Boundary Layer Meshes). After that, points and line segments are used to bridge the gaps between the armor wires. Here too, mapped meshes are used, see section Meshing the Gaps. Finally, more boundary layers are added to the armor wires — along with some polygons helping out with the mesh in the wire center.

You can start by adding the first polygon.

Mesh Control Entities (wp3)>Polygon 1 (pol1)

1 In the **Work Plane** toolbar, click **Polygon**.

2 In the **Settings** window for **Polygon**, locate the **Object Type** section.

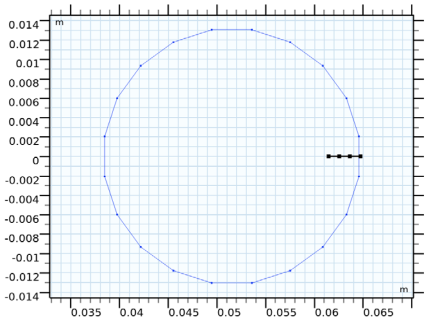
3 From the **Type** list, choose **Open curve**.

4 Locate the **Coordinates** section. From the **Data source** list, choose **Vectors**.

5 In the **xw** text field, type `Dpha/sqrt(3)+TCFp20*Dcon/2-Dscup/8*{0,1,2,3}`.

6 In the **yw** text field, type `0`.

7 Click **Build Selected**.

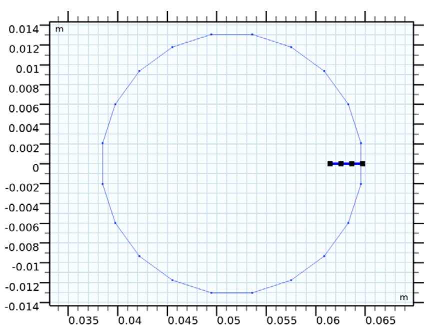


Notice that this polygon is defined in terms of phase dimensions and skin depth: The expression `Dpha/sqrt(3)+TCFp20*Dcon/2` is the phase location, plus the truncation-corrected radius. The expression `-Dscup/8*{0,1,2,3}` creates four equidistant points, each $\delta_{cup}/8$ apart.

Mesh Control Entities (wp3)>Rotate 1 (rot1)

1 In the **Work Plane** toolbar, click **Transforms** and choose **Rotate**.

2 Select the object **pol1** only.

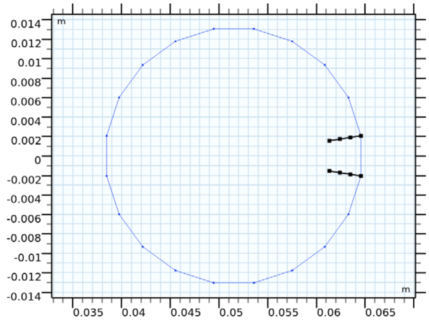


3 In the **Settings** window for **Rotate**, locate the **Rotation** section.

4 In the **Angle** text field, type `360[deg]*{-1/40,1/40}`.

5 Locate the **Center of Rotation** section. In the **xw** text field, type `Dpha/sqrt(3)`.

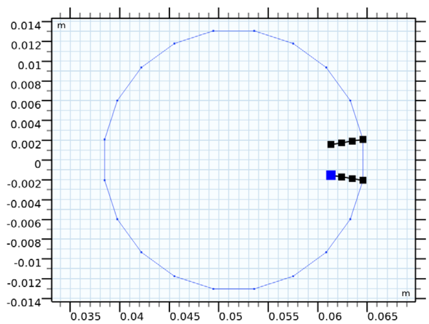
6 Click **Build Selected**.



Mesh Control Entities (wp3)>Line Segment 1 (ls1)

1 In the **Work Plane** toolbar, click **More Primitives** and choose **Line Segment**.

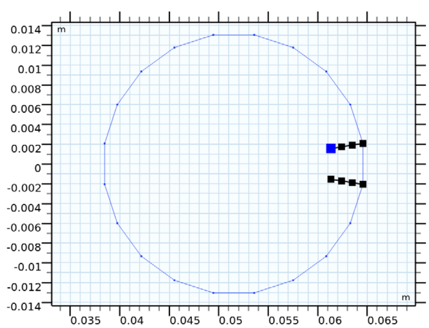
2 On the object **rot1(1)**, select Point 1 only.



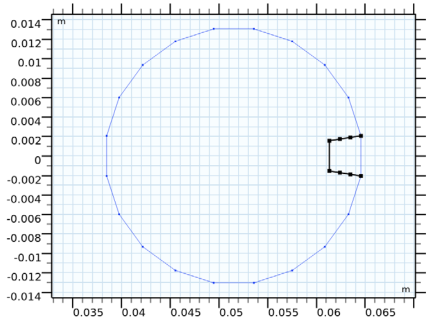
3 In the **Settings** window for **Line Segment**, locate the **Endpoint** section.

4 Find the **End vertex** subsection. Click to select the **Activate Selection** toggle button.

5 On the object **rot1(2)**, select Point 1 only.



6 Click **Build Selected**.



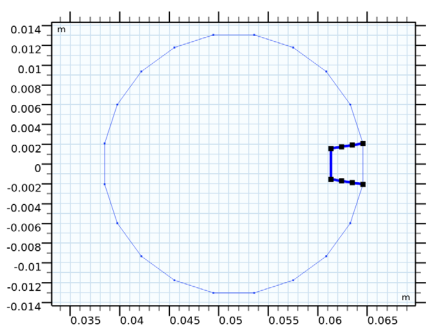
This shape could actually have been made as one single polygon. Still, we chose to have a polygon with four points on the line $y = 0$, create two rotated duplicates, and connect those duplicates with a line segment.

Note that this is not by any means the "best" or "only" way to achieve this kind of result. The chosen strategy simply seems to give a fair compromise between the complexity of the used expressions, and the number of required operations. Please keep in mind when building geometries, you will often have three or four alternative routes — typically with minor advantages, or disadvantages. The same holds for the following operations:

Mesh Control Entities (wp3)>Rotate 2 (rot2)

1 In the **Work Plane** toolbar, click **Transforms** and choose **Rotate**.

2 Click in the **Graphics** window and then press Ctrl+A to select all objects.

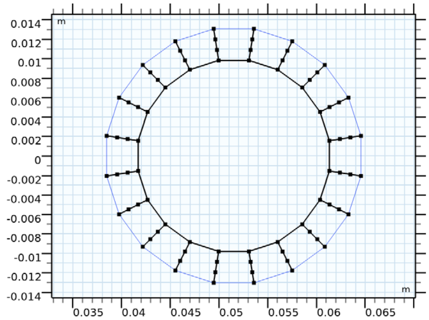


3 In the **Settings** window for **Rotate**, locate the **Rotation** section.

4 In the **Angle** text field, type `360[deg]*range(1/20,1/20,1)`.

5 Locate the **Center of Rotation** section. In the **xw** text field, type `Dpha/sqrt(3)`.

6 Click **Build Selected**.



Mesh Control Entities (wp3)>Polygon 2 (pol2)

1 In the **Work Plane** toolbar, click **Polygon**.

2 In the **Settings** window for **Polygon**, locate the **Object Type** section.

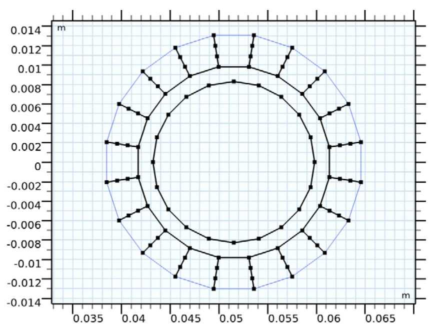
3 From the **Type** list, choose **Closed curve**.

4 Locate the **Coordinates** section. From the **Data source** list, choose **Vectors**.

5 In the **xw** text field, type `Dpha/sqrt(3)+cos(360[deg]*range(1/20,1/20,1))*(TCFp20*Dcon/2-Dscup*9/16)`.

6 In the **yw** text field, type `sin(360[deg]*range(1/20,1/20,1))*(TCFp20*Dcon/2-Dscup*9/16)`.

7 Click **Build Selected**.



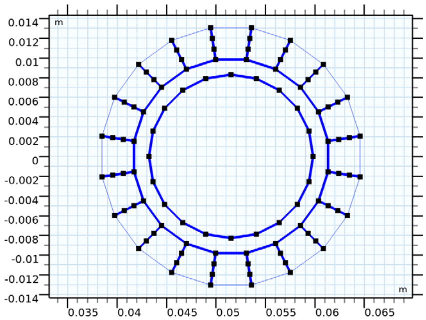
The resulting polygon will cause a layer of "flat" triangles, which represents a transition zone between the boundary layers and the "equilateral" triangles in the phase center.

To create a proper overlay between these mesh control entities and the original shape **Phases and Screens (wp1)>Polygon 1 (pol1)**, they must be subjected to the same transformations and correction factors. The truncation correction factor has already been included. The slant correction is added next.

Mesh Control Entities (wp3)>Scale 1 (sca1)

1 In the **Work Plane** toolbar, click **Transforms** and choose **Scale**.

2 Click in the **Graphics** window and then press Ctrl+A to select all objects.

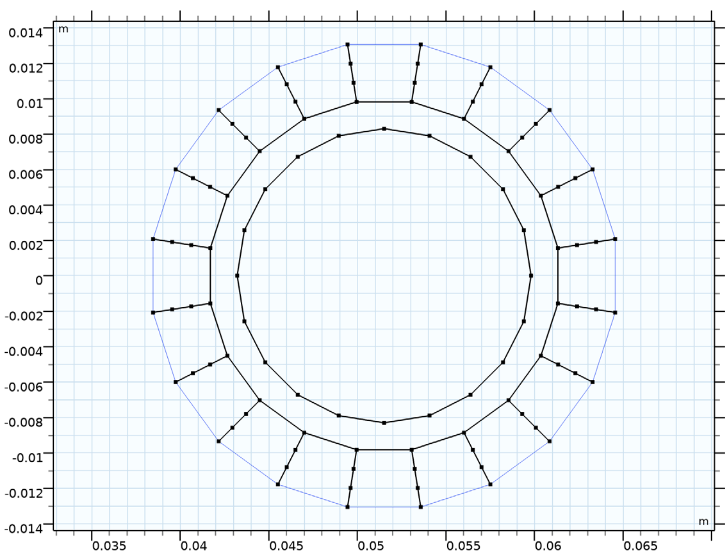


3 In the **Settings** window for **Scale**, locate the **Scale Factor** section.

4 From the **Scaling** list, choose **Anisotropic**.

5 In the **yw** text field, type `SCFpha`.

6 Click **Build Selected**.



Now, a similar reasoning holds for the mesh control edges that are to be added between the armor wires. An easy way to construct them is to add a point, and then subject that point to all transformations needed to let it coincide with the points of the armor wire;

Cable Armor and Sea Bed (wp2)>Polygon 2 (pol2). The resulting points are then connected using line segments to bridge the gap.

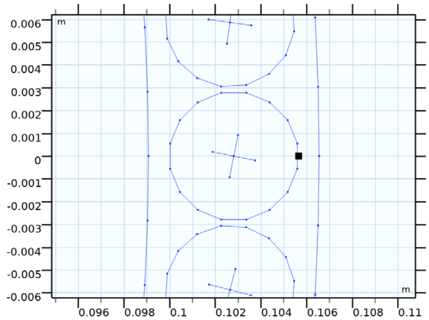
Mesh Control Entities (wp3)>Point 1 (pt1)

1 In the **Work Plane** toolbar, click **Point**.

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

3 In the **xw** text field, type `Darm/2+TCFp16*Tarm/2`.

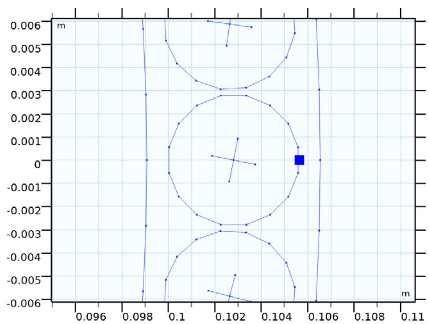
4 Click **Build Selected**.



Mesh Control Entities (wp3)>Rotate 3 (rot3)

1 In the **Work Plane** toolbar, click **Transforms** and choose **Rotate**.

2 Select the object **pt1** only.



3 In the **Settings** window for **Rotate**, locate the **Rotation** section.

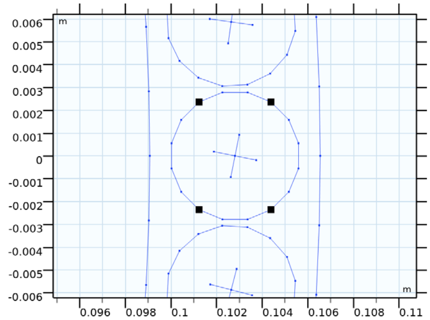
4 In the **Angle** text field, type `360[deg]*{5/32,11/32,21/32,27/32}`.

5 Locate the **Center of Rotation** section. In the **xw** text field, type `Darm/2`.

6 Locate the **Selections of Resulting Entities** section. Select the **Resulting objects selection** check box.

7 From the **Show in 3D** list, choose **Off**.

8 Click **Build Selected**.

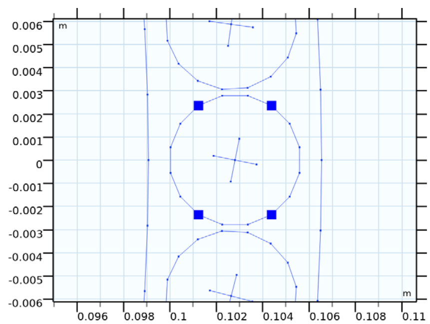


Mesh Control Entities (wp3)>Scale 2 (sca2)

1 In the **Work Plane** toolbar, click **Transforms** and choose **Scale**.

2 In the **Settings** window for **Scale**, locate the **Input** section.

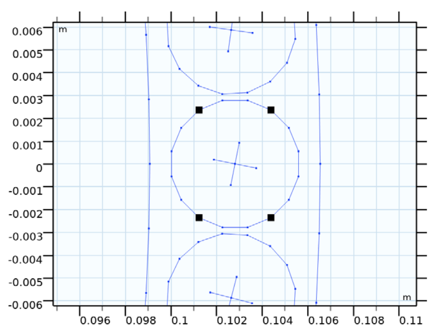
3 From the **Input objects** list, choose **Rotate 3**.



4 Locate the **Scale Factor** section. From the **Scaling** list, choose **Anisotropic**.

5 In the **yw** text field, type `SCFarm`.

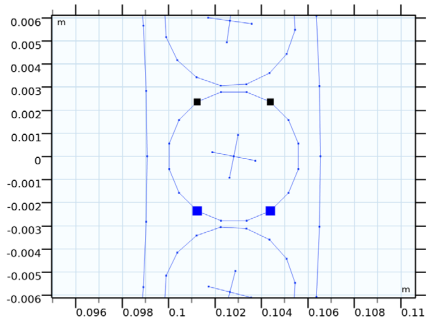
6 Click **Build Selected**.



Mesh Control Entities (wp3)>Rotate 4 (rot4)

1 In the **Work Plane** toolbar, click **Transforms** and choose **Rotate**.

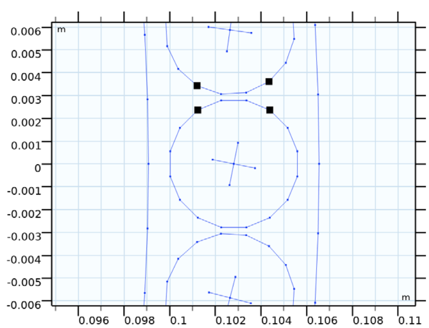
2 Select the objects **sca2(3)** and **sca2(4)** only.



3 In the **Settings** window for **Rotate**, locate the **Rotation** section.

4 In the **Angle** text field, type `360[deg]/Narm`.

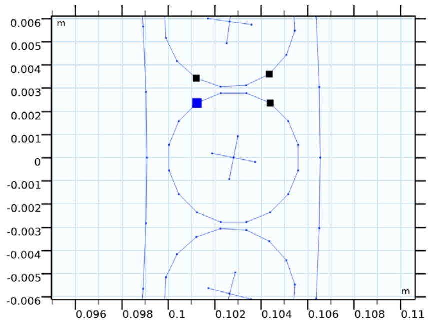
5 Click **Build Selected**.



Mesh Control Entities (wp3)>Line Segment 2 (ls2)

1 In the **Work Plane** toolbar, click **More Primitives** and choose **Line Segment**.

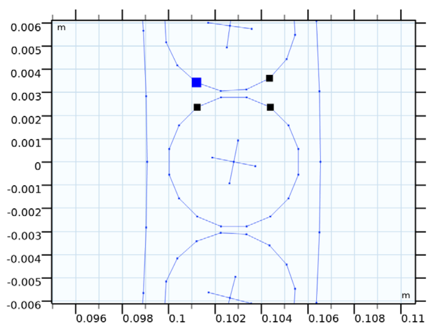
2 On the object **sca2(2)**, select Point 1 only.



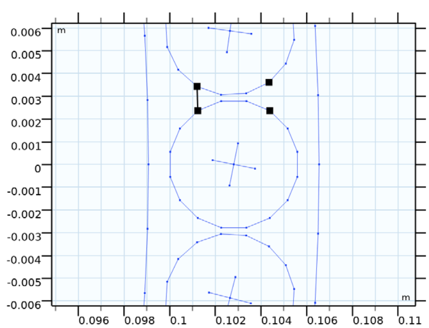
3 In the **Settings** window for **Line Segment**, locate the **Endpoint** section.

4 Find the **End vertex** subsection. Click to select the **Activate Selection** toggle button.

5 On the object **rot4(1)**, select Point 1 only.



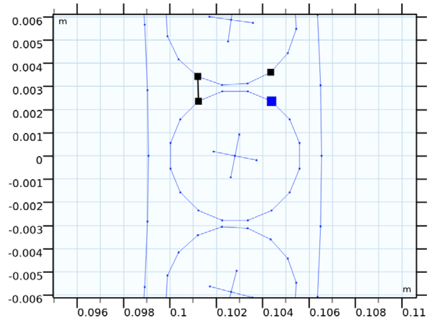
6 Click **Build Selected**.



Mesh Control Entities (wp3)>Line Segment 3 (ls3)

1 In the **Work Plane** toolbar, click **More Primitives** and choose **Line Segment**.

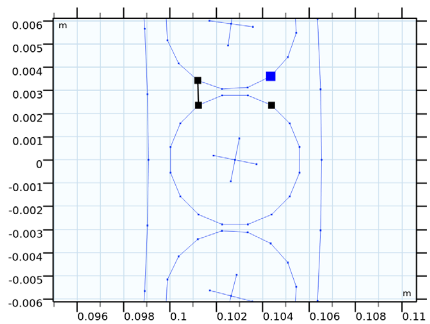
2 On the object **sca2(1)**, select Point 1 only.



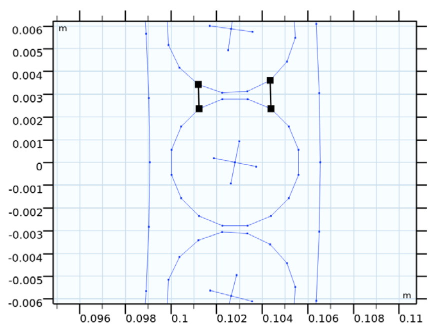
3 In the **Settings** window for **Line Segment**, locate the **Endpoint** section.

4 Find the **End vertex** subsection. Click to select the **Activate Selection** toggle button.

5 On the object **rot4(2)**, select Point 1 only.



6 Click **Build Selected**.



An alternative to this solution is to add two line segments in the second work plane directly, without having to go through the hassle of defining and transforming points. That solution would cause the helper edges to be swept too, resulting in interior boundaries between the wires. You might prefer that approach actually — an additional advantage is that you can use the resulting domains to assign a “gap permeability”.

Again, there are many ways to achieve your goal. One advantage of the approach presented here, is increased robustness with respect to *topological changes*. The points on the armor wire polygons are numbered, and so are the polygons themselves. If you would use a line segment to connect two polygons directly — as done with **Mesh Control Entities (wp3)>Line Segment 1 (ls1)** for example — the line segment will refer to those numbers. If you then change the number of armor wires or the number of sides per polygon, the numbering will change and the geometry sequence may become “broken”.

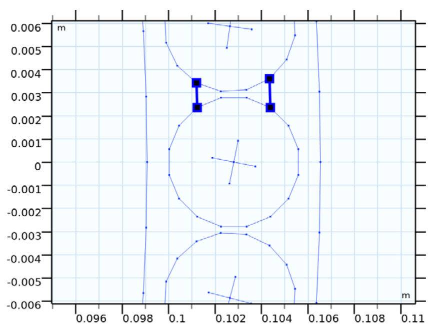
In short, this approach is slightly more robust when varying the number of wires N_{arm}, making it a bit easier for you to use this geometry sequence for your own cable designs. Let us proceed by copying these edges to the rest of the gaps.

