CST Studio Suite

Low Frequency Simulation

3DEXPERIENCE®

Workflow &
Solver Overview

Version 2020.0 - 8/16/2019

Copyright

Trademarks

3DS.com/SIMULIA

Table of contents

Chapter 1 – Introduction

Welcome

Welcome to CST EM Studio®, the powerful and easy-to-use electromagnetic field simulation software. This program combines a user-friendly interface with unsurpassed simulation performance.

CST EM Studio is part of CST Studio Suite®. Please refer to the *CST Studio Suite Getting Started* manual first. The following explanations assume that you already installed the software and familiarized yourself with the basic concepts of the user interface.

How to Get Started Quickly

We recommend that you proceed as follows:

1. Read the *CST Studio Suite Getting Started* manual.
2. Work through this document carefully. It provides all the basic information necessary to understand the advanced documentation.
3. Look at the examples provided in the Component Library (*File: Component Library ⇨ Examples*). Especially the examples which are tagged as *Tutorial* provide detailed information of a specific simulation workflow. Press the *Help* button of the individual component to get to the help page of this component. Please note that all these examples are designed to give you a basic insight into a particular application domain. Real-world applications are typically much more complex and harder to understand if you are not familiar with the basic concepts.
4. Start with your own first example. Choose a reasonably simple example, which will allow you to quickly become familiar with the software.
5. After you have worked through your first example, contact technical support for hints on possible improvements to achieve even more efficient usage of the software.

What is CST EM Studio?

CST EM Studio is a fully featured software package for electromagnetic analysis and design of electrostatic, magnetostatic, stationary current and low-frequency devices. It simplifies the process of creating the structure by providing a powerful graphical solid modeling front end, which is based on the ACIS modeling kernel. After the model has been constructed, a fully automatic meshing procedure is applied before a simulation engine is started.

A key feature of CST EM Studio is the *Method on Demand* approach, which allows using the solver or mesh type that is best suited to a particular problem. Most solvers support two different meshing strategies:

- ☐ Classic tetrahedral meshes, which provide an explicit representation of the geometry and material interface by a surface mesh. Thus, material interfaces are explicitly resolved by the mesh. Curvilinear mesh elements are especially suited to discretize curved geometries. The geometry resolution is continually improved during an adaptive mesh refinement using the *True Geometry Adaptation* technique.
- ☐ Hexahedral grids in combination with the Perfect Boundary Approximation (PBA®) feature. With hexahedral (Cartesian) meshes, interfaces of materials and solids are not represented by surface mesh cells. Therefore, the meshing algorithm is very robust, and meshes can be generated even for very complex CAD geometries. The PBA feature significantly increases the accuracy of the simulation in comparison to conventional Cartesian mesh simulators.

The software contains five different solvers that best fit their particular applications:

- Electrostatic solver
- Magnetostatic solver
- Stationary current solver

- LF Frequency Domain solver
  - magnetoquasistatic
  - electroquasistatic
  - full-wave
- LF Time Domain solver
  - magnetoquasistatic
  - electroquasistatic

If you are unsure which solver best suits your needs, please consult the online help or contact your local sales office for further assistance.

Simulation results from each solver can be visualized with a variety of different options. Again, a strongly interactive interface will help you quickly achieve the desired insight into your device.

The last – but certainly not least – outstanding feature is the full parameterization of the structure modeler, which enables the use of variables in the definition of your component. In combination with the built-in optimizer and parameter sweep tools, CST EM Studio is capable of both the analysis and the design of electromagnetic devices.

Who Uses CST EM Studio?

Anyone who seeks to solve static or low-frequency electromagnetic problems can use CST EM Studio. The program is especially suited to the fast, efficient analysis and design of components like actuators, insulators, shielding problems, sensors, transformers, electrical machines, etc. Since the underlying method is a general 3D approach, CST EM Studio can solve virtually any static and low-frequency field problem.

CST EM Studio Key Features

The following list gives you an overview of CST EM Studio's main features. Note that not all of these features may be available to you because of license restrictions. Contact a sales office for more information.

General

- ☐ Native graphical user interface based on Windows 7, Windows 2008 Server R2, Windows 8, Windows 2012 Server, Windows 8.1, Windows 2012 Server R2 or Windows 10
- ☐ The structure can be viewed either as a 3D model or as a schematic. The latter allows for easy coupling of EM simulation with circuit simulation.
- ☐ Various independent types of solver strategies (based on hexahedral as well as tetrahedral meshes) allow accurate results with a high performance for all kind of low frequency applications.
- ☐ For specific solvers, highly advanced numerical techniques offer features like PERFECT BOUNDARY APPROXIMATION (PBA)® for hexahedral grids and curved and higher order elements for tetrahedral meshes.

Structure Modeling

- ☐ Advanced ACIS-based, parametric solid modeling front end with excellent structure visualization
- ☐ Feature-based hybrid modeler allows quick structural changes.
- ☐ Import of 3D CAD data from ACIS SAT (e.g. AutoCAD®), ACIS SAB, Autodesk Inventor®, IGES, VDA-FS, STEP, Pro/ENGINEER®, CATIA®, Siemens NX, Parasolid, Solid Edge, SolidWorks, CoventorWare®, Mecadtron®, NASTRAN, STL or OBJ files
- ☐ Import of 2D CAD data from DXF™, GDSII and Gerber RS274X, RS274D files
- ☐ Import of EDA data from design flows including Cadence Allegro® / APD® / SiP®, Mentor Graphics Expedition®, Mentor Graphics PADS®, Mentor Graphics HyperLynx®, Zuken CR-5000® / CR-8000®, IPC-2581 and ODB++® (e.g. Altium Designer, Mentor Graphics Boardstation®, CADSTAR®, Visula®)
- ☐ Import of OpenAccess and GDSII-based integrated-circuit layouts via CST Chip Interface
- ☐ Import of PCB designs originating from CST PCB Studio®
- ☐ Import of 2D and 3D sub models
- ☐ Import of Agilent ADS® layouts

- ☐ Import of Sonnet® EM models
- ☐ Import of a visible human model dataset or other voxel datasets
- ☐ Export of CAD data by ACIS SAT, ACIS SAB, IGES, STEP, NASTRAN, STL, DXF™, GDSII, Gerber or POV files
- ☐ Parameterization for imported CAD files
- ☐ Material database
- ☐ Structure templates for simplified problem setup

Electrostatic Solver

- ☐ Isotropic and (coordinate-dependent) anisotropic material properties
- ☐ Support of hexahedral meshes as well as linear and curved tetrahedral meshes
- ☐ Sources: potentials, charges on conductors (floating potentials), uniform volume- and surface-charge densities, capacitive field grading
- ☐ Force calculation
- ☐ Capacitance calculation
- ☐ Electric / magnetic / tangential / normal / open / fixed-potential boundary-conditions
- ☐ Perfect conducting sheets and wires
- ☐ Discrete edge capacitive elements at any location in the structure
- ☐ Adaptive mesh refinement in 3D
- ☐ Higher order representation of the solution with tetrahedral mesh
- ☐ Automatic parameter studies using built-in parameter sweep tool
- ☐ Automatic structure optimization for arbitrary goals using built-in optimizer
- ☐ Network distributed computing for optimizations, parameter sweeps and remote calculations
- ☐ Coupled simulations with Mechanical Solver from CST MPhysics Studio®
- ☐ Equivalent Circuit EMS/DS Co-Simulation for constant material properties

Magnetostatic Solver

- ☐ 3D- and 2D[1]- problem support.
- ☐ Isotropic and (coordinate-dependent) anisotropic material properties
- ☐ Nonlinear ferromagnetic material properties
- ☐ Laminated material properties
- ☐ Support of hexahedral meshes as well as linear and curved tetrahedral meshes
- ☐ Sources: coils, coil segments, linear and non-linear permanent magnets, current paths, external magnetic fields, stationary current fields, current ports
- ☐ Force and force density calculation
- ☐ Apparent and incremental inductance calculation
- ☐ Flux linkages
- ☐ Electric / magnetic / tangential / normal / open boundary-conditions
- ☐ Rotational periodicity for 2D and 3D problems
- ☐ Adaptive mesh refinement for 2D and 3D solver
- ☐ Higher order representation of the solution with tetrahedral and triangular meshes
- ☐ Automatic parameter studies using built-in parameter sweep tool
- ☐ Automatic structure optimization for arbitrary goals using built-in optimizer
- ☐ Network distributed computing for optimizations, parameter sweeps and remote calculations
- ☐ Coupled simulations with Mechanical Solver from CST MPhysics Studio
- ☐ Equivalent Circuit EMS/DS Co-Simulation for constant and nonlinear material properties

Stationary Current Solver

- ☐ Isotropic and (coordinate-dependent) anisotropic material properties
- ☐ Nonlinear electrical conductivity properties
- ☐ Temperature dependent materials with coupling to CST MPhysics Studio
- ☐ Electric contact resistance
- ☐ Support of hexahedral meshes as well as linear and curved tetrahedral meshes
- ☐ Sources: current paths, potentials, current ports, coil segments including those created from solids
- ☐ Conductance calculation

[1] The 2D solver supports rotational and translational symmetric problems.

- Discrete edge resistances at any location in the structure
- Perfect conducting sheets and wires
- Electric / magnetic / normal / tangential boundary-conditions
- Adaptive mesh refinement in 3D
- Higher order representation of the solution with tetrahedral mesh
- Automatic parameter studies using built-in parameter sweep tool
- Automatic structure optimization for arbitrary goals using built-in optimizer
- Network distributed computing for optimizations, parameter sweeps and remote calculations
- Coupled simulations with Thermal Solver from CST MPhysics Studio
- Equivalent Circuit EMS/DS Co-Simulation for constant material properties

LF Frequency Domain Solver

- Isotropic and (coordinate-dependent) anisotropic material properties
- Nonlinear material properties (B(H)) and linear equivalent permeability computation
- Temperature dependent nonlinear (B(H)) and linear materials with coupling to CST MPhysics Studio
- Support of hexahedral meshes as well as linear and curved tetrahedral meshes
- Electroquasistatic analysis
- Magnetoquasistatic analysis (eddy current approximation)
- Full wave analysis
- Sources for electroquasistatic analysis: potentials
- Sources for full wave and magnetoquasistatic analysis: coils, coil segments including those created from solids, current ports, current paths, voltage paths, external magnetic source fields
- Impedance calculation
- Broadband impedance calculation for magnetoquasistatic analysis for the setups with stranded coils and segments, current ports and coil segments created from solids
- Broadband source parameters for magnetoquasistatic analysis for the setups with stranded coils and/or current ports
- Authoring of Reduced Order Models as Functional Mockup Units according to FMI standard
- Fast frequency sweep in the broadband solver regime
- Force calculation
- Perfect conducting sheets and wires
- Lumped R, L, C elements at any location in the structure
- Surface impedance model for good conducting metals
- Electric / magnetic / open boundary-conditions
- Adaptive mesh refinement in 3D
- Higher order representation of the solution with tetrahedral mesh
- Automatic parameter studies using built-in parameter sweep tool
- Automatic structure optimization for arbitrary goals using built-in optimizer
- Network distributed computing for optimizations, parameter sweeps and remote calculations
- Coupled simulations with Thermal Solver from CST MPhysics Studio
- Coupled simulations with SIMULIA Abaqus

LF Time Domain Solver

- Isotropic and (coordinate-dependent) anisotropic material properties
- Magnetoquasistatic analysis (eddy current approximation), 3D- and 2D[2]-problem support
- Electroquasistatic analysis
- Nonlinear material properties (B(H), E(J), J(H, T))
- Recoil model for nonlinear hard magnetic material properties (permanent magnets)
- Iron Loss computation
- Support of linear and curved tetrahedral meshes
- Sources for the magnetoquasistatic analysis: coils, coil segments including those created from solids, current ports, current paths, voltage paths, permanent magnets, external magnetic source field

[2] The 2D solver supports rotational and translational symmetric problems.

- ☐ Sources for electroquasistatic analysis: potentials
- ☐ Magnetoquasistatic analysis: perfect conducting sheets and wires
- ☐ Electric / magnetic / open boundary-conditions
- ☐ Higher order representation of the solution with tetrahedral mesh
- ☐ User defined excitation signals and signal database
- ☐ Adaptive time stepping
- ☐ Dedicated time stepping algorithm for time periodic problems
- ☐ Rigid body motion for 2D and 3D models with nested rotation and translation
- ☐ Periodic boundary condition (translation) and subvolume (rotation)
- ☐ Demagnetization monitors
- ☐ Network distributed computing remote calculations
- ☐ Coupled simulations with Thermal Solver from CST MPhysics Studio
- ☐ Coupled simulations with SIMULIA Abaqus

Note: some solvers or features may be available for hexahedral and some may be available for tetrahedral meshes only.

CST Design Studio View

- ☐ Schematic view that shows the circuit level description of the current CST EM Studio project
- ☐ Allows additional wiring, including active and passive circuit elements as well as more complex circuit models coming from measured data (e.g. Touchstone or IBIS files), analytical or semi-analytical descriptions, or from simulated results (e.g. CST Microwave Studio, CST Cable Studio or CST PCB Studio projects)
- ☐ Offers many different circuit simulation methods
- ☐ All schematic elements as well as all defined parameters of the connected CST EM Studio project can be parameterized and are ready for optimization runs
- ☐ Geometry creation by assembling the components on the schematic in 3D
- ☐ Flexible and powerful hierarchical task concept offering nested parameter sweep / optimizer setups

SAM (System and Assembly Modeling)

- ☐ 3D representations for individual components
- ☐ Automatic project creation by assembling the schematic's elements into a full 3D representation
- ☐ Manage project variations derived from one common 3D geometry setup
- ☐ Coupled multi-physics simulations by using different combinations of coupled Circuit/EM/Thermal/Stress projects

Visualization and Secondary Result Calculation

- ☐ Multiple 1D result view support
- ☐ Online visualization of intermediate results during transient simulations
- ☐ Import and visualization of external xy-data
- ☐ Copy & paste of xy-datasets
- ☐ Fast access to parametric data via interactive tuning sliders
- ☐ Automatic parametric 1D result storage

- ☐ Various field visualization options in 2D and 3D for electric fields, magnetic fields, potentials, current densities, energy densities, etc.
- ☐ Animation of field distributions
- ☐ Display of source definitions in 3D
- ☐ Display of nonlinear material curves in xy-plots
- ☐ Display of material distribution for nonlinear materials

- ☐ Display and integration of 2D and 3D fields along arbitrary curves
- ☐ Integration of 3D fields across arbitrary faces

- Hierarchical result templates for automated extraction and visualization of arbitrary results from various simulation runs. These data can also be used for the definition of optimization goals.

Result Export

- Export of result data such as fields, curves, etc.
- Export of result data as ASCII files
- Export screen shots of result field plots

Automation

- Powerful VBA (Visual Basic for Applications) compatible macro language including editor and macro debugger
- OLE automation for seamless integration into the Windows environment (Microsoft Office®, MATLAB®, AutoCAD®, MathCAD®, Windows Scripting Host, etc.)

About This Manual

This manual is primarily designed to enable a quick start of CST EM Studio. It is not intended to be a complete reference guide to all the available features but will give you an overview of key concepts. Understanding these concepts will allow you to learn how to use the software efficiently with the help of the online documentation.

The main part of the manual is the *Simulation Workflow* (Chapter 2), which will guide you through the most important features of CST EM Studio. We strongly encourage you to study this chapter carefully.

Document Conventions

- Buttons that should be pressed within dialog boxes are always written in italics, e.g. *OK*.

- Key combinations are always joined with a plus (+) sign. Ctrl+S means that you should hold down the Ctrl key while pressing the S key.

- The program's features can be accessed through a Ribbon command bar at the top of the main window. The commands are organized in a series of tabs within the Ribbon. In this document, a command is printed as follows*: Tab name: Group name ⇨ Button name ⇨ Command name.* This means that you should activate the proper tab first and then press the button *Command name*, which belongs to the group *Group name*. If a keyboard shortcut exists, it is shown in brackets after the command.
  Example: *View: Visibility ⇨ Wire Frame (Ctrl+W)*

- The project data is accessible through the navigation tree on the left side of the application's main window. An item of the navigation tree is referenced in the following way: *NT: Tree folder ⇨ Sub folder ⇨ Tree item*.
  Example: *NT: 2D/3D Results ⇨ E-Field [Es] ⇨ Abs*

Your Feedback

We are constantly striving to improve the quality of our software documentation. If you have any comments regarding the documentation, please send them to your support center: 3DS.com/Support.

Chapter 2 – Simulation Workflow

The following example shows a fairly simple magnetostatic simulation. Studying this example carefully will allow you to become familiar with many standard operations that are necessary to perform a simulation within CST EM Studio.

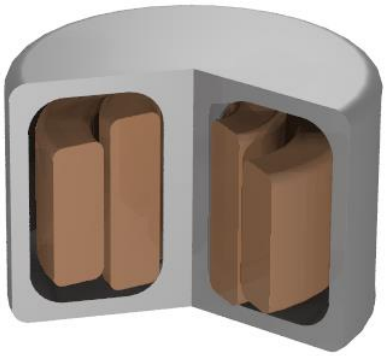
Go through the following explanations carefully even if you are not planning to use the software for magnetostatic computations. Only a small portion of the example is specific to this particular application type since most of the considerations are quite general to all solvers and application domains.

At the end of this example, you will find some remarks concerning the differences between the typical sources and simulation procedures for electrostatic, stationary current, magnetostatic, and low-frequency calculations.

The following explanations always describe the “long” way to open a particular dialog box or to launch a particular command. Whenever available, the corresponding toolbar item will be displayed next to the command description. In order to limit the space in this manual, the shortest way to activate a particular command (i.e. either by pressing a shortcut key or by activating the command from the context menu) is omitted. You should regularly open the context menu to check the available commands for the currently active mode.

The Structure

In the example, you will model a simple sealed transformer consisting of two coils and an iron core in a cylindrical box. Then you will set up the simulation to compute the magnetic field distribution and the inductances. The following picture shows the current structure of interest (it has been sliced open purely to aid visualization). The picture was produced using the POV export option in CST EM Studio.



Before you start modeling the structure, let us spend a few moments discussing how to describe this structure efficiently.

CST EM Studio allows you to define the properties of the background material. Anything you do not fill with a particular material will automatically be considered as the background material. For this structure, it is sufficient to model only the cylinder box, the iron core and the two coils. The background properties will be set to vacuum.

Your method of describing the structure should therefore be as follows:

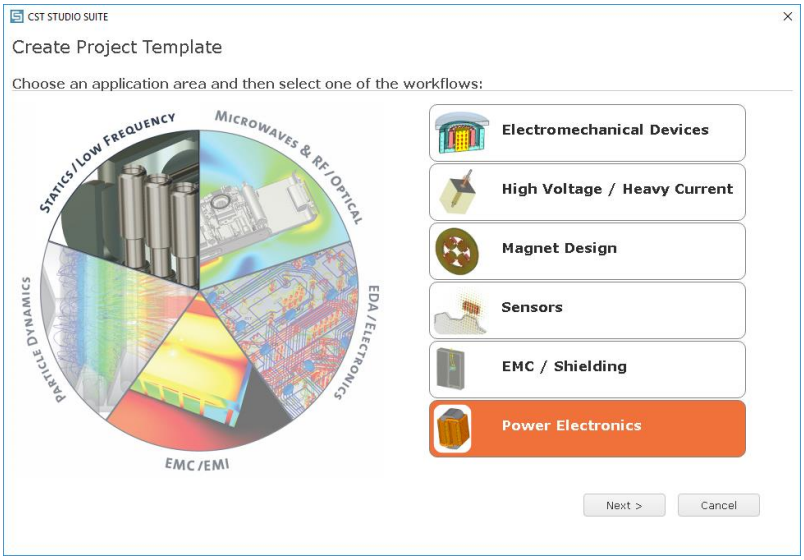
1. Create the cylindrical box.
2. Model the iron core inside the box.
3. Define the coils.

Create a New Project

After launching the CST Studio Suite, you will enter the start screen showing you a list of recently opened projects and allowing you to specify the application which suits your requirements best. The easiest way to get started is to configure a project template, which sets the basic settings

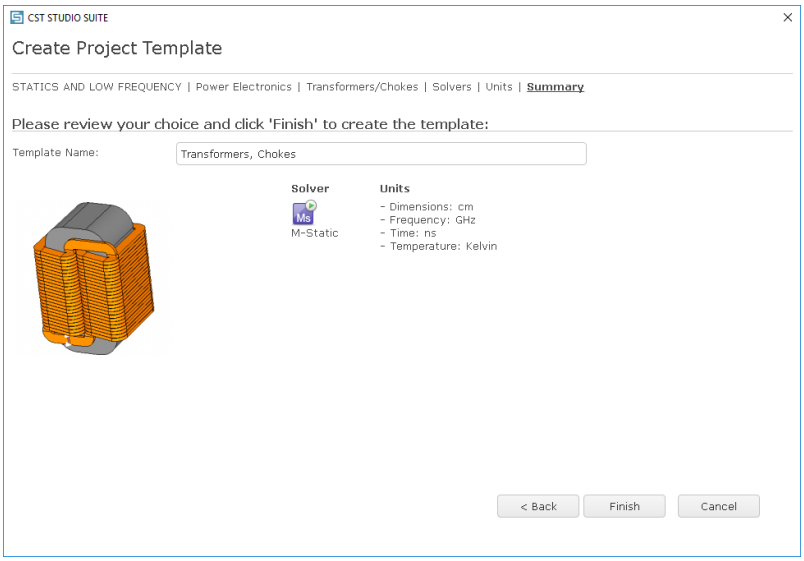
that are meaningful for your typical application. Therefore, click on the *New Template* button in the *New Project from Template* section.

Next you should choose the application area, which is *Statics and Low Frequency* for the example in this tutorial and then select the workflow by double-clicking on the corresponding entry.



For the sealed transformer, please select *Power Electronics* ⇨ *Transformers/Chokes* ⇨ *M-Static* .

At last you are requested to select the units which fit your application best. For the sealed transformer all dimensions will be given in cm. Therefore, select cm from the *Dimensions* drop-down list. For the specific application in this tutorial, the other settings can be left unchanged. After clicking the *Next* button, you can give the project template a name and review a summary of your initial settings.



Finally, click the *Finish* button to save the project template and to create a new project with appropriate settings. CST EM Studio will be launched automatically due to the choice of the application area *Statics and Low Frequency*.

Please note: When you click again on *File: New and Recent* you will see that the recently defined template appears below the New *Project from Template* section. For additional projects in the same application area you can simply click on this template entry to launch CST EM Studio with

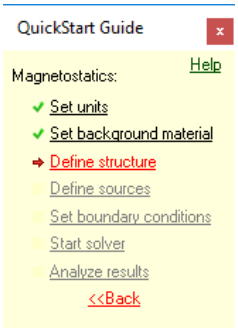
useful basic settings. It is not necessary to define a new template each time. You are now able to start the software with reasonable initial settings quickly with just one click on the corresponding template.

Please note: All settings made for a project template can be modified later on during the construction of your model. For example, the units can be modified in the units dialog box (*Home: Settings ⇨ Units*) and the solver type can be selected in the *Home: Simulation ⇨ Setup Solver* drop-down list.

Open the QuickStart Guide

An interesting feature of the online help system is the QuickStart Guide, an electronic assistant that will guide you through your simulation. If it does not show up automatically, you can open this assistant by selecting *QuickStart Guide* from the Help contents drop-down menu in the upper right corner.

The following dialog box should be positioned in the upper right corner of the main view:



As the project template has already set the solver type, units and background material, the *Magnetostatic Analysis* is preselected, and some entries are marked as done. The red arrow always indicates the next step necessary for your problem definition. You do not have to follow the steps in this order, but we recommend you follow this guide at the beginning to ensure that all necessary steps have been completed.

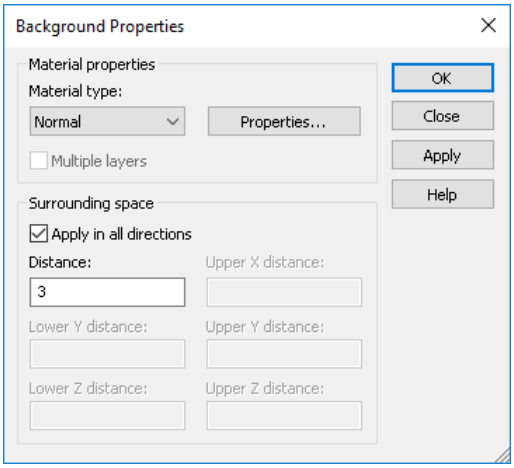
Look at the dialog box as you follow the various steps in this example. You may close the assistant at any time. Even if you re-open the window later, it will always indicate the next required step.

If you are unsure of how to access a certain operation, click on the corresponding line. The QuickStart Guide will then either run an animation showing the location of the related menu entry or open the corresponding help page.

Define the Background Material

As discussed above, the structure will be described within a vacuum world with some surrounding space. The project template has set some typical default values already. Select *Modeling: Materials ⇨ Background* to check or modify the background material settings. For this example enter 3 cm for all directions by checking *Apply in all directions* and enter the *Distance* value.

Confirm by clicking the *OK* button. (Remember: according to the predefined unit, all geometric settings are in cm.)

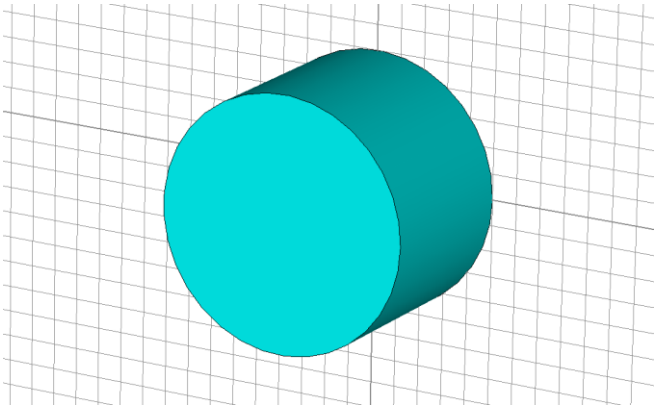


Model the Structure

First create a cylinder along the z-axis of the coordinate system by the following steps:

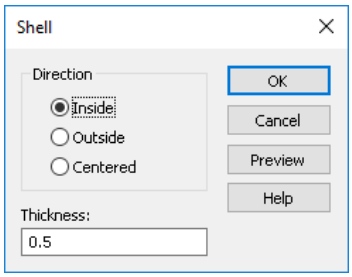
1. Select the cylinder creation tool from *Modeling: Shapes ⇨ Cylinder* .
2. Press the *Shift+Tab* key, and enter the center point (0,0) in the xy-plane before pressing the *Enter* key to store this setting.
3. Press the *Tab* key again, enter the radius 5 and press the *Enter* key.
4. Press the *Tab* key, enter the height as 7 and press the *Enter* key.
5. Press *Esc* to create a solid cylinder (skip the definition of the inner radius).
6. In the shape dialog box, enter “cylinder box” in the *Name* field.
7. Select *component1* from the Component drop-down list.
8. Select *[New Material]* from the Material drop-down list. The Material dialog box opens where you should enter the material name “Iron”, select *Normal* properties (*Type*) and set the material properties *Epsilon* = 1.0 and *Mu* = 1000. Now you can select a color and close the dialog box by clicking *OK*.
9. Back in the cylinder creation dialog box, click *OK* to create the cylinder.
10. Finally, save the structure by selecting *File: Save (Ctrl+S)* and entering a name, e.g. "Transformer.cst" in a folder of your choice.

The result of all these operations should look like the picture below. You can press the *Space* bar to zoom to a full screen view.



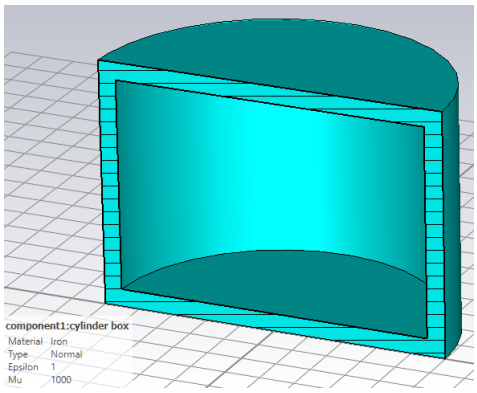
Please note that you can switch on or off the multicolored axes or the axes at the origin in the *View Options* dialog box (*View: Options ⇨ View Options (Alt+V)*).

The next step is to shell the cylinder. Select the cylinder in the navigation tree (*NT: Components ⇨ component1 ⇨ cylinder box*) and open the shell dialog by selecting *Modeling: Tools ⇨ Shape Tools ⇨ Shell Solid or Thicken Sheet*. Enter the *Thickness* 0.5 and select *Inside* as the direction.



To observe the result, activate the cutting plane view via *View: Sectional View ⇨ Cutting Plane ⇨ Cutting Plane (Shift+C)* . You can adjust the cutting plane settings either by using the *Up/Down* arrow keys or by entering the *Cutting Plane Properties* dialog box (*View: Sectional View ⇨ Cutting Plane ⇨ Properties*).

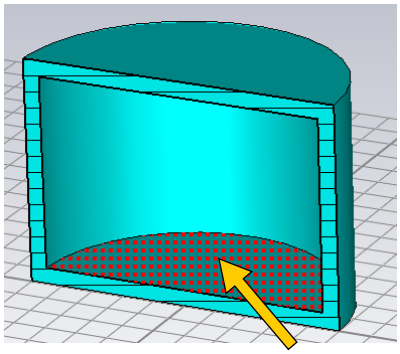
To look into the box, you might have to rotate the view. Activate the rotation mode by selecting *View: Mouse Control ⇨ Rotate ⇨ Smart (Mouse Pointer)* . Then press the left mouse button and move the mouse until the view looks like this:



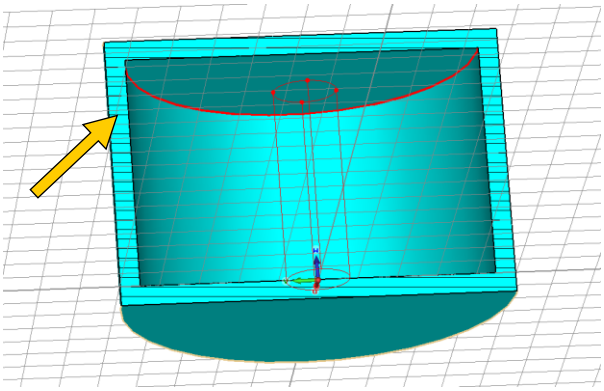
It is also possible to hold down the *Ctrl* button to activate the rotation mode for as long as *Ctrl* is pressed.

The next step is to create a second cylinder inside the box. The center of the new cylinder's base should align with the center of the box's inside face. To this end, first align the local coordinate system (WCS) with the lower inside z face of the box:

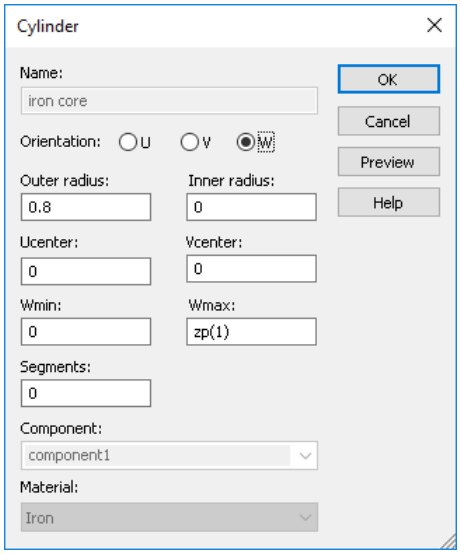
1. Select *Modeling: Picks* ⇨ *Picks* .
2. Double-click on the box's lower inside z-plane. Note: Pickable faces or edges are automatically highlighted, when the mouse is in an appropriate position. They sometimes hide other objects. With the *Tab* key, it is possible to switch through the relevant objects until the desired face is marked for picking. The selected face should now be highlighted:



3. Now choose *Modeling: WCS* ⇨ *Align WCS* (Shortcut: *W*).
4. Select the cylinder creation tool *Modeling: Shapes* ⇨ *Cylinder* .
5. Press the *Shift+Tab* key and enter the center point (0,0) in the uv-plane and press the *Enter* key.
6. Press the *Tab* key again and enter a radius of 0.8 and press the *Enter* key.
7. Select *Modeling: Picks* ⇨ *Pick Points* ⇨ *Pick Circle Center* .
8. Set the cylinder's height by picking the highlighted circle of the upper inner face of the box with a double-click. You might have to rotate the structure a little bit to get a better view:

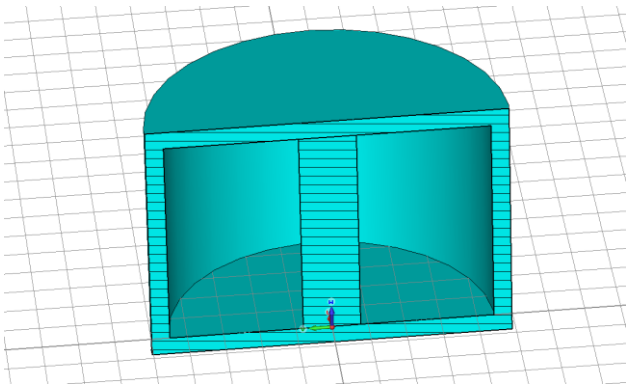


9. Press *Esc* to create a solid cylinder (skip the definition of the inner radius).
10. In the shape dialog box, enter "iron core" in the *Name* field.
11. Select the component "component1" from the component list.
12. Select the material "Iron" from the material list.



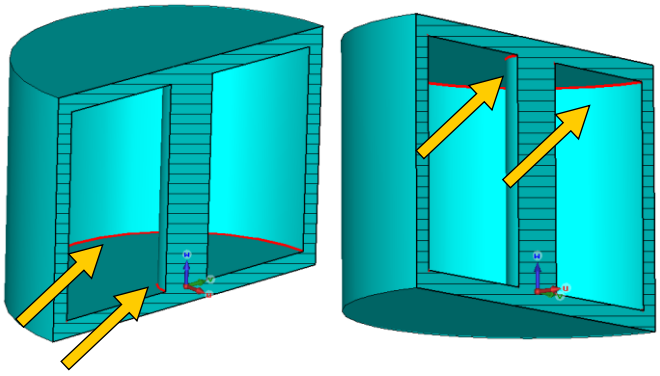
13. Click the *OK* button.

The result of these operations should look like this:

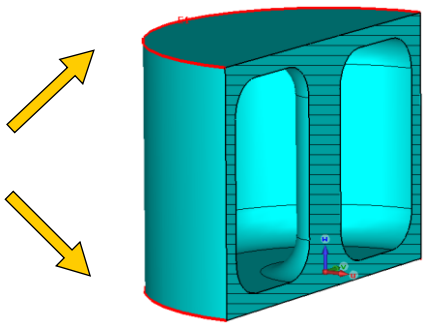


Sharp edges are, in general, responsible for field singularities. Therefore, we will blend the edges of the iron core and the cylinder box. Before we can do this, the two bodies have to be united. Thus, select the cylinder box (either in the navigation tree or by double-clicking on it in the main view). Then choose *Modeling: Tools* ⇨ *Boolean* ⇨ *Add* and select the iron core. Confirm the operation by pressing the *Enter* key. The iron core entry will vanish from the navigation tree and only the cylinder box remains in the *NT: Components* ⇨ *component1* folder.

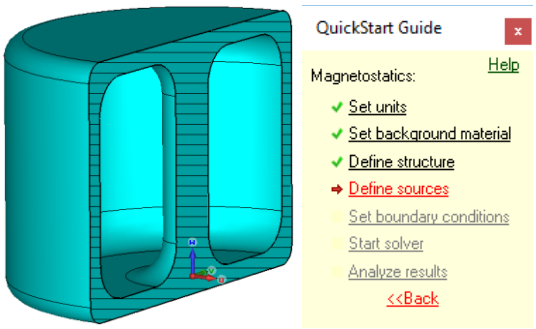
Now you can select the edges to blend. All inner edges shall be blended with radius 1, the outer edges of the cylinder box with radius 0.5. Hence, activate the pick edge tool Modeling: *Picks* ⇨ *Picks* ⇨ *Pick Edge* (Shortcut: *e*) and pick first all inner edges (multiple activations of the pick edge tool might be necessary, you can see the selected edges in the lower left corner):



Finally, enter the *Blend Edges* dialog box via *Modeling: Tools* ⇨ *Blend* ⇨ *Blend Edges* and enter the radius 1.0. Confirm this setting by pressing *OK*. Next pick the two outer edges of the cylinder box.



Open the *Blend Edges* dialog again and enter the radius 0.5. Leave the dialog via the *OK* button. The cylinder box should look now as depicted below:



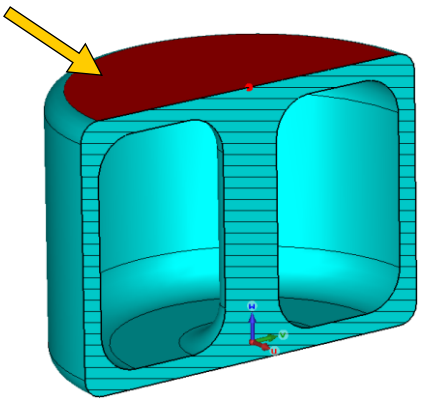
Looking at the *QuickStart Guide*, you will see that now it is time to define the sources for the magnetic field simulation.

Define Coils

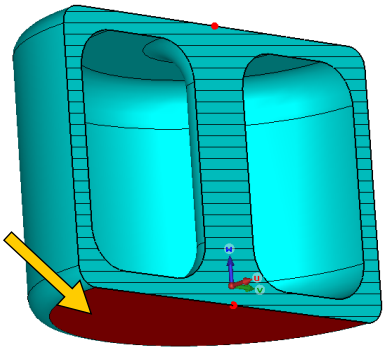
In CST EM Studio, a coil can be defined as an *a-priori* known current- or voltage-distribution which is constant over the cross-section of the coil body for this example. Consequently, the coil represents the equivalent distribution of the current in a realistic coil with many turns, where small-scale variations are averaged out.

The creation of a coil is quite similar to the definition of a solid by curves. First of all, you have to move the working coordinate system to the right position:

1. Select *Modeling: Picks* ⇨ *Pick Points* ⇨ *Pick Face Center* (shortcut: *A*).

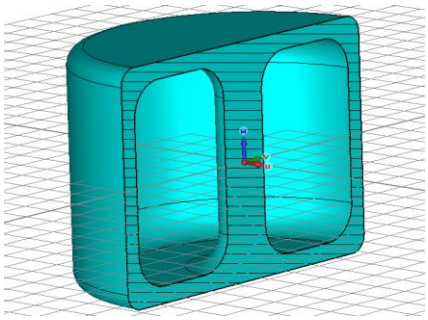


2. Double-click on the upper outside face of the box as highlighted.
3. Select *Modeling: Picks* ⇨ *Pick Points* ⇨ *Pick Face Center* again.



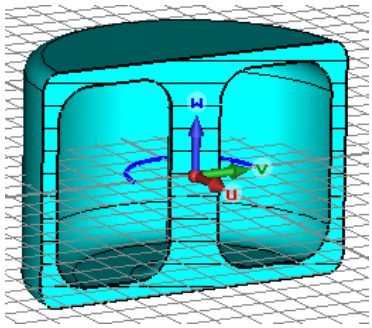
4. Double-click on the lower outside face of the box as highlighted.
5. Select *Modeling: Picks ⇨ Pick Points ⇨ Mean Last Two Points* (*Ctrl+Shift+M*).
6. Select *Modeling: WCS ⇨ Align WCS* .

Now, the working coordinate system should be placed as depicted in the next figure. At any time the *Working Plane* can be enabled or disabled using *View: Visibility ⇨ Working Plane* (*Alt+W*).



To define the path of the first coil, carry out the following:

1. Select *Modeling: Curves ⇨ Curves ⇨ Circle* .
2. Press the *Shift+Tab* key and enter the center point (0,0) in the uv-plane. Then press the *Enter* key to store this setting.
3. Press the *Tab* key again, and enter the radius 2.
4. In the circle dialog box, enter “coil path 1” in the *Name* field.
5. Click *OK* to create the circle.



The path for the second coil is created in the same way:

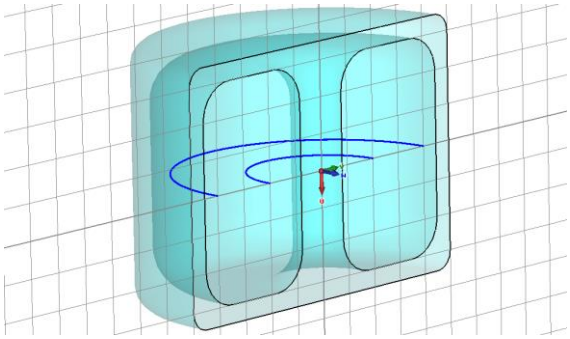
1. Select *Modeling: Curves ⇨ Curves ⇨ Circle* .
2. Press the *Shift+Tab* key, and enter the center point (0,0) in the uv-plane before pressing the *Enter* key to store this setting.
3. Press the *Tab* key again, and enter the radius 4.

4. In the circle dialog box, enter “coil path 2” in the *Name* field.
5. Select *[New Curve]* from the *Curve* drop-down list.
6. Click *OK* to create the circle.

Please note: We put all path and profile curves into separate *Curve* folders just to simplify blending the coils’ edges afterwards.

To define the profile paths of both coils, you first need to rotate the working coordinate system around the v-axis:

1. Press *Shift+V* or select *Modeling: WCS ⇨ Transform WCS* and activate the *Rotate* control in the *Transform Local Coordinate System* dialog box and enter 90 for the V component.



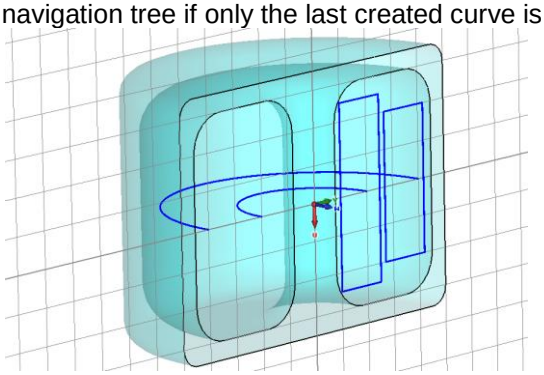
For the definition of the first profile curve, perform the following steps:

1. Select *Modeling: Curves ⇨ Curves ⇨ Rectangle* .
2. Press the *Tab* key, and enter the first point (-2.5, 1) in the uv-plane before pressing the *Enter* key to store this setting.
3. Press the *Tab* key again, and enter the second point (2.5, 2.5) and press the *Enter* key.
4. In the rectangle dialog box, enter “profile path 1” in the *Name* field.
5. Select *[New Curve]* from the *Curve* drop-down list.
6. Click *OK* to create the rectangle.

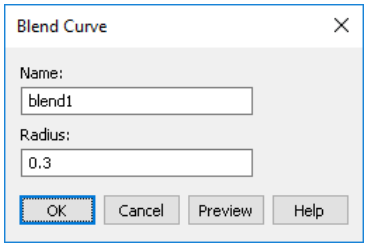
The second profile can be created as follows:

1. Select *Modeling: Curves ⇨ Curves ⇨ Rectangle* .
2. Press the *Tab* key, and enter the first point (-2, 2.7) in the uv-plane before pressing the *Enter* key to store this setting.
3. Press the *Tab* key again, and enter the second point (2, 4.2) and press the *Enter* key.
4. In the rectangle dialog box, enter “profile path 2” in the *Name* field.
5. Select *[New Curve]* from the *Curve* drop-down list.
6. Click *OK* to create the rectangle.

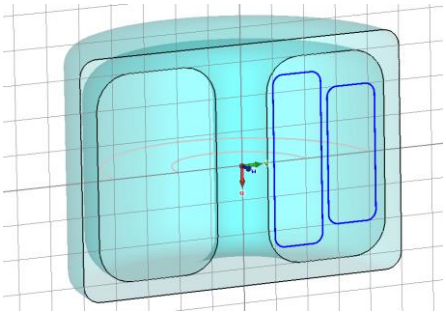
Now your model should look like the one depicted below. You may need to click on the components folder in the navigation tree if only the last created curve is still highlighted.



Like for the cylinder box, it is meaningful to blend the coil edges as well. This can be done by blending the corners of the profile paths. Select *NT: Curves ⇨ curve3 ⇨ profile path 1* and pick its four corners (*Modeling: Picks ⇨ Picks* . , shortcut p). Now choose *Modeling: Curves ⇨ Curves ⇨ Blend Curve* . The *Blend Curve* dialog box will pop up. Enter the radius 0.3 and confirm this setting by pressing OK.

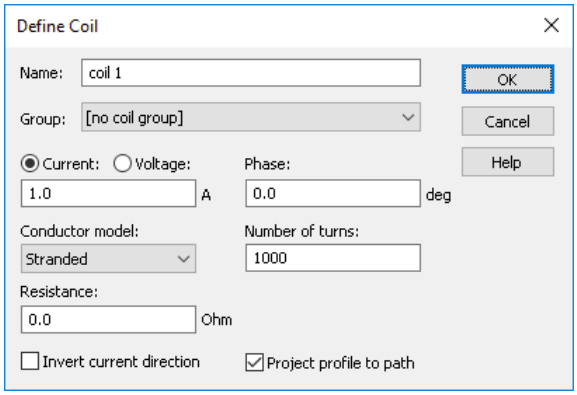


Next, the corners of the *profile path 2* rectangle need to be blended in completely the same manner. Select *NT: Curves ⇨ curve4 ⇨ profile path 2*, choose *Modeling: Picks ⇨ Picks* .repeatedly and pick the four corners of the selected rectangle. Next use *Modeling: Curves ⇨ Curve Tools ⇨ Blend Curve* to blend the corners with the radius 0.3. The profile curves should then look as depicted below:



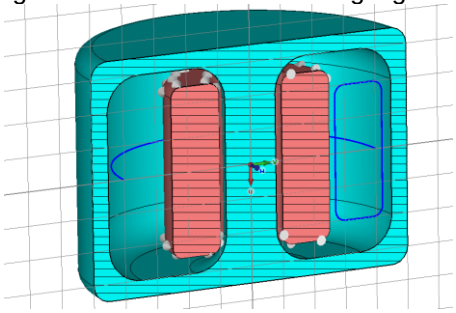
Finally, the coils can be created from the profile and path curves:

1. Select *Simulation: Sources and Loads ⇨ Coil ⇨ Coil* .
2. Move the mouse cursor to “profile path 1” until it the entire curve is highlighted (or select “curve3” in the navigation tree). Then double-click on it in the main view to select it (the inner profile curve).
3. Move the mouse cursor to “coil path 1" and select it by double-clicking.
4. In the *Define Coil* dialog box, enter “coil 1” in the *Name* field, select the type *Current*, enter 1 A for the current value and 1000 in the *Number of turns* field. (Do not change the *Conductor Type*, *Phase* or *Resistance* values.) Coils can be gathered into the so-called coil groups. For more information about this, please refer to the online help.



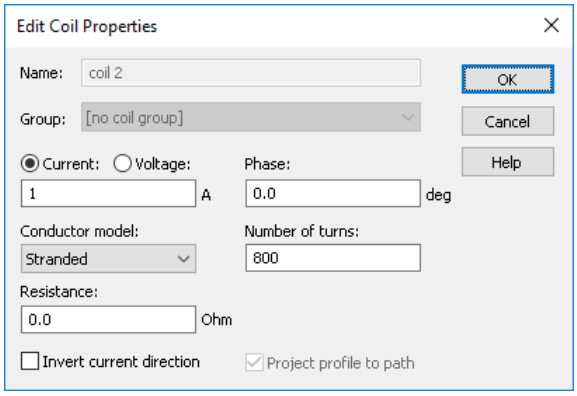
5. Click *OK* to create the coil.

Now your model should look like the one depicted below. You may need to click on the components folder in the navigation tree if the coil is not highlighted.



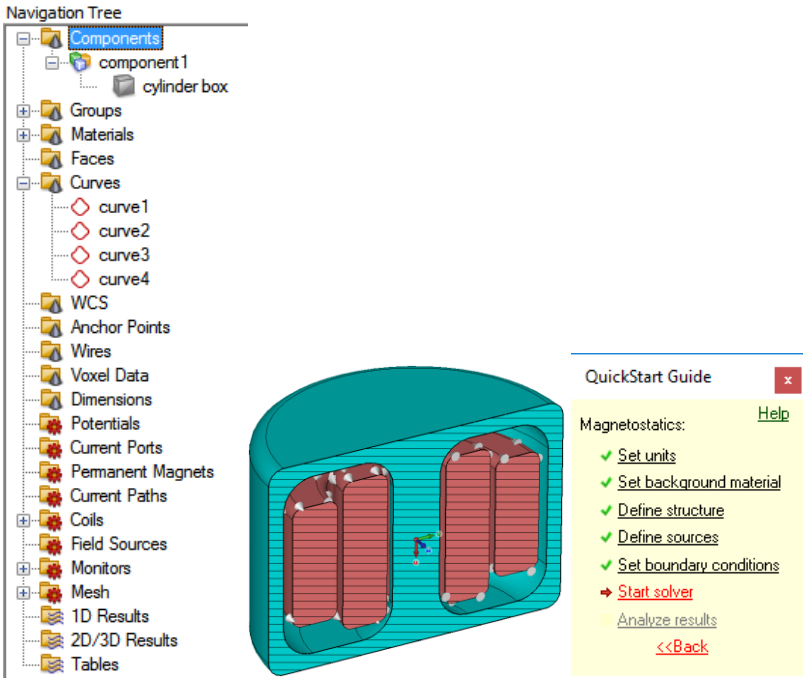
The same procedure can be applied for the second coil:

1. Select *Simulation: Sources and Loads* ⇨ *Coil* ⇨ *Coil* from the main menu.
2. Move the mouse cursor to “profile path 2” until it is highlighted. Then double-click to select it.
3. Move the mouse cursor to “coil path 2,” and select it by double-clicking.
4. In the *Define Coil* dialog box, enter “coil 2” in the *Name* field, 1 A for the value of the current and 800 in the *Number of turns* field.



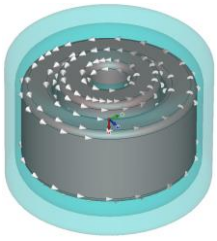
5. Click *OK* to create the coil.

Congratulations! You have just created your first structure within CST EM Studio. The view should now look like this after the working plane (*View: Visibility* ⇨ *Working Plane* (*Alt+W*)) has been switched off:

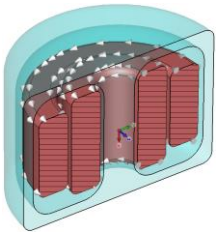


Please note: As the project template has set some default boundary conditions applicable in most use cases, the corresponding entry in the *QuickStart Guide* is already checked. Nevertheless, you should always check if the model can be simplified, e.g. by symmetry conditions. We will discuss this in the next section.