Mesh Control Entities (wp3)>Rotate 5 (rot5)

1 In the **Work Plane** toolbar, click **Transforms** and choose **Rotate**.

2 Click the **Select Box** button in the **Graphics** toolbar.

3 Select the objects **ls2**, **ls3**, **rot4(1)**, **rot4(2)**, **sca2(1)**, and **sca2(2)** only.



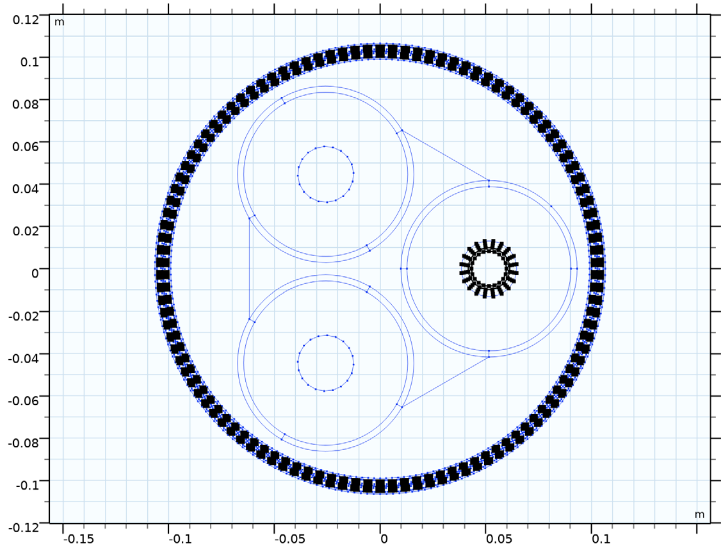
4 In the **Settings** window for **Rotate**, locate the **Rotation** section.

5 In the **Angle** text field, type `360[deg]*range(1/Narm,1/Narm,1)`.

6 Click **Build Selected**.

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

8 Click the **Zoom In** button in the **Graphics** toolbar, twice.



Finally, add some mesh control entities to the armor wire, to help out with the boundary layer mesh and the triangles in the wire center.

Mesh Control Entities (wp3)>Polygon 3 (pol3)

1 In the **Work Plane** toolbar, click **Polygon**.

2 In the **Settings** window for **Polygon**, locate the **Object Type** section.

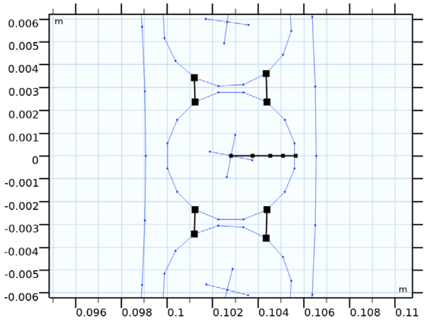
3 From the **Type** list, choose **Open curve**.

4 Locate the **Coordinates** section. From the **Data source** list, choose **Vectors**.

5 In the **xw** text field, type `Darm/2+TCFp16*Tarm/2-{0,Dsarm/5,2*Dsarm/5,TCFp16*Tarm/3,TCFp16*Tarm/2}`.

6 In the **yw** text field, type `0`.

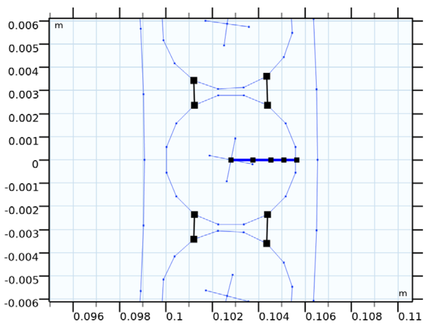
7 Click **Build Selected**.



Mesh Control Entities (wp3)>Rotate 6 (rot6)

1 In the **Work Plane** toolbar, click **Transforms** and choose **Rotate**.

2 Select the object **pol3** only.

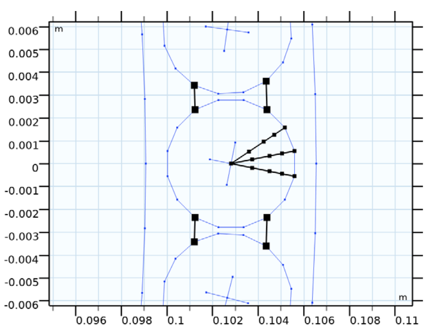


3 In the **Settings** window for **Rotate**, locate the **Rotation** section.

4 In the **Angle** text field, type `360[deg]*{-1/32,1/32,3/32}`.

5 Locate the **Center of Rotation** section. In the **xw** text field, type `Darm/2`.

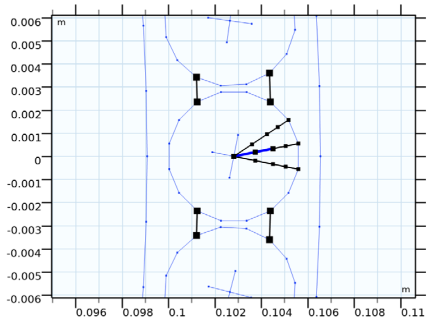
6 Click **Build Selected**.



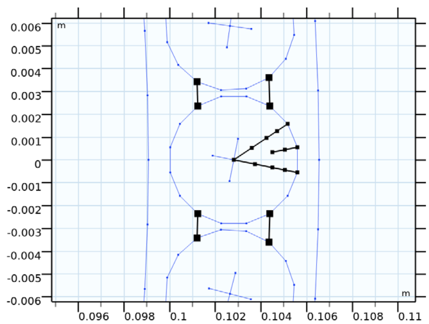
Mesh Control Entities (wp3)>Delete Entities 1 (del1)

1 Right-click **Plane Geometry** and choose **Delete Entities**.

2 On the object **rot6(2)**, select Boundaries 1 and 2 only.



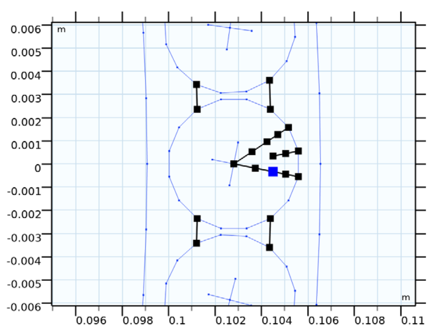
3 In the **Settings** window for **Delete Entities**, click **Build Selected**.



Mesh Control Entities (wp3)>Line Segment 4 (ls4)

1 In the **Work Plane** toolbar, click **More Primitives** and choose **Line Segment**.

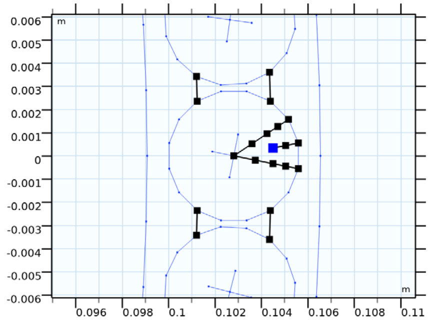
2 On the object **rot6(1)**, select Point 3 only.



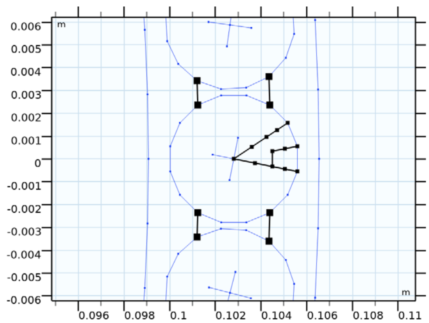
3 In the **Settings** window for **Line Segment**, locate the **Endpoint** section.

4 Find the **End vertex** subsection. Click to select the **Activate Selection** toggle button.

5 On the object **del1**, select Point 1 only.



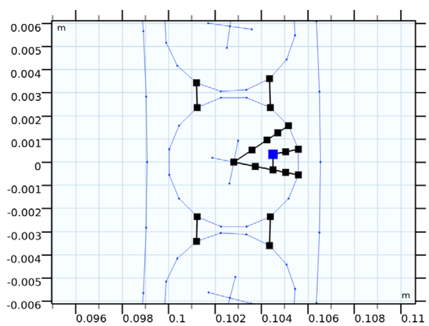
6 Click **Build Selected**.



Mesh Control Entities (wp3)>Line Segment 5 (ls5)

1 In the **Work Plane** toolbar, click **More Primitives** and choose **Line Segment**.

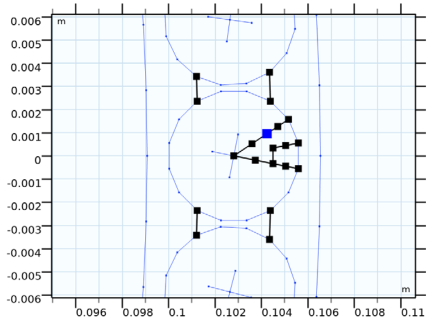
2 On the object **ls4**, select Point 2 only.



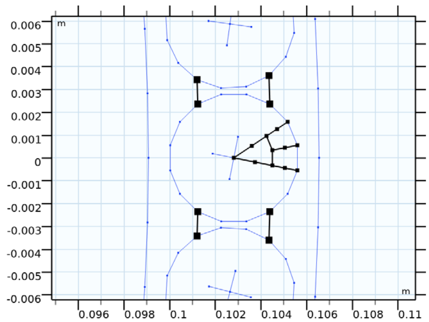
3 In the **Settings** window for **Line Segment**, locate the **Endpoint** section.

4 Find the **End vertex** subsection. Click to select the **Activate Selection** toggle button.

5 On the object **rot6(3)**, select Point 3 only.



6 Click **Build Selected**.

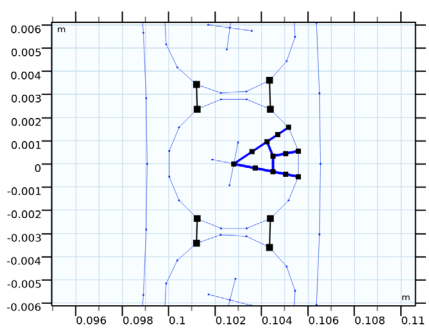


Mesh Control Entities (wp3)>Rotate 7 (rot7)

1 In the **Work Plane** toolbar, click **Transforms** and choose **Rotate**.

2 Click the **Select Box** button in the **Graphics** toolbar.

3 Select the objects **del1**, **ls4**, **ls5**, **rot6(1)**, and **rot6(3)** only.



4 In the **Settings** window for **Rotate**, locate the **Rotation** section.

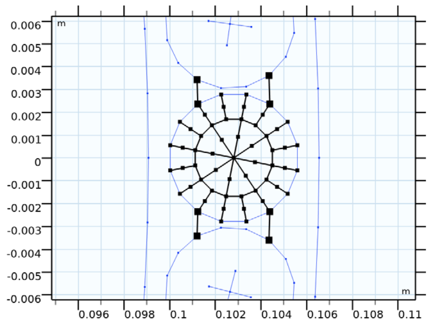
5 In the **Angle** text field, type `360[deg]*range(1/8,1/8,1)`.

6 Locate the **Center of Rotation** section. In the **xw** text field, type `Darm/2`.

7 Locate the **Selections of Resulting Entities** section. Select the **Resulting objects selection** check box.

8 From the **Show in 3D** list, choose **Off**.

9 Click **Build Selected**.

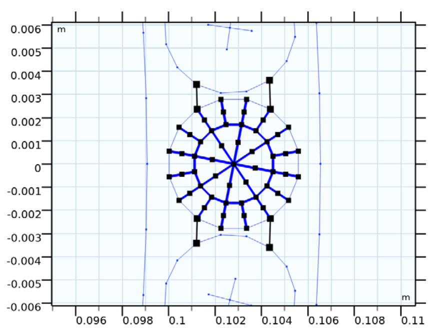


Mesh Control Entities (wp3)>Scale 3 (sca3)

1 In the **Work Plane** toolbar, click **Transforms** and choose **Scale**.

2 In the **Settings** window for **Scale**, locate the **Input** section.

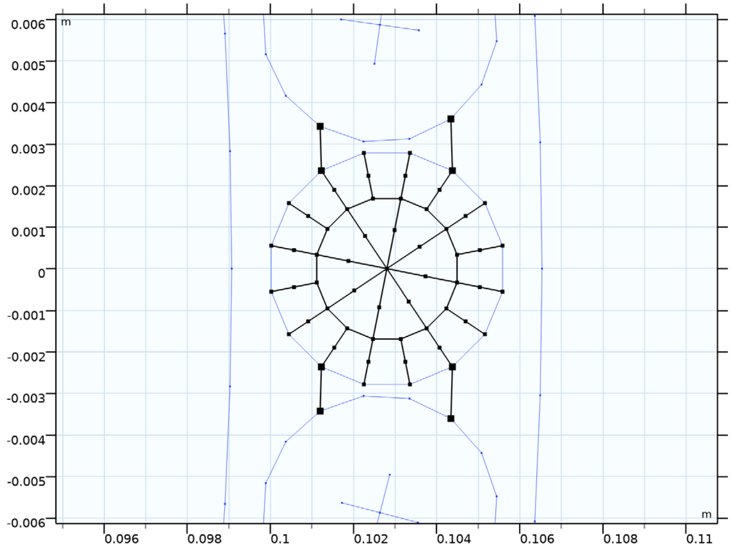
3 From the **Input objects** list, choose **Rotate 7**.



4 Locate the **Scale Factor** section. From the **Scaling** list, choose **Anisotropic**.

5 In the **yw** text field, type `SCFarm`.

6 Click **Build Selected**.

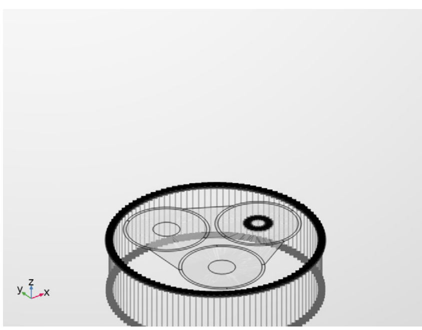


Note that these edges do not need to be copied to the other armor wires to get similar meshes there. A **Copy Face** operation in the mesh sequence will see to that.

Mesh Control Entities (wp3)

1 In the **Model Builder** window, under **Component 1 (comp1)>Geometry 1** click **Mesh Control Entities (wp3)**.

2 In the **Settings** window for **Work Plane**, click **Build Selected**.



3 In the **Model Builder** window, collapse the **Mesh Control Entities (wp3)** node.

Now that all parts of the cross section have been prepared on work planes, it is time to extrude some of them using a **Sweep** operation. In a sense, a sweep is more versatile than a plain **Extrude** or **Revolve** operation. The direction of the sweep is determined by an edge, or *spine curve*. The edge does not need to be straight (a quality used extensively in the *AC/DC Coil Geometry Parts Library*). While following the spine, the cross section can be rotated around it, creating a number of helical paths or a "twist".

Note that the extrude operator supports a twist too — and even scaling and displacement of the cross section is supported — but that twist is more like a straight projection. That is; both ends of the extrusion are connected by straight lines, creating an hourglass-like shape. Instead, the sweep operation attempts to resolve the helices in detail, using *Bézier* or *Spline* curves. So the sweep is more sophisticated and computationally heavy, while the extrude operator is more lightweight and robust.

At this point the twist is still inactive though, so you will get a plain extrusion either way. Let us proceed by adding the first spine polygon.

Polygon 2 (pol2)

1 In the **Geometry** toolbar, click **More Primitives** and choose **Polygon**.

2 In the **Settings** window for **Polygon**, locate the **Coordinates** section.

3 From the **Data source** list, choose **Vectors**.

4 In the **x** text field, type `0`.

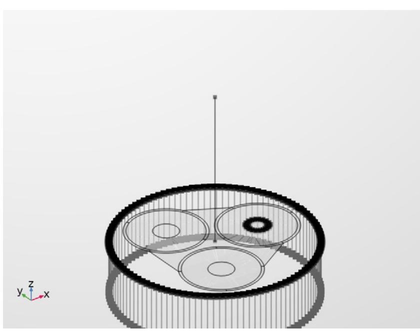
5 In the **y** text field, type `0`.

6 In the **z** text field, type `Lsec*{-1/2,1/2}`.

7 Locate the **Selections of Resulting Entities** section. Select the **Resulting objects selection** check box.

8 From the **Show in physics** list, choose **Off**.

9 Click **Build Selected**.

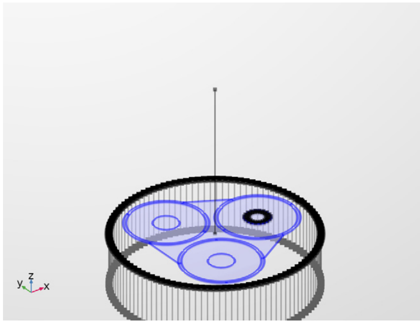


Sweep 1 (swe1)

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

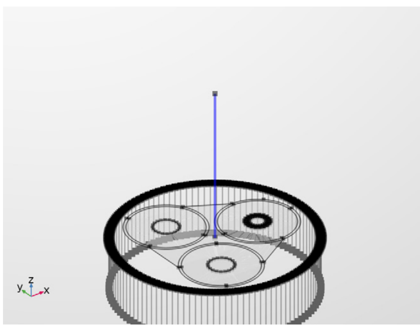
2 In the **Settings** window for **Sweep**, locate the **Cross Section** section.

3 From the **Faces to sweep** list, choose **Phases and Screens**.



4 Locate the **Spine Curve** section. Find the **Edges to follow** subsection. Click to select the **Activate Selection** toggle button.

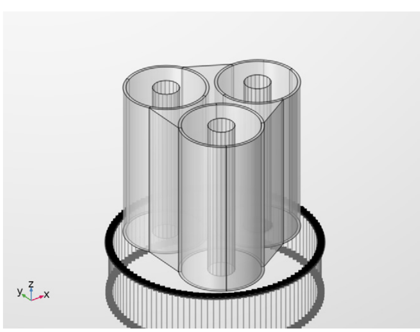
5 From the **Edges to follow** list, choose **Polygon 2**.



6 Locate the **Keep Input** section. Clear the **Keep input objects** check box.

7 Locate the **Motion of Cross Section** section. In the **Twist angle** text field, type `360[deg]*Tenab*s[m]/LLpha`.

8 Click **Build Selected**.

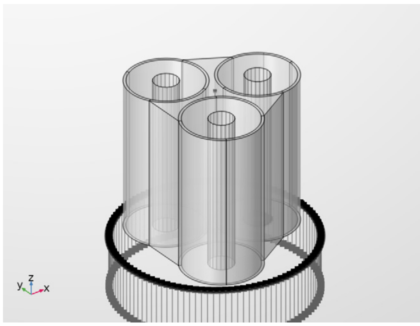


The details of the twist and the related angular expressions will be discussed later. Continue with the second sweep.

Polygon 3 (pol3)

1 In the **Model Builder** window, under **Component 1 (comp1)>Geometry 1** right-click **Polygon 2 (pol2)** and choose **Duplicate**.

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



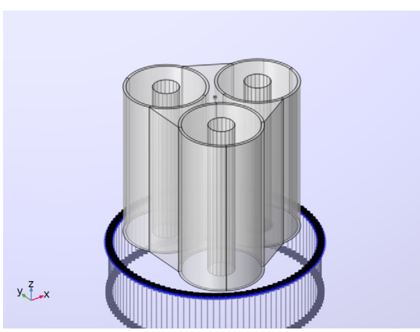
Sweep 2 (swe2)

1 In the **Model Builder** window, under **Component 1 (comp1)>Geometry 1** right-click **Sweep 1 (swe1)** and choose **Duplicate**.

2 In the **Settings** window for **Sweep**, locate the **Cross Section** section.

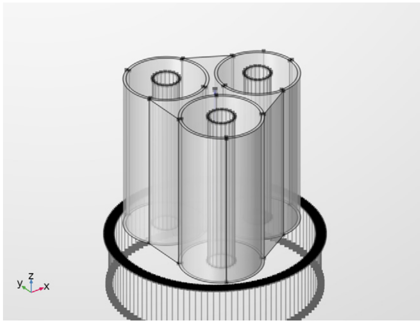
3 Find the **Faces to sweep** subsection. Click to select the **Activate Selection** toggle button.

4 From the **Faces to sweep** list, choose **Cable Armor and Sea Bed**.



5 Locate the **Spine Curve** section. Find the **Edges to follow** subsection. Click to select the **Activate Selection** toggle button.

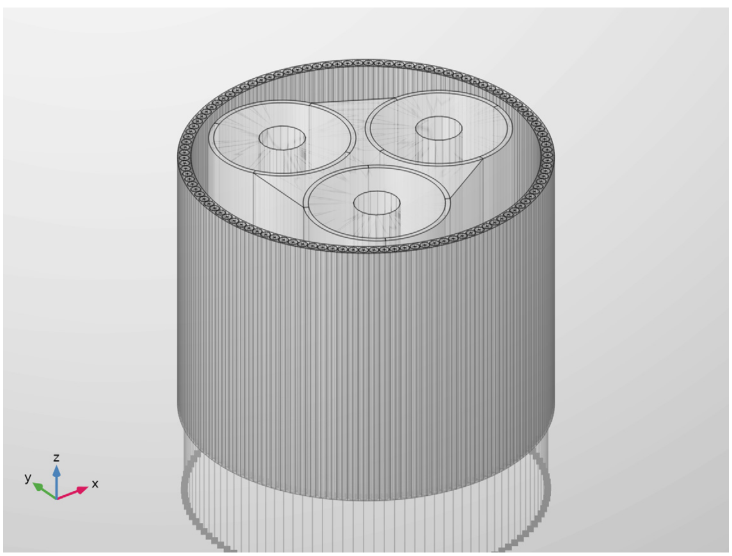
6 From the **Edges to follow** list, choose **Polygon 3**.