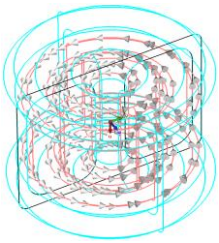
The following gallery shows some views of the structure using different visualization options:



Shaded view, (deactivated working plane iron material properties: 50% transparency)



Shaded view, (cutting plane active)



Wireframe view, (*View: Visibility* ⇨ *Wireframe*)

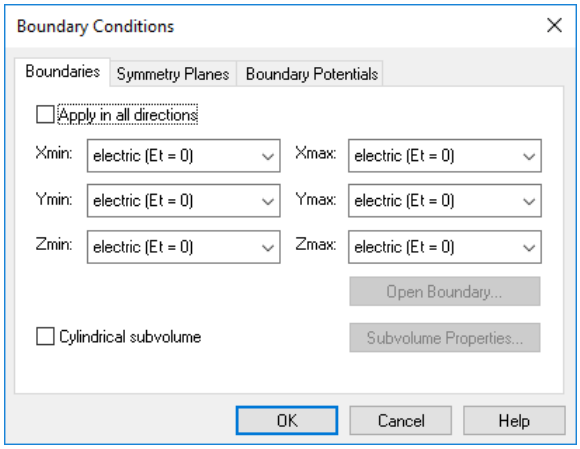
Define Boundary Conditions

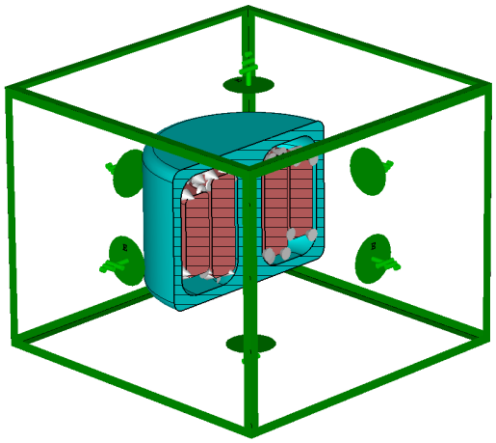
The simulation of this structure is performed only within the bounding box enclosing the structure together with some background material. The space occupied by the structure and background material is called the "computational domain" in the sequel.

Note that the restriction to a bounded computational domain is artificial for our example (keeping in mind the transformer structure in open space). However, in this simple case, the magnetic flux is concentrated in the core material. Therefore, the artificial boundary will not considerably disturb the solution though the added space around the structure is not very large.

In order to get a well-defined problem, you must specify the behavior of the field at the boundary of the computational domain by setting a boundary condition for each plane (Xmin/Xmax/Ymin/Ymax/Zmin/Zmax).

The boundary conditions are specified in a dialog box which you can bring up by choosing *Simulation: Settings ⇨ Boundaries* .





While the boundary dialog box is open, the boundary conditions will be visualized in the structure view as in the picture above. You can change boundary conditions within the dialog box or interactively in the view by double-clicking on the corresponding boundary symbol, and then select the appropriate type from the context menu.

The project template has already set "electric (Et = 0)" boundary conditions for every face. You do not need to change the default setting.

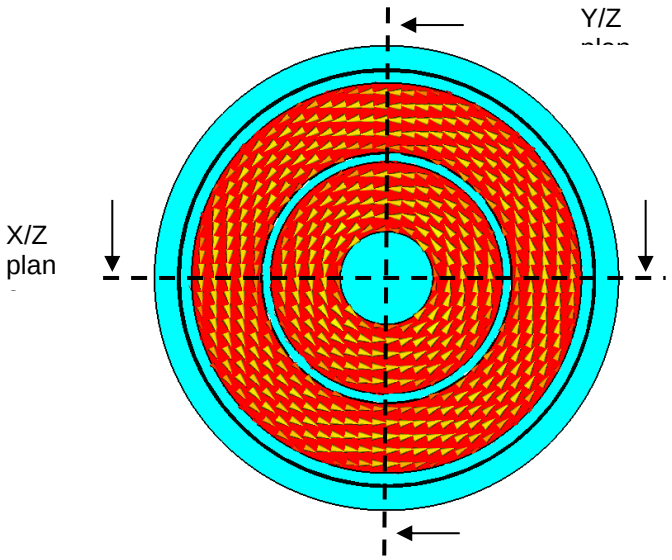
***Background information**: Electric boundary conditions ("electric (Et = 0)") force the tangential electric field to be zero. For non-zero frequencies, Faraday's Law implies a zero normal component of the magnetic flux density B. Viewing magnetostatics as a static limit of Maxwell's equations justifies this implication even for the magnetostatic case. Consequently, an electric boundary condition always forces a zero normal component of the magnetic flux density, i.e. the B-field is purely tangential, and no flux can leave the computational domain at this face. Note that this also applies to the boundary of perfect electric conductors (PECs), which play the role of interior boundary conditions.*

Another important boundary condition is the "magnetic (Ht = 0)"-condition, which forces a zero tangential magnetic field, i.e. the magnetic field is purely normal to a face defined as "magnetic." This consideration is used in the next sub-section.

Define Symmetry Conditions

In addition to the boundary planes, you can specify “symmetry planes". Each specified symmetry plane reduces the simulation time and the required memory by a factor of two. In our example, the structure is symmetric with respect to the Y/Z plane (perpendicular to the x-axis). A second symmetry plane applies to the X/Z plane.

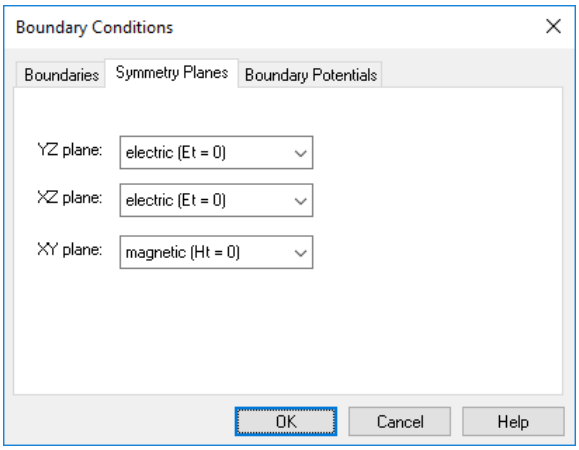
The excitation of the fields is performed by the currents in the coils for which the current pattern is shown below:

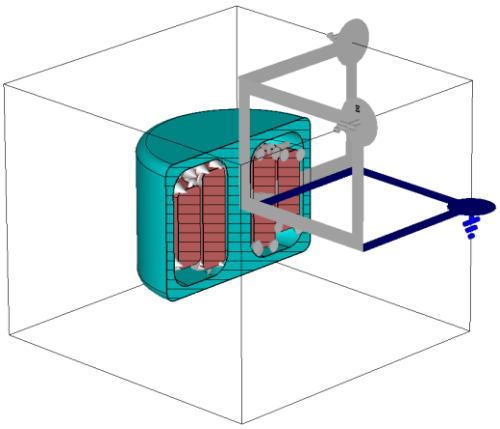


The electric symmetry planes for the magnetic field can be applied if the current pattern is normal to the plane.

The resulting magnetic field has no component normal to the X/Z and Y/Z planes (the entire field is oriented tangential to these planes). Moreover, the fields have no component tangential to the X/Y plane. If you specify X/Z and Y/Z planes as “electric” and X/Y as “magnetic” symmetry planes, you can advise CST EM Studio to limit the simulation to 1/8 of the actual structure while taking these symmetry conditions into account.

To specify the symmetry condition, click on the *Symmetry Planes* tab in the *Boundary Conditions* dialog box. For the YZ- and XZ-plane symmetry, you can choose "electric (Et = 0)" by either selecting the appropriate choice in the dialog box, or by double-clicking on the corresponding symmetry plane visualization in the view and selecting the appropriate choice from the context menu. For XY-plane symmetry, choose "magnetic (Ht = 0)." Once you have done this, your model and the dialog box will appear as follows:





Finally, click *OK* in the dialog box to store the settings. The boundary visualization will then disappear.

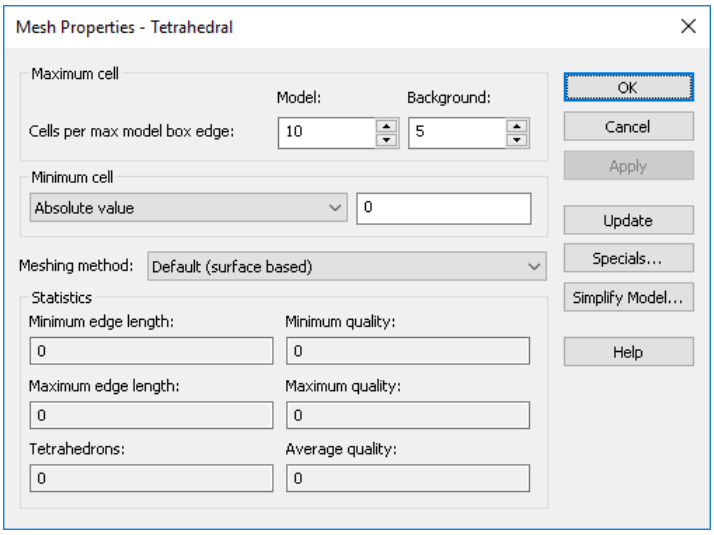
As shown by the *QuickStart Guide*, the model is now completely defined, and you are ready to start the magnetostatic solver.

In order to get a discrete version of the defined model that can be solved numerically, a *mesh* must be provided for the computational domain. CST EM Studio features two independent solvers based on tetrahedral and hexahedral meshes, respectively. Let us start with the tetrahedral solver.

Generate and Visualize a Tetrahedral Mesh

The tetrahedral mesh generation for the structure is performed fully automatically when the tetrahedral magnetostatic solver is started.

It is also possible to generate the mesh separately before starting the solver. This may be helpful in order to get an impression of the mesh quality and mesh resolution. Furthermore, it is possible to fine-tune the mesh before running the computation using *a-priori* knowledge about the solution. Let us use this second possibility and generate the mesh separately.

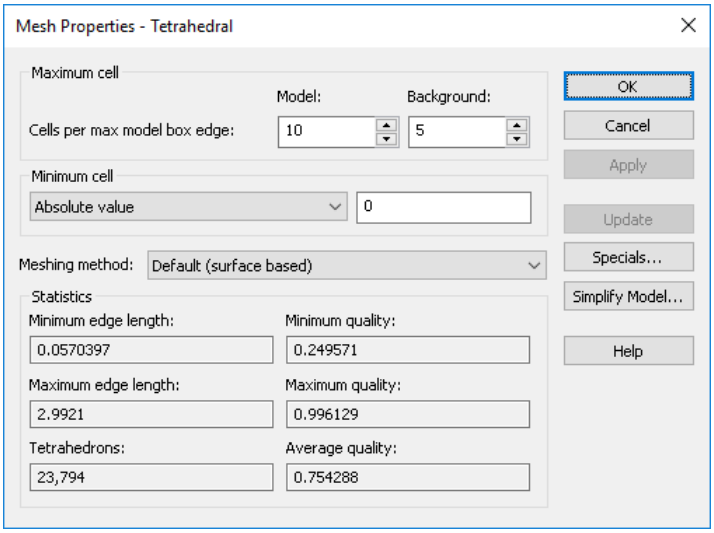


First, open the *Mesh Properties* dialog by selecting *Simulation: Mesh* ⇨ *Global Properties* ⇨ *Tetrahedral* . The dialog box "*Mesh Properties – Tetrahedral*" will open. In order to get a reasonable overall mesh resolution of the problem, you can increase the values for *Maximum cell*. In general this is sufficient, however, to refine the mesh locally, i.e. only at certain critical parts of the geometry. This can be achieved by running the solver with the fully automatic energy-based adaptive refinement. Thus, we start with a rather coarse mesh and leave the *Cells per max model box edge* at the value 10 for the model and at 5 for the background. Additionally, press the *Specials* button and switch off the option *Anisotropic Curvature Refinement* in the *Special Mesh Properties – Tetrahedral* dialog box.

Background information: *The results are strongly influenced by the mesh resolution. The automatic mesh generator analyzes the geometry and tries to refine the mesh locally taking geometric features into account (e.g. curvature-based refinement with tetrahedral meshes or expert system-based approach with hexahedral meshes). However, due to the complexity of electromagnetic problems, this approach may not be able to determine all critical domains in the structure. To circumvent this problem, CST EM Studio features an adaptive mesh refinement that uses the results of a previous solver run in order to optimize the mesh. The adaptive mesh refinement can be activated by checking the corresponding option in the Solver Parameters dialog box.*

Now click the *Update* button in the *Mesh Properties* dialog box to start the mesh generation. You will see a progress bar displaying the current status of the mesh generation.

When the mesh generation process has finished, the progress bar disappears. You will see that the entries in the *Mesh summary* frame of the *Mesh Properties* dialog box have been updated.



In the *Statistics* frame, you can get information about the minimum and maximum mesh-edge lengths, the number of tetrahedrons, and the maximum / minimum, and average mesh quality. The number of tetrahedrons and the edge lengths give you information about the size and resolution of the discretized model.

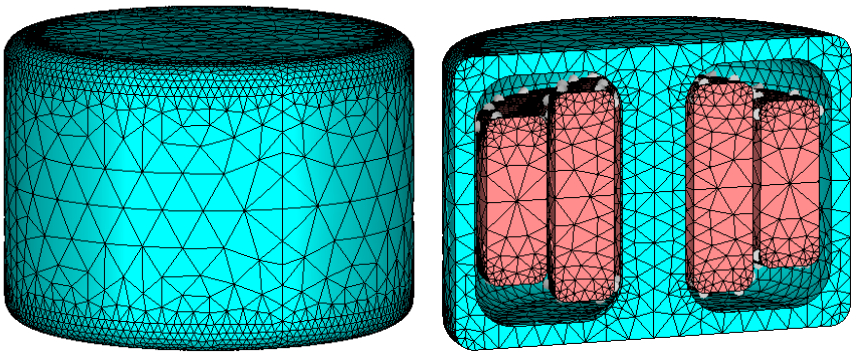
Please note that the mesh size and the results might differ slightly depending on the operating system and the architecture of the machine with which they are calculated.

***Background information**: Generally, due to the finiteness of the mesh density, the computed results differ from the exact solution. The introduced error is called the discretization error. Increasing the mesh density will usually lead to more precise results, yet the computation time and the necessary memory size will increase.*

The quality of a tetrahedron is positive and less than or equal to one. The value “1” indicates the highest (equilateral tetrahedron), the value “0” the lowest quality (zero volume tetrahedron). Please refer to the online help for an exact definition of quality.

***Background information**: Not only the mesh density but also the mesh quality has a strong influence on the results. A very low mesh quality may lead to a poor approximation of the model. Moreover, a low mesh quality may reduce the speed of an iterative solver. This is the reason why it is always meaningful to have a look at the mesh when running a simulation.*

Now close the *Mesh Properties* dialog box by clicking the *OK* button. You can visualize the mesh by entering the mesh view (*Simulation: Mesh ⇨ Mesh View*). The mesh should look similar to the illustration below. To inspect the mesh in the interior of the structure, activate the cutting plane by selecting *View: Sectional View ⇨ Cutting Plane ⇨ Cutting Plane (Shift+C)*.

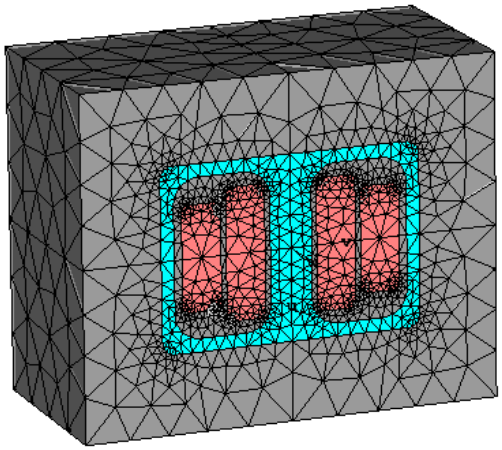


The automatic curvature refinement leads to a local refinement along the blended edges. By default, the mesh transition from the coarser to the finer mesh regions is very rapid. This transition

can be smoothed in the *Specials* dialog-box of the Global Mesh Properties dialog box (*Mesh: Mesh Control ⇨ Global Properties ⇨ Tetrahedral*), which may also improve the mesh quality. Please refer to the online help for more details.

Remember that you have reduced the computational model by defining symmetry planes. Therefore, only 1/8 of the computational domain is meshed. Nevertheless, the mesh is visualized for the complete structure by mirroring the missing parts. You can easily see the symmetry planes in the mesh-view.

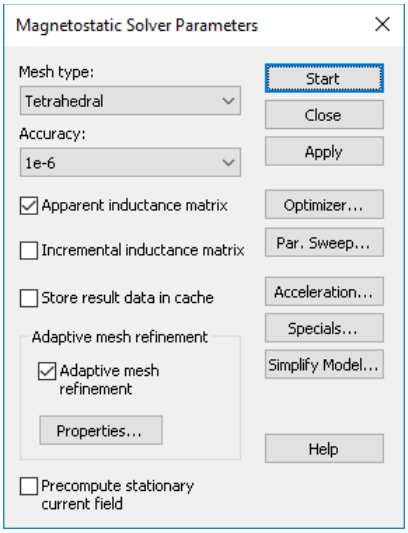
Finally, let us take a look at the mesh of the surrounding space. Activate the visualization of the background material by selecting *View: Options ⇨ View Options (Alt+V)* , and then select the *Background material* checkbox in the *Draw* frame of the *General* Tab. After you click *OK,* the displayed mesh should look similar to the following picture:



Before you go on, you should deactivate the visualization of the background material by selecting *View: Options ⇨ View Options (Alt+V)* again and un-checking *Background material*. Leave the mesh view by selecting *Mesh: Close ⇨ Close Mesh View* .

Run the Tetrahedral Magnetostatic Solver

The simulation is started from the *Magnetostatic Solver Parameters* dialog box which can be opened via *Home: Simulation* ⇨ *Setup Solver* :

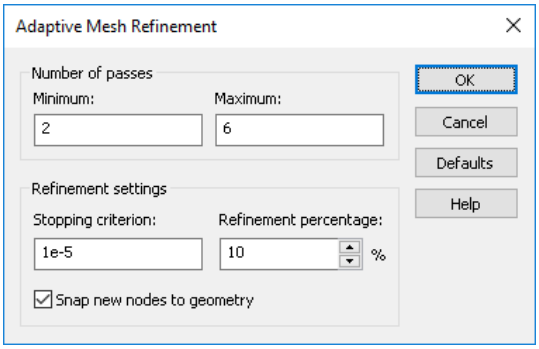


Make sure the *Mesh Type* "Tetrahedral" is selected. In the *Accuracy* drop-down list, a stopping criterion for the iterative linear equation system solver can be selected. For the example model, leave the *Accuracy* value at 1e-6.

Background Information: *While the solution accuracy mainly depends on the discretization of the structure and can be improved by refining the mesh, the numerical error of the linear equation system solver introduces a second error source in field simulations (iteration error). Choosing a small Accuracy value reduces this error at the expense of a longer calculation time. Usually, an Accuracy setting of "1e-6" is sufficient, but in some cases it might be necessary to select a smaller value, particularly if you receive a warning that the results are not accurate. Furthermore, with increasing mesh density (i.e. smaller discretization error) you should also increase the solver accuracy by selecting a smaller Accuracy value*.

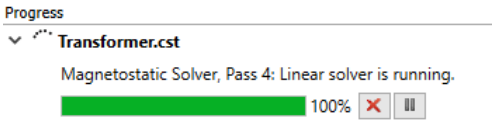
Furthermore, activate the calculation of the *Apparent inductance matrix*. Please note that the *Adaptive mesh refinement* is switched on already. This setting is meaningful as the initial mesh is rather coarse. During the solver run, several mesh refinement passes are performed automatically until the energy value does not change significantly between two subsequent passes. The default termination criterion is an energy deviation of 1% (or less). You can fine-tune these settings in the *Adaptive Mesh Refinement* dialog box.

Click the *Properties…* button to enter the *Adaptive Mesh Refinement* dialog box. Change the *Stopping criterion* to 1e-5 and verify that the checkbox *Snap new nodes to geometry* is checked. This feature will ensure that new nodes that are generated on the surface mesh during the mesh adaption will be projected to the original geometry, so that the approximation of curved surfaces is improved after each adaption step. The dialog box should now look as follows:



Close the dialog with the *OK* button and finally start the simulation procedure by clicking *Start*.

Several progress bars like the one depicted below will appear in the status bar informing you about the current solver status:



These are the steps of the tetrahedral magnetostatic solver run:

1. **Computing coil(s)**: This first calculation step must be performed to calculate the discrete representation of coil current patterns.

2. **Initializing magnetostatic solver:** During this step, your input model is checked for errors such as invalid overlapping materials, not well-defined sources, etc.

3. **Assembling system:** The linear system of equations is generated.

4. **Constructing pre-conditioner:** This includes construction steps for the pre-conditioner of the solver, e.g. an LU-decomposition, a construction of hierarchy for a multigrid solver etc.

5. **Solving linear system:** During this stage, the equation system is solved yielding the unknown field.

6. **Estimating error** (only during mesh adaption pass): The local error for each element is estimated (error distribution).

7. **Marking elements for refinement** (only during mesh adaption pass): Based on the computed error, a certain number of elements will be marked for refinement.

8. **Adapting mesh** (only during mesh adaption pass): The mesh is refined taking the marked elements into account.

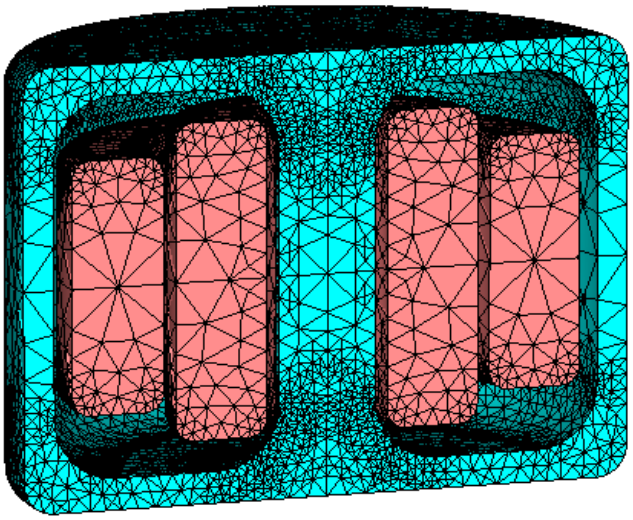
9. **Inductance computation** (only if switched on): The apparent and/or incremental inductance matrix is calculated.

10. **Postprocessing stage:** From the field solution other fields and additional results like the energy within the structure are computed.

If the adaptive mesh refinement is switched on, some of the steps are repeated until a predefined stopping criterion is met.

For this simple structure, the entire analysis (including adaptive mesh refinement) usually takes only a few minutes to complete on a today's standard computer.

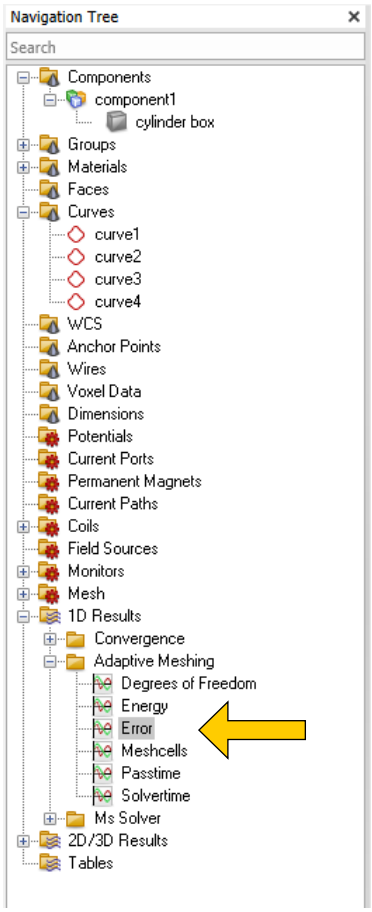
If you activate the mesh view (*Home: Mesh ⇨ Mesh View*) while the adaptive solver is running, you can observe how and where the mesh is refined after each pass. After the solver has finished, the mesh should look like depicted in the following picture (deviations are possible since the initial mesh can differ slightly depending on the operating system and the architecture of the machine):



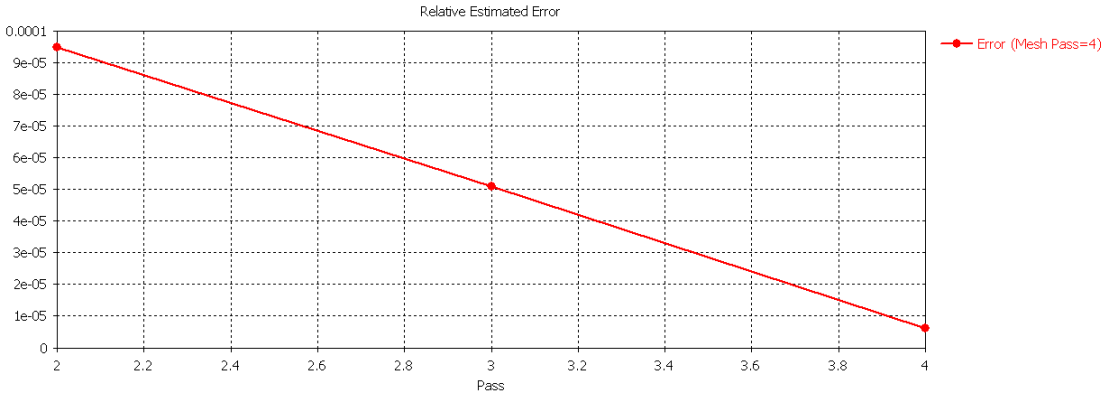
Analyze the Results of the Tetrahedral Solver

After the solver run you can access the results via the navigation tree, see below.

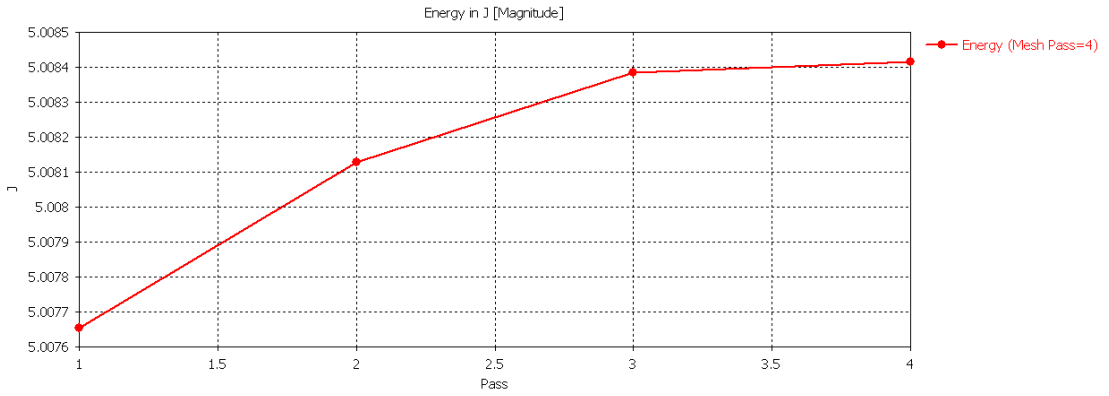
While the adaptive solver is running, you can already watch the progress of the mesh refinement and the convergence behavior in the *NT: 1D Results ⇨ Adaptive Meshing* folder:



Click, for instance, on *NT: 1D Results ⇨ Adaptive Meshing ⇨ Error*. This folder contains a curve which displays the change of the relative energy of two subsequent simulations. The curve below shows that the maximum difference of the relative change of the energy is below the desired stopping criterion of 1e-5.



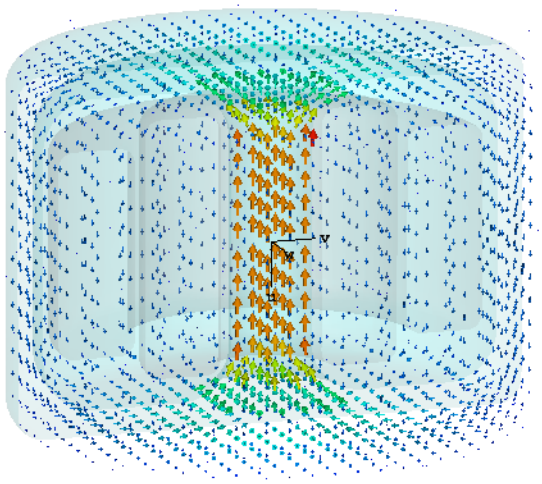
Additionally, the convergence of the energy can be visualized by selecting *NT: 1D Results ⇨ Adaptive Meshing ⇨ Energy*.



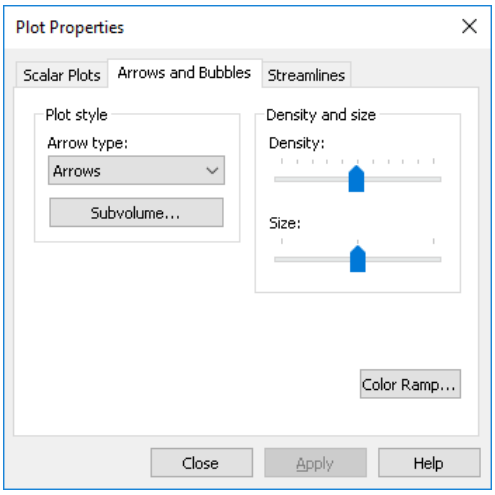
Please remember that the curves can differ slightly when computed on a 32 bit or 64 bit machine. Furthermore, the number of passes needed for convergence can deviate owing to the machine architecture.

In practice it often proves judicious to activate the adaptive mesh refinement to ensure convergence of the results. (This might not be necessary for structures with which you are already familiar and where you can use your experience to refine the automatic mesh manually.)

You can visualize the magnetic flux density by choosing *NT: 2D/3D Results ⇨ B-Field [Ms]* to get an impression of the B-field inside the transformer. After you select this folder, a plot similar to the following should appear:

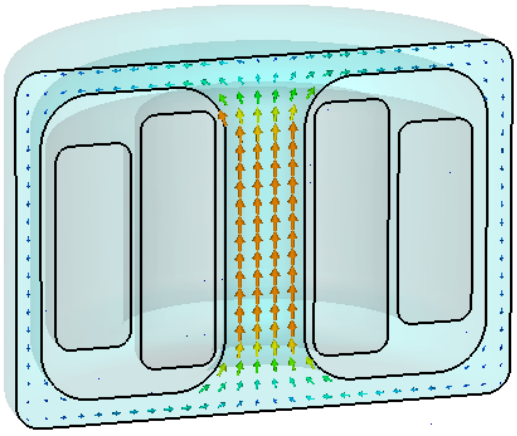


It might be necessary to adjust the size (scaling) and the density of the arrow objects to obtain a better view. You can modify the plot properties by selecting *2D/3D Plot: Plot Properties* ⇨ *Properties* (or by selecting *Plot Properties* from the context menu in the main view). The following dialog box will open:



To decrease the number of arrows, move the *Density* slider slightly to the left. If necessary, the number or size scaling of the drawn arrow objects can be adjusted with the *Size* slider.

To get an even better view, you can plot the field on a 2D plane. Select *2D/3D Plot: Sectional View* ⇨ *Fields on Plane*. Again, to adjust the plot quality, you can select *2D/3D Plot: Plot Properties* ⇨ *Properties*, and move the *Density* and *Size* sliders.

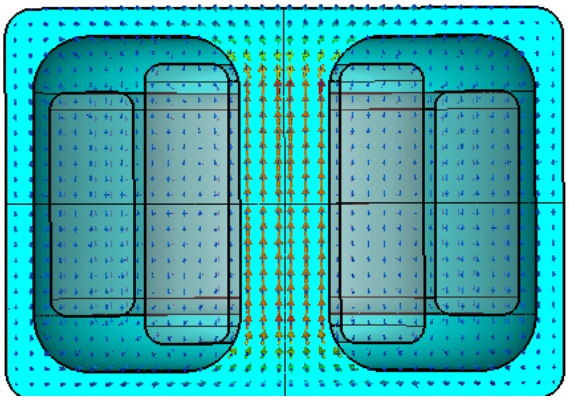


Before you continue, ensure that the local coordinate system is not active. In order to deactivate the local coordinate system, deselect *Modeling: WCS* ⇨ *Local WCS* ⇨ *Local WCS*. Note that it may be necessary to click on the *NT: Components* folder first.

After reselecting *NT: 2D/3D Results* ⇨ *B-Field [Ms],* switch off the “All Transparent” mode by clicking on *2D/3D Plot: Plot Properties* ⇨ *All Transparent*. Furthermore, use the *View* tab to adjust the view properly:

1. Select “Right” from the drop-down list in *View: Change View* ⇨ *Select View*.
2. Activate the Plane Rotation Mode (*View: Mouse Control* ⇨ *Rotate in Plane*).
3. Turn the plot 90 degrees by holding the left mouse button and moving the mouse.
4. Select *View: Change View* ⇨ *Reset View* to adjust the plot size.

A plot similar to the following should appear:



Afterwards, switch on the “All Transparent” mode again via *2D/3D Plot: Plot Properties ⇨ All Transparent* and deactivate the 2D plot mode by deselecting *2D/3D Plot: Sectional View ⇨ Fields on Plane*.

Please note: *At the right top corner in the main view, you can usually see a color ramp, which you can adjust by dragging its small markers, or in the plot properties. By default, it is scaled to the overall maximum of the 3D Field you are viewing. From time to time it may happen that, for example, the maximum of an active 2D cut plane is much smaller than the 3D maximum. In order to get a meaningful impression of the field then, it might be necessary to rescale the color ramp. This can be done, for example, in the context menu (right-click in the main view) by selecting Smart Scaling. To reset the view to the default, select Reset Scaling from the context menu.*

The inductance matrix was computed after the last adaptive run. The results are located in *NT: 1D Results ⇨ Ms Solver ⇨ Inductance Matrix* and contain both, the self- and the mutual inductances. You can either obtain a visual representation or inspect the numerical values in the *Result Navigator*, which is by default located as a tab in the window below the main view. You can select multiple results within one folder by holding down the Shift key and then clicking them with the left mouse button.

Result Navigator

3D Run ID	L coil 1, coil 1	L coil 1, coil 2	L coil 2, coil 1	L coil 2, coil 2
1	3.08538	2.46559	2.46559	2.00027
0: Current Run	3.08538	2.46559	2.46559	2.00027

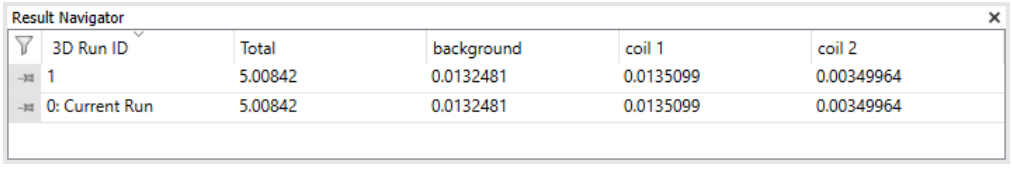
Note: For your convenience, you can move around and detach all visual tabs and windows by Drag&Drop. You can control which windows are visible by selecting or deselecting them via the drop-down list accessible through View: Window ⇨ Windows.

Finally, let us take a look at the total magnetic energy in the computational domain. Select all entries under *NT: 1D Results ⇨ Ms Solver ⇨ Energy* to obtain:

Result Navigator

3D Run ID	Total	background	coil 1	coil 2
1	5.00842	0.0132481	0.0135099	0.00349964
0: Current Run	5.00842	0.0132481	0.0135099	0.00349964

You can find the co-energy in a similar way under *NT: 1D Results ⇨ Ms Solver ⇨ Co-Energy*:



The energy and the co-energy are shown for each solid separately. Note that energy and co-energy are exactly the same since only linear materials have been used in the model.

Remember that the major advantage of the tetrahedral mesh is the explicit representation of the geometry, even in the course of adaptive refinement. A proper resolution of non-planar surfaces is very important, in particular, to model jumps in the field components at material interfaces. For very complex geometries, however, the generation of the tetrahedral mesh is sometimes rather time-consuming and requires a sufficient quality of the CAD data. An optional method is available which combines the simplicity of hexahedral meshes with the Perfect Boundary Approximation technique.

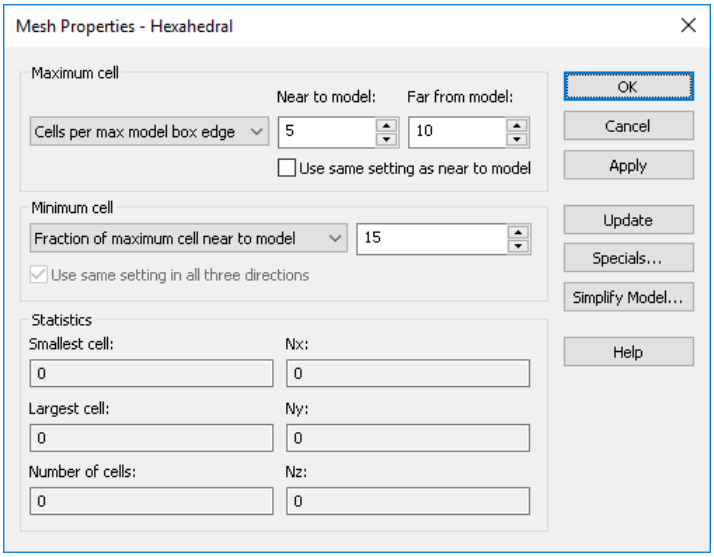
In the following subsections, let us compute the same model applying the hexahedral magnetostatic solver. Again, we will look at the mesh parameters and the visualization and then turn to the solver itself.

Visualize a Hexahedral Mesh

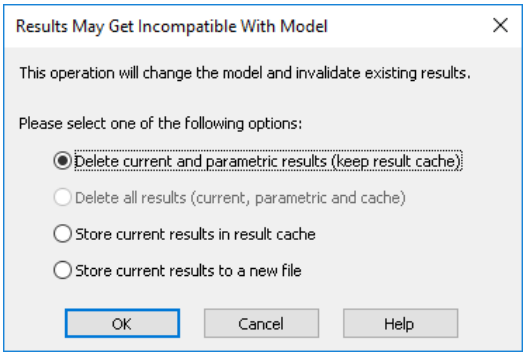
The hexahedral mesh generation for the structure analysis is performed fully automatically based on an expert system. As for tetrahedral meshes, it may be helpful in some situations to inspect the mesh before starting the solver in order to improve the simulation speed by changing the parameters for the mesh generation.

Note that in CST EM Studio generating hexahedral meshes is very fast compared to generating tetrahedral meshes. The reason is that by applying the Perfect Boundary Approximation feature, hexahedral meshes do not need to resolve the geometry: i.e. interfaces of materials and solids are not represented by a surface mesh as they are for tetrahedral meshes.

First, you must switch from tetrahedral to hexahedral meshing. Select *Home: Mesh ⇨ Global Properties ⇨ Hexahedral*  . Then, the Global *Mesh Properties - Hexahedral* dialog box will open automatically. For the purpose of this tutorial, the *Maximum cell - Near to model* value can be changed to 5.

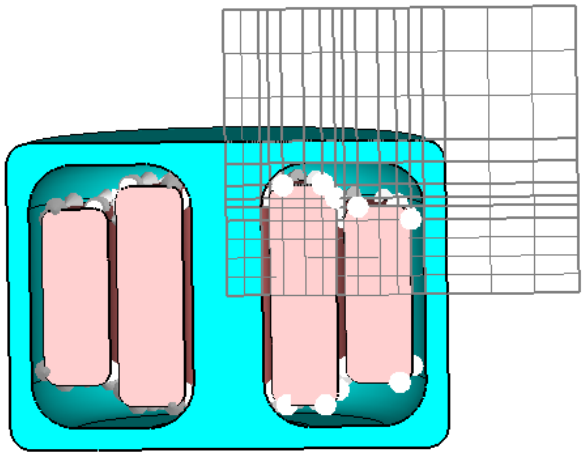


When you click the *OK* button you will be informed that the results have to be deleted:



Confirm the deletion of the results by clicking *OK*.

A hexahedral mesh will be generated automatically without further action. You can visualize the mesh by entering the mesh view (*Home: Mesh* ⇨ *Mesh View*). For this structure, the mesh information will be displayed as follows:



One 2D mesh plane will always be kept in view. Because of the symmetry settings, the mesh only extends across 1/8 of the structure (the mesh plane extends to 1/4). You can modify the orientation of the mesh plane by choosing *Mesh: Sectional View* ⇨ *Normal: X/Y/Z* (shortcut: X/Y/Z). You can move the plane along its normal direction with *Mesh: Sectional View* ⇨ *Position* or by pressing the *Up* / *Down* cursor keys.

In most cases, the automatic mesh generation produces a sufficient mesh, but we recommend that you spend some time later on studying the mesh generation procedures in the online documentation once you feel familiar with the standard simulation procedure.

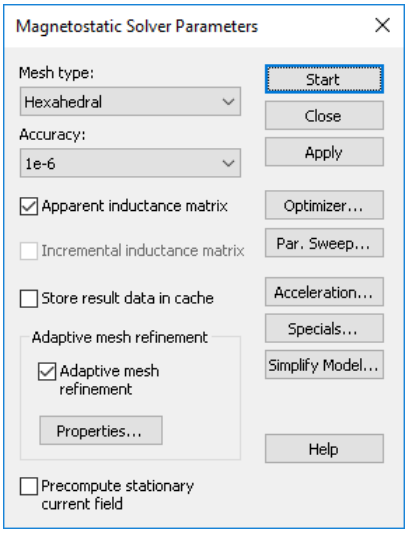
Leave the mesh inspection view by *Mesh: Close* ⇨ *Close Mesh View*.

Start the Hexahedral Solver

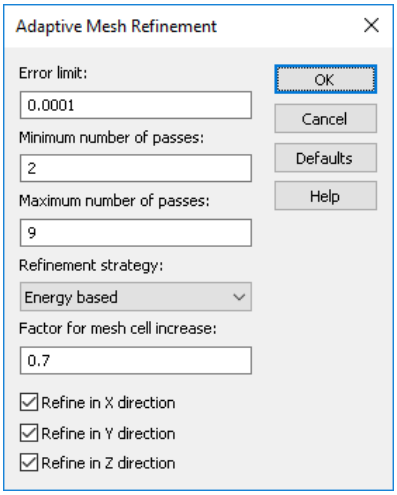
After you have defined all necessary parameters, you are ready to start your first simulation using the hexahedral solver. Again, start the simulation from the magnetostatic solver dialog box: *Home: Simulation* ⇨ *Setup Solver*. Within the solver dialog box, the "Hexahedral" mesh should be selected in the *Mesh Type* drop-down list. In order to compute inductances from the magnetic field, the box *Apparent inductance matrix* has to be checked. Ensure that the *Adaptive mesh refinement* is switched on (this is not the default for hexahedral meshes). Please recall the remarks on adaptive mesh refinement made in the section *Generate and Visualize a Tetrahedral Mesh*. They apply to hexahedral meshes as well.

The *Accuracy* value can be left unchanged. Please note that what is mentioned concerning the accuracy value in the tetrahedral solver subsection (e.g. its dependence on the discretization)

also applies to the hexahedral solver. After you set all these parameters, the dialog box should look like this:



Next enter the *Properties* dialog of the adaptive mesh refinement. The *Error limit* should be changed to 0.0001 and the *Maximum number of passes* to 9. The other settings can be kept at their default values.



Confirm your setting by pressing *OK*. Now start the simulation procedure by clicking *Start*. A few progress bars will appear in the status bar to keep you up-to-date with the solver's progress:

1. **Calculating coil excitations**: This first calculation step must be performed to calculate the discrete representation of coil current patterns.

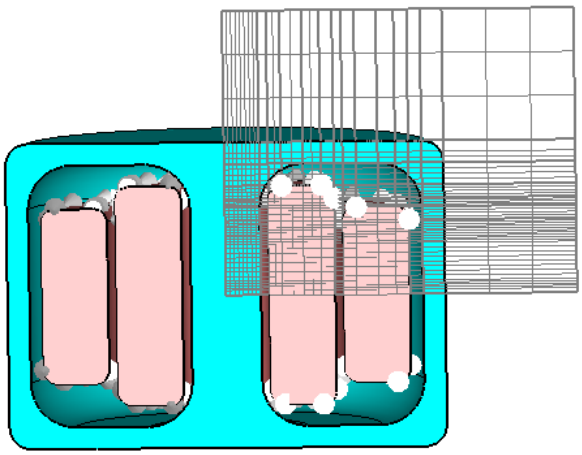
2. **Checking model:** During this step, your input model is checked for errors such as invalid overlapping materials, etc.

3. **Calculating matrix and dual matrix:** During these steps, the system of equations is set up, which will be solved subsequently.

4. **Solving linear system:** During this stage, a linear equation solver calculates the field distribution inside the structure.

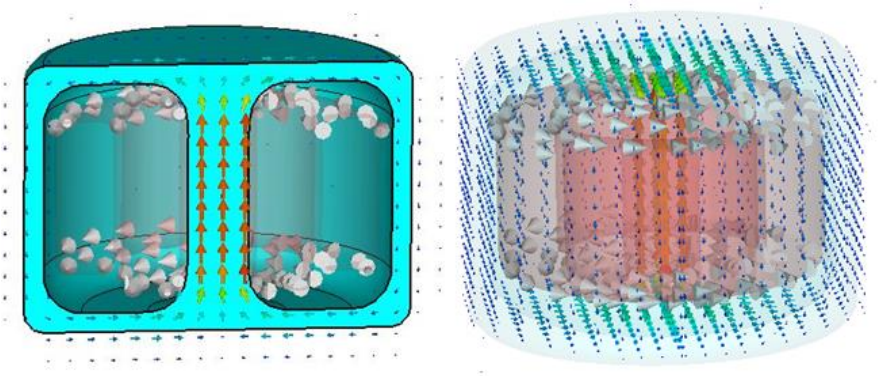
5. **Postprocessing:** From the field distribution, additional results like the inductance matrix or the energy within the calculation domain are calculated.

As for the tetrahedral solver, some error estimation and mesh refinement steps are performed in the case of adaptive mesh refinement. Note that several linear systems will be solved during the computation in order to compute all entries of the inductance matrix. For this simple structure, the entire analysis takes only a few seconds per adaption pass. After the simulation the mesh (*Home: Mesh ⇨ Mesh View*) should look similar to this:



Analyze the Results of the Hexahedral Solver

Now you can generate similar result plots as you did for the tetrahedral solver-run: Visualize the magnetic flux density by choosing *NT: 2D/3D Results ⇨ B-Field*. After you select this item a plot similar to the following should appear (possibly after some fine-tuning of the plot properties in *2D/3D Plot: Plot Properties ⇨ Properties*):

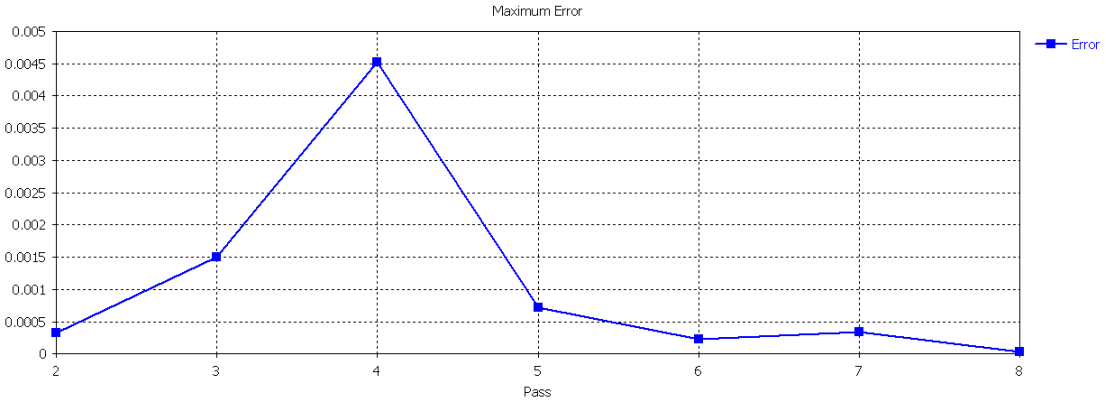


Again, you can switch between 2D and 3D-view as well as transparency mode via *NT: 2D/3D Results ⇨ Sectional View ⇨ Fields on Plane* and *2D/3D Plot: Plot Properties ⇨ All Transparent*, respectively.

To observe field values at certain positions within a 2D plot, activate *2D/3D Plot: Tools ⇨ Field at Cursor*. The field values will be displayed in the lower right corner of the main view. Note that for the scalar fields and for the vector fields projected on the plane you can add points to a *List of Field Values* with a double click in the main view.

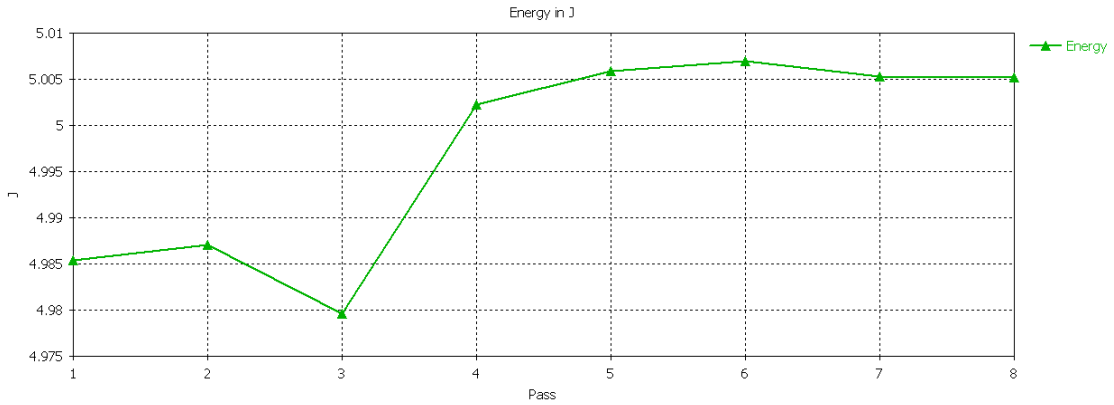
Several mesh refinement passes were performed automatically until the energy value did not change significantly between two subsequent passes. The default termination criterion is an energy deviation of 1% (or less).

The progress of the mesh refinement can be checked in the *NT: 1D Results ⇨ Adaptive Meshing* folder. This plot can be viewed by selecting *NT: 1D Results ⇨ Adaptive Meshing ⇨ Error*:



This result shows that the maximum difference of the energy error is below 0.05 %, i.e. below the error limit prescribed in the adaptive mesh refinement properties.

Additionally, the convergence of the energy can be visualized by selecting *NT: 1D Results ⇨ Adaptive Meshing ⇨ Energy*:



It can be seen that the hexahedral mesh generator already provides a good mesh for a first calculation. The small energy error shows that the adaptive mesh refinement is able to confirm that variations are reduced to a minimum.