7 Locate the **Motion of Cross Section** section. In the **Twist angle** text field, modify the expression.

8 Type `360[deg]*Tenab*s[m]/LLarm`, that is; replace "`LLpha`" with "`LLarm`".

9 Click **Build Selected**.



This is the geometry as it will be used for the *Extruded 2D Model* in the *Inductive Effects 3D* tutorial. It builds instantly and the related models solve in minutes. It is ideal for quick tests and validation.

Finally, it is time to check that the *full 3D twist* geometry works.

GLOBAL DEFINITIONS

Please be aware that the geometry has been set to "Automatic rebuild". Setting Nper to a large value and subsequently browsing in the Model Builder tree may freeze the COMSOL Desktop temporarily.

Geometric Parameters 3

1 In the **Model Builder** window, under **Global Definitions** click **Geometric Parameters 3**.

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

3 In the table, set the parameter `Nper` back to `1`.

With the parameter `Nper` set to "`1`", the following logic will come into motion:

- The value of `Nper` will become an integer, causing the parameter `Tenab` (defined as `round(Nper)==Nper`) to become "`1`" (true).
- The length of the cable section included in the model `Lsec` (defined as `CPcab*Nper`) will increase tenfold. This sets the length of the geometry to be precisely one times the cable's cross pitch (for more on this, see the *Inductive Effects 3D* tutorial).
- All the cameras in the model will apply a new anisotropic scaling, causing the geometry to look just as long as it did before. This allows you to keep a good overview and get consistent plots.
- The new parameter value `Tenab` will force the twist angle of the cable section `Tsec` (defined as `360[deg]*Tenab*Lsec/LLpha`) to become "`Lsec/LLpha`" times one full revolution.
- In addition to this, `Tenab` will enable the slant correction factors `SCFpha`, and `SCFarm`, and will cause the two sweep operations in the geometry to generate helices instead of straight extrusions.

Hint: If you press Ctrl+F and search for the strings "Nper", "Tenab", or "Lsec" in the model, you can see where these parameters have an effect.

Now, proceed by rebuilding **Sweep 1** and **Sweep 2**.

GEOMETRY 1

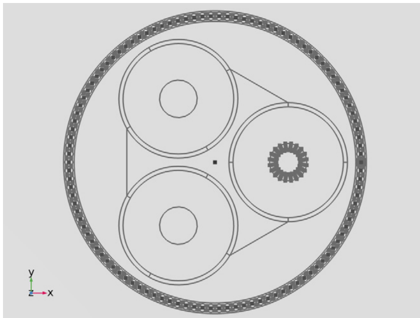
Polygon 2 (pol2)

1 In the **Model Builder** window, under **Component 1 (comp1)>Geometry 1** click **Polygon 2 (pol2)**.

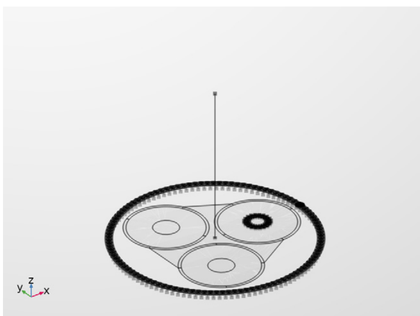
2 In the **Settings** window for **Polygon**, click **Build Selected**.

If the geometry looks stretched, it could be that the camera needs to be refreshed after the parameter change. This can be done by switching between views:

3 In the **Graphics** window toolbar, click ▾ next to **Go to Default View**, then choose **Go to View 2 (Orthographic, Top)**.



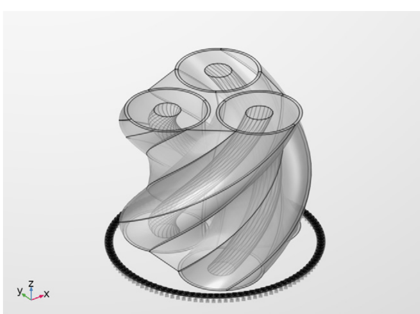
4 In the **Graphics** window toolbar, click ▾ next to **Go to Default View**, then choose **Go to View 1 (Orthographic)**.



Sweep 1 (swe1)

1 In the **Model Builder** window, click **Sweep 1 (swe1)**.

2 In the **Settings** window for **Sweep**, click **Build Selected** (takes a couple of seconds).

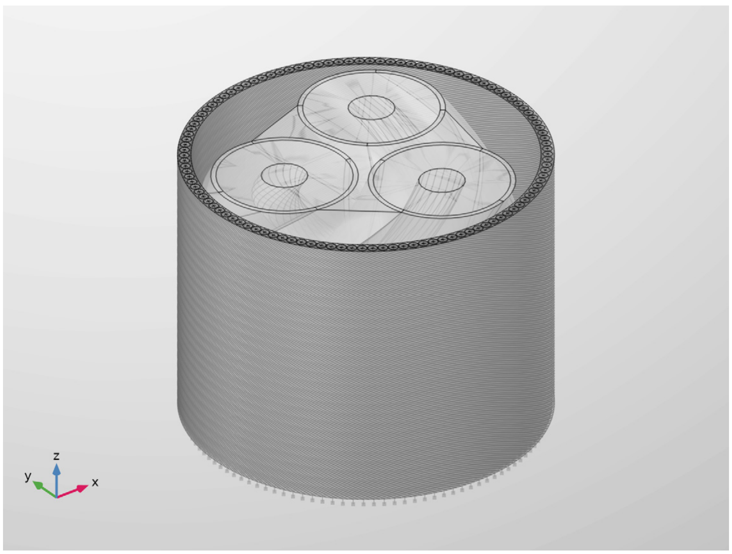


The twist works as follows: In the **Spine Curve** section, you have the setting **Parameterization**: *Arc length*. In other words, the value of the parameter "s" — as set in the section **Motion of Cross Section** — is equal to the distance in meters from the start of the curve **Polygon 2 (pol2)**. This curve starts at $z = -L_{sec}/2$, and ends at $z = +L_{sec}/2$. Now that the parameter `Tenab` has become "`1`", the twist angle (being a function of "s") will increase linearly from zero to `360[deg]*Lsec/LLpha`, or `Tsec`; the twist angle corresponding to a cable section of length `CPcab`. For more on this, see the *Inductive Effects 3D* tutorial.

Sweep 2 (swe2)

1 In the **Model Builder** window, click **Sweep 2 (swe2)**.

2 In the **Settings** window for **Sweep**, click **Build Selected** (should take a minute or so).

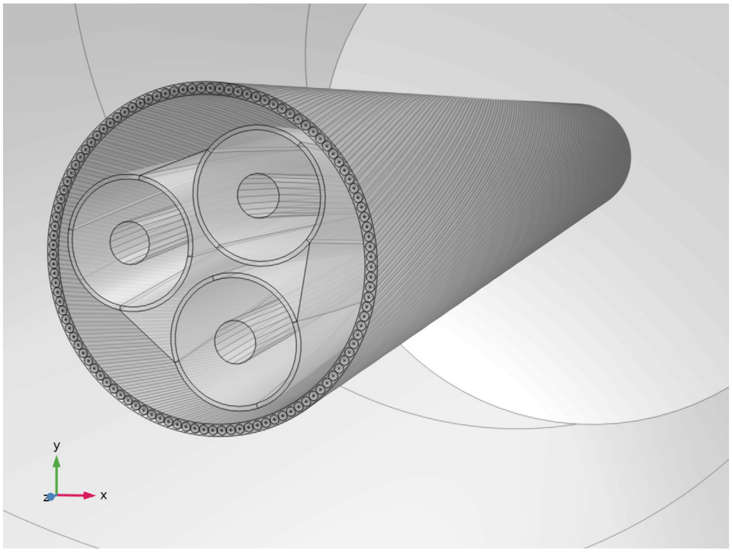


For the armor the twist works very much the same. It does turn in the opposite direction though, as `LLarm` is negative. Note that the final angle will be identical. That is; the expression `360[deg]*Lsec/LLarm` evaluates to the same angle: `Tsec` — apart from a shift equal to minus one full revolution.

3 In the **Model Builder** window, click **Geometry 1**.

4 In the **Settings** window for **Geometry 1**, click **Build All** (this will take a couple of minutes).

5 In the **Settings** window for **Geometry**, in the **Graphics** window toolbar, click ▾ next to **Go to Default View**, then choose **Go to View 5 (Perspective)**.



The resulting geometry will be used for the *3D Twist Model* in the *Inductive Effects 3D* tutorial. It takes more time to build, but represents a much more realistic approximation of the actual device. It will allow you to investigate phenomena that the 2D models cannot reproduce.

You have now finished building and testing the geometry sequence. Save the resulting file, so that you can use it as a basis for the next part.

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

Modeling Instructions — Selections

With the geometry built and tested, you are ready to configure the *selections*. Here, a "selection" is basically a stored reference to a number of domains, boundaries, or edges. The most simple and direct approach is to specify a selection manually by clicking in the graphics window (when adding a mesh or physics feature, for example). This is convenient for quick testing, or prototyping — when the selection is intended as a one-time effort.

When your models contain sophisticated geometries or when you have many (parameterized) variants of these geometries, manual selections may be insufficient. Alternatively, selections can be generated by the geometry sequence itself — as used extensively in the *Introduction* tutorial, section *Appendix — Modeling Instructions (Geometry)* — or they can be added as separate entities under the **Definitions** node. The latter will be used in this tutorial.

ROOT

You can either continue where you left off, or open a reference file from the Application Libraries folder.

1 From the **File** menu, choose **Open**.
2 Browse to the model's Application Libraries folder and double-click the file `submarine_cable_07_b_geom_mesh_3d.mph`.
3 From the **File** menu, choose **Save As**.
4 Browse to a suitable folder and type the filename `submarine_cable_07_c_geom_mesh_3d.mph`.

DEFINITIONS

The selections used in this tutorial are primarily *coordinate-based*: Parameterized coordinate expressions are used to define simple shapes such as cylinders, boxes or spheres. Domains, boundaries, and edges are then selected based on certain criteria — whether they lie inside the shape, outside the shape, or intersect with the shape.

Geometry-based selections on the other hand, use operations in the geometry sequence to specify a selection. For instance, the polygon creating one armor wire can also define its selection. As the armor wire is then copied to create the armor ring, the selection is inherited. Consequently, all armor wires end up in the selection "armor wire". This allows for selections that would be difficult to achieve otherwise.

However, if a model relies heavily on selections generated by the geometry and at some point you decide to redesign the geometry sequence from scratch — or to import it instead

— the applied selections may become empty or “broken” throughout the model. Coordinate-based selections are more robust in this regard.

Proceed by adding your first coordinate-based selection.

Cross Section, Top

1 In the **Definitions** toolbar, click **Cylinder**.

2 In the **Settings** window for **Cylinder**, type `Cross Section, Top` in the **Label** text field.

3 Locate the **Geometric Entity Level** section. From the **Level** list, choose **Boundary**.

4 Locate the **Size and Shape** section. In the **Outer radius** text field, type `6*Dcab/2`.

5 In the **Top distance** text field, type `Lsec/4`.

6 In the **Bottom distance** text field, type `-Lsec/4`.

7 Locate the **Position** section. In the **z** text field, type `Lsec/2`.

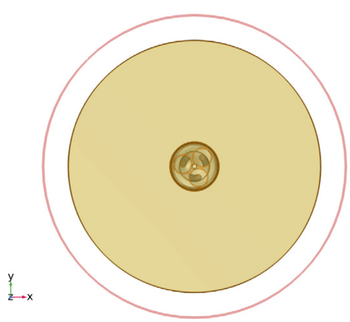
8 Locate the **Output Entities** section. From the **Include entity if** list, choose **All vertices inside cylinder**.

This setting improves both performance and reliability. Checking whether all vertices adjacent to a certain boundary lie inside a cylinder is much easier then checking whether the boundary itself lies inside the cylinder. The boundary could have double curvature after all, so an evaluation mesh would need to be constructed: For all points that lie on the boundary one would have to evaluate whether they also lie inside the cylinder.

Instead, the coordinates of the vertices are readily available and what vertices are adjacent to any given boundary is known too.

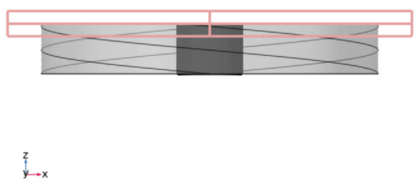
9 In the **Graphics** window toolbar, click ▾ next to **Go to Default View**, then choose **Go to View 2 (Orthographic, Top)**.

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



11 In the **Graphics** window toolbar, click ▾ next to **Go to Default View**, then choose **Go to View 4 (Orthographic, Side)**.

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



By using a cylinder $6D_{cab}$ in diameter, $L_{sec}/2$ in height, and centered around the coordinate $z = +L_{sec}/2$, you have a clear definition of what is considered to be the *top cross section plane*.

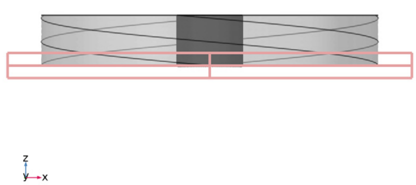
The *bottom cross section plane* is simply a shifted duplicate.

Cross Section, Bottom

1 Right-click **Cross Section, Top** and choose **Duplicate**.

2 In the **Settings** window for **Cylinder**, type `Cross Section, Bottom` in the **Label** text field.

3 Locate the **Position** section. In the **z** text field, type `-Lsec/2`.



Now that both the top- and bottom cross section selections are known, you can unify the two to create a new selection containing all cross-sectional faces:

Cross Section, Top and Bottom

1 In the **Definitions** toolbar, click **Union**.

2 In the **Settings** window for **Union**, type `Cross Section, Top and Bottom` in the **Label** text field.

3 Locate the **Geometric Entity Level** section. From the **Level** list, choose **Boundary**.

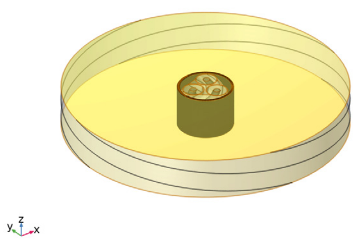
4 Locate the **Input Entities** section. Under **Selections to add**, click + **Add**.

5 In the **Add** dialog box, in the **Selections to add** list, choose **Cross Section, Top** and **Cross Section, Bottom**.

6 Click **OK**.

7 In the **Settings** window for **Union**, in the **Graphics** window toolbar, click ▾ next to **Go to Default View**, then choose **Go to View 1 (Orthographic)**.

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



Hint: if during this procedure (or any of the following procedures) you get a persistent red highlighting, it may be you need to clear your selections manually by clicking "Clear Selection" in the graphics toolbar.

You could consider **Cross Section, Top and Bottom** a "derived" selection (based on a *union*), rather than a coordinate-based primary selection. Notice that this selection would have been difficult to obtain using a single coordinate-based selection — any cylinder containing all top and bottom faces would automatically contain anything in-between.

Phases

1 In the **Definitions** toolbar, click **Cylinder**.

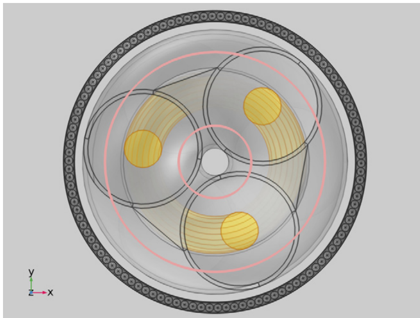
2 In the **Settings** window for **Cylinder**, type `Phases` in the **Label** text field.

3 Locate the **Size and Shape** section. In the **Outer radius** text field, type `Dpha/sqrt(3)+(Dcon/2+Dins/2)/2`.

4 In the **Inner radius** text field, type `Dpha/sqrt(3)-(Dcon/2+Dins/2)/2`.

5 Locate the **Output Entities** section. From the **Include entity if** list, choose **All vertices inside cylinder**.

6 In the **Graphics** window toolbar, click ▾ next to **Go to Default View**, then choose **Go to View 2 (Orthographic, Top)**.



The orthographic views are ideal for checking the selections. This (among other things) is what made section *Modeling Instructions — Camera Setup* worth the effort.

Because of the helical structure of the geometry, selecting all the phases was rather easy. Selecting an individual phase is more difficult. What helps in this case, is to use "all phases" as input entities for the "single phase" selection. This prevents the coordinate-based selection from picking anything other than a phase domain. The only thing that remains to be done, is to specify which one.

Phase 1

1 In the **Model Builder** window, right-click **Selections** and choose **Ball**.

2 In the **Settings** window for **Ball**, type `Phase 1` in the **Label** text field.

3 Locate the **Input Entities** section. From the **Entities** list, choose **From selections**.

4 Under **Selections**, click + **Add**.

5 In the **Add** dialog box, select **Phases** in the **Selections** list.

6 Click **OK**.

7 In the **Settings** window for **Ball**, locate the **Ball Center** section.

8 In the **x** text field, type `-(Dpha/2)/sqrt(3)`.

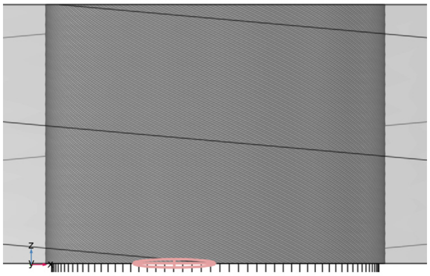
9 In the **y** text field, type `Dpha/2`.

10 In the **z** text field, type `-Lsec/2`.

11 Locate the **Ball Radius** section. In the **Radius** text field, type `(Dcon/2+Dins/2)/2`.

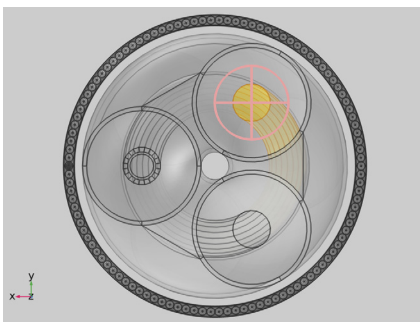
12 Locate the **Output Entities** section. From the **Include entity if** list, choose **Some vertex inside ball**.

13 In the **Graphics** window toolbar, click ▾ next to **Go to Default View**, then choose **Go to View 4 (Orthographic, Side)**.



The ball looks distorted because of the anisotropic scaling used in the camera settings.

14 In the **Graphics** window toolbar, click ▾ next to **Go to Default View**, then choose **Go to View 3 (Orthographic, Bottom)**.



The ball is chosen to coincide with the work planes in the geometry, located at coordinate $z = -L_{sec}/2$. Here, the expressions for the phase's x,y-coordinates are fairly straightforward — they do not depend on the twist angle.

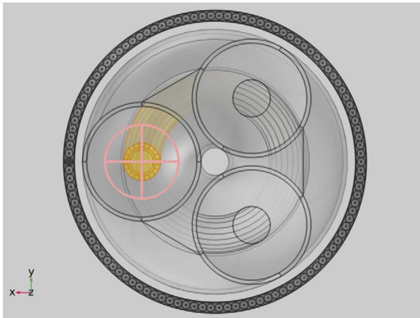
Phase 2

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

2 In the **Settings** window for **Ball**, type `Phase 2` in the **Label** text field.

3 Locate the **Ball Center** section. In the **x** text field, type `Dpha/sqrt(3)`.

4 In the **y** text field, type `0`.



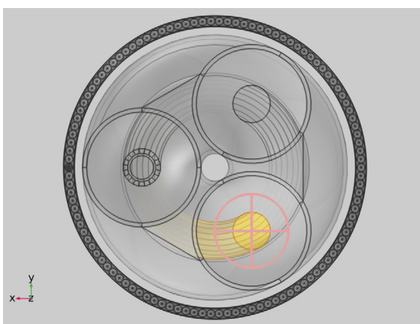
Phase 3

1 Right-click **Phase 2** and choose **Duplicate**.

2 In the **Settings** window for **Ball**, type `Phase 3` in the **Label** text field.

3 Locate the **Ball Center** section. In the **x** text field, type `-(Dpha/2)/sqrt(3)`.

4 In the **y** text field, type `-Dpha/2`.



If at this point you are wondering why you should be using parameterized expressions if you can just use an explicit selection and click the domain you want directly, please remember this approach is demonstrated for use in advanced parameterized models. *When building a model from scratch, one typically starts off using explicit selections.* More sophisticated selection methods are introduced later, when deemed necessary.

Coming up with a simple yet reliable definition of the screen selection is a bit tricky. Instead of a coordinate-based selection, the *topology* of the cable is used. One **Adjacent** operation with the phases as input will give you a good definition of the cross-linked polyethylene (XLPE) domains. One more **Adjacent** operation will give you both the phases and the screens. What then remains is to remove the phases from the selection, and you are left with the screens. Let us proceed.