In practice, it often proves judicious to activate the adaptive mesh refinement to ensure convergence of the results. (This might not be necessary for structures with which you are already familiar where you can use your experience to manually refine the automatic mesh.)

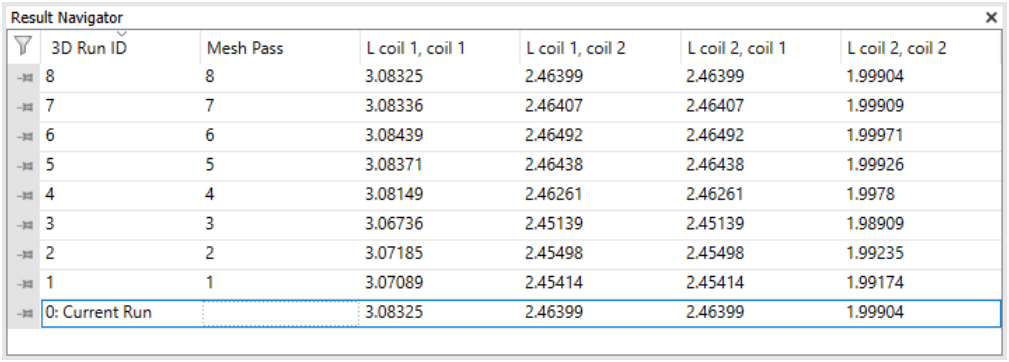
Now let us compare the magnetic energy computed by the hexahedral solver to the one computed by the tetrahedral solver. Select *NT: 1D Results ⇨ Ms Solver ⇨ Energy* to obtain the "Current Run" value for the total energy in the *Result Navigator:*

Result Navigator

3D Run ID	Mesh Pass	Energy
8	8	5.00514
7	7	5.00529
6	6	5.00697
5	5	5.00586
4	4	5.00225
3	3	4.97961
2	2	4.98708
1	1	4.98545
0: Current Run		5.00514

This is very similar to the value computed by the tetrahedral solver. The difference comes from the non-zero discretization errors. Moreover, fewer meshcells have been used for the hexahedral discretization.

In the solver dialog box, you have chosen to calculate the inductance matrix. To view the values of the inductance matrix, select all entries in *NT: 1D Results ⇨ Ms Solver ⇨ Apparent Inductance Matrix* to see them in the *Result Navigator.*


Result Navigator

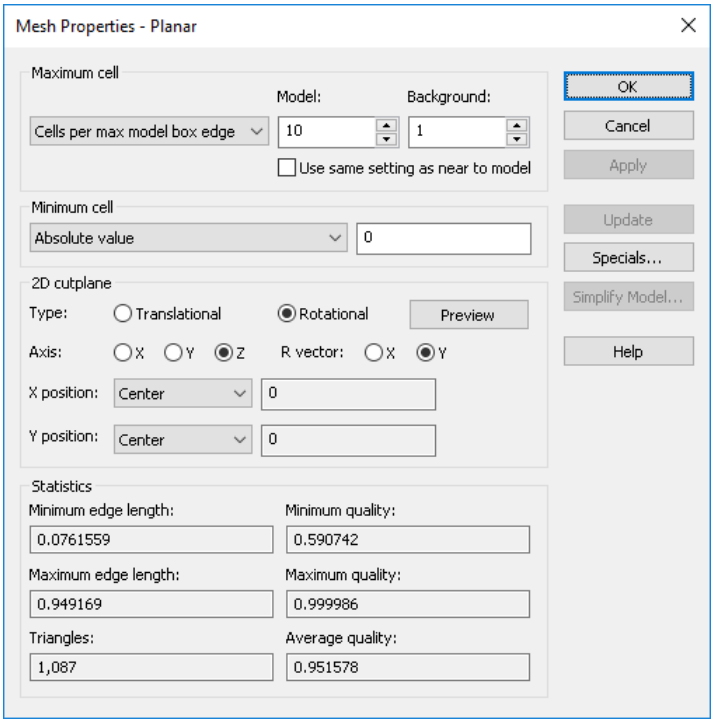
3D Run ID	Mesh Pass	L coil 1, coil 1	L coil 1, coil 2	L coil 2, coil 1	L coil 2, coil 2
8	8	3.08325	2.46399	2.46399	1.99904
7	7	3.08336	2.46407	2.46407	1.99909
6	6	3.08439	2.46492	2.46492	1.99971
5	5	3.08371	2.46438	2.46438	1.99926
4	4	3.08149	2.46261	2.46261	1.9978
3	3	3.06736	2.45139	2.45139	1.98909
2	2	3.07185	2.45498	2.45498	1.99235
1	1	3.07089	2.45414	2.45414	1.99174
0: Current Run		3.08325	2.46399	2.46399	1.99904

The results are in good agreement with those obtained with the tetrahedral mesh.

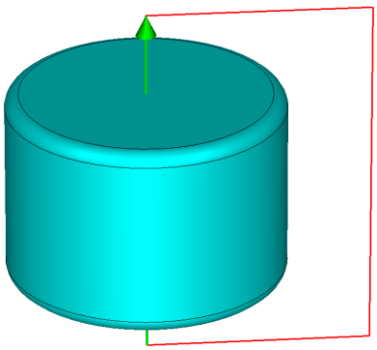
Create a Planar Mesh

For axis symmetric structures or structures for which boundary effects for one spatial dimension can be neglected, the 2D solver can be applied. The structure is designed as a 3D model and cut by a user defined plane. Compared to the 3D solvers, choosing this option might save a lot of computation time. Even if your model is not perfectly symmetric, this solver can give good estimates when starting with a new design.

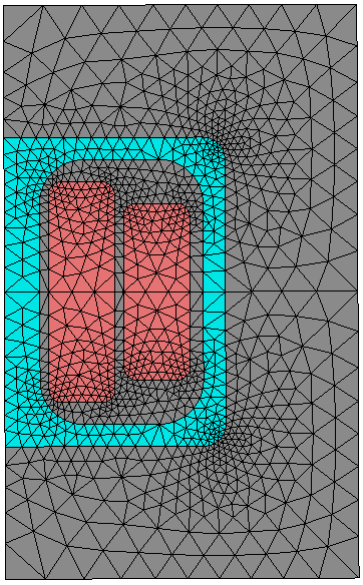
First, you must switch from hexahedral to planar meshing. Select *Home: Mesh* ⇨ *Global Properties* ⇨ *Planar* . The cutting plane alignment description as well as the 2D mesh setting are available then in the Mesh Properties dialog box, which will open automatically.



Select *Rotational* for the symmetry type and *Z* for the axis. The axis should be centered in the 3D domain, therefore select *Center* for the *X*- and *Y*-position. Finally select *Y* for the *R vector*. Also change the *Maximum cell – Model* to 10 and *Background* to 1. The *Preview* button allows checking the settings in the main view:



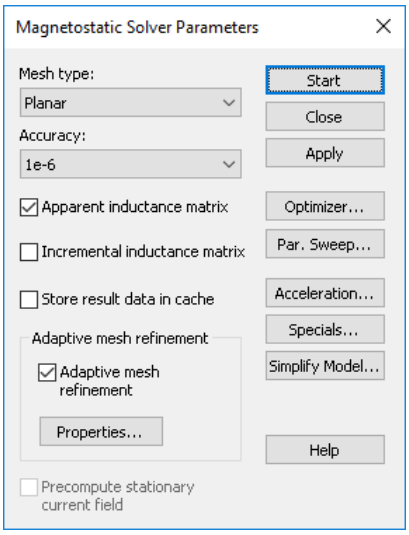
A first mesh can be created directly with the *Update* button. It will be shown a few seconds later:



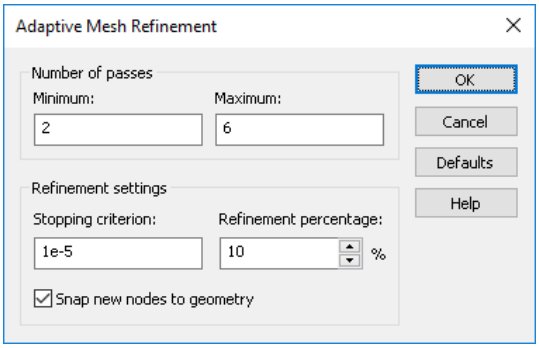
Finally, leave the Mesh Properties dialog box by pressing *OK*.

Start the Planar Solver

After you have defined all the necessary parameters, you are ready to start your first simulation using the planar solver. Again, start the simulation from the magnetostatic solver dialog box: *Home: Simulation ⇨ Setup Solver* . Within the solver setup menu, the "*Planar*" mesh should be selected in the *Mesh type* drop-down list. In order to compute the apparent inductances, the box *Apparent inductance matrix* has to be checked. Ensure that the *Adaptive mesh refinement* is switched on. The *Accuracy* value can be left unchanged.
After you set all these parameters, the dialog box should look like this:



Next, enter the *Properties* dialog of the adaptive mesh refinement. If necessary, change the *Stopping criterion* to 1e-5. The other settings can be kept at their default values.



Finally, close the dialog with the *OK* button and start the simulation procedure by clicking on *Start*. Like in the case of the previous simulations, several progress bars will appear in the status bar informing you about the current solver status.

These are the steps of the planar magnetostatic solver run:

1. **Computing coil(s)**: This first calculation step must be performed to calculate the discrete representation of coil current patterns.

2. **Initializing magnetostatic solver:** During this step, your input model is checked for errors such as invalid overlapping materials, not well-defined sources, etc.

3. **Assembling system:** The linear system of equations is generated.

4. **Constructing pre-conditioner:** This includes construction steps for the pre-conditioner of the solver, e.g. an LU-decomposition, a construction of hierarchy for a multigrid solver etc.

5. **Solving linear system:** During this stage, the equation system is solved yielding the unknown field.

6. **Estimating error** (only during mesh adaption pass): The local error for each element is estimated (error distribution).

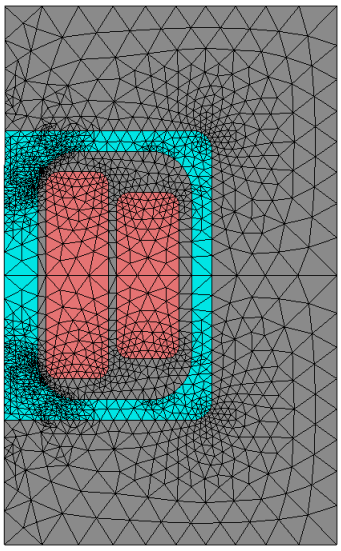
7. **Marking elements for refinement** (only during mesh adaption pass): Based on the computed error, a certain number of elements will be marked for refinement.

8. **Adapting mesh** (only during mesh adaption pass): The mesh is refined taking the marked elements into account.

9. **Inductance computation** (only if switched on): The apparent and/or incremental inductance matrix is calculated.

10. **Postprocessing stage:** From the field solution other fields and additional results like the energy within the structure are computed.

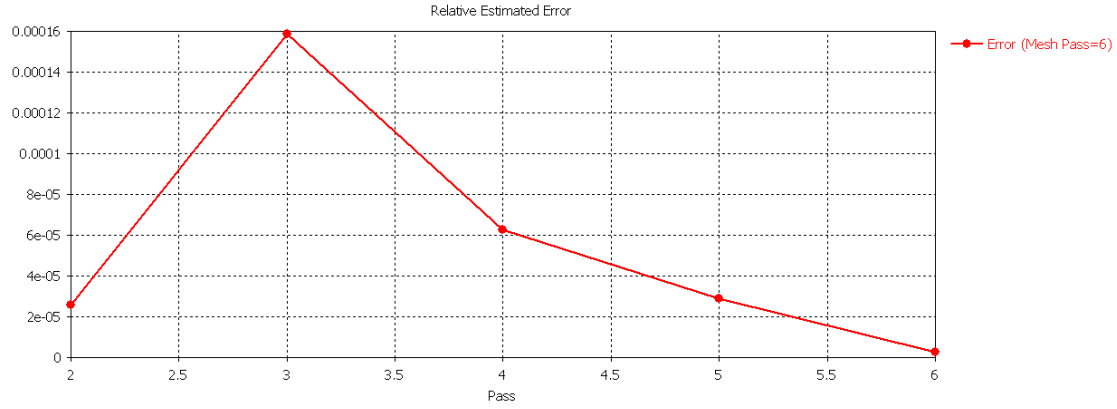
After the solver has finished, the mesh should look similar to the one depicted in the following picture (deviations are possible since the initial mesh can differ slightly depending on the operating system and the architecture of the machine):



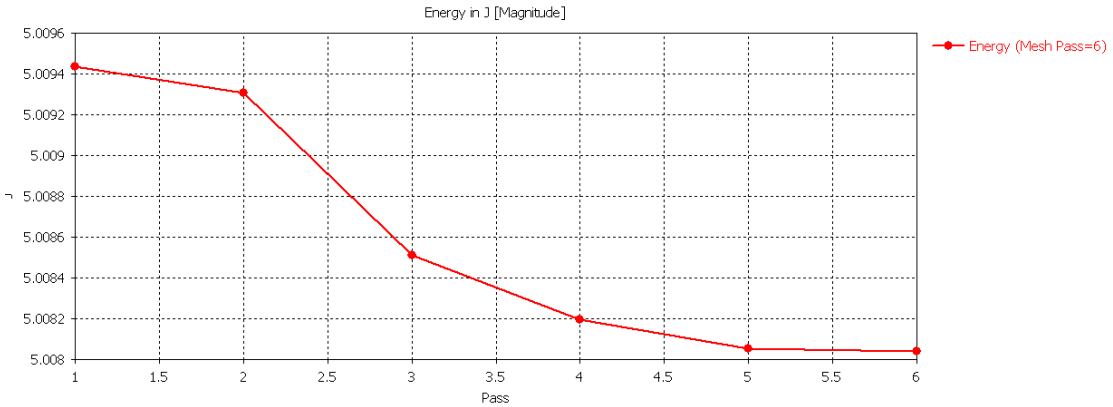
Analyze the Results of the Planar Solver

Already during the planar solver run, you can watch the progress of the mesh refinement and the convergence behavior in the *NT: 1D Results ⇨ Adaptive Meshing* folder.

Click, for instance, on *NT: 1D Results ⇨ Adaptive Meshing ⇨ Error.* This folder contains a curve which displays the change of the relative energy of two subsequent simulations. From this result, we can observe that the maximum difference of the relative change of the energy is below the desired stopping criterion 1e-5:

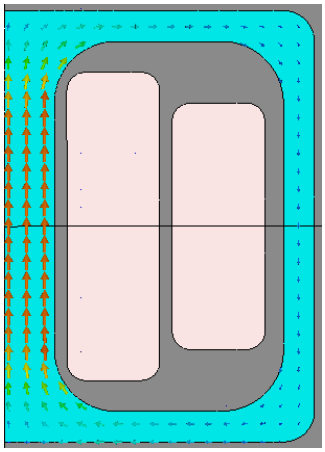


Additionally, the energy convergence can be visualized by selecting in the *NT: 1D Results ⇨ Adaptive Meshing ⇨ Energy:*



The curves can slightly differ when computed on a 32 bit or a 64 bit machine. The number of adaptation passes needed for convergence can also deviate depending on the machine architecture.

Now, you can visualize the magnetic flux density by choosing *NT: 2D/3D Results ⇨ B-Field*. After you select this item and fine-tune the plot properties in *2D/3D Plot: Plot Properties*, a plot similar to the following one should appear:



Completing the analysis of the planar solver results, let us compare the magnetic energy and the apparent inductance values computed by this solver to the ones computed by the 3D solvers.

To view the magnetic energy result, select all the results in *NT: 1D Results ⇨ Ms Solver ⇨ Energy* and check them in the *Result Navigator:*

Result Navigator

3D Run ID	Total	background	coil 1	coil 2
1	5.00804	0.0127107	0.0135074	0.003499
0: Current Run	5.00804	0.0127107	0.0135074	0.003499

The co-energy results you can similarly find in *NT: 1D Results ⇨ Ms Solver ⇨ Co-Energy:*

Result Navigator

3D Run ID	Total	background	coil 1	coil 2
1	5.00804	0.0127107	0.0135074	0.003499
0: Current Run	5.00804	0.0127107	0.0135074	0.003499

These results are very similar to the ones computed by the tetrahedral and hexahedral solvers. The results for the apparent inductance computation can be found under *NT: 1D Results ⇨ Ms Solver ⇨ Inductance Matrix*:

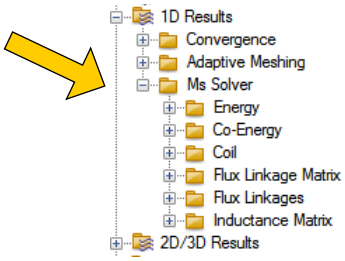
Result Navigator

3D Run ID	L coil 1, coil 1	L coil 1, coil 2	L coil 2, coil 1	L coil 2, coil 2
1	3.08514	2.46541	2.46541	2.00012
0: Current Run	3.08514	2.46541	2.46541	2.00012

These results are also in good agreement with those computed with the 3D solvers.

Accessing the Single-Value Results

All single-value results can be found in the *NT*. For the magnetostatic solver, result values for energy, co-energy, coil characteristics, flux linkages and inductances are quickly accessible from *NT: 1D Results ⇨ Ms Solver* folder:



The same data and more complex post processing results are also available via the *Template Based Post Processing* tool.

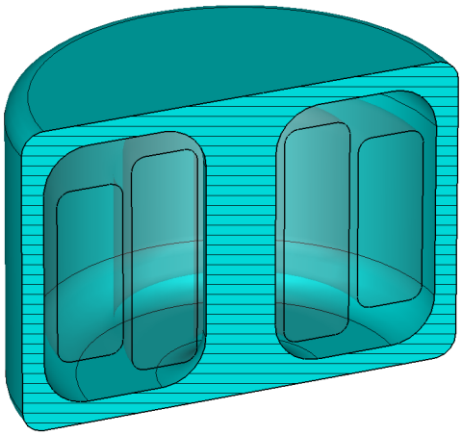
Parameterization and Automatic Optimization of the Structure

The steps above demonstrate how to enter and analyze a simple structure. However, structures are usually analyzed to improve their performance. This procedure is called “design” in contrast to “analysis”.

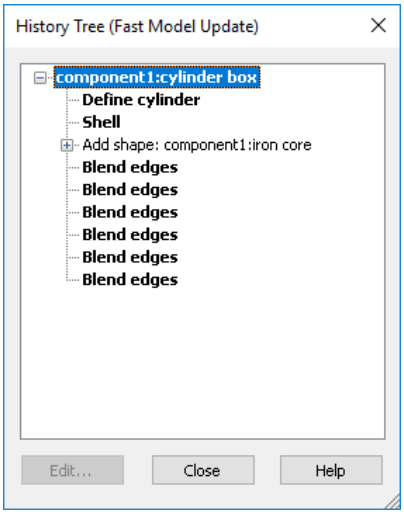
After you receive some information on how to improve the structure, you will need to change the structure’s parameters. This could be done by simply re-entering the structure, but this is not the most efficient solution.

CST EM Studio offers various options to describe the structure parametrically in order to change the parameters easily. The *History List* function, described in the *CST Studio Suite Getting Started* manual, is a general option, but for simple parameter changes there is an easier solution, which is described below.

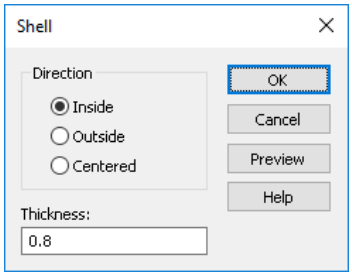
Let us assume you want to change the thickness of the transformer’s box. The easiest way to do this is to select the box by clicking on *NT: Components ⇨ component1 ⇨ cylinder box*. You may also need to rotate the structure in order to see a plot similar to the following (the cutting plane is still switched on):



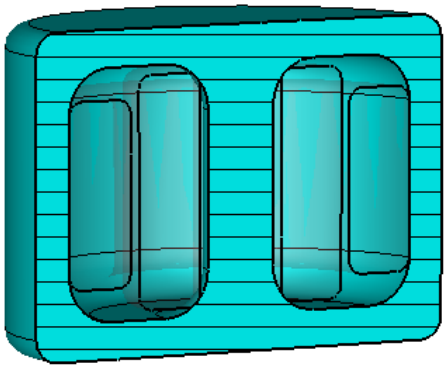
You can now choose *Modeling: Edit ⇨ Properties (Ctrl+E)* to open a list showing the history of the shape’s creation:



Select the “Shell” operation from the history tree (see above). After you click *Edit*, the shell dialog will appear. In this dialog box, you will find the thickness of the box (*Thickness* = 0.5) as specified during the shape creation. Change this parameter to a value of 0.8 and click *OK*.



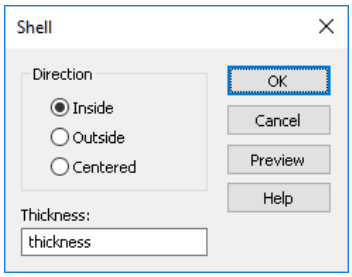
Confirm the deletion of the results by clicking *OK*.
The structure plot will change showing the new structure with the new box thickness:



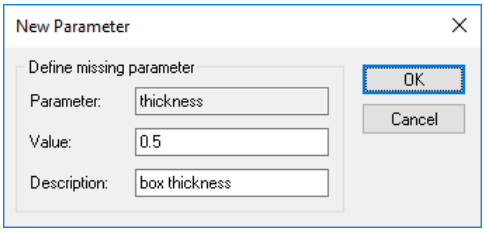
You can generally change all parameters of any shape by selecting the shape and editing its properties. This fully parametric structural modeling is one of CST EM Studio’s most outstanding features.
The parametric structure definition also works if some objects have been constructed relative to each other using local coordinate systems. In this case, the program will try to identify all the picked faces according to their topological order rather than their absolute position in space.

The changes in parameters occasionally alter the topology of the structure too severely, so the structure update may fail. In this case, the History List function offers powerful options to circumvent these problems. Please refer to the online documentation, or contact technical support.

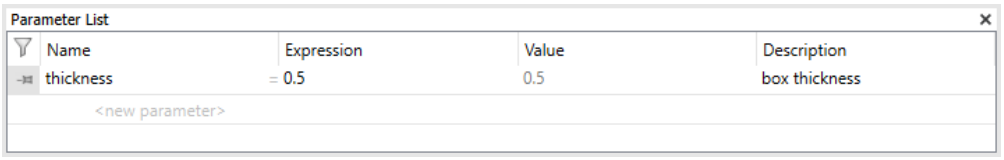
You may also assign variables to the structure parameters: Select the “Shell” operation from the history tree again (the dialog box should be still open) and click *Edit.* Now enter the string "thickness" as depicted below:



Then click *OK*. A new dialog box will open asking you to define the new parameter "thickness". Here enter 0.5 in the *Value* field. You may also provide a text in the *Description* field so that you can later remember the meaning of the parameter:



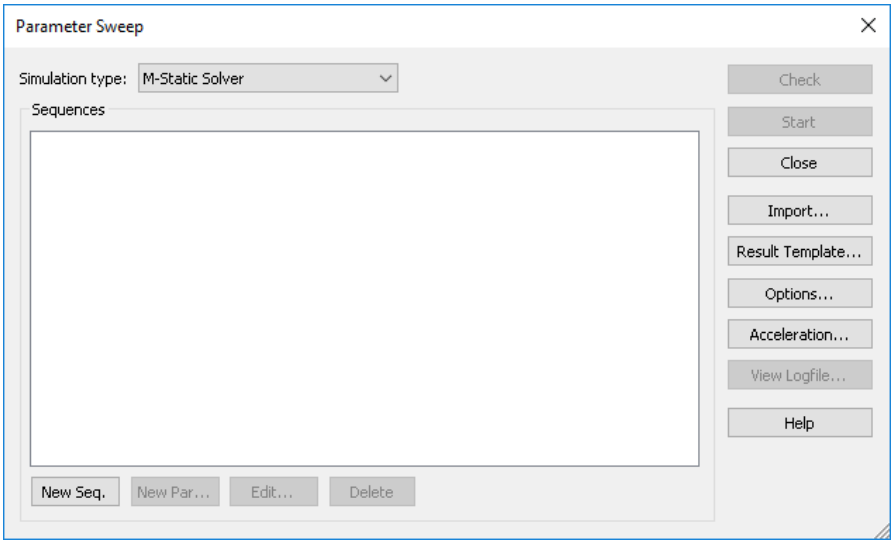
Closing this dialog box by clicking *OK* defines the parameter and updates the model. Now also close the History Tree window. Note that all defined parameters are listed in the parameter docking window:



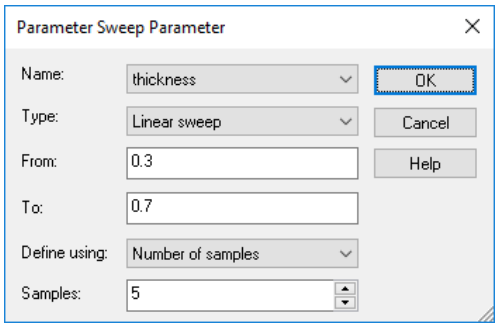
The *Parameter List* shares the same space with the *Result Navigator* and it may be necessary to select the *Parameter List* tab in in the lower part of the CST Studio Suite.