Cross-Linked Polyethylene (XLPE)

1 In the **Definitions** toolbar, click **Adjacent**.

2 In the **Settings** window for **Adjacent**, type `Cross-Linked Polyethylene (XLPE)` in the **Label** text field.

3 Locate the **Input Entities** section. Under **Input selections**, click + **Add**.

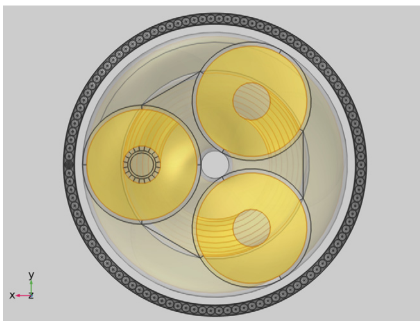
4 In the **Add** dialog box, select **Phases** in the **Input selections** list.

5 Click **OK**.

6 In the **Settings** window for **Adjacent**, locate the **Output Entities** section.

7 From the **Geometric entity level** list, choose **Adjacent domains**.

8 Click the **Clear Selection** button in the **Graphics** toolbar.



Phases and Screens

1 In the **Definitions** toolbar, click **Adjacent**.

2 In the **Settings** window for **Adjacent**, type `Phases and Screens` in the **Label** text field.

3 Locate the **Input Entities** section. Under **Input selections**, click + **Add**.

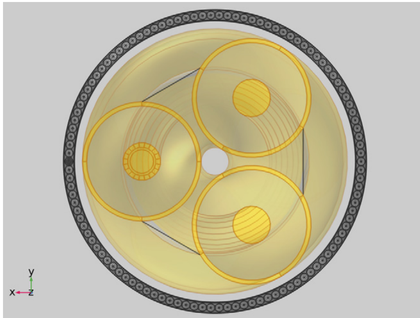
4 In the **Add** dialog box, select **Cross-Linked Polyethylene (XLPE)** in the **Input selections** list.

5 Click **OK**.

6 In the **Settings** window for **Adjacent**, locate the **Output Entities** section.

7 From the **Geometric entity level** list, choose **Adjacent domains**.

8 Click the **Clear Selection** button in the **Graphics** toolbar.



Screens

1 In the **Definitions** toolbar, click **Difference**.

2 In the **Settings** window for **Difference**, type `Screens` in the **Label** text field.

3 Locate the **Input Entities** section. Under **Selections to add**, click **Add**.

4 In the **Add** dialog box, select **Phases and Screens** in the **Selections to add** list.

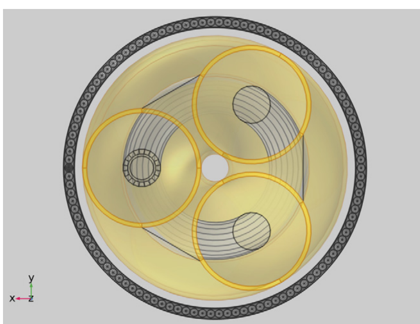
5 Click **OK**.

6 In the **Settings** window for **Difference**, locate the **Input Entities** section.

7 Under **Selections to subtract**, click **Add**.

8 In the **Add** dialog box, select **Phases** in the **Selections to subtract** list.

9 Click **OK**.



Next, will be the armor wire *centerlines*. The centerline selection is a nice example of something that would be tedious to do by hand: There are a 100 or so armor wires, each of them having 16 exterior edges and five interior ones (including the centerline). So the challenge is to pick a hundred needles from a 2100-needle hay stack. This is what the edges sticking out of the bottom of the geometry are good for.

Cable Armor, Centerline

1 In the **Definitions** toolbar, click **Cylinder**.

2 In the **Settings** window for **Cylinder**, type `Cable Armor, Centerline` in the **Label** text field.

3 Locate the **Geometric Entity Level** section. From the **Level** list, choose **Edge**.

4 Select the **Group by continuous tangent** check box.

5 In the **Angular tolerance** text field, type `60`.

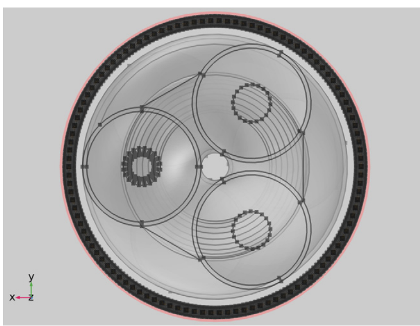
6 Locate the **Size and Shape** section. In the **Outer radius** text field, type `Darm/2+Tarm`.

7 In the **Top distance** text field, type `5*Tarm`.

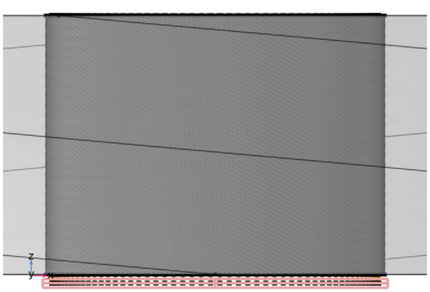
8 In the **Bottom distance** text field, type `-5*Tarm`.

9 Locate the **Position** section. In the **z** text field, type `-(Lsec/2+10*Tarm)`.

10 Locate the **Output Entities** section. From the **Include entity if** list, choose **Some vertex inside cylinder**.



11 In the **Graphics** window toolbar, click next to **Go to Default View**, then choose **Go to View 4 (Orthographic, Side)**.



The **Group by continuous tangent** setting is something you might want to keep in mind. It works for both boundary and edge selections.

The cylinder itself only selects the edges sticking out at the bottom. Those edges however, are an "extension" of the armor wire center lines (within a 60° tolerance, that is). As a consequence of the **Group by continuous tangent** setting, the centerlines are selected too.

Note that this check box is especially useful if you have a continuous curved surface consisting of many faces, and you want to avoid having to select every face individually.

Let us continue, and use the centerline selection to get to the armor-, conductor- and insulator selections.

Cable Armor

1 In the **Definitions** toolbar, click **Adjacent**.

2 In the **Settings** window for **Adjacent**, type `Cable Armor` in the **Label** text field.

3 Locate the **Input Entities** section. From the **Geometric entity level** list, choose **Edge**.

4 Under **Input selections**, click + **Add**.

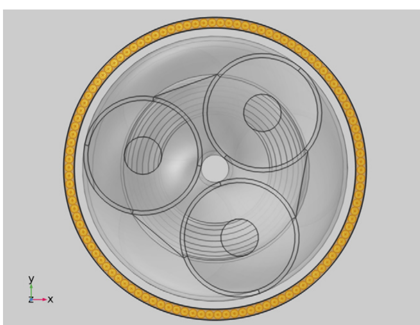
5 In the **Add** dialog box, select **Cable Armor, Centerline** in the **Input selections** list.

6 Click **OK**.

7 In the **Settings** window for **Adjacent**, locate the **Output Entities** section.

8 From the **Geometric entity level** list, choose **Adjacent domains**.

9 In the **Graphics** window toolbar, click ▾ next to **Go to Default View**, then choose **Go to View 2 (Orthographic, Top)**.



Conductors

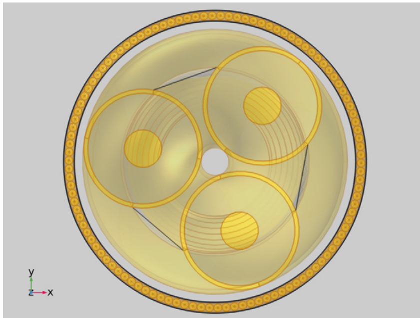
1 In the **Definitions** toolbar, click **Union**.

2 In the **Settings** window for **Union**, type `Conductors` in the **Label** text field.

3 Locate the **Input Entities** section. Under **Selections to add**, click + **Add**.

4 In the **Add** dialog box, in the **Selections to add** list, choose **Phases and Screens** and **Cable Armor**.

5 Click **OK**.



Insulators

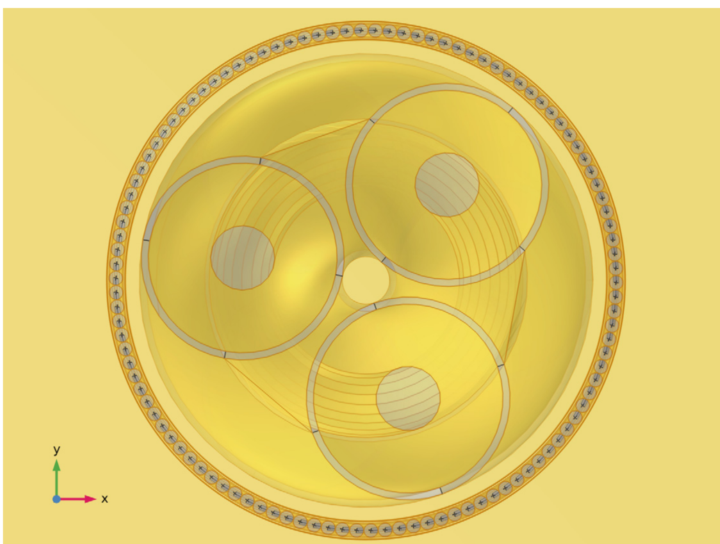
1 In the **Definitions** toolbar, click **Complement**.

2 In the **Settings** window for **Complement**, type `Insulators` in the **Label** text field.

3 Locate the **Input Entities** section. Under **Selections to invert**, click **Add**.

4 In the **Add** dialog box, select **Conductors** in the **Selections to invert** list.

5 Click **OK**.



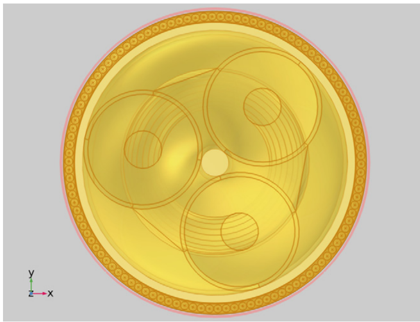
Cable Domains

1 In the **Definitions** toolbar, click **Cylinder**.

2 In the **Settings** window for **Cylinder**, type `Cable Domains` in the **Label** text field.

3 Locate the **Size and Shape** section. In the **Outer radius** text field, type `Darm/2+Tarm`.

4 Locate the **Output Entities** section. From the **Include entity if** list, choose **All vertices inside cylinder**.



This selection is convenient for postprocessing among other things; it separates the main area of interest (the cable) from the environment.

MESH SELECTIONS

The next 26 selections are all related to *meshing* (they will be grouped accordingly). First, the different parts of the bottom cross-sectional plane will need to be identified. This will help you build the cross-sectional surface mesh before it is swept through the cable geometry. The basic operations are the same as the ones you have used before.

Phases, Exterior Boundaries

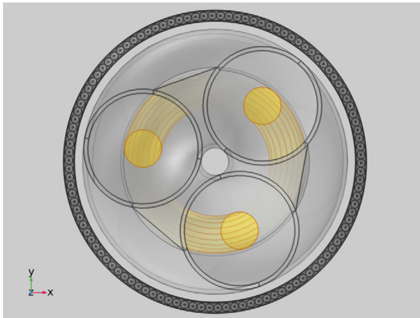
1 In the **Definitions** toolbar, click **Adjacent**.

2 In the **Settings** window for **Adjacent**, type `Phases, Exterior Boundaries` in the **Label** text field.

3 Locate the **Input Entities** section. Under **Input selections**, click **Add**.

4 In the **Add** dialog box, select **Phases** in the **Input selections** list.

5 Click **OK**.



6 Right-click **Phases, Exterior Boundaries** and choose **Group**.

Mesh Selections

In the **Settings** window for **Group**, type `Mesh Selections` in the **Label** text field.

Screens, Exterior Boundaries

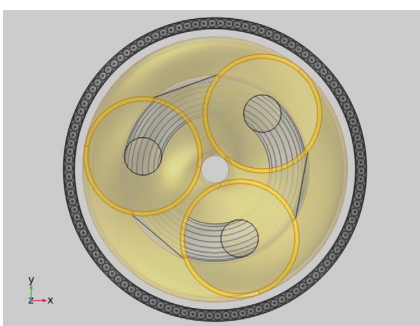
1 In the **Definitions** toolbar, click **Adjacent**.

2 In the **Settings** window for **Adjacent**, type `Screens, Exterior Boundaries` in the **Label** text field.

3 Locate the **Input Entities** section. Under **Input selections**, click **Add**.

4 In the **Add** dialog box, select **Screens** in the **Input selections** list.

5 Click **OK**.



Armor, Exterior Boundaries

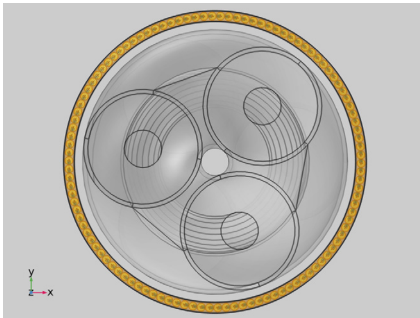
1 In the **Definitions** toolbar, click **Adjacent**.

2 In the **Settings** window for **Adjacent**, type `Armor, Exterior Boundaries` in the **Label** text field.

3 Locate the **Input Entities** section. Under **Input selections**, click **Add**.

4 In the **Add** dialog box, select **Cable Armor** in the **Input selections** list.

5 Click **OK**.



Phases, Boundaries Bottom

1 In the **Definitions** toolbar, click **Intersection**.

2 In the **Settings** window for **Intersection**, type `Phases, Boundaries Bottom` in the **Label** text field.

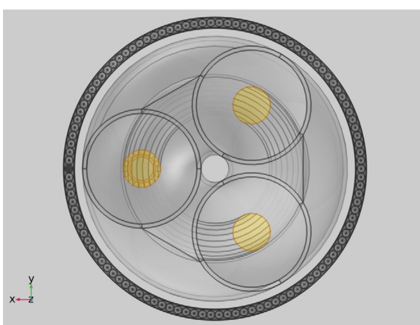
3 Locate the **Geometric Entity Level** section. From the **Level** list, choose **Boundary**.

4 Locate the **Input Entities** section. Under **Selections to intersect**, click **Add**.

5 In the **Add** dialog box, in the **Selections to intersect** list, choose **Cross Section, Bottom** and **Phases, Exterior Boundaries**.

6 Click **OK**.

7 In the **Settings** window for **Intersection**, in the **Graphics** window toolbar, click next to **Go to Default View**, then choose **Go to View 3 (Orthographic, Bottom)**.



Screens, Boundaries Bottom (Mapped 3)

1 In the **Definitions** toolbar, click **Intersection**.

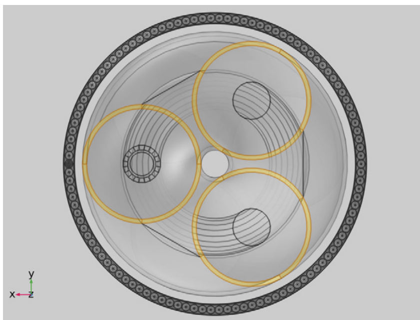
2 In the **Settings** window for **Intersection**, type `Screens, Boundaries Bottom (Mapped 3)` in the **Label** text field.

3 Locate the **Geometric Entity Level** section. From the **Level** list, choose **Boundary**.

4 Locate the **Input Entities** section. Under **Selections to intersect**, click + **Add**.

5 In the **Add** dialog box, in the **Selections to intersect** list, choose **Cross Section, Bottom** and **Screens, Exterior Boundaries**.

6 Click **OK**.



Armor, Boundaries Bottom

1 In the **Definitions** toolbar, click **Intersection**.

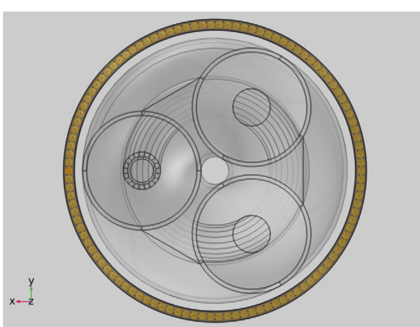
2 In the **Settings** window for **Intersection**, type `Armor, Boundaries Bottom` in the **Label** text field.

3 Locate the **Geometric Entity Level** section. From the **Level** list, choose **Boundary**.

4 Locate the **Input Entities** section. Under **Selections to intersect**, click + **Add**.

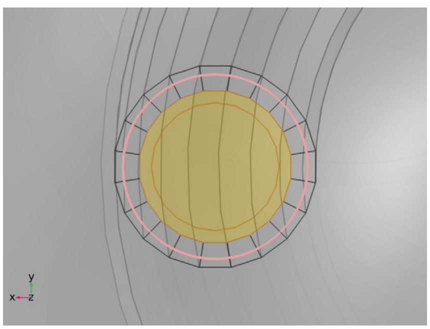
5 In the **Add** dialog box, in the **Selections to intersect** list, choose **Cross Section, Bottom** and **Armor, Exterior Boundaries**.

6 Click **OK**.



Free Triangular 1

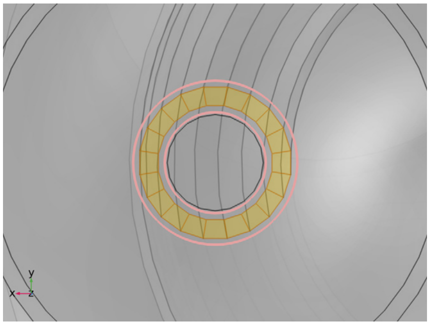
1 In the **Definitions** toolbar, click **Cylinder**.
2 In the **Settings** window for **Cylinder**, type `Free Triangular 1` in the **Label** text field.
3 Locate the **Geometric Entity Level** section. From the **Level** list, choose **Boundary**.
4 Locate the **Input Entities** section. From the **Entities** list, choose **From selections**.
5 Under **Selections**, click **Add**.
6 In the **Add** dialog box, select **Cross Section, Bottom** in the **Selections** list.
7 Click **OK**.
8 In the **Settings** window for **Cylinder**, locate the **Size and Shape** section.
9 In the **Outer radius** text field, type `Dcon/2-Dscup/8`.
10 Locate the **Position** section. In the **x** text field, type `Dpha/sqrt(3)`.
11 Locate the **Output Entities** section. From the **Include entity if** list, choose **All vertices inside cylinder**.



Mapped 1

1 Right-click **Free Triangular 1** and choose **Duplicate**.
2 In the **Settings** window for **Cylinder**, type `Mapped 1` in the **Label** text field.
3 Locate the **Size and Shape** section. In the **Outer radius** text field, type `Dcon/2+Dscup/8`.

4 In the **Inner radius** text field, type `Dcon/2-Dscup/2`.



Copy Face 1, Source Boundaries

1 In the **Definitions** toolbar, click **Union**.

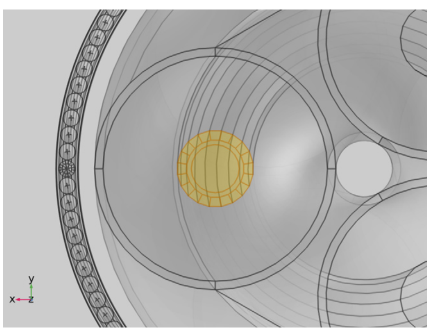
2 In the **Settings** window for **Union**, type `Copy Face 1, Source Boundaries` in the **Label** text field.

3 Locate the **Geometric Entity Level** section. From the **Level** list, choose **Boundary**.

4 Locate the **Input Entities** section. Under **Selections to add**, click **Add**.

5 In the **Add** dialog box, in the **Selections to add** list, choose **Free Triangular 1** and **Mapped 1**.

6 Click **OK**.



Copy Face 1, Destination Boundaries

1 In the **Definitions** toolbar, click **Difference**.

2 In the **Settings** window for **Difference**, type `Copy Face 1, Destination Boundaries` in the **Label** text field.

3 Locate the **Geometric Entity Level** section. From the **Level** list, choose **Boundary**.

4 Locate the **Input Entities** section. Under **Selections to add**, click **Add**.

5 In the **Add** dialog box, select **Phases, Boundaries Bottom** in the **Selections to add** list.

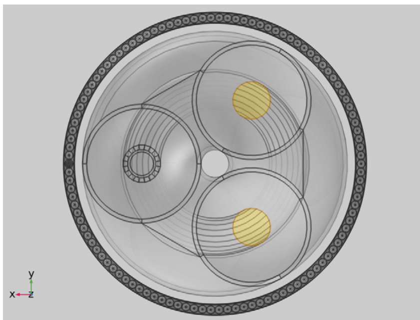
6 Click **OK**.

7 In the **Settings** window for **Difference**, locate the **Input Entities** section.

8 Under **Selections to subtract**, click + **Add**.

9 In the **Add** dialog box, select **Copy Face 1, Source Boundaries** in the **Selections to subtract** list.

10 Click **OK**.



Free Triangular 2

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

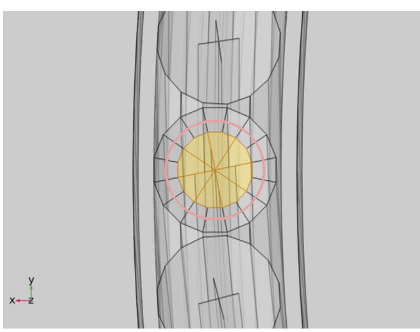
2 In the **Settings** window for **Cylinder**, type `Free Triangular 2` in the **Label** text field.

3 Locate the **Size and Shape** section. In the **Outer radius** text field, type `Tarm/2-Dsarm/5`.

4 Locate the **Position** section. In the **x** text field, type `Darm/2`.

5 Click the **Zoom to Selection** button in the **Graphics** toolbar.

6 Click the **Zoom Out** button in the **Graphics** toolbar, twice.



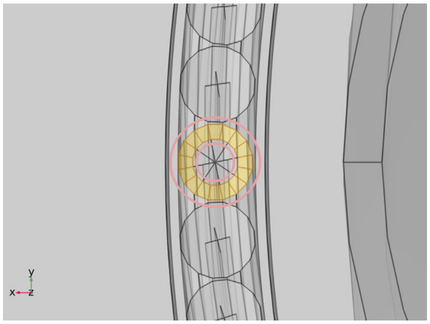
Mapped 2

1 Right-click **Free Triangular 2** and choose **Duplicate**.

2 In the **Settings** window for **Cylinder**, type `Mapped 2` in the **Label** text field.

3 Locate the **Size and Shape** section. In the **Outer radius** text field, type `Tarm/2+Dsarm/5`.

4 In the **Inner radius** text field, type `Tarm/2-Dsarm/2`.



Copy Face 2, Source Boundaries

1 In the **Definitions** toolbar, click **Union**.

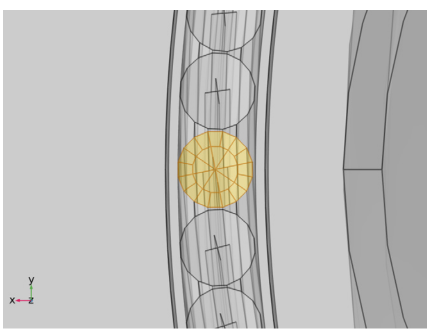
2 In the **Settings** window for **Union**, type `Copy Face 2, Source Boundaries` in the **Label** text field.

3 Locate the **Geometric Entity Level** section. From the **Level** list, choose **Boundary**.

4 Locate the **Input Entities** section. Under **Selections to add**, click **Add**.

5 In the **Add** dialog box, in the **Selections to add** list, choose **Free Triangular 2** and **Mapped 2**.

6 Click **OK**.



Copy Face 2, Destination Boundaries

1 In the **Definitions** toolbar, click **Difference**.

2 In the **Settings** window for **Difference**, type `Copy Face 2, Destination Boundaries` in the **Label** text field.

3 Locate the **Geometric Entity Level** section. From the **Level** list, choose **Boundary**.

4 Locate the **Input Entities** section. Under **Selections to add**, click + **Add**.

5 In the **Add** dialog box, select **Armor, Boundaries Bottom** in the **Selections to add** list.

6 Click **OK**.

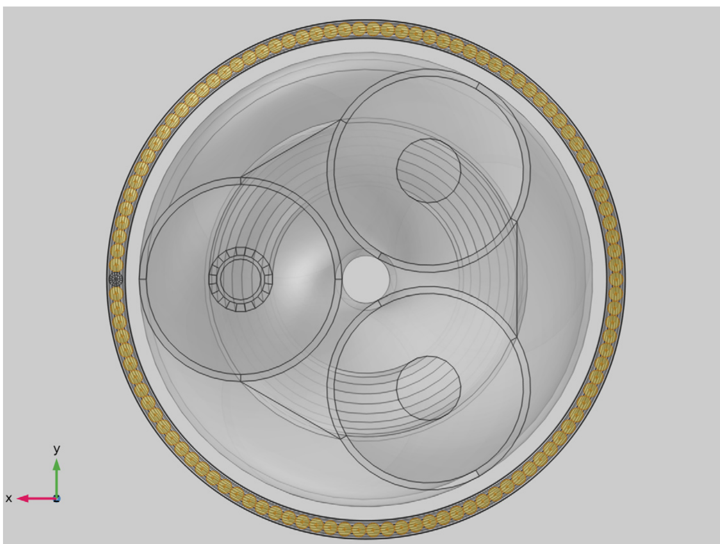
7 In the **Settings** window for **Difference**, locate the **Input Entities** section.

8 Under **Selections to subtract**, click + **Add**.

9 In the **Add** dialog box, select **Copy Face 2, Source Boundaries** in the **Selections to subtract** list.

10 Click **OK**.

11 Click the **Zoom to Selection** button in the **Graphics** toolbar.



Not Armor, Boundaries Bottom

1 In the **Definitions** toolbar, click **Difference**.

2 In the **Settings** window for **Difference**, type `Not Armor, Boundaries Bottom` in the **Label** text field.

3 Locate the **Geometric Entity Level** section. From the **Level** list, choose **Boundary**.

4 Locate the **Input Entities** section. Under **Selections to add**, click + **Add**.

5 In the **Add** dialog box, select **Cross Section, Bottom** in the **Selections to add** list.

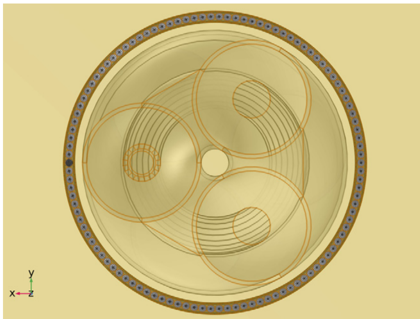
6 Click **OK**.

7 In the **Settings** window for **Difference**, locate the **Input Entities** section.

8 Under **Selections to subtract**, click **Add**.

9 In the **Add** dialog box, select **Armor, Boundaries Bottom** in the **Selections to subtract** list.

10 Click **OK**.



Mapped 4

1 In the **Definitions** toolbar, click **Cylinder**.

2 In the **Settings** window for **Cylinder**, type `Mapped 4` in the **Label** text field.

3 Locate the **Geometric Entity Level** section. From the **Level** list, choose **Boundary**.

4 Locate the **Input Entities** section. From the **Entities** list, choose **From selections**.

5 Under **Selections**, click **Add**.

6 In the **Add** dialog box, select **Not Armor, Boundaries Bottom** in the **Selections** list.

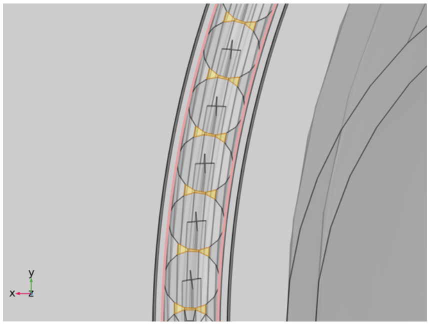
7 Click **OK**.

8 In the **Settings** window for **Cylinder**, locate the **Size and Shape** section.

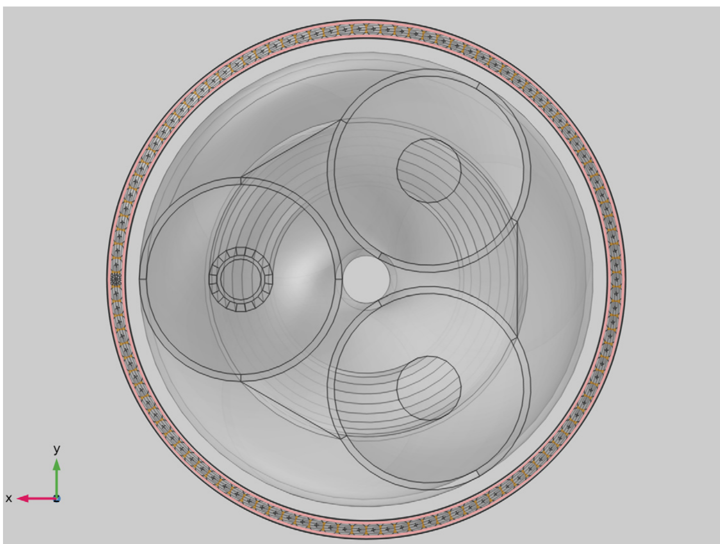
9 In the **Outer radius** text field, type `Darm/2+Tarm/2`.

10 In the **Inner radius** text field, type `Darm/2-Tarm/2`.

11 Locate the **Output Entities** section. From the **Include entity if** list, choose **All vertices inside cylinder**.



12 Click the **Zoom to Selection** button in the **Graphics** toolbar.



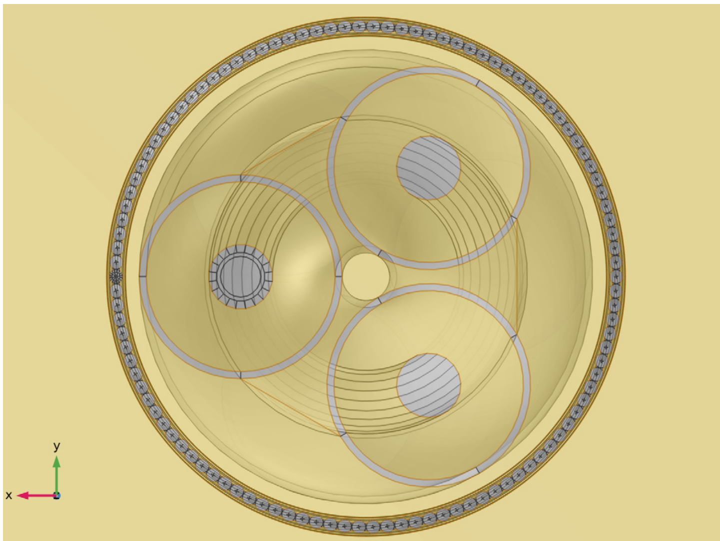
Free Triangular 3

1. In the **Definitions** toolbar, click **Difference**.
2. In the **Settings** window for **Difference**, type `Free Triangular 3` in the **Label** text field.
3. Locate the **Geometric Entity Level** section. From the **Level** list, choose **Boundary**.
4. Locate the **Input Entities** section. Under **Selections to add**, click **Add**.
5. In the **Add** dialog box, select **Cross Section, Bottom** in the **Selections to add** list.
6. Click **OK**.
7. In the **Settings** window for **Difference**, locate the **Input Entities** section.

8 Under **Selections to subtract**, click **Add**.

9 In the **Add** dialog box, in the **Selections to subtract** list, choose **Phases, Boundaries Bottom**, **Screens, Boundaries Bottom (Mapped 3)**, **Armor, Boundaries Bottom**, and **Mapped 4**.

10 Click **OK**.



Free Triangular 3, Size 1

1 In the **Definitions** toolbar, click **Cylinder**.

2 In the **Settings** window for **Cylinder**, type `Free Triangular 3, Size 1` in the **Label** text field.

3 Locate the **Geometric Entity Level** section. From the **Level** list, choose **Boundary**.

4 Locate the **Input Entities** section. From the **Entities** list, choose **From selections**.

5 Under **Selections**, click **Add**.

6 In the **Add** dialog box, select **Free Triangular 3** in the **Selections** list.

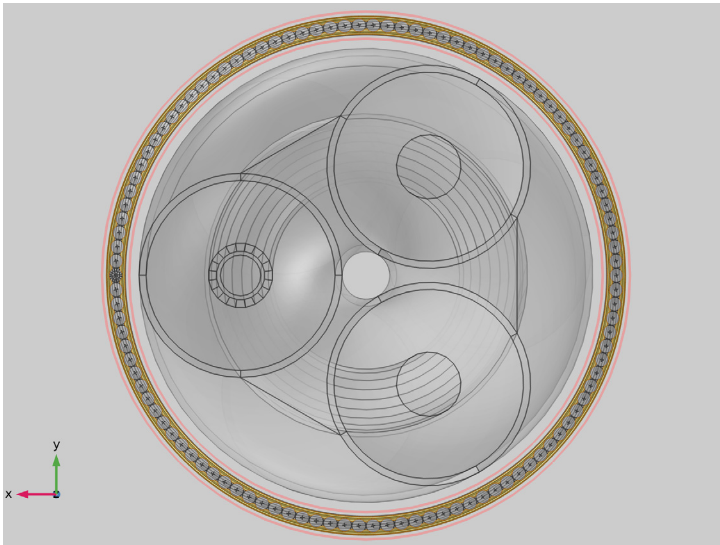
7 Click **OK**.

8 In the **Settings** window for **Cylinder**, locate the **Size and Shape** section.

9 In the **Outer radius** text field, type `Darm/2+Tarm`.

10 In the **Inner radius** text field, type `Darm/2-Tarm`.

11 Locate the **Output Entities** section. From the **Include entity if** list, choose **All vertices inside cylinder**.

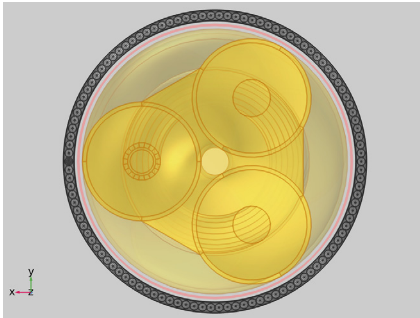


Now that you have analyzed and tagged all the faces in the bottom cross-sectional plane, it is time for some mesh *domain selections*. There will be a swept mesh on the inside (the phases and screens), a swept mesh on the outside (the armor and the environment), and in-between there will be a free tetrahedral mesh.

Swept 1, Distribution 1

1 In the **Definitions** toolbar, click **Cylinder**.
2 In the **Settings** window for **Cylinder**, type `Swept 1, Distribution 1` in the **Label** text field.
3 Locate the **Size and Shape** section. In the **Outer radius** text field, type `Dpha3/2`.
4 Locate the **Output Entities** section. From the **Include entity if** list, choose **All vertices inside cylinder**.

5 Click the **Clear Selection** button in the **Graphics** toolbar.



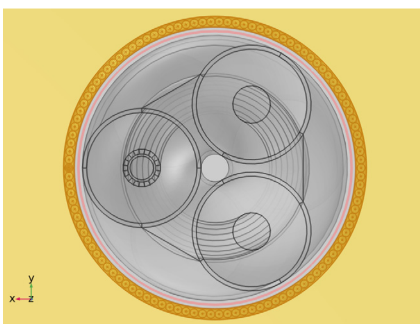
Swept 1, Distribution 2

1 Right-click **Swept 1, Distribution 1** and choose **Duplicate**.

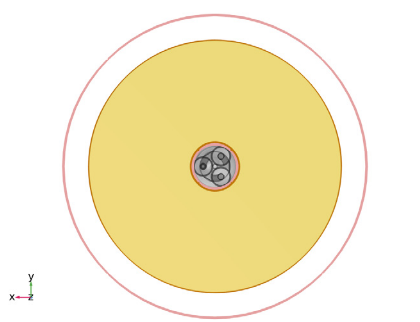
2 In the **Settings** window for **Cylinder**, type `Swept 1, Distribution 2` in the **Label** text field.

3 Locate the **Size and Shape** section. In the **Outer radius** text field, type `6*Dcab/2`.

4 In the **Inner radius** text field, type `Dpha3/2`.



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



Swept 1

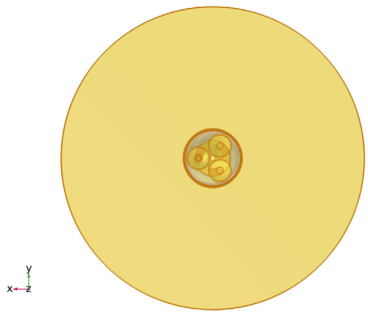
1 In the **Definitions** toolbar, click **Union**.

2 In the **Settings** window for **Union**, type `Swept 1` in the **Label** text field.

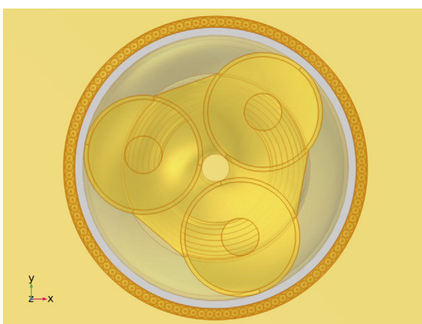
3 Locate the **Input Entities** section. Under **Selections to add**, click **Add**.

4 In the **Add** dialog box, in the **Selections to add** list, choose **Swept 1, Distribution 1** and **Swept 1, Distribution 2**.

5 Click **OK**.



6 In the **Settings** window for **Union**, in the **Graphics** window toolbar, click next to **Go to Default View**, then choose **Go to View 2 (Orthographic, Top)**.



Remaining Domains (Free Tetrahedral 1)

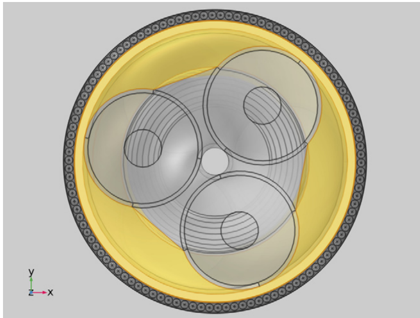
1 In the **Definitions** toolbar, click **Complement**.

2 In the **Settings** window for **Complement**, type `Remaining Domains (Free Tetrahedral 1)` in the **Label** text field.

3 Locate the **Input Entities** section. Under **Selections to invert**, click **Add**.

4 In the **Add** dialog box, select **Swept 1** in the **Selections to invert** list.

5 Click **OK**.



The next bit is about the exterior boundaries of the free tetrahedral domain. For those boundaries that lie on the interface between the swept- and tetrahedral meshes, a mesh conversion is required to make the meshes "connect". For the top- and bottom boundaries on the other hand, special care needs to be taken to make the surface meshes identical. Selections are prepared for both these cases.

Remaining Domains, Exterior Boundaries

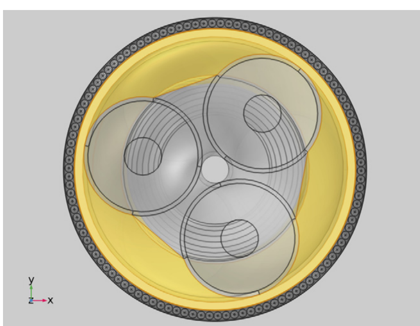
1 In the **Definitions** toolbar, click **Adjacent**.

2 In the **Settings** window for **Adjacent**, type `Remaining Domains, Exterior Boundaries` in the **Label** text field.

3 Locate the **Input Entities** section. Under **Input selections**, click **Add**.

4 In the **Add** dialog box, select **Remaining Domains (Free Tetrahedral 1)** in the **Input selections** list.

5 Click **OK**.



Convert 1

1 In the **Definitions** toolbar, click **Difference**.

2 In the **Settings** window for **Difference**, type `Convert 1` in the **Label** text field.

3 Locate the **Geometric Entity Level** section. From the **Level** list, choose **Boundary**.

4 Locate the **Input Entities** section. Under **Selections to add**, click + **Add**.

5 In the **Add** dialog box, select **Remaining Domains, Exterior Boundaries** in the **Selections to add** list.

6 Click **OK**.

7 In the **Settings** window for **Difference**, locate the **Input Entities** section.

8 Under **Selections to subtract**, click + **Add**.

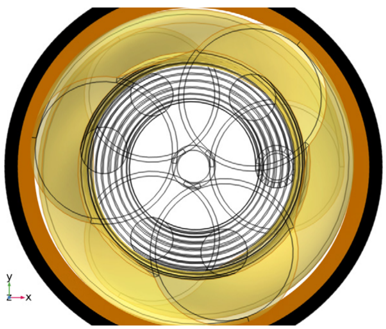
9 In the **Add** dialog box, select **Cross Section, Top and Bottom** in the **Selections to subtract** list.

10 Click **OK**.

Modify the view settings temporarily, to get a better look at the interior boundaries:

11 Click the **Wireframe Rendering** button in the **Graphics** toolbar once (to enable it).

12 Click the **Orthographic Projection** button in the **Graphics** toolbar once (disable it).



You might need to turn the camera a bit to see what is going on here: These are the boundaries that will be exterior to the free tetrahedral mesh, *excluding* the top and bottom periodicity planes. That is; this selection represents the interface between the tetrahedral- and the swept meshes.

The next selections represent the periodicity planes of the free tetrahedral mesh domain.

Copy Face 3, Source Boundaries

1 In the **Definitions** toolbar, click **Intersection**.

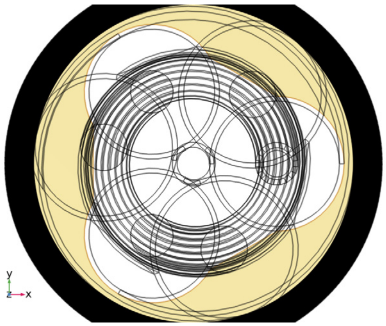
2 In the **Settings** window for **Intersection**, type `Copy Face 3, Source Boundaries` in the **Label** text field.

3 Locate the **Geometric Entity Level** section. From the **Level** list, choose **Boundary**.

4 Locate the **Input Entities** section. Under **Selections to intersect**, click + **Add**.

5 In the **Add** dialog box, in the **Selections to intersect** list, choose **Cross Section, Bottom** and **Remaining Domains, Exterior Boundaries**.

6 Click **OK**.



Copy Face 3, Destination Boundaries

1 In the **Definitions** toolbar, click **Intersection**.

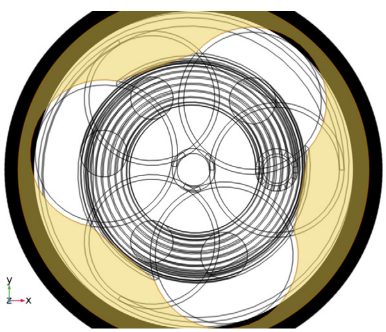
2 In the **Settings** window for **Intersection**, type `Copy Face 3, Destination Boundaries` in the **Label** text field.