You can change the value of parameters by clicking on the corresponding entry in the *Expression* column of the parameter window and entering a new value. If you do this, the message “*Some variables have been modified. Press ‘Home: Edit ⇨ Parametric Update (F7)’* ” will appear in the main view. Then, if you perform this update operation, the structure will be regenerated according to the current parameter value. You can verify that parameter values between 0.3 and 0.7 give useful results. The function *Modeling: Edit ⇨ Parameters ⇨ Animate Parameter* is also useful in this regard. It is also possible to define a new parameter by entering it in the parameter window.

Since you now successfully parameterized your structure, it might be interesting to see how the apparent inductance values change when the thickness of the box is varied. The easiest way to obtain these variation results is to use the Parameter Sweep tool accessible from within the magnetostatic solver dialog box (*Simulation: Solver ⇨ Setup Solver*). Note that the *Planar Mesh type* with *Adaptive mesh refinement* is still selected. Click the *Par. Sweep* button to open the following dialog box:



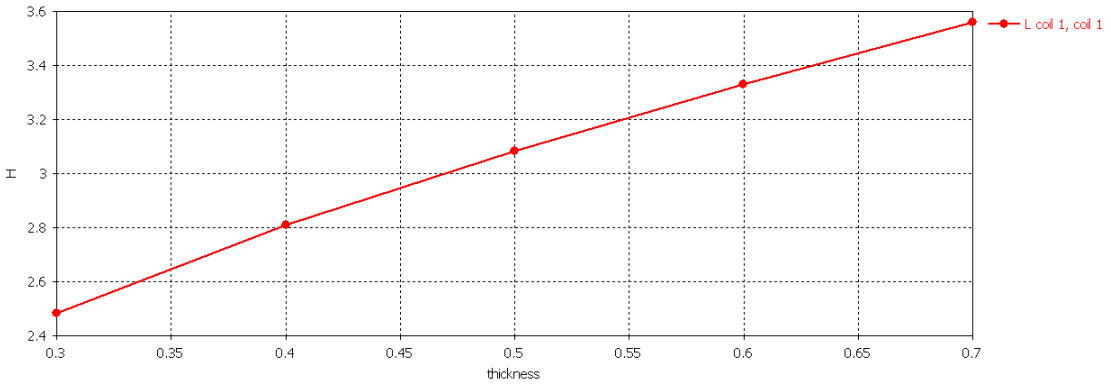
In this dialog box, you can specify calculation “sequences”, which consist of various parameter combinations. To add such a sequence, click the *New Seq.* button. Then click the *New Par* button to add a parameter variation to the sequence:



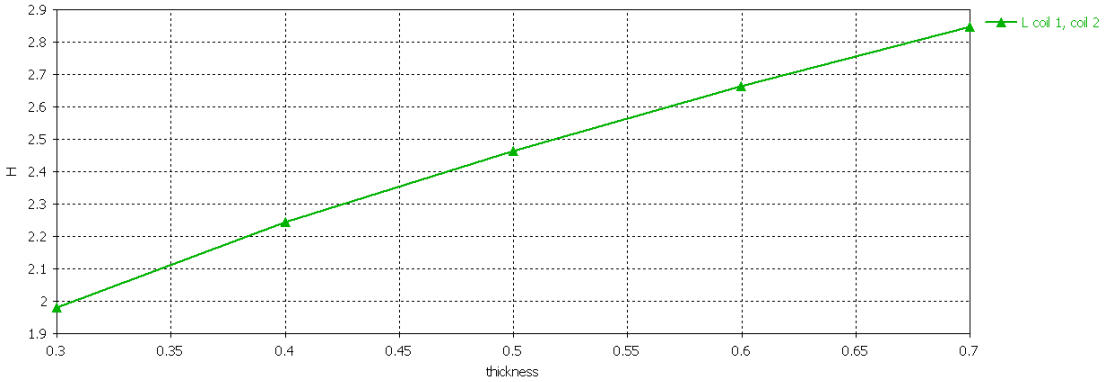
In the dialog box that arises, you can select the name of the parameter to vary in the *Name* drop-down list. After selecting the item to sweep, you can specify the lower (*From*) and upper (*To*) bounds for the parameter variation. Finally, enter the number of steps in which the parameter should be varied in the *Samples* field.

In this example, the *thickness* of the box should be swept *From* 0.3 *To* 0.7 in 5 *Samples*. After you click *OK*, the parameter sweep setting will appear in the *Sequences* frame. Note that you can define an arbitrary number of sequences each containing an unlimited number of different parameter combinations.

Now run the parameter sweep by clicking *Start*. A progress bar in the *Progress* window shows the current status of the parameter sweep. After the solver has finished its work, you will find the results in the navigation tree*: NT: 1D Results ⇨ Ms Solver ⇨ Inductance Matrix*, where the respective inductance values can be plotted against the parameter values covered by the sweep. Selecting *L coil 1, coil 1*. If the X axis is based on Run IDs (0..5), switch to the parametric X axis by selecting “*Parametric*” from the dropdown list available at *1D Plot: 0D Result Axis ⇨ X Axis*. You should get a graph similar to the following:



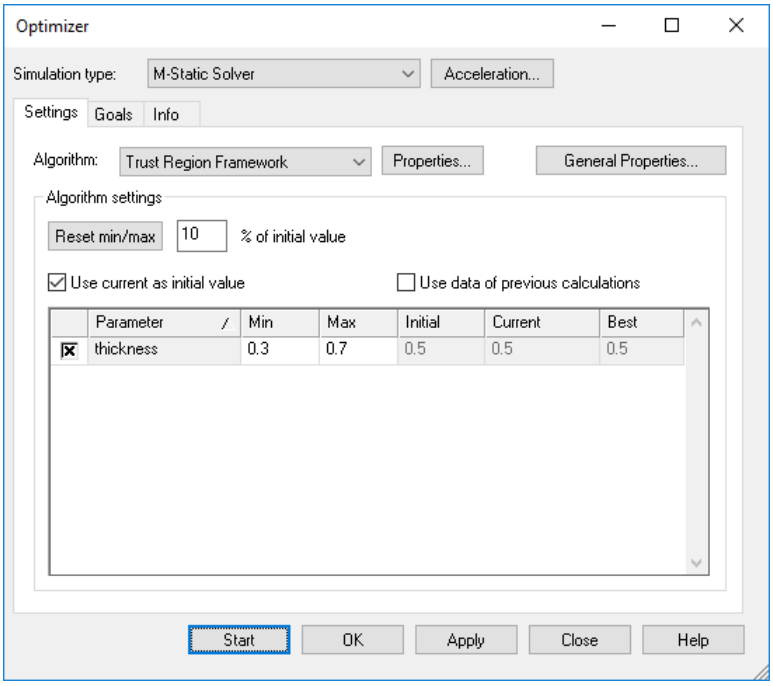
Choosing *NT: 1D Results ⇨ Ms Solver ⇨ Inductance Matrix ⇨ L coil 1, coil 2, y*ou can inspect the mutual inductance in the same way:



Assume that you now want to adjust the self-inductance of coil 1 to a value of 3.2 H (which can be achieved within a parameter range of 0.3 to 0.7 according to the parameter sweep). However, figuring out the proper parameter may be a lengthy task that can be performed equally well automatically.

Before you continue to optimize this structure, ensure that the thickness parameter is within the valid parameter range (e.g. 0.5). If you have to modify the value, do not forget to update (*Home: Edit ⇨ Parametric Update* *(F7)*) the structure afterwards. Note that you must enter the modeler mode, e.g. by clicking on the "*Components*" item in the navigation tree, before you modify the parameters.

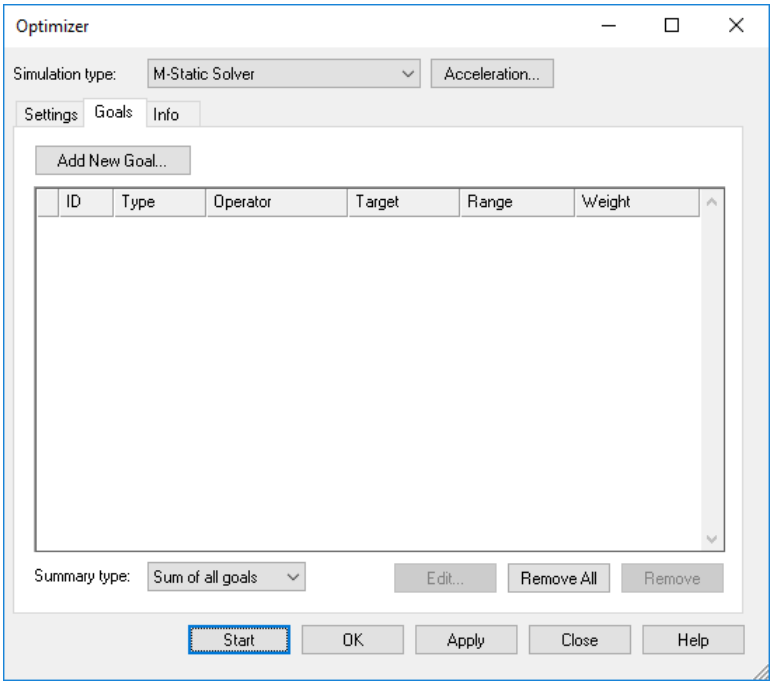
CST EM Studio offers a very powerful built-in optimizer feature for parametrical optimizations. To open the optimizer control dialog box, select *Simulation: Solver ⇨ Optimizer*:



First, check the desired parameter(s) for the optimization in the *Settings* tab of the optimization dialog box (here the “thickness” parameter should be checked). Next specify the minimum and maximum values for this parameter during the optimization. Here you should enter a parameter range between 0.3 and 0.7. Refer to the online documentation for more information on these settings.

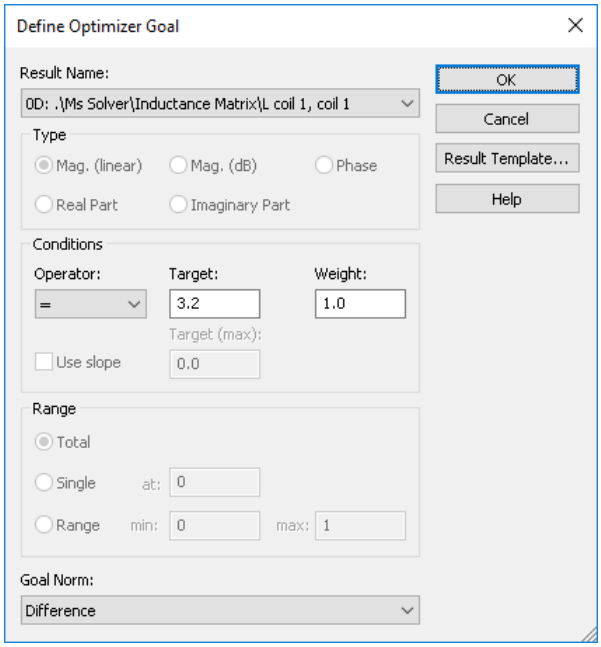
To store the parametric results calculated during the optimizer run, the Result storage settings should be changed in the *General Properties* dialog from the default “*None*” to “*Automatic*”.

Next specify the optimization goal. Hence, please click on the *Goals* tab:

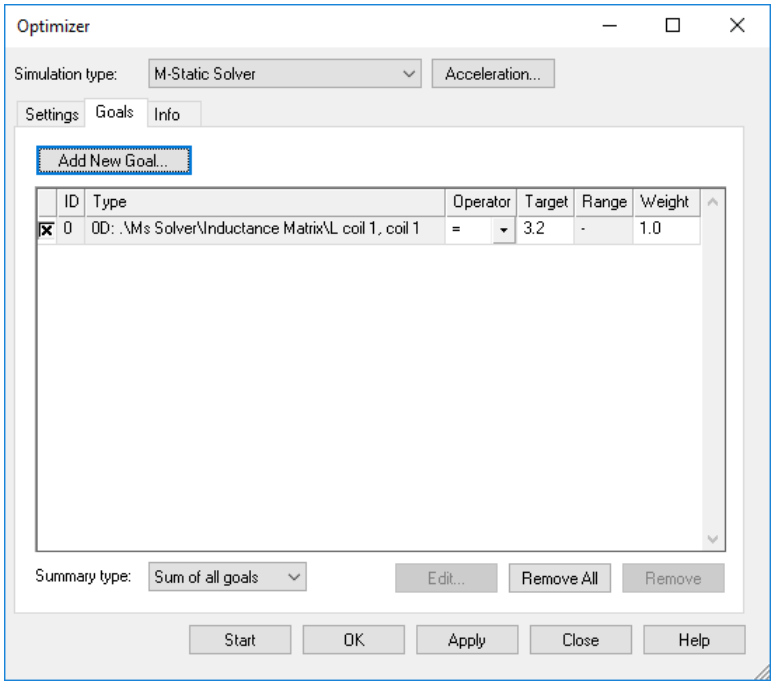


Here you can specify a list of goals to be achieved during the optimization. In this example, the target is to find a parameter value for which the self-inductance of coil 1 is 3.2 H.

Therefore, click on the *Add New Goal* button. A new dialog box will open: *Define Optimizer Goal*. Since you want to find the thickness value for a self-inductance of 3.2 H, select the corresponding result name (*0D: .\Ms Solver\Inductance Matrix\L coil 1, coil 1*) and the equal operator in the *Conditions* frame and set the *Target* to 3.2:

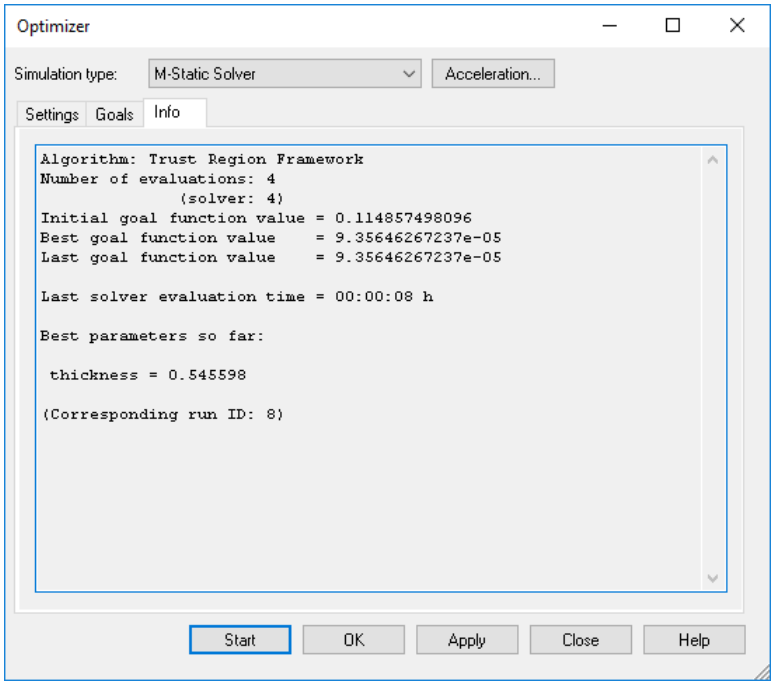


After you click *OK*, the optimizer dialog box should look as follows:



Since you now specified which parameters to optimize and set the goal for the optimization, the next step is to start the optimization procedure by clicking *Start*. The optimizer will show the progress of the optimization in an output window in the *Info* tab, which is activated automatically.

When the optimization is done, the optimizer output window shows the best parameter settings to achieve the optimization goal:



Note that due to the sophisticated optimization technology, only a few solver runs were necessary to find the optimal solution with high accuracy. It even reuses your previously computed results for a more efficient use of resources.

Now check the inductance value for the optimal parameter setting (thickness = 0.5456) by clicking *NT: 1D Results ⇨ Ms Solver ⇨ Inductance Matrix ⇨ L coil 1, coil 1*. The computed inductance is very close to the target value:

Result Navigator

3D Run ID	thickness	Mesh Pass	L coil 1, coil 1
8	0.545598	6	3.19991
7	0.544715	6	3.19773
6	0.504	6	3.09542
5	0.7	4	3.56067
4	0.6	6	3.33091
3	0.5	6	3.08514
2	0.4	6	2.81035
1	0.3	4	2.4843
0: Current Run			3.19991

This ends the first application example.

Summary

This example should have given you an overview of the key concepts of CST EM Studio. You should now have a basic idea of how to do the following:

1. Model the structures by using the solid modeler
2. Specify the solver parameters, check the mesh and start the simulation using the tetrahedral solver with the adaptive mesh refinement feature
3. Specify the solver parameters, check the mesh and start the simulation using the hexahedral solver with the adaptive mesh refinement feature
4. Visualize the magnetic field distributions
5. Specify the solver parameters, check the mesh and start the simulation using the planar solver with the adaptive mesh refinement feature

6. Define the structure using structure parameters
7. Use the parameter sweep tool and visualize parametric results
8. Perform automatic optimizations

If you are familiar with all these topics, you have a very good starting point for an even more productive use of CST EM Studio.

For more information on a particular topic, we recommend that you browse through the online help system which can be opened by selecting *File: Help ⇨ Help Contents – Get help using CST Studio Suite* . If you have any further questions or remarks, please do not hesitate to contact your technical support team. We also strongly recommend that you participate in one of our special training classes held regularly at a location near you. Please ask your support center for details.

Chapter 3 – Solver Overview

Solvers and Sources

The example in the previous chapter demonstrates how to define a coil source for a magnetostatic simulation. The general workflow of electrostatic, stationary current or low-frequency problems is quite similar to a magnetostatic application.

The different simulation types differ in the definition of materials, boundary conditions and excitation sources. The way to define materials and boundary conditions in CST EM Studio is quite similar for all solvers, whereas there are larger differences in the definition of sources. For this reason, an overview of the sources that are interpreted by each solver is given below.

Magnetostatic Solver:

- Permanent magnet:
  Simulation: Sources and Loads ⇨ Permanent Magnet
- Current or voltage coil:
  Simulation: Sources and Loads ⇨ Coil
- Coil segment:
  Simulation: Sources and Loads ⇨ Coil ⇨ Coil Segment
- Coil group:
  Simulation: Sources and Loads ⇨ Coil ⇨ Coil Group
- Current path:
  Simulation: Sources and Loads ⇨ Current Path
- External magnetic field:
  Simulation: Sources and Loads ⇨ Magnetic Source Field
- Stationary current field (via Solver checkbox) – the most important stationary current sources (*Electric Potential* and *Current Port*) are available from the *Simulation: Sources and Loads* toolbar as well

Typical applications are: magnets, magnetic valves, actuators, motors, generators and sensors.

Electrostatic Solver:

- Potential definition on a PEC (perfect electric conductor) solid:
  Simulation: Sources and Loads ⇨ Electric Potential
- Capacitive field grading on a PEC:
  Simulation: Sources and Loads ⇨ Electric Potential ⇨ Field Grading
- Potential definition on a normal/electric boundary:
  Simulation: Settings ⇨ Boundaries
  (select the *Boundary Potentials* tab from within the Boundary dialog box)
- Charge definition on a PEC:
  Simulation: Sources and Loads ⇨ Electric Charge on PEC
- Uniform volume- or surface-charge distribution:
  Simulation: Sources and Loads ⇨ Electric Charge Distribution

Typical applications are: high voltage devices, capacitors, MEMS and sensors.

Stationary Current Solver:

- Potential definition on a PEC solid:
  Simulation: Sources and Loads ⇨ Electric Potential
- Current port:
  Simulation: Sources and Loads ⇨ Current Port
- Field import:
  Simulation: Sources and Loads ⇨ Field Import
- Current path:
  Simulation: Sources and Loads ⇨ Current Path
- Coil segment:
  Simulation: Sources and Loads ⇨ Coil segment
- Coil segment from Solid:
  Simulation: Sources and Loads ⇨ Coil Segment from Solid

Typical applications are: sensors, coils, circuit breakers, IR drop simulations and grounding problems.

LF Frequency Domain Solver (Full Wave and Magnetoquasistatics):

- Current or voltage coil:
  Simulation: Sources and Loads ⇨ Coil
- Coil segment:
  Simulation: Sources and Loads ⇨ Coil ⇨ Coil Segment
- Coil segment from Solid (broadband only):
  Simulation: Sources and Loads ⇨ Coil Segment from Solid
- Coil group:
  Simulation: Sources and Loads ⇨ Coil ⇨ Coil Group
- Current port:
  Simulation: Sources and Loads ⇨ Current Port
- Current path:
  Simulation: Sources and Loads ⇨ Path Sources⇨ Current Path From Curve
- Voltage path:
  Simulation: Sources and Loads ⇨ Path Sources ⇨ Voltage Path from Curve
- External magnetic field:
  Simulation: Sources and Loads ⇨ Magnetic Source Field
- Field import:
  Simulation: Sources and Loads ⇨ Field Import

LF Frequency Domain Solver (Electroquasistatics):

- Potential definition on a PEC solid:
  Simulation: Sources and Loads ⇨ Electric Potential

Typical applications are: NDT, proximity sensors, inductively coupled power transfer, induction heating, magnetic and electric design of transformers.

LF Time Domain Solver (Magnetoquasistatics):

- Permanent magnet:
  Simulation: Sources and Loads ⇨ Permanent Magnet
- Current or voltage coil:
  Simulation: Sources and Loads ⇨ Coil
- Coil segment:
  Simulation: Sources and Loads ⇨ Coil ⇨ Coil Segment
- Coil segment from Solid:
  Simulation: Sources and Loads ⇨ Coil Segment from Solid
- Coil group:
  Simulation: Sources and Loads ⇨ Coil ⇨ Coil Group
- Current port:
  Simulation: Sources and Loads ⇨ Current Port

- Current path:
  Simulation: Sources and Loads ⇨ Path Sources ⇨ Current Path From Curve
- Voltage path:
  Simulation: Sources and Loads ⇨ Path Sources ⇨ Voltage Path from Curve
- External magnetic field:
  Simulation: Sources and Loads ⇨ Magnetic Source Field
- Rotational motion:
  Simulation: Motion ⇨ Motion ⇨ New Rotation
- Translational motion:
  Simulation: Motion ⇨ Motion ⇨ New Translation

LF Time Domain Solver (Electroquasistatics):

- Potential definition on a PEC solid:
  Simulation: Sources and Loads ⇨ Electric Potential

Typical applications are: transient device switching, nonlinear time-dependent problems such as electrical machines, sensors and high-voltage transformers.

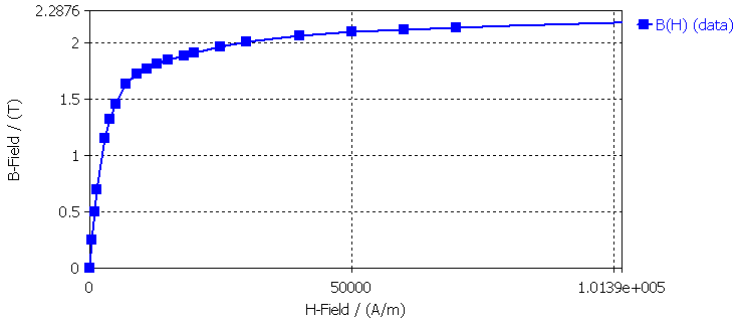
Magnetostatic Solver

The magnetostatic solver can be used for static magnetic problems. Available sources are current paths, current or voltage coils, coil segments, coil groups, permanent magnets and homogeneous magnetic source fields as well as the current density field previously calculated by the *stationary current solver*. To use the J-static current density field as magnetostatic source, activate the checkbox *Precompute stationary current field* in the Magnetostatic Solver dialog box. The stationary current field will then be precomputed automatically.

The main task of the solver is to calculate the magnetic field strength and the flux density. These results appear automatically in the navigation tree after the solver run.

Nonlinear ferromagnetic Materials

The magnetostatic solver also features nonlinear ferromagnetic materials. These can be defined by creating a BH-curve describing a soft-magnetic material behavior or by creating a JH-curve describing a hard-magnetic material behavior. A nonlinear solver will use a smoothed version of this curve in order to improve the convergence. The resulting permeability distribution is also stored and can be accessed in the navigation tree. Below an example of a soft-magnetic BH-curve is shown.



Inductance Calculation

The magnetostatic solver can extract the inductance matrices of coils and coil segments. For nonlinear material properties, the nonlinear characteristic of the material is taken into account. The user may choose the extraction of the apparent inductance matrix and/or the incremental inductance matrix. For *n* coils and coil segments, the computation of the inductance matrix requires the solution of *n* equation systems. If all material properties are constant (i.e. type is

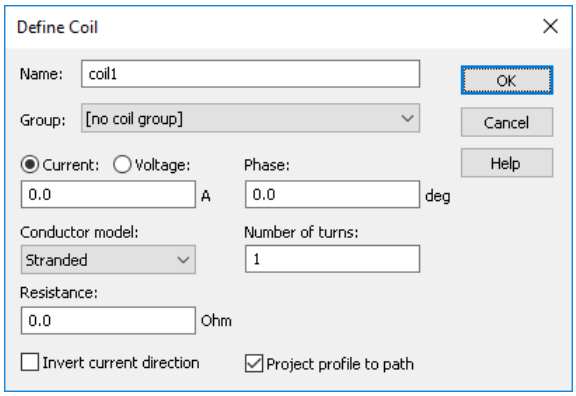
Normal and no nonlinear properties have been defined), the apparent and the incremental inductances are identical.

Current or Voltage Coils

In the section *Define Coils* of the previous chapter, the main ideas of the simulation of coils in CST EM Studio are already outlined. Moreover, you can find a step-by-step description of a coil creation there.

Remember that a current and voltage coil is defined as an *a-priori* known current distribution (also for voltage driven coils) which is constant over the cross-section of the coil body. The supporting material has no influence on the source current distribution.