3 Locate the **Geometric Entity Level** section. From the **Level** list, choose **Boundary**.

4 Locate the **Input Entities** section. Under **Selections to intersect**, click **Add**.

5 In the **Add** dialog box, in the **Selections to intersect** list, choose **Cross Section, Top** and **Remaining Domains, Exterior Boundaries**.

6 Click **OK**.



Re-enable the orthographic projection to restore the camera to its original state:

7 Click the **Orthographic Projection** button in the **Graphics** toolbar once (to enable it).

8 Click the **Wireframe Rendering** button in the **Graphics** toolbar once (disable it).

Mesh Selections

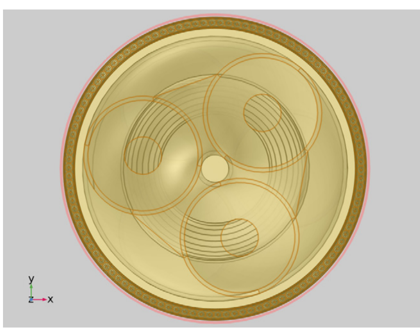
In the **Model Builder** window, collapse the **Mesh Selections** node.

All selections from **Phases, Exterior Boundaries**, to **Copy Face 3, Destination Boundaries** are now part of the group **Mesh Selections**. Note that these *node groups* do not have a special meaning; they merely help you to organize your model. The last group will consist of three boundary selections. They are a subset of the top periodicity plane, and are mainly intended for use in postprocessing.

POSTPROCESSING SELECTIONS

Cable Face, Top

1 In the **Definitions** toolbar, click **Cylinder**.
2 In the **Settings** window for **Cylinder**, type `Cable Face, Top` in the **Label** text field.
3 Locate the **Geometric Entity Level** section. From the **Level** list, choose **Boundary**.
4 Locate the **Input Entities** section. From the **Entities** list, choose **From selections**.
5 Under **Selections**, click + **Add**.
6 In the **Add** dialog box, select **Cross Section, Top** in the **Selections** list.
7 Click **OK**.
8 In the **Settings** window for **Cylinder**, locate the **Size and Shape** section.
9 In the **Outer radius** text field, type `Darm/2+Tarm`.
10 Locate the **Output Entities** section. From the **Include entity if** list, choose **All vertices inside cylinder**.



11 Right-click **Cable Face, Top** and choose **Group**.

Postprocessing Selections

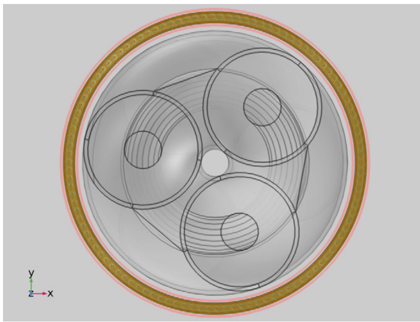
In the **Settings** window for **Group**, type `Postprocessing Selections` in the **Label** text field.

Cable Ring, Top

1 In the **Model Builder** window, right-click **Cable Face, Top** and choose **Duplicate**.

2 In the **Settings** window for **Cylinder**, type `Cable Ring, Top` in the **Label** text field.

3 Locate the **Size and Shape** section. In the **Inner radius** text field, type `Darm/2-Tarm`.



Armor Wire Trio

1 Right-click **Cable Ring, Top** and choose **Duplicate**.

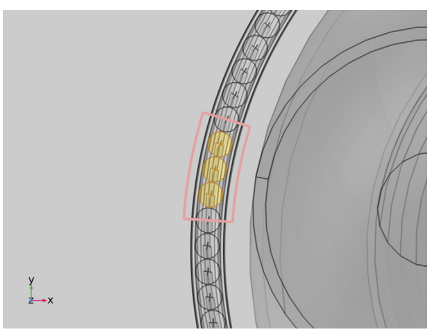
2 In the **Settings** window for **Cylinder**, type `Armor Wire Trio` in the **Label** text field.

3 Locate the **Size and Shape** section. In the **Start angle** text field, type `Tsec-2*360[deg]/Narm`.

4 In the **End angle** text field, type `Tsec+2*360[deg]/Narm`.

5 Click the **Zoom to Selection** button in the **Graphics** toolbar.

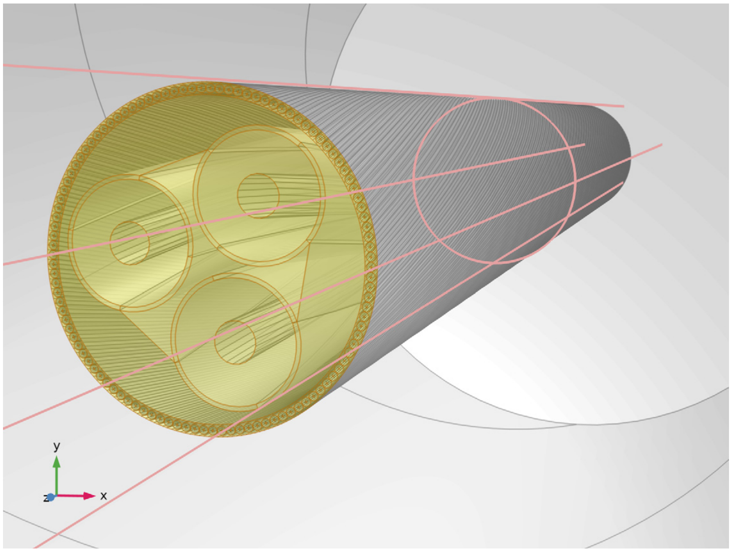
6 Click the **Zoom Out** button in the **Graphics** toolbar, twice.



Cable Face, Top

1 In the **Model Builder** window, click **Cable Face, Top**.

2 Click the **Go to View 5 (Perspective)** button in the **Graphics** toolbar.



The resulting set of selections will be used extensively in the mesh, the physics, and in postprocessing. Preparing them may have taken you some time, but in the subsequent sections — *and the next tutorial* — you will see it was worth the effort.

You have now finished preparing the selections. Save the resulting file, so that you can use it as a basis for the next part.

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

Modeling Instructions — Mesh Sequence

With the geometry and the selections prepared, the biggest hurdle has been taken. Building the mesh should now be pretty straightforward. The procedure consists of four main steps:

- Preparing a surface mesh on the bottom periodicity plane.
- Sweeping the surface mesh through the central core and the outer domains.
- Constructing a free tetrahedral mesh in-between the core and the armor.
- Investigating the mesh quality in postprocessing.

The surface mesh contains details such as boundary layers in the phases and the armor, quadrilateral elements in the screens and between the armor wires, and special features enforcing rapid element growth from the armor wires outwards. The swept mesh extrudes or "projects" this surface mesh along helical paths into the domains, turning the triangles into prisms, and the quad elements into hexahedra.

Since the core (consisting of the phases and the screens) and the armor are twisted in opposite directions, they will have to be swept independently. In-between the core and the armor, a void region will appear where no swept mesh can be used. This region is filled with an unstructured tetrahedral mesh. Special care needs to be taken to ensure the surface meshes on the periodicity planes coincide — this is to obtain a *conforming mesh*. For more details on meshing, see Meshing Considerations.

If you have just finished the previous section, you can continue where you left off. If you intend to start here, you will have to open a reference file from the Application Library. In both cases, it is convenient to resave the file under a new name.

ROOT

1 From the **File** menu, choose **Open**.

2 Browse to the model's Application Libraries folder and double-click the file `submarine_cable_07_c_geom_mesh_3d.mph`.

3 From the **File** menu, choose **Save As**.

4 Browse to a suitable folder and type the filename `submarine_cable_07_geom_mesh_3d.mph`.

MESH 1

Let us start by adding a free triangular mesh to the central parts of the cross section of **Phase 2** (the one with the extra *mesh control entities*).

Free Triangular 1

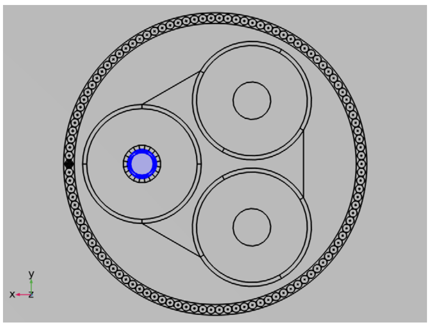
1 In the **Mesh** toolbar, click **Boundary** and choose **Free Triangular**.

2 In the **Settings** window for **Free Triangular**, locate the **Boundary Selection** section.

3 From the **Selection** list, choose **Free Triangular 1**.

4 In the **Graphics** window toolbar, click next to **Go to Default View**, then choose **Go to View 3 (Orthographic, Bottom)**.

5 Click the **Transparency** button in the **Graphics** toolbar once (to disable it).



Disabling the transparency temporarily, allows you to get a clear view of the bottom surface mesh without the rest of the geometry shining through.

Size 1

1 Right-click **Free Triangular 1** and choose **Size**.

2 In the **Settings** window for **Size**, locate the **Element Size** section.

3 Click the **Custom** button.

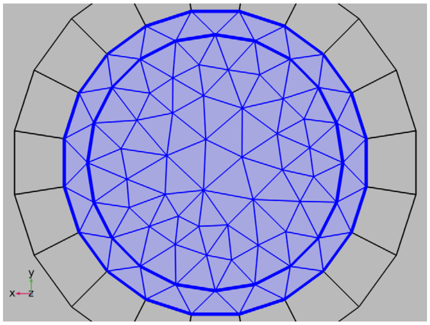
4 Locate the **Element Size Parameters** section.

5 Select the **Maximum element size** check box. In the associated text field, type `Dcon/9`.

6 Select the **Minimum element size** check box. In the associated text field, type `Dcon/9`.

7 Click **Build Selected**.

8 Click the **Zoom to Selection** button in the **Graphics** toolbar.



The element size is set to be nine times smaller than the conductor thickness D_{con}, or about three times smaller than the copper skin depth at 50 Hz. Notice that the element size is chosen not to depend on the skin depth directly — instead, it is geometry dependent. After all, resolving the skin depth is less crucial for the central parts of the phase mesh.

Mapped 1

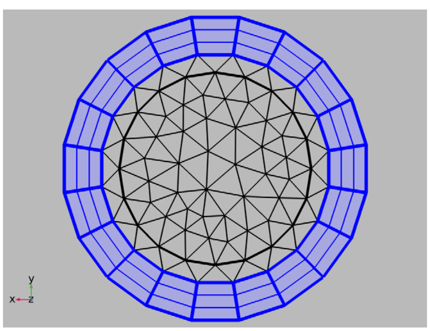
1 In the **Mesh** toolbar, click **Boundary** and choose **Mapped**.

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

3 From the **Selection** list, choose **Mapped 1**.

4 Click **Build Selected**.

5 Click the **Zoom to Selection** button in the **Graphics** toolbar.



This is a different matter for the boundary layer mesh. The boundary layers have been added specifically to help resolve the skin- and proximity effects, while keeping the amount of *degrees of freedom* (the DOF count) low. The layers are set to be eight times thinner than the copper skin depth. For more on this, see the section Boundary Layer Meshes.

You may remember the **Mesh Control Entities (wp3)** work plane in the geometry sequence preparing for this. In fact, every aspect of this boundary layer mesh has been constructed in advance in the geometry- and selection sections of this tutorial. The only thing left to do, was to fill-in the quadrilaterals. This is what **Mapped 1** just did.

Note that the mesh sequence itself is equipped with boundary layer functionality too, but at the time of writing (2019) the approach presented here — adding the layers before sweeping them — turns out to be a bit more efficient. There is not a strict line between what meshing aspects should be considered in the geometry, and what aspects should be considered in the mesh. It partially depends on your own preferences.

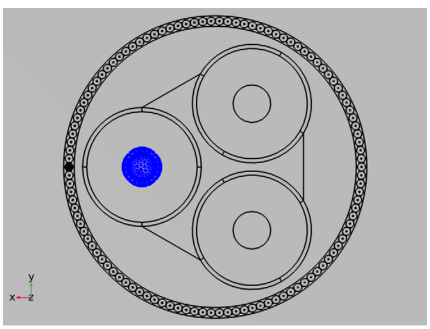
Proceed by copying the surface mesh to the other two phases.

Copy Face 1

1 In the **Mesh** toolbar, click **Copy** and choose **Copy Face**.

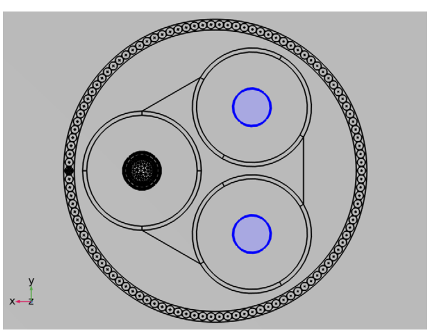
2 In the **Settings** window for **Copy Face**, locate the **Source Boundaries** section.

3 From the **Selection** list, choose **Copy Face 1, Source Boundaries**.

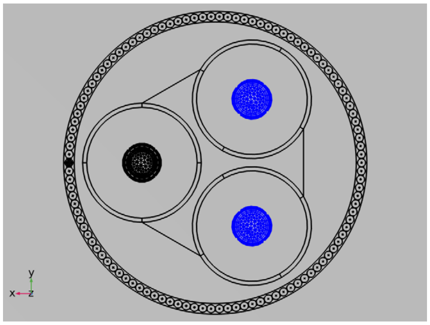


4 Locate the **Destination Boundaries** section. Click to select the **Activate Selection** toggle button.

5 From the **Selection** list, choose **Copy Face 1, Destination Boundaries**.



6 Click **Build Selected**.



Having identical meshes in the three phases is not critically important to obtain conforming meshes on the periodicity planes or anything like that (the swept mesh will see to that), rather, it is a matter of *convenience*: Because of the **Copy Face** operation, you will need to add mesh control entities in the geometry for one phase only.

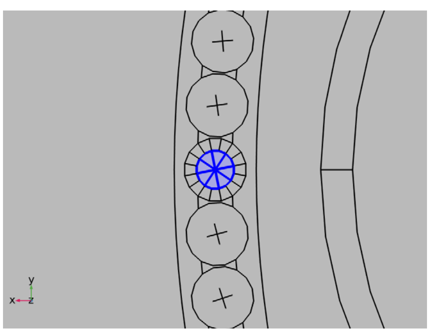
This is especially true for the armor wires. For the armor wires the general procedure is the same, but with even less freedom for the mesh to deviate from the structure dictated by the geometry. Let us continue.

Free Triangular 2

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

2 In the **Settings** window for **Free Triangular**, locate the **Boundary Selection** section.

3 From the **Selection** list, choose **Free Triangular 2**.



4 Click to expand the **Tessellation** section. From the **Method** list, choose **Delaunay**.

This setting affects the structure of the mesh. In this case, the difference between **Advancing front** and **Delaunay** does not really show, but when checked for a whole range of different armor wire diameters, **Delaunay** turns out to give more consistent results. If you wish to see the effect more clearly, you should try it for larger surface areas.

Size 1

1 Right-click **Free Triangular 2** and choose **Size**.

2 In the **Settings** window for **Size**, locate the **Element Size** section.

3 Click the **Custom** button.

4 Locate the **Element Size Parameters** section.

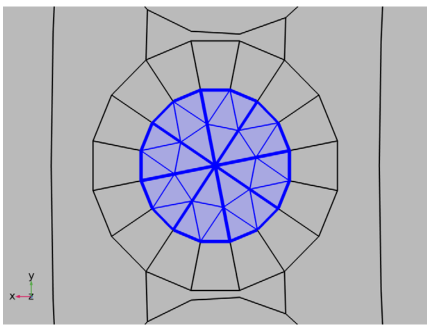
5 Select the **Maximum element size** check box. In the associated text field, type `Tarm/4`.

6 Select the **Minimum element size** check box. In the associated text field, type `Tarm/4`.

7 Click **Build Selected**.

8 Click the **Zoom to Selection** button in the **Graphics** toolbar.

9 Click the **Zoom Out** button in the **Graphics** toolbar.



Mapped 2

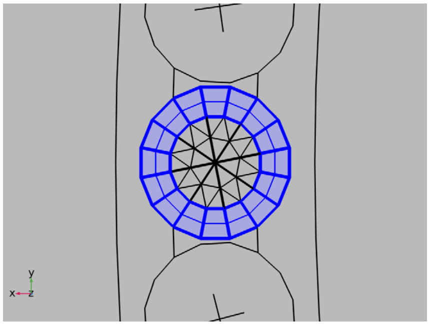
1 In the **Mesh** toolbar, click **Boundary** and choose **Mapped**.

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

3 From the **Selection** list, choose **Mapped 2**.

4 Click **Build Selected**.

5 Click the **Zoom Out** button in the **Graphics** toolbar.



This mesh is actually quite important. It has been optimized to resolve a cone-shaped eddy (in-plane), a linear current density distribution (out-of-plane), as well as the magnetic flux densities associated with them (both out-of-plane and in-plane) — see the *Inductive Effects 3D* tutorial.

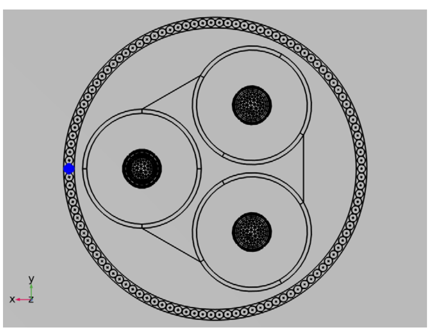
Note that this is also the part of the mesh where most degrees of freedom are generated: *If you wish to cut down on the DOF count without losing too much accuracy, switching to a single boundary layer here would be most effective.*

Copy Face 2

1 In the **Mesh** toolbar, click **Copy** and choose **Copy Face**.

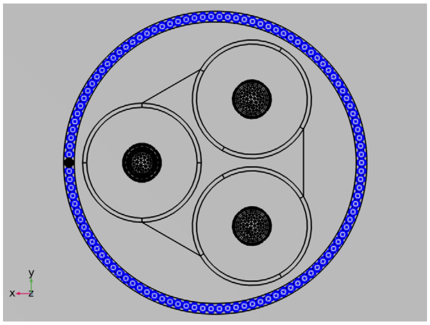
2 In the **Settings** window for **Copy Face**, locate the **Source Boundaries** section.

3 From the **Selection** list, choose **Copy Face 2, Source Boundaries**.

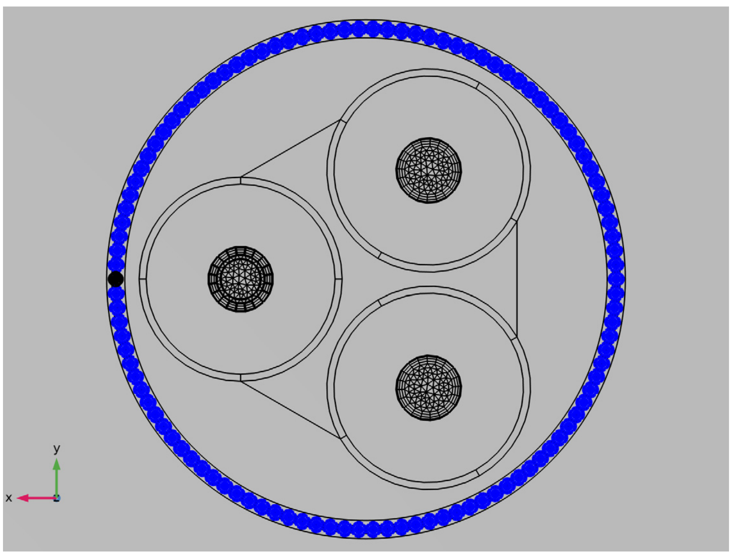


4 Locate the **Destination Boundaries** section. Click to select the **Activate Selection** toggle button.

5 From the **Selection** list, choose **Copy Face 2, Destination Boundaries**.



6 Click **Build Selected**.

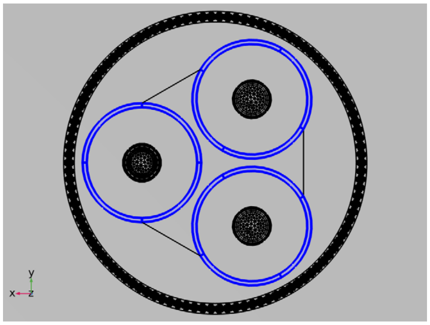


Mapped 3

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

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

3 From the **Selection** list, choose **Screens, Boundaries Bottom (Mapped 3)**.



For the screens, the skin depth is so large (twelve times larger than the screen thickness) that a single layer of elements should suffice.

Size 1

1 Right-click **Mapped 3** and choose **Size**.

2 In the **Settings** window for **Size**, locate the **Element Size** section.

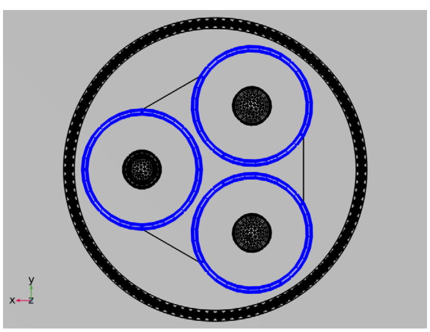
3 Click the **Custom** button.

4 Locate the **Element Size Parameters** section.

5 Select the **Maximum element size** check box. In the associated text field, type `3*Tpbs`.

6 Select the **Minimum element size** check box. In the associated text field, type `3*Tpbs`.

7 Click **Build Selected**.



Indirectly, the size expression "`3*Tpbs`" sets the aspect ratio of the quadrilaterals, to three. The size is primarily set to get a nice resolution in the angular direction. In other words, the size node determines whether the circular screens turn into hexadecagons, or icosagons, for example.

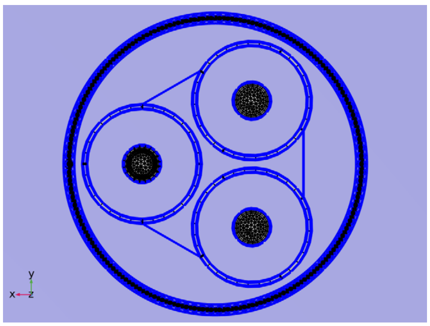
Now, the metallic parts of the cross section have been equipped with an appropriate mesh — *taking the to-be-resolved fields and the DOF count into consideration*. What remains are the insulators. Of particular importance are the gaps between the armor wires, but before switching to those, let us have a look at the rest of the periodicity plane:

Free Triangular 3

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

2 In the **Settings** window for **Free Triangular**, locate the **Boundary Selection** section.

3 From the **Selection** list, choose **Free Triangular 3**.

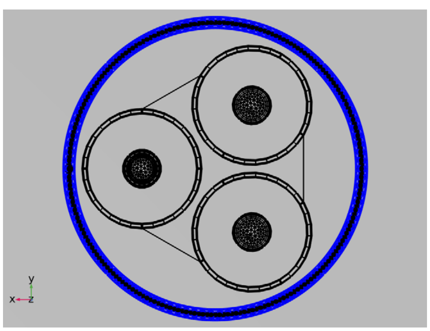


Size 1

1 Right-click **Free Triangular 3** and choose **Size**.

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

3 From the **Selection** list, choose **Free Triangular 3, Size 1**.



4 Locate the **Element Size** section. Click the **Custom** button.

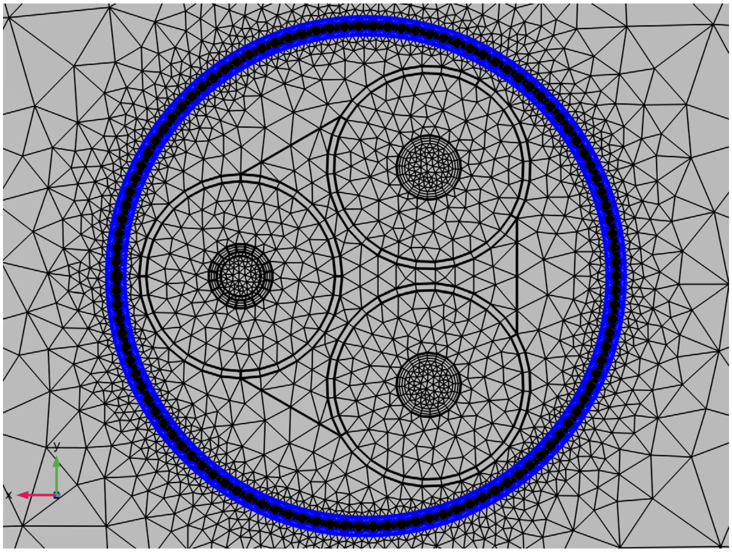
5 Locate the **Element Size Parameters** section.

6 Select the **Maximum element size** check box. In the associated text field, type `Tarm`.

7 Select the **Minimum element size** check box. In the associated text field, type `Tarm`.

8 Select the **Maximum element growth rate** check box. In the associated text field, type 2.

9 Click **Build Selected**.



You may remember the two rings enclosing the armor wires, as drawn in the work plane **Cable Armor and Sea Bed (wp2)**, see Modeling Instructions — Geometry Sequence. These rings, together with the settings used here, in **Free Triangular 3>Size 1**, see to it that the mesh grows very rapidly from the armor outwards — a factor of three within one layer of mesh elements. We will get back to this shortly.

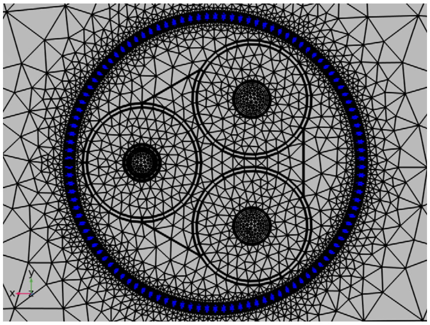
First, you can finish the periodicity plane by putting some quadrilateral elements in the gaps between the wires.

Mapped 4

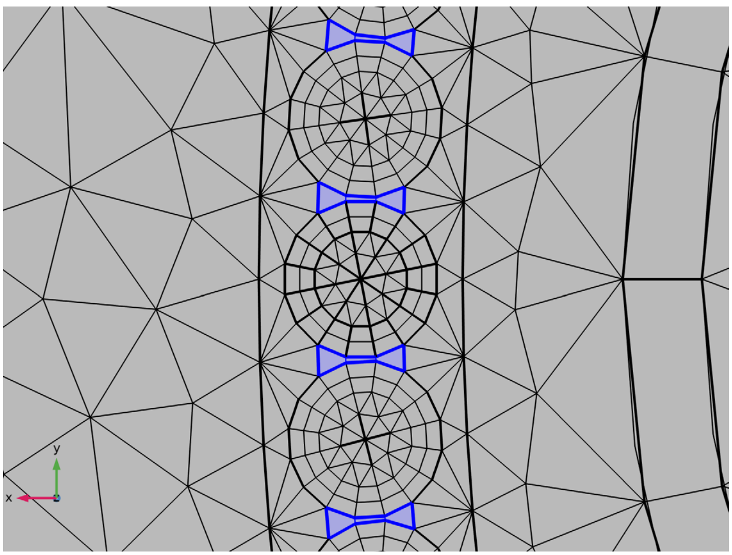
1 In the **Mesh** toolbar, click **Boundary** and choose **Mapped**.

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

3 From the **Selection** list, choose **Mapped 4**.



4 Click **Build Selected**.



As clearly seen in this close-up, there is a rapid growth close to the armor wires.

This element growth is important for several reasons (for more on this, see section Swept Meshes and Tetrahedral Meshes):

- First of all, the rapid growth will reduce the DOF count. This is the area where most DOFs come from, while the field does not require a very fine mesh outside the wires.
- Secondly, it is beneficial for the quality of the tetrahedra. The armor wires and their neighboring domains will be using a *structured* (swept) mesh, and so will the screens. In-between these domains, an *unstructured* (free tetrahedral) mesh is used. Every bit of growth that can be squeezed into the first layer of mesh elements close to the armor

wires — still within the swept mesh — does not need to be provided by the tetrahedra. This is advantageous since structured meshes are more robust when it comes to rapid growth.

- Furthermore, it improves the interface between the swept mesh and the tetrahedra. When the swept mesh for the armor is built, a number of out-of-plane elements is generated (the surface mesh is extruded). To limit the DOF count the amount of elements in the direction of extrusion is kept low, so the resulting elements will have a high *aspect ratio*. Since a large in-plane element size will lower this aspect ratio, and since having a low aspect ratio on the swept-tetrahedral interface is preferable, rapid growth within the swept mesh is needed.

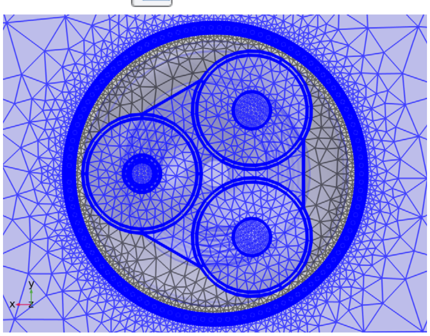
Finally, in the close-up, notice the extremely thin quadrilateral element in the gaps. Putting a triangular mesh here, is expected to be less practical. The triangles would introduce more degrees of freedom for sure, but perhaps more importantly; they would be of *very poor quality* — they would have angles close to zero. The quadrilateral elements do not suffer from this issue (for more, see section Meshing the Gaps).

SWEPT MESH AND FREE TETRAHEDRAL MESH

Now that the surface mesh is finished, you can start preparing for the swept mesh. Reintroduce transparency, so that you can see the interior of the geometry.

Swept 1

1 In the **Mesh** toolbar, click **Swept**.
2 In the **Settings** window for **Swept**, locate the **Domain Selection** section.
3 From the **Geometric entity level** list, choose **Domain**.
4 From the **Selection** list, choose **Swept 1**.
5 Click the **Transparency** button in the **Graphics** toolbar.

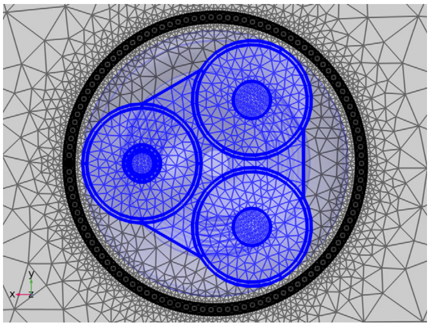


Distribution 1

1 Right-click **Swept 1** and choose **Distribution**.

2 In the **Settings** window for **Distribution**, locate the **Domain Selection** section.

3 From the **Selection** list, choose **Swept 1, Distribution 1**.



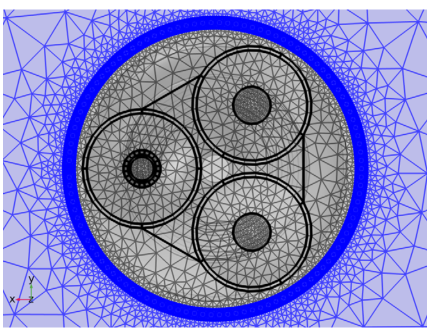
4 Locate the **Distribution** section. In the **Number of elements** text field, type `round(Lsec/(Dpha/3))`.

Distribution 2

1 In the **Model Builder** window, right-click **Swept 1** and choose **Distribution**.

2 In the **Settings** window for **Distribution**, locate the **Domain Selection** section.

3 From the **Selection** list, choose **Swept 1, Distribution 2**.

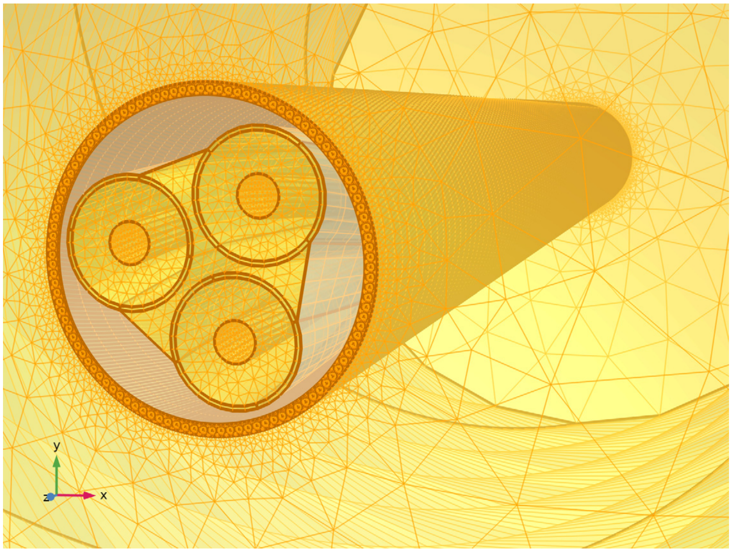


4 Locate the **Distribution** section. In the **Number of elements** text field, type `round(Lsec/(3*Tarm))`.

Swept 1

1 Right-click **Swept 1** and choose **Build Selected** (this should take a minute or so).

2 In the **Settings** window for **Swept**, in the **Graphics** window toolbar, click ▼ next to **Go to Default View**, then choose **Go to View 5 (Perspective)**.



Here, you have applied a single swept mesh, to two separate groups of domains.

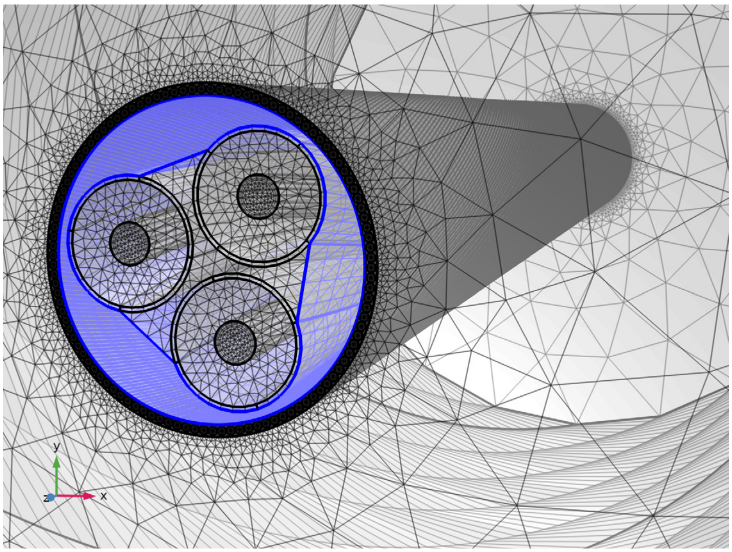
Each group uses its own distribution settings. For the main conductors and the screens, the number of elements is set to "`round(Lsec/(Dpha/3))`", so the generated prisms and hexahedra will be about $D_{pha}/3$ long. The width of the hexahedra in the screens is $3T_{pbs}$, as set in **Mapped 3>Size 1**. Consequently, the aspect ratio of the rectangular faces that form on the exterior of the central swept mesh, is about $D_{pha}/(9T_{pbs})$, or "about three and a half".

A similar reasoning holds for the outer swept mesh. Here, the faces will be $3T_{arm}$ long, and $T_{arm}/2$ wide (thanks to the rapid growth discussed earlier). This results in an aspect ratio of six. So for both distributions the out-of-plane size of the elements has been given in terms of cross-sectional dimensions, to ensure a certain aspect ratio on the swept-tetrahedral interface.

This interface will need some further preparation though. The tetrahedra cannot connect to these swept meshes directly, as the boundary mesh should be triangular for that (not quadrilateral; tetrahedra do not have any rectangular faces). If you would add a free tetrahedral mesh now, COMSOL would cover the interface with *pyramid elements* to make the conversion. In most cases this works fine, but in this geometry there is only a very small distance over which the conversion needs to take place. The alternative is to use a mesh conversion.

Convert 1

1 In the **Mesh** toolbar, click **Modify** and choose **Convert**.
2 In the **Settings** window for **Convert**, locate the **Geometric Entity Selection** section.
3 From the **Geometric entity level** list, choose **Boundary**.
4 From the **Selection** list, choose **Convert 1**.
5 Click **Build Selected**.



On the interface, diagonals are inserted into the rectangular faces. The prisms and hexahedra that are in direct contact with the interface, have been cut into tetrahedral elements and pyramid elements. The quality of these elements is not particularly good. In fact, this is where the lowest element quality comes from. Even so, it is still preferable to having the conversion in the *unstructured* tetrahedral domain.

Having the conversion there, would mean going from hexahedron ($D_{\mathrm{pha}}/3$ long), to pyramid, to tetrahedron, to pyramid, to prism ($3T_{\mathrm{arm}}$ long), all within a 2–3 element wide space (that is; where the screens and the armor meet). Doing this in an unstructured way — where the mesh nodes are expected to distribute themselves such that on average the best element quality is achieved — is a challenging optimization problem to say the least. And it may not always give the same outcome.

So the conversion step applied to the structured mesh may not give a very good element quality in this case, but at least it is reliable in the sense that it is *deterministic*: The elements are constructed in a specific, well-controlled way.

Now, before adding the free tetrahedral mesh there is one thing left to do; to ensure conforming meshes on the periodicity planes. Basically this means that — apart from a rotation — the surface meshes need to be identical (see section Conforming Meshes). For the swept mesh, this property comes out-of-the-box: The swept mesh already does a direct projection of the mesh from one periodicity plane to the other. For the tetrahedral mesh however, this needs to be enforced by means of a **Copy Face** operation.

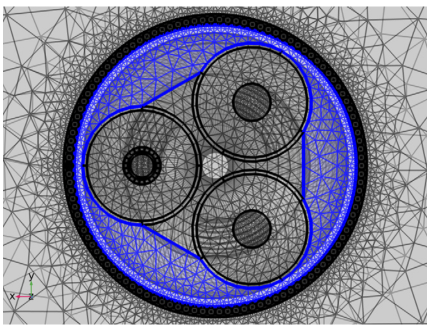
Copy Face 3

1 In the **Mesh** toolbar, click **Copy** and choose **Copy Face**.

2 In the **Settings** window for **Copy Face**, locate the **Source Boundaries** section.

3 From the **Selection** list, choose **Copy Face 3, Source Boundaries**.

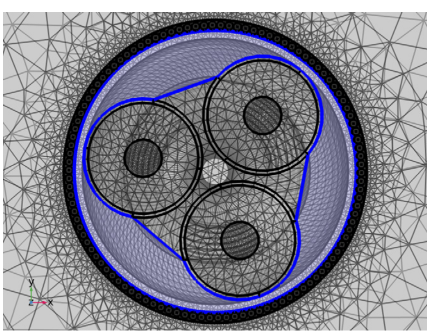
4 In the **Graphics** window toolbar, click ▾ next to **Go to Default View**, then choose **Go to View 3 (Orthographic, Bottom)**.



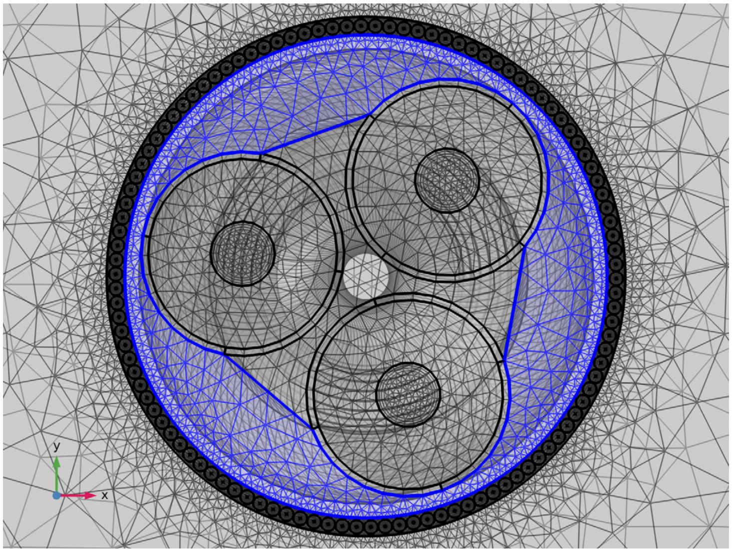
5 Locate the **Destination Boundaries** section. Click to select the **Activate Selection** toggle button.

6 From the **Selection** list, choose **Copy Face 3, Destination Boundaries**.

7 In the **Graphics** window toolbar, click ▾ next to **Go to Default View**, then choose **Go to View 2 (Orthographic, Top)**.



8 Click **Build Selected**.



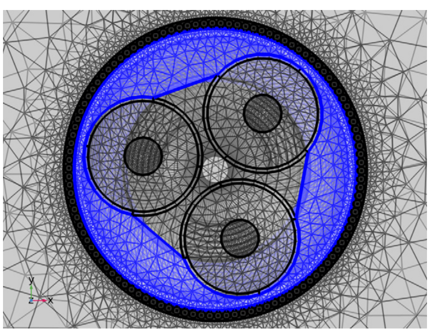
Free Tetrahedral 1

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

2 In the **Settings** window for **Free Tetrahedral**, locate the **Domain Selection** section.

3 From the **Geometric entity level** list, choose **Domain**.

4 From the **Selection** list, choose **Remaining Domains (Free Tetrahedral 1)**.



5 Click to expand the **Scale Geometry** section. In the **z-direction scale** text field, type 1/6.

6 Click to expand the **Element Quality Optimization** section. From the **Optimization level** list, choose **High**.

The **z direction scale** setting basically “squeezes” the geometry (and the mesh) by a factor of six, when putting in the tetrahedra. The tetrahedra will then perceive the rectangles on the swept-tetrahedral interface as squares — or parallelograms with a modest aspect ratio and a diagonal, rather — resulting in a smooth connection. When the procedure is finished, the tetrahedra will have been stretched by a factor of six in the longitudinal direction of the cable. As a bonus, this helps to cut down the DOF count.

Size 1

1 Right-click **Free Tetrahedral 1** and choose **Size**.

2 In the **Settings** window for **Size**, locate the **Element Size** section.

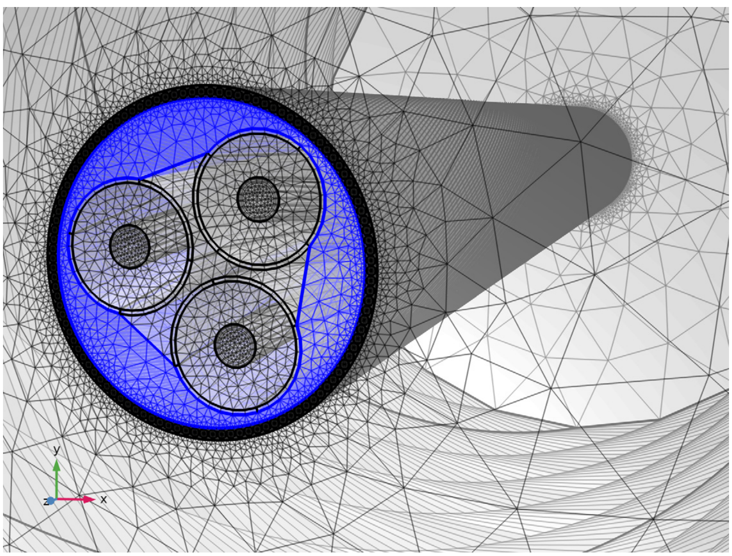
3 Click the **Custom** button.