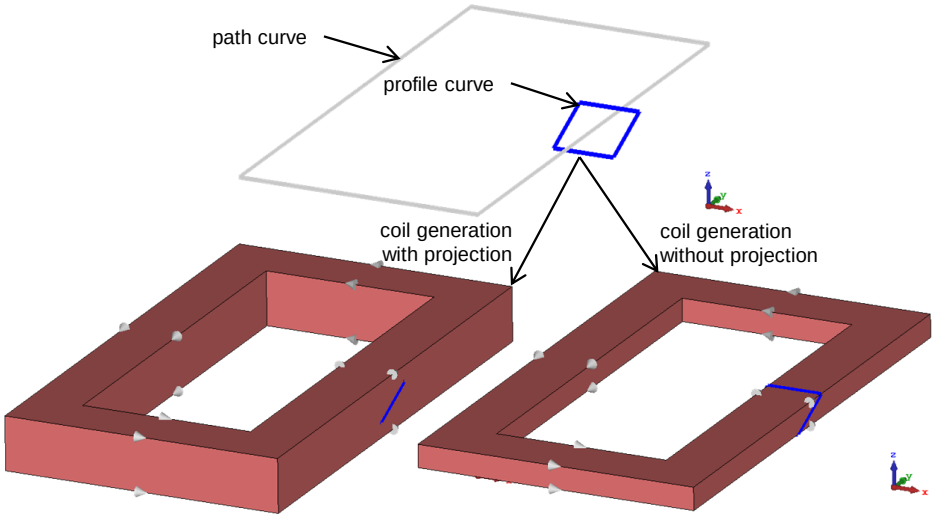
A coil in CST EM Studio can be constructed from two curves – the *profile curve* and the *path curve*. To create a current coil, you must define these two curves and then select *Simulation: Sources and Loads ⇨ Coil* . You will be prompted to select the coil profile curve and then the coil path curve. When the profile curve can be swept along the path curve successfully, the *Define Coil* dialog box will open automatically:



In this dialog box, you can specify the *Name,* the *Group* and the *Conductor Type* (*Stranded* or *Solid*) as well as the *current* or *voltage* value, the *Number of turns* and the ohmic *Resistance* of a coil. The *Phase* value is relevant only for LF Frequency Domain simulations. The current direction can be reverted by checking *Invert Current Direction*.

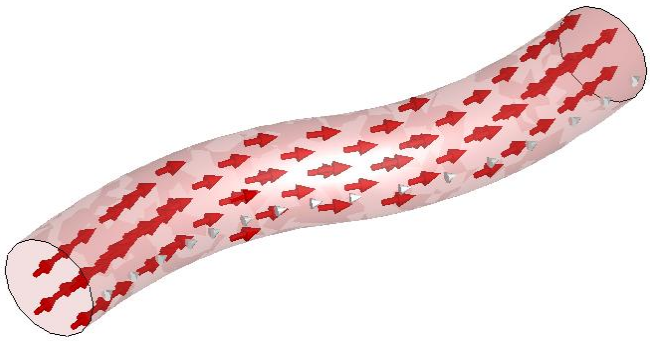
Depending on the physical connections, coil sources can be gathered into so-called coil groups. A current or voltage coil group is represented by a series connection of coils characterized by a common current flowing through them. For the voltage coil groups, the total voltage is defined by the individual coil group voltages. A coil group can be understood as a single conductor, only a single flux linkage embraces the coil group, and the coil group will only contribute to an inductance matrix as a single entity.

When the *Project profile to path* checkbox is activated, the profile curve is aligned with the plane which is normal to the path curve. In the following example you can see the profile curve, which includes an angle of 10 degrees with the path curve. The coil on the left hand side will be obtained if the alignment is activated. To generate the coil displayed on the right hand side, the alignment is switched off so that the profile is swept unchanged along the path curve.



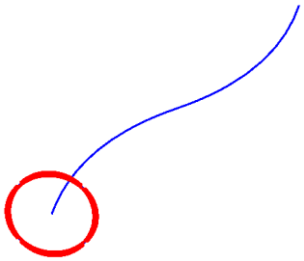
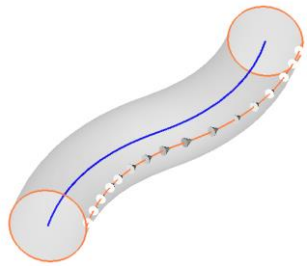
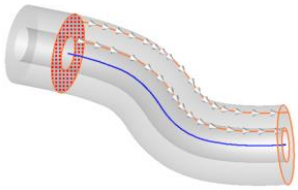
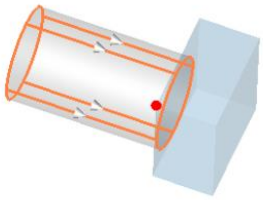
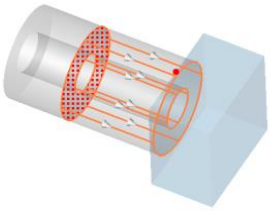
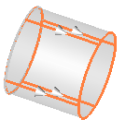
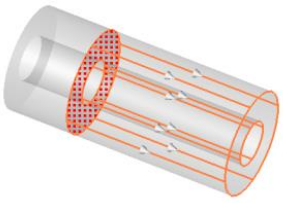
Coil Segments

A coil segment is a stranded conductor with a homogeneous current distribution in its cross section. This source type feeds the calculation domain with a given current.



Please note that coil segment sources are available for the tetrahedral-based solvers only – if a hexahedral solver is used, a thin path is created at the position of the path curve instead.

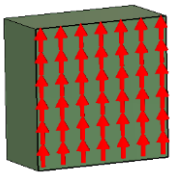
There are various ways to define a coil segment. To define the profile of the source, one can either pick a planar face before activating this mode or select a planar profile curve in the main plot window. If the tool is activated with a picked planar face, the interactive mode will start with the definition of the path or extrusion. The second step of the construction is to select a path curve. Alternatively, a numerical value could be used for the extrusion of the profile. To skip this step one can press *ESC*. If the profile is to be extruded to a picked point, it is necessary to pick this point before activating the construction mode. After the path selection was completed (either by selection or pressing *ESC*) a dialog box opens where all other settings can be defined. In total, there are six different ways to define a coil segment, which are summarized in the table below.

	Profile: **selected curve**	Profile: **picked face** (needs to be picked beforehand)
Path: **selected curve**	1. **Activate the creation tool** 2. Select the closed profile curve 3. Select the open path curve  Coil segment created from a profile curve (red) and a path curve (blue) 	1. Pick a planar profile face 2. **Activate the creation tool** 3. Select the closed profile curve  Coil segment created from a picked face (red dots) and a path curve (blue)
Path: **extruded to picked point** (needs to be picked beforehand)	1. Pick a point 2. **Activate the creation tool** 3. Select the closed profile curve 4. Press *ESC* to open the dialog box  Coil segment created from a profile curve and a picked point	1. or 2. Pick a planar profile face 1. or 2. Pick a point 3. **Activate the creation tool** 4. Press *ESC* to open the dialog box  Coil segment created from a picked face (red dots) and a picked point (red)
Path: **extrude with given numerical height value**	1. **Activate the creation tool** 2. Select the closed profile curve 3. Press *ESC* to open the dialog box  Coil segment created from a selected profile curve and a numerical value for the extrusion height	1. Pick a planar profile face 2. **Activate the creation tool** 3. Press *ESC* to open the dialog box  Coil segment created from a picked face (red dots) and a numerical value for the extrusion height

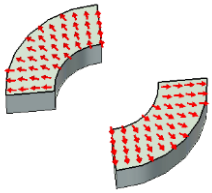
Permanent Magnets

To define a permanent magnet, you must activate the permanent magnet tool by selecting *Simulation: Sources and Loads* ⇨ *Permanent Magnet* . You will be prompted to select a face of a solid in order to select the magnet's geometry. Pick any solid with "Normal" material properties, possibly associated with a nonlinear, temperature dependent, hard magnetic J-H curve.

You can define constant, radial or azimuthal magnetizations. For details refer to the online help.



Constant magnetization

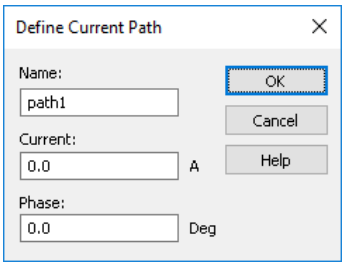


Radial magnetization

Azimuthal magnetization

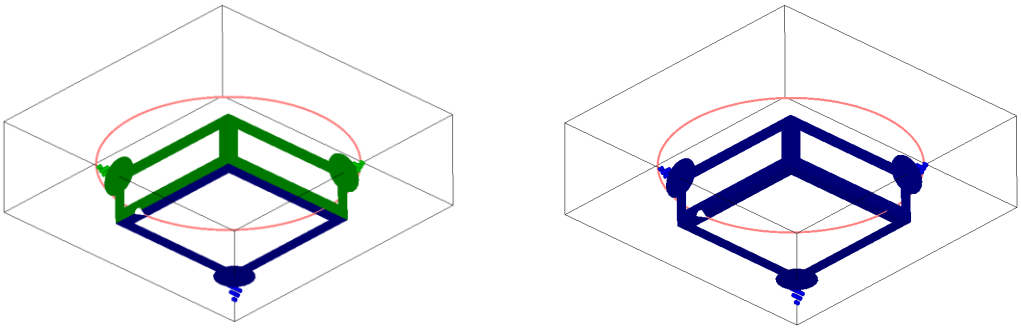
Current Paths

The definition of a current path is very similar to a coil definition. A single curve must be defined before the current path tool can be activated by selecting *Simulation: Sources and Loads* ⇨ *Current Path* . You will be prompted to select a curve. Then a dialog box arises in order to define the total current through the loop:



The phase value is only relevant for the LF Frequency Domain solver.

It is important that the current path is closed or that it terminates on a union of perfect electric conductors (PEC) and electric boundary conditions or conductive domains (generating a stationary current field) such that this union forms a closed loop with the current path. Otherwise the problem is not solvable since such a source violates the continuity equation in a magnetostatic context.

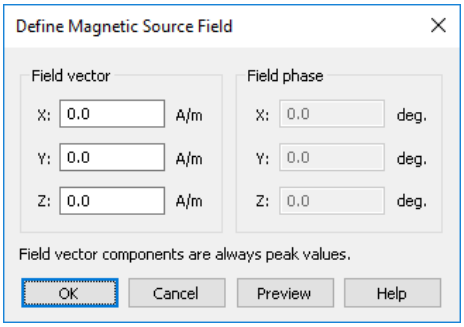


Left: A circular current path leaves the calculation domain through two electric boundaries – a solvable situation. Due to symmetries, only 1/4 of the structure has to be calculated.

Right: A circular current path leaves the calculation domain through two magnetic boundaries – not a solvable situation in magnetostatics.

Homogeneous Magnetic Field

To simulate structures in a homogeneous magnetic field, it is possible to define such a source by selecting *Simulation: Sources and Loads* ⇨ *Magnetic Source Field* . The following dialog box allows you to define the magnetic field vector:



Boundaries along the direction of the source field (i.e. boundary faces for which the source field has non-zero flux) have to be set to type “magnetic”. Moreover, to set a valid problem using the tetrahedral solver, one of the remaining faces may also be set to type “magnetic”.

The *Field phase* value is relevant only for LF Frequency Domain simulations.

Electrostatic Solver

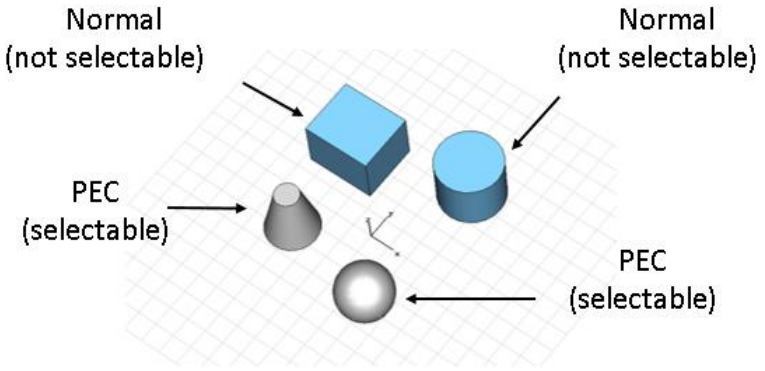
The electrostatic solver can be used for the simulation of static electric problems. Available sources comprise fixed and floating potentials, boundary potentials, charges on PEC solids and homogeneous volume and surface charges. The main task for the solver is to calculate the potential, the electric field strength and the electric flux density. These results appear automatically in the navigation tree after the solver run.

Open Boundaries

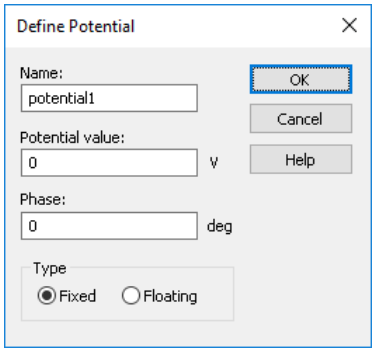
The electrostatic solver features open boundary conditions. These help to reduce the number of mesh nodes when problems in free space are simulated.

Potential Sources

The most important electrostatic source type is a potential definition. To define a potential on a perfect electric conductor (the solid has to be assigned to PEC material) you must activate the potential tool first via *Simulation: Sources and Loads* ⇨ *Electric Potential* . The first step is to select the surface of a perfect electric conductor carrying the new potential:



After a PEC surface has been selected, the potential dialog appears to assign a *Name*, a *Potential value* and a *Type* for the new source:

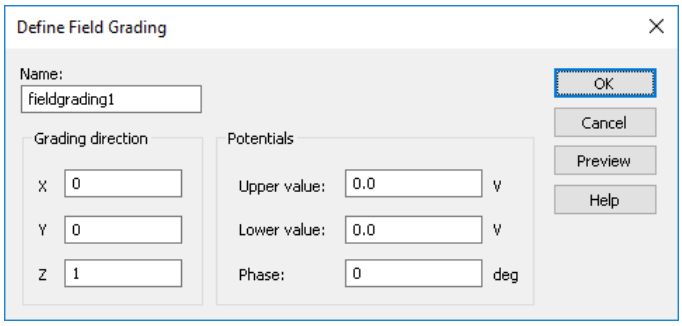


The *Phase* value is relevant only for LF Frequency Domain simulations and thus will be ignored by the described solver.

Note that for a potential of *Type* "Floating", the value itself is not prescribed, but the resulting constant potential at the solid will obtain a value such that the resulting total charge of the conductor is zero. Consequently, defining a floating potential is equivalent to assigning a zero charge. The charge definition will be discussed later.

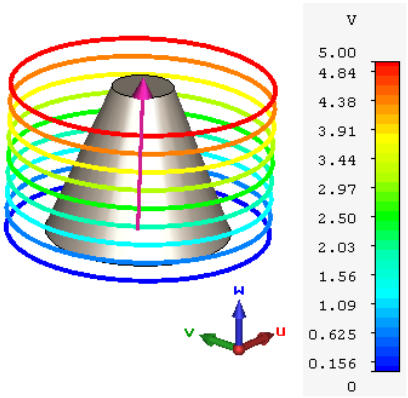
Field Grading

Capacitive field grading is an electrostatic source characterized by a linear distribution of potential on the PEC solid surface. This source can be created by selecting *Simulation: Sources and Loads* ⇨ *Electric Potential* ⇨ *Field Grading* . Afterwards, a surface of a PEC solid can be picked, on which the field grading source is to be created. Then the field grading definition dialog box appears, where all the settings for the source can be defined.



The *Grading direction* is the vector along which the potential value must change linearly. In any plane perpendicular to this vector the potential value on the surface of the PEC object is constant. Upper and lower potential values define the range within which the electrical potential is changing on the surface of the PEC solid.

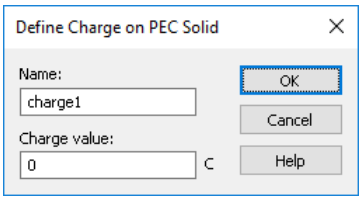
After the necessary values are set, press the OK button. A new field grading source is created.



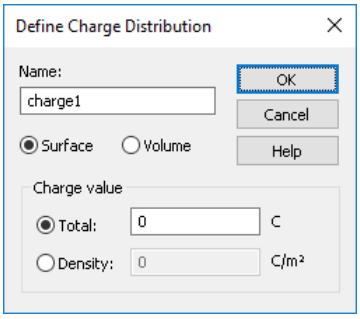
Charge Sources

Two different charge types exist in CST EM Studio: total charges on perfect conductors (resulting generally in a non-uniform surface-charge distribution along the PEC surfaces) and uniform charge distributions on normal material solids.

For the charge definition based on PEC, the first step is very similar to the one carried out with the potential definition. After activating the charge tool via *Simulation: Sources and Loads* ⇨ *Electric Charge on PEC*, you can pick a surface to which the charge will be applied. Then the charge dialog appears to determine the name and the charge value:



For the definition of a uniform charge-distribution definition, the first step is similar again - the only difference is that the source must be assigned to a normal material solid. You cannot define an uniform charge distribution on a PEC material. Use *Simulation: Sources and Loads* ⇨ *Electric Charge Distribution* and select a normal material solid. Then the following dialog will appear:



Here you can specify a name, a type and a value for the charge distribution. You can define a volume as well as a surface charge distribution. Remember that the latter will generate a jump in the normal component of the electric flux density. Furthermore, you can define the total charge or the charge density value.

Boundary Potentials

Finally, you can also assign an electrostatic potential to an electric boundary condition from within the boundary dialog. Open the boundary dialog box via *Simulation: Settings* ⇨ *Boundaries* and select the *Boundary Potentials* tab: In order to specify a boundary potential, select the "Floating" type from the drop-down list or select the "Fixed" type and enter a value in the edit field.

A boundary potential can be defined on normal or electric boundary conditions only. Boundaries with different potential values must not be adjacent. Again, you can define a fixed or floating potential.

Stationary Current Solver

The stationary current solver can be used to simulate DC current distributions. Available sources are potentials, boundary potentials, current paths, current ports and coil segments. Additionally, to the modeled structure with defined material properties, lumped network elements, i.e. resistors, may be added into the computational domain. The main task for the solver is to calculate the electric field strength, current density and ohmic losses. These results appear automatically in the navigation tree after the solver run.

Since the process of defining potential and current path sources is discussed in the two previous sections, we will focus on the definition of current ports, contact properties and coils segments created from solids. For a more detailed description of the lumped network element, we refer to the subsection *Lumped Network Elements* in the section *LF Frequency Domain Solver*.

Parameterized Electrical Conductivity

The stationary current solver supports not only fixed electrical conductivity values (isotropic or anisotropic) but also temperature-dependent and nonlinear characteristics:

- Temperature-dependent electrical conductivity can be defined by setting the material *Type* in the *General* tab of the *Material Parameters* dialog box to *Temp. dependent.* Then press the *Properties* button in this tab to open the *Temperature-Dependent Materials* dialog box, where you can define the temperature dependency slope of electrical conductivity. A temperature field must be imported from a thermal project via *Simulation: Sources and Loads* ⇨ *Field Import* .

- Nonlinear electrical conductivity is defined by creating an E(J) curve in the *Electrical Conductivity Properties* dialog box. This dialog is accessible via the *Conductivity* tab of the *Material Properties* dialog box. Here, in the group for *Electrical conductivity* check *Advanced* and press the button *Parameters*.

Before setting either parameterization, a default non-zero value of electrical conductivity must be set.

Current Ports

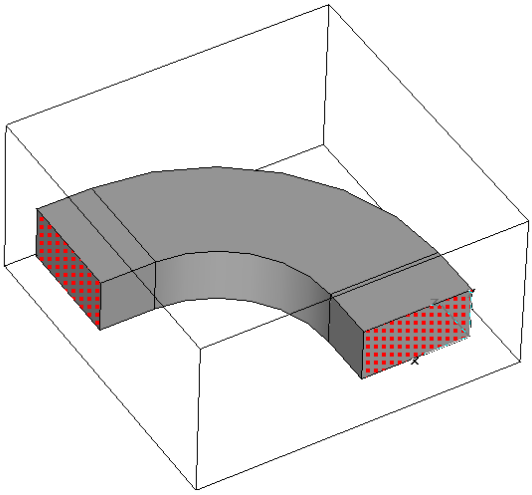
A current port is a face on a conductive material surface, characterized by its normal direction and the total electric current flowing through it. The usage of current ports is somewhat different depending on the mesh type utilized by the stationary current solvers:

- When using hexahedral meshes, the current port must be located on the computational domain's boundary.
- For tetrahedral meshes, this limitation does not apply. Besides, for such meshes the current port can be placed onto a surface between two conductive domains. In this case the solution guarantees the continuity of the normal component of current density on both sides of the current port surface.

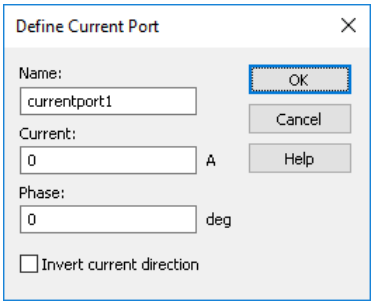
Note that if no sources with fixed potentials are defined, the sum of the prescribed currents entering and leaving the computational domain must be zero. Otherwise the problem does not have a stationary solution.

If the stationary current solution is intended to be used as a pre-computation step for a magnetostatic solution, all the current ports must be located either on the computation domain's boundary or between two conductive domains, in order to ensure the divergence-free current density distribution.

The following picture shows a simple conductive bend inside the computational domain. The two conducting faces are highlighted.

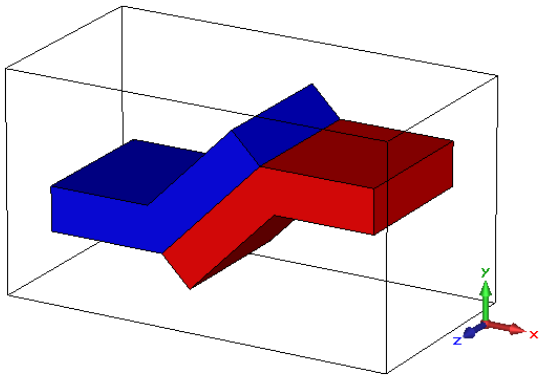


In order to define a current port on one of these faces, select the current port tool via *Simulation: Sources and Loads* ⇨ *Current Port* . Next pick an appropriate face on a conductive material. A dialog box opens where you can define the port's name and the magnitude of the current:



Contact Properties

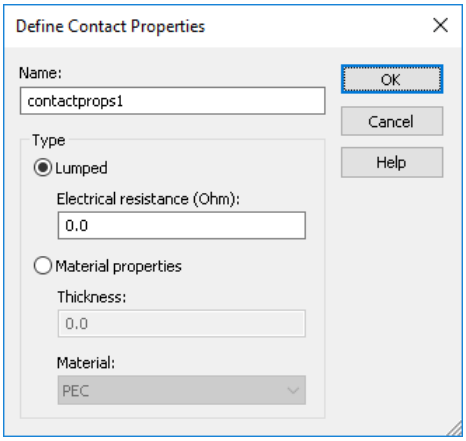
A contact resistance is defined via *Simulation: Sources and Loads* ⇨ *Contact Properties* . It is equivalent to a thin layer of conductive material at the interface between two (or several) solids. The definition is performed by selecting the solids associated with the "first" and then with the "second" side of the contact surface.



A contact resistance can be characterized either by a lumped parameter (integral electrical resistance in Ohm) or by its thickness and conductivity of the material assigned. Both definitions are equivalent and can be converted into each other:

$$R = \frac{1}{\sigma}\frac{l}{A}$$

Here R is the lumped parameter representing integral resistance. In the material-based representation, electrical conductivity σ and layer thickness l are used. Contact area A is calculated by the solver.



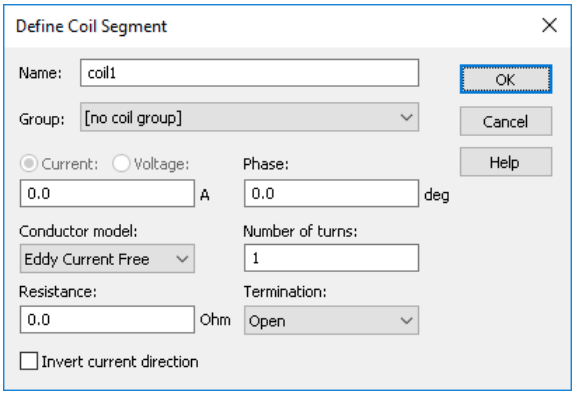
The advantage of contact resistance definition through integral resistance is that it is independent on the contact area A which may vary in case of intersections of associated solids or depending on the mesher settings. On the other hand, the material-based definition offers much more flexibility, for example, it supports nonlinear or temperature-dependent electrical conductivity via the material definition.

Electrical losses which take place within the contact region are calculated and saved by the stationary current solver as surface losses, so they can be utilized afterwards for a thermal analysis.

Contact resistances are only supported by the tetrahedral-based stationary current solver.

Coil Segments from Solids

Another way to create a coil segment is to create it from a previously defined solid. After creation of a solid, a definition of a coil can be initiated via *Simulation: Sources and Loads* ⇨ *Coil* ⇨ *Coil Segment from Solid* . You will be asked to specify a planar current entry and exit face. Finally, the coil segment characteristics are defined in the dialog box which opens as soon as both required current faces are specified:



Depending on the chosen conductor type (*solid* or *eddy current free*), a coil segment can be characterized either by a lumped value for the electrical resistance (in Ohm) or an electrical conductivity (in S/m). Within the stationary current solver, both definitions are equivalent and can be converted into each other. However, this statement does not hold within the quasistatic

regimes of the frequency domain and time domain solvers. We will elaborate on this in the following two subsections.

The advantage of coil segment definition through a resistance is that this value can be chosen independently from the coil segment geometry, which may vary in case of intersections of associated solids or depending on mesh settings. On the other hand, a coil of conductor model *solid* with associated conductivity allows for skin-effect and eddy current analysis (in magnetoquasistatic simulation simulations).

LF Frequency Domain Solver

The LF Frequency Domain solver can be used to solve electromagnetic field problems with time-harmonic sources and linear materials. In this case, all quantities are time-harmonic and it is possible to solve a complex-valued problem in the frequency domain.

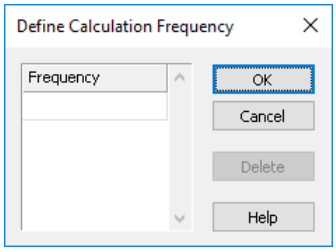
The main task for the solver is to calculate electromagnetic fields and the resulting currents, losses, and energies. These results appear automatically in the navigation tree after the solver run has been finished.

The LF Frequency Domain solver includes the following simulators:

- Full Wave simulator
- Magnetoquasistatic simulator
- Electroquasistatic simulator

The Full Wave simulator solves the full Maxwell's equations. The magnetoquasistatic and electroquasistatic simulators can solve low frequency problems with dominating magnetic (e.g. eddy current problems) or electric energy, respectively. A typical application is the computation of AC current and loss distributions.

In contrast to the static solvers, one or more calculation frequencies must be defined before the LF frequency domain solver can start. In order to do that, open the frequency dialog box *Simulation: Settings* ⇨ *Frequency* for the modelled task:



To add a new frequency to the list, double-click on the empty edit field, enter the value and confirm with the *Enter* key. The list becomes operative when you leave the dialog box by clicking *OK*.

Full Wave and Magnetoquasistatic Simulator

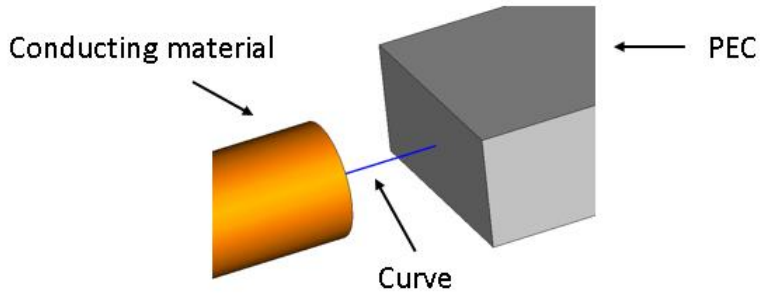
Available sources are current and voltage paths, current ports, coils and coil segments including those created from solids. Coils and coil segments can be collected into the groups.

Coil and current path definitions are discussed in the magnetostatic solver section. Current ports have been introduced in the stationary current solver section. One minor difference exists: in addition to the current (or voltage) value, it is possible to assign a phase value to a current path or a coil (for magnetostatic calculations, this setting is ignored). Coil segments created from solids have been also presented in the stationary current solver section. Within the magnetoquasistatic simulations however, the two conductor models exhibit significantly different behavior: solid coil segments are massive conductors carrying eddy currents and stipulating losses, whereas eddy

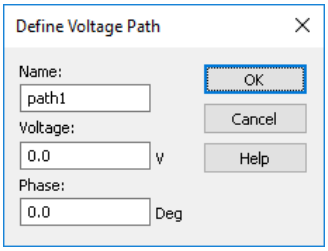
current free coil segment sources are not affected by eddy current effects. Presently, coil segments created from solids are supported for broadband calculations within the frequency domain solver.

Voltage Paths

Voltage paths are similar to the previously described current paths. They are created from a curve path. A typical application is a voltage path connecting two conducting regions, defining a voltage between the conductors:



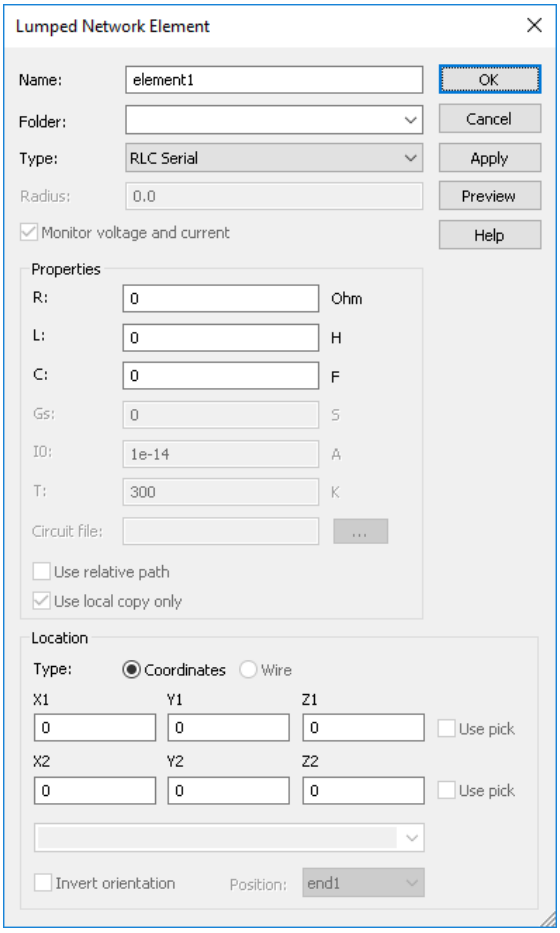
To define a voltage source, activate the appropriate tool via *Simulation: Sources and Loads* ⇨ *Path Sources* ⇨ *Voltage Path from Curve*. The curve selection modus enables the selection of the curve that is to be transformed into a voltage path. After the appropriate curve has been selected, the voltage path dialog box appears. Here you can determine the element's name, its voltage and phase values.



After the definition is complete, the voltage source is listed in the navigation tree folder *Voltage Paths.*

Lumped Network Elements

The full wave and the magnetoquasistatic formulations of the LF Frequency Domain solver account for the inclusion of the lumped network elements in the simulation domain. In this context, one can make use of any parallel or serial circuits consisting of one resistor, one capacitor and one inductor. To add a new network, open the lumped element dialog box, *Simulation: Sources and Loads* ⇨ *Lumped Element*:



In the lumped network element dialog box, the element values as well as the connection type for a lumped element – serial or parallel – are defined. Furthermore, the geometrical location of the lumped element is set in the dialog box, i.e. the starting point and ending point of the network in the computational domain.

Nonlinear equivalent permeability

The magnentoquasistatic frequency domain solver supports nonlinear material properties (B(H)) via a linear equivalent permeability computation. Note that this is an approximation; for fully nonlinear time-dependent calculations the LF time domain solver should be employed. Additionally, the LF frequency domain magnetoquasistatic solver supports time dependent nonlinear (B(H)) and linear material properties with coupling to CST MPhysics Studio. More information on these topics can be found in the online help.

Broadband simulation regime

For the magnentoquasistatic simulator, a broadband calculation regime is also available, which allows the broadband calculation of impedance matrices as well source und lumped parameters. Additionally, instead of a standard full 3D frequency sweep where the solution of a large linear system is required for each calculation frequency, a fast frequency sweep is available where only a much smaller system has to be solved for each frequency value. Thus, this calculation mode should be the method of choice if solutions for multiple frequencies and/or broadband 1D results are required.

Furthermore, the broadband formulation allows for a deduction of a macro-model representation of a field model, which finally results in the authoring of a reduced order models as a Functional Mockup Units according to the FMI standard. The created .fmu archive is issued automatically as soon as a broadband simulation is chosen and can be imported into any simulation tool capable to interpret the FMI standard for Model Exchange. The System Simulator, which is a component of the CST Studio Suite®, supports importing Functional Mockup Units and can be applied for the simulation of macro-models on a system level.

More information on these topics can be found in the online help and in the help system of System Simulator.

Electroquasistatic Simulator

In the electroquasistatic approximation of the full Maxwell's equations, the time derivative of the magnetic field is ignored in the Faraday-law. Hence, the computed electric field is curl-free in the whole space. Consequently, electroquasistatic problems can be described by a complex scalar potential, which reduces the number of unknowns in the equation system to be solved.

Thus, running the electroquasistatic simulator is usually much faster and more robust than running the full wave simulator on the same mesh. Whenever the time derivative of the magnetic field is negligible in Faraday's law, you should use the electroquasistatic solver to solve your low frequency problem. Typical applications are insulator problems, where the conductivities and magnetic field energies are very low.

Potentials are available as excitation sources. These are already discussed in the electrostatic solver section. Again, a minor difference exists: In addition to the potential value, it is possible to assign a phase value (for electrostatic calculations this setting is ignored). Please refer to the online help for further details.

LF Time Domain Solver

The LF Time Domain solver can be used to solve electromagnetic field problems with the time-dependent sources driven at low frequencies. This solver includes the following simulators:

- Magnetoquasistatic simulator
- Electroquasistatic simulator

The solver features both a constant and an adaptive *implicit* time-stepping algorithm. The adaptive time-stepping scheme requires solving four linear or nonlinear systems of equations in each time step.
Furthermore, if the solution of the investigated problem is known to be periodic in time, the LF time domain solver provides a dedicated steady state time-stepping algorithm which may accelerate the calculation of the steady state solution. The online help provides further information on the steady state solver.

Magnetoquasistatic Simulator

In the magnetoquasistatic approximation of the Maxwell's equations, the time derivative of the displacement current can be omitted with respect to the conduction currents. Typical use cases are the nonlinear eddy current problems or transient simulations (e.g. switching devices, actuators, sensors).

Within the simulator, supported excitation sources are permanent magnets, current- and voltage-driven coils and wires, coil segments including those created from solids, coil groups, current ports, transient external magnetic source fields and rigid body motions.

The main task for the simulator is to calculate the time evolution of the magnetic and current fields as well as the resulting losses, energies and other derived quantities like e.g., forces.

Electroquasistatic Simulator

The electroquasistatic approximation of the Maxwell's equations is employed when the influence of the magnetic induction can be neglected. Thus, a description of an electroquasistatic field is completed by a scalar potential function which reduces the number of unknowns in the equation system to be solved. Typical use case includes, e.g., a high-voltage bushing.

Electrical potentials are available as excitation sources. These are already discussed in the electrostatic solver section. Please note here again: The *Phase* value, which can be defined

together with the potential value, is relevant only for LF Frequency Domain simulations and thus will be ignored by the described simulator.

Workflow

The workflow for a time domain simulation is very similar to the workflow of static and time harmonic simulations. However, some additional steps must be performed before the solver is started:

1. One or more excitation signals must be defined.
2. Excitation signals must be associated with the sources.
3. Monitors must be defined.
4. A simulation duration must be set.

These differences result from the fact that additional information is necessary about the time evolution of the excitations and the size of the time interval of interest. Furthermore, storing the whole evolution of all computationally available results needs a lot of disk space. For this reason, the concept of time monitors is introduced, which allows a more specific definition of the results of interest.

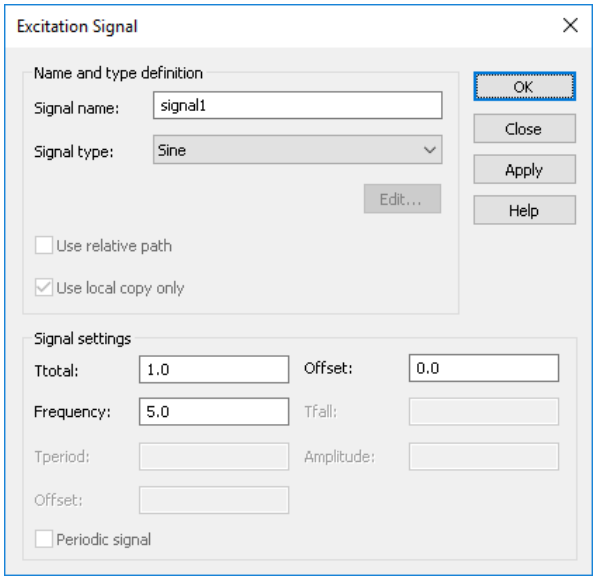
Note: The excitation definition as well as the usage of monitors in CST EM Studio is very similar to those available in CST Microwave Studio.

The following subsections will describe these additional steps in short. For more detailed information, please refer to the online help.

Signal Definition

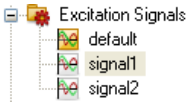
In a new project, only a constant "default" signal is defined. For a meaningful simulation with the LF Time Domain Solver, at least one non-constant signal should be defined.

A new signal can be defined via *Simulation: Sources and Loads ⇨ Signals ⇨ New Excitation Signal* . A dialog box opens where a signal type, its parameters and a name can be set:

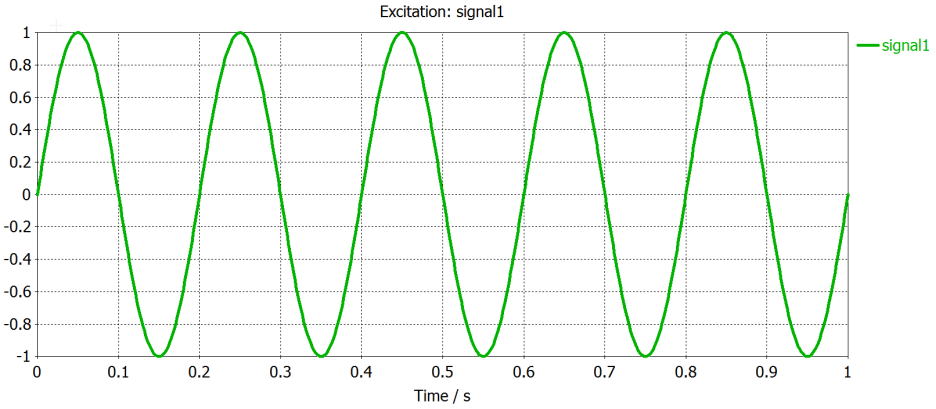


The parameters of the signal depend on the individual signal type and are described in the online help. The parameter *Ttotal* must be set for almost all signal types and defines the size of the definition interval. For time values larger than *Ttotal,* the signal is, in general, continued by a constant value. It is also possible to import a signal or to create a user defined signal or to select a pre-defined signal from the signal database.

All defined signals are visible in the *Excitation Signal* folder in the navigation tree.



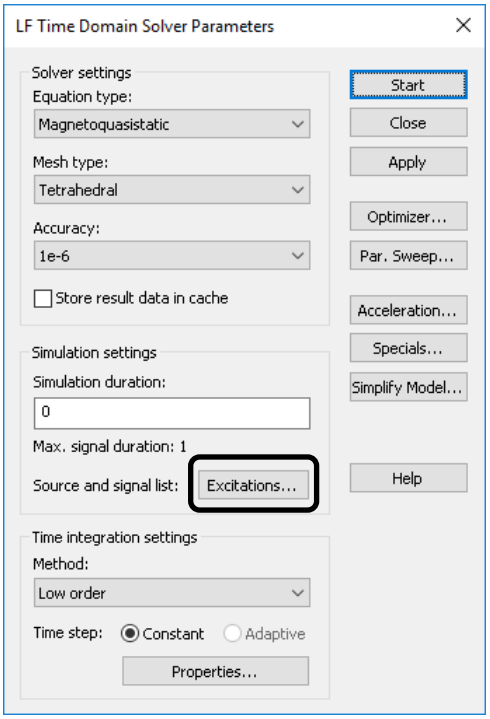
A signal can be displayed by selecting it in the navigation tree:



Excitations: Assigning Signals to Sources

As for the static solvers, the source value defines the strength of a source field. The time evolution of a source is defined by assigning a signal to it.

This can be done by opening the solver dialog box via *Home: Simulation* ⇨ *Setup Solver* and pressing the *Excitations…* button:



A sub-dialog opens showing each defined source that can be interpreted by the solver. Also the source values are displayed. Each source can be switched on or off for the simulation. By default, all sources are switched on.

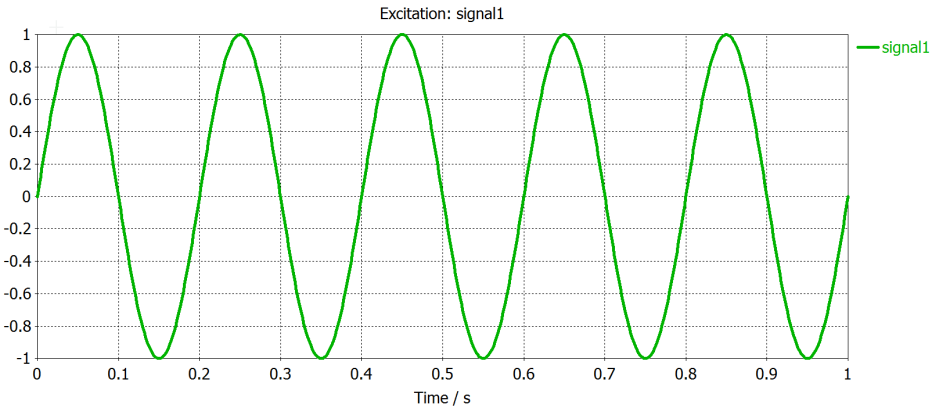
For each source, a signal can be assigned via a drop-down list. The same signal can be assigned to several sources. Optionally, an individual time delay Δt can be defined for each source.

The resulting time dependent excitation f is the product of the source value v (e.g. the coil current) and the (possibly shifted) assigned signal s :

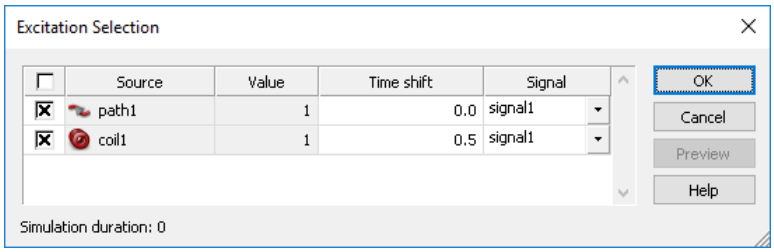
$$f(t) = s(t - \Delta t) \cdot v \ .$$

Example

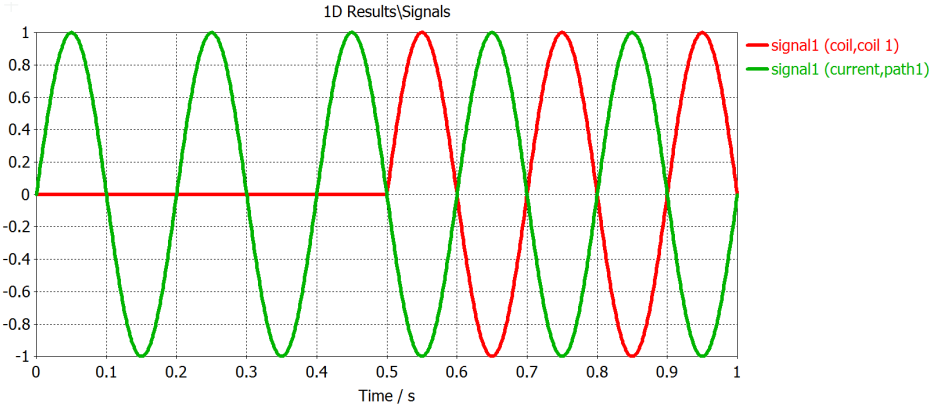
Two sources are defined, one current path with source current 1 A and one coil, also carrying 1 A in each turn. A previously defined signal "signal1" (see image below) is assigned to both sources.



The signal of the coil is shifted by 0.5 s by clicking in the field in column *Time shift*. With these settings, the *Excitation Selection* dialog will look like this:



For this example, the resulting excitations used by the solver look like this:



Reference Signal

There is always one signal tagged as the 'reference signal'. This signal is highlighted in the navigation tree by a yellow background. The reference signal can be changed by marking another

signal in the navigation tree and selecting *Simulation: Sources and Loads ⇨ Signal ⇨ Use as Reference.*

By default, all sources are set to use the currently defined reference signal. Hence, it is not necessary to visit the *Excitations* sub-dialog of the solver dialog if only one source or only one signal shall be used for the simulation. Then, it is sufficient to select the desired signal being the reference signal and by default all sources are automatically assigned to this signal.

Rigid Body Motion Definition

The 2D and 3D magnetoquasistatic time domain solver allows for the definition of periodic rotational or translational rigid body motions, which can be used for the simulation of electrical machines and actuators. The movement is described by the mechanical motion definition and the motion *Gap*. The mechanical motion definition defines the absolute movement in time of the moving objects and the moving direction for translations or rotation axis and its center for rotations. The absolute movement in time can be defined by a constant motion, by a time signal or by an equation of motion. The motion *Gap* is a closed surface that surrounds the moving objects and is located in the air gap between moving and static objects. Multiple motions including nested gaps can be defined if the gap surfaces do not intersect and the following limitations do apply.

Limitations for nested gaps definition

If nested gaps are defined, the absolute value of the speed defined in the dialog applies for the gap parts which are not part of any other gap nested inside the gap. It is possible to have one of the following combinations of nested gaps:

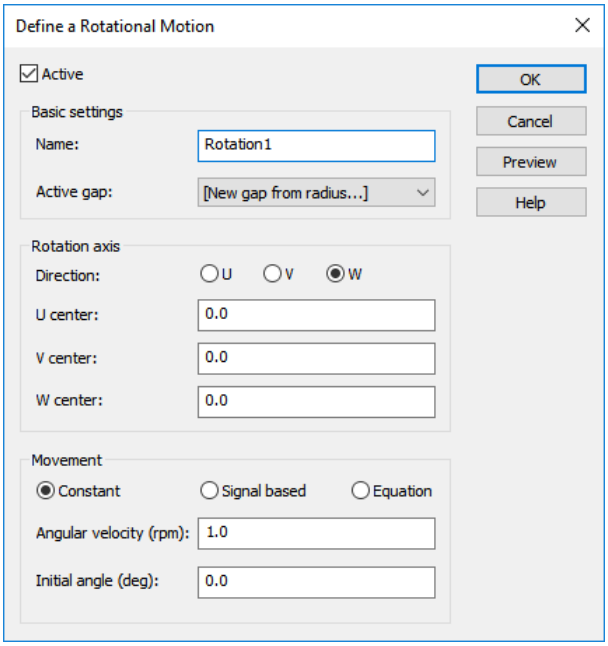
- Rotation gaps inside rotation gaps with possibly nested gaps
- Translation gaps inside translation gaps with possibly nested gaps
- Rotation gaps inside translation gaps with possibly nested rotation gaps

It is not possible to define the following (intersecting) combination of gaps:

- Translation gap inside a rotation gap, because the translation gap is required to touch the model boundaries

Rotation definition

A new rotational motion is defined by opening the *Define a Rotational Motion* dialog via *Simulation: Motion ⇨ Motion*  *⇨ New Rotation*  :

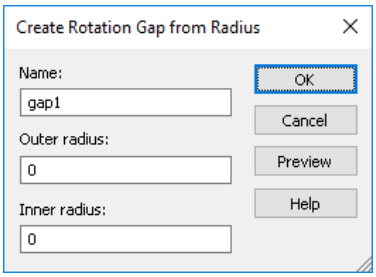


The rotation axis is defined in the active working coordinate system and must be aligned with one of the global axes, and in case of 2D simulations, with the normal of the 2D planar mesh. The center of the rotation axis is defined by the coordinates *U center*, *V center* and *W center*.
The movement can be specified as one of the following:

- *Constant* defined by the *Angular velocity* (revolutions per minute, rpm) and the *Initial angle* (degrees)
- *Signal based,* which allows the selection of a previously defined excitation signal (with the *y*-component in radians and the time axis in user units)
- *Equation of motion* defined by the solid parameters *Moment of inertia* (kg·m^2), *Damping constant* (kg·m^2/(s·rad)), *Torsion spring* (N·m/rad), *External torque* (N·m), *Initial position* (degree) and *Initial speed* (rpm). At least the *Moment of inertia* must be non-zero in order to allow the calculation of motion.

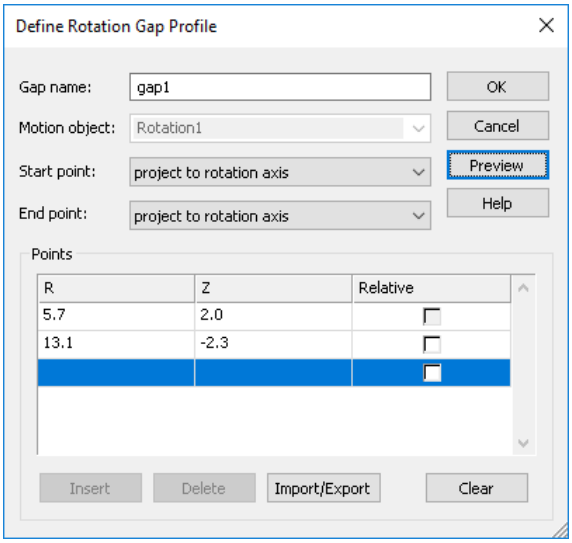
The rotational motion gap is defined by clicking on one of the options on the *Active gap* drop-down menu ([*New gap from polygon]* or [*New gap from radius*]). If you already have closed the dialog box, you can define a new gap by selecting the corresponding entry from the context menu, when selecting the newly defined rotation item in the navigation tree.

While the *Radius* gap definition mode is active, the specified rotation axis is shown and you can define an outer and an inner radius in the plane normal to the rotation axis. All values can be reviewed and edited in the *Create Rotation Gap from Radius* dialog box after closing the gap definition mode.



While the polygon gap definition mode is active, a working coordinate system is shown with the *Z* axis pointing in the direction of the specified rotation axis. Now you can define a (closed) polygon in the plane normal to *Z,* which then will be rotated around the *Z