4 Locate the **Element Size Parameters** section.

5 Select the **Maximum element growth rate** check box. In the associated text field, type 2.

6 Click **Build Selected** (this should take a minute or so).

7 In the **Graphics** window toolbar, click ▼ next to **Go to Default View**, then choose **Go to View 5 (Perspective)**.



8 Right-click **Component 1 (comp1)>Mesh 1** and choose **Statistics**.

9 In the **Settings** window for **Mesh**, check the following statistics:

Property	Value
Number of elements	1400118
Minimum element quality	0.02807
Average element quality	0.6293
Element volume ratio	1.159E-5
Mesh volume	1.51[m^3]

The actual numbers you get will probably deviate (by less than a percent), and will depend on your operating system, COMSOL version, and geometry representation — *COMSOL kernel*, or *CAD kernel*.

As far as meshes go, this one does not give the best statistics. Not when considering the **Element Quality Histogram** at least. Ideally, it should be a nice bell shape with a not-too-large standard deviation and a good average element quality. Here, "good" means the aspect ratio of the elements should be close to 1. Assuming your material properties and your physics are fairly isotropic, an isotropic mesh will lead to a friendly matrix structure; iterative solvers should like it.

This reasoning is very general however — *as COMSOL is able to handle a vast amount of different kinds of physics* — and the element quality measure does not take into account the extreme aspect ratio of the features included in the cable. If you would put a nice isotropic tetrahedral mesh in this model (while still resolving the skin depth of course), the amount of degrees of freedom would quickly rise to 30 million or more. Although iterative solvers may still be able to handle that, it turns out not to be the most efficient way to go about.

Instead, this mesh has been optimized to a great extent, to be used with a direct solver — more precisely: *PARDISO* (*MUMPS* should work too, if pivoting is limited). Direct solvers are more forgiving when it comes to the mesh quality, but they do consume more memory. Luckily, desktop machines with 32–64 GB of RAM and several hundred gigabytes of SSD swap drive capacity are very affordable nowadays (2019).

So the main goal has been to find a mesh that resolves the geometry and the physics as best as possible, while still keeping the DOF count low enough to have a reasonable solving time on a modern desktop machine. *This is actually one of the more challenging exercises in this tutorial series.*

MESH 1

Let us investigate the mesh further using some postprocessing tools.

1 In the **Mesh** toolbar, click **Plot** (this should take a minute or so).

RESULTS

Mesh 1

On the following pages some postprocessing tools are demonstrated that will help you to get more insight in the structure of the mesh. They will show you how mesh quality can be investigated. To finish the tutorial, a plot is generated that checks the mesh conformity on the periodicity planes.

The default plot shows the quality by skewness for all element types. Start by modifying it.

Mesh Quality, Volume Elements

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

2 In the **Settings** window for **3D Plot Group**, type `Mesh Quality, Volume Elements` in the **Label** text field.

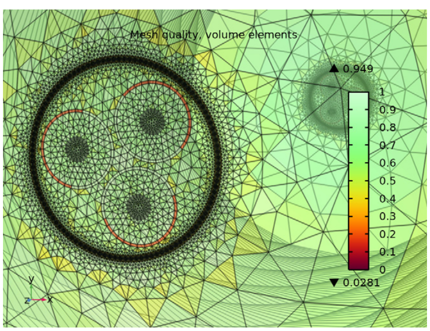
3 Click to expand the **Title** section. From the **Title type** list, choose **Manual**.

4 In the **Title** text area, type `Mesh quality, volume elements`.

5 Locate the **Plot Settings** section. From the **View** list, choose **View 5 (Perspective)**.

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

7 In the **Mesh Quality, Volume Elements** toolbar, click **Plot**.



Mesh 1

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

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

3 Click **Change Color Table**.

4 In the **Color Table** dialog box, select **Wave>WaveLight** in the tree.

5 Click **OK**.

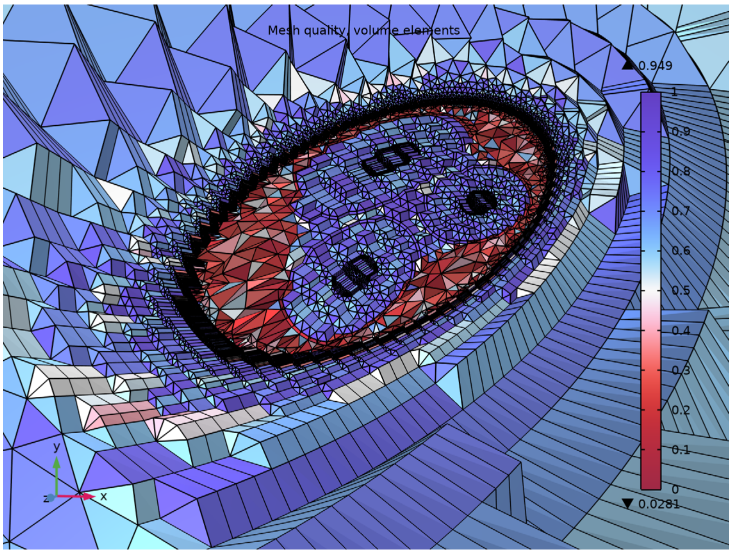
6 In the **Settings** window for **Mesh**, click to expand the **Element Filter** section.

7 Select the **Enable filter** check box.

8 In the **Expression** text field, type `z<Lsec/2-2*Nper*(Dcab+x+y)`.

9 In the **Mesh Quality, Volume Elements** toolbar, click **Plot**.

10 Click the **Transparency** button in the **Graphics** toolbar.



Most of the settings you have modified here are pretty common and do not really need a further explanation. The main thing to keep in mind, is the **Element Filter**. In this case the element filter is coordinate-based, but it could be based on element type, size, or quality for example. *Hint: use the green-red arrows to find interesting quantities for your expression.*

Speaking of which, let us have a further look at the poor quality elements.

Mesh Quality, Poor Quality Elements

1 In the **Model Builder** window, right-click **Mesh Quality, Volume Elements** and choose **Duplicate**.

2 In the **Settings** window for **3D Plot Group**, type `Mesh Quality, Poor Quality Elements` in the **Label** text field.

3 Locate the **Title** section. In the **Title** text area, type `Mesh quality, poor quality elements`.

Mesh 1

1 In the **Model Builder** window, expand the **Mesh Quality, Poor Quality Elements** node, then click **Mesh 1**.

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

3 From the **Element color** list, choose **White**.

4 From the **Wireframe color** list, choose **None**.

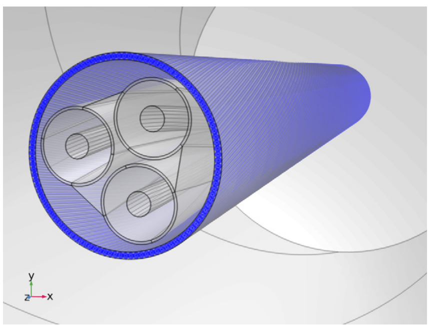
Selection 1

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

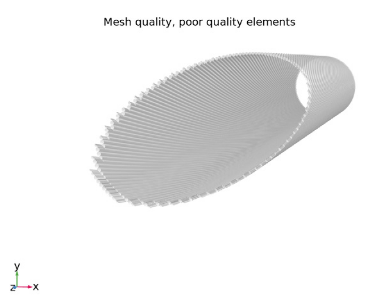
2 In the **Settings** window for **Selection**, locate the **Selection** section.

3 From the **Selection** list, choose **Cable Armor**.

4 Click the **Transparency** button in the **Graphics** toolbar.



5 In the **Mesh Quality, Poor Quality Elements** toolbar, click **Plot**.



Mesh 2

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

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

3 From the **Wireframe color** list, choose **Black**.

4 Locate the **Element Filter** section. Clear the **Enable filter** check box.

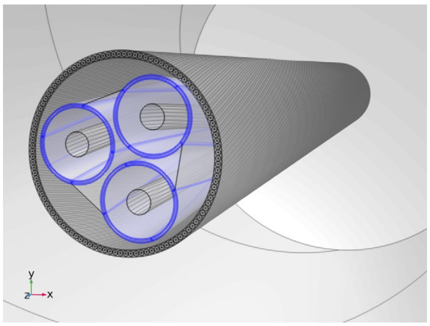
Selection 1

1 In the **Model Builder** window, expand the **Mesh 2** node, then click **Selection 1**.

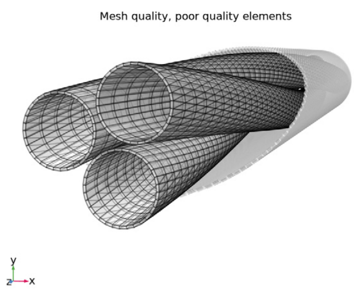
2 In the **Settings** window for **Selection**, locate the **Selection** section.

3 Click to select the **Activate Selection** toggle button.

4 From the **Selection** list, choose **Screens**.



5 In the **Mesh Quality, Poor Quality Elements** toolbar, click **Plot**.



Notice that this intermediate result contains no real information yet — apart from the structure of the armor and the screens. While most plots you encounter will display a field quantity of some kind, having a surface, volume, or mesh plot that shows nothing more than a solid color, is not at all uncommon. In fact, it is a great way to make your plots more insightful (or in some cases just *more appealing*).

Next, will be the part of the plot showing the actual information: What the poor quality elements look like, and where they are located.

Mesh 3

1 In the **Model Builder** window, right-click **Mesh Quality, Poor Quality Elements** and choose **Mesh**.

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

3 From the **Level** list, choose **Volume**.

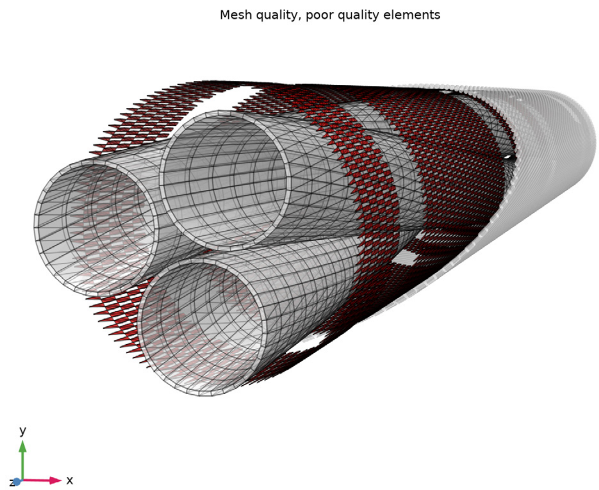
4 Locate the **Coloring and Style** section. From the **Element color** list, choose **Red**.

5 Locate the **Element Filter** section. Select the **Enable filter** check box.

6 From the **Criterion** list, choose **Worst quality**.

7 In the **Fraction** text field, type `0.01`.

8 In the **Mesh Quality, Poor Quality Elements** toolbar, click **Plot**.



The lowest quality elements are all adjacent to the swept-tetrahedral interface. These elements are generated by **Convert 1**. Since they all have the same quality (because of the structured mesh), the “rings” of poor quality elements in the plot actually show an arbitrary subset (because of the **Fraction** setting).

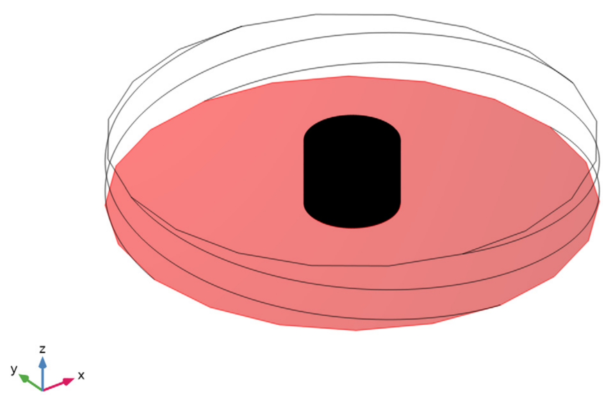
But these elements we knew about. We know what causes them, we know what quality they have, and thanks to the *Inductive Effects 3D* tutorial, we know their quality is sufficient to build a numerically stable cable model. The main thing of interest here, is whether there are any *unexpected* poor quality elements. Feel free to crank up the **Fraction** setting a bit, to `0.04`, `0.05`, or `0.06`. You can also look for certain element types, such as tetrahedron, pyramid, prism, or hexahedron.

CONFORMING MESHES

For checking the surface mesh on the periodicity planes, a 2D plot is more suitable. This 2D plot will need a 2D dataset. Two of them actually, one for the bottom plane, and one for the top.

Cut Plane 1

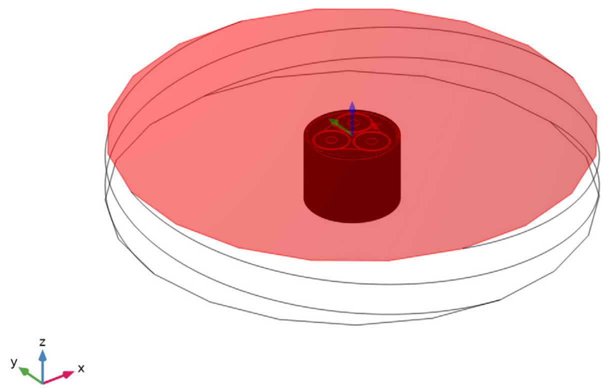
1 In the **Results** toolbar, click **Cut Plane**.
2 In the **Settings** window for **Cut Plane**, locate the **Plane Data** section.
3 From the **Plane** list, choose **xy-planes**.
4 In the **z-coordinate** text field, type `-Lsec/2`.
5 Click **Plot**.
6 In the **Graphics** window toolbar, click ▾ next to **Go to Default View**, then choose **Go to View 1 (Orthographic)**.
7 Click the **Zoom Extents** button in the **Graphics** toolbar.



Cut Plane 2

1 In the **Results** toolbar, click **Cut Plane**.
2 In the **Settings** window for **Cut Plane**, locate the **Plane Data** section.
3 From the **Plane** list, choose **xy-planes**.
4 In the **z-coordinate** text field, type `Lsec/2`.

5 Click **Plot**.



Mesh Comparison, Source and Destination

1 In the **Results** toolbar, click **2D Plot Group**.

2 In the **Settings** window for **2D Plot Group**, type `Mesh Comparison, Source and Destination` in the **Label** text field.

3 Click to expand the **Title** section. From the **Title type** list, choose **Manual**.

4 In the **Title** text area, type `Mesh comparison, source and destination`.

Surface 1

1 Right-click **Mesh Comparison, Source and Destination** and choose **Surface**.

2 In the **Settings** window for **Surface**, locate the **Expression** section.

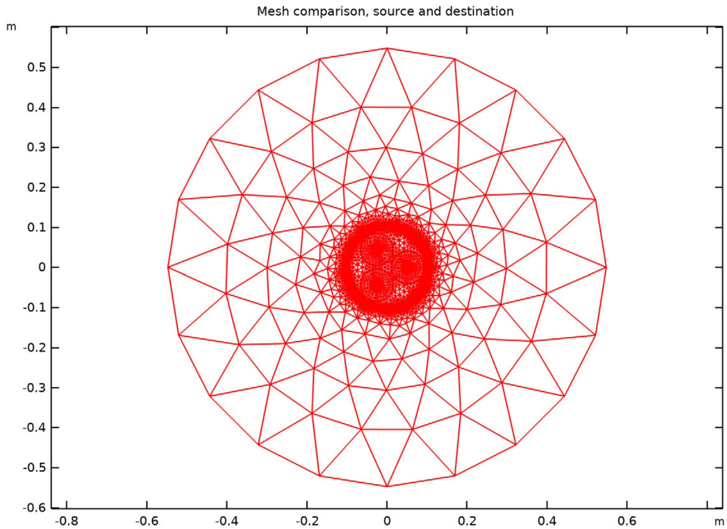
3 In the **Expression** text field, type `1`.

(Note that the actual expression does not really matter here, use "`1`" to have something reasonable).

4 Locate the **Coloring and Style** section. From the **Coloring** list, choose **Uniform**.

5 Select the **Wireframe** check box.

6 In the **Mesh Comparison, Source and Destination** toolbar, click **Plot**.



The **Wireframe** setting can be really convenient. It can be used in any surface plot, to plot the used mesh (not just plots based on mesh datasets).

Surface 2

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

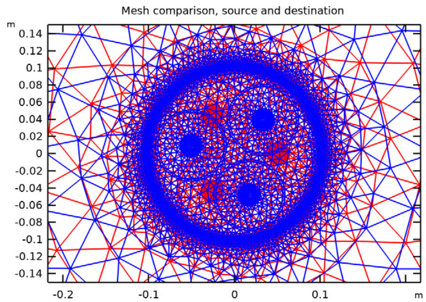
2 In the **Settings** window for **Surface**, locate the **Data** section.

3 From the **Dataset** list, choose **Cut Plane 2**.

4 Locate the **Coloring and Style** section. From the **Color** list, choose **Blue**.

5 In the **Mesh Comparison, Source and Destination** toolbar, click **Plot**.

6 Click the **Zoom In** button in the **Graphics** toolbar, twice.



In the 2D datasets **Cut Plane 1** and **Cut Plane 2**, you have selected the *x,y*-plane from the global coordinate system in the 3D geometry to provide the *x,y*-coordinates for the 2D plot. Consequently, this plot can be seen as an orthographic projection (straight from the top). As you can see, the two plots do not coincide: One is rotated with respect to the other, by an angle equal to `Tsec`, or "about 170°". This is caused by the cable's twist — *the logic behind this, is explained in the next tutorial.*

The surface plot allows you to add a **Deformation**. Deformations are typically used when plotting physical quantities that actually involve a deformation, such as those occurring in *Structural Mechanics* models. In this case, you can use the **Deformation** to apply a rotation to the second surface plot, to compensate for the twist.

In 2D, a rotation about the origin is given by the coordinate transformation:

$$x' = x\ \cos(\theta) - y\ \sin(\theta)$$

$$y' = x\ \sin(\theta) + y\ \cos(\theta)$$

The angle will be entered with a minus sign, to compensate for the rotation. Furthermore, since the **Deformation** requires a displacement as input — rather than a location — you will have to subtract the original coordinates. Finally, a factor of 0.002 times *x,y* is added to include some scaling.

Deformation 1

1 Right-click **Surface 2** and choose **Deformation**.

2 In the **Settings** window for **Deformation**, locate the **Expression** section.

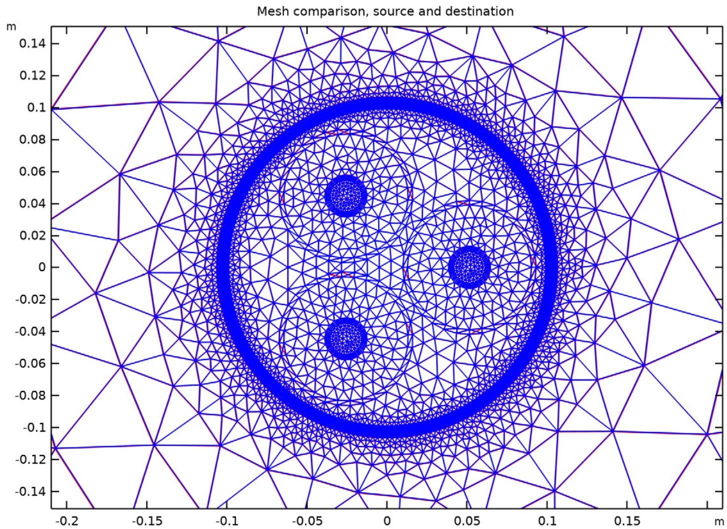
3 In the **x-component** text field, type `(x*cos(-Tsec)-y*sin(-Tsec))-x+0.002*x`.

4 In the **y-component** text field, type `(x*sin(-Tsec)+y*cos(-Tsec))-y+0.002*y`.

5 Locate the **Scale** section.

6 Select the **Scale factor** check box. In the associated text field, type `1`.

7 In the **Mesh Comparison, Source and Destination** toolbar, click **Plot**.



Apart from a few diagonals in the quadrilateral elements (a plotting artifact), the meshes should coincide almost perfectly — "almost", because of the deliberate scaling. The scaling is there for your convenience: If you set the constant "`0.002`" to zero the plots will match perfectly, and it may not be clear whether the red one is present in the first place (note that the built-in **Scale factor** is effectively disabled, because it would scale the rotation too). Feel free to toy with these settings, to gain some further insight.

Having checked the mesh conformity on the periodicity planes, you have now completed this tutorial. Subsequent tutorials will refer to the resulting file as `submarine_cable_07_geom_mesh_3d.mph`.

The next tutorial in this series will take this file as a starting point, and add a detailed electromagnetic analysis. Currents will be applied and the resulting inductive effects will be investigated, *including the magnetic flux, the eddies, and the losses inside the armor*.

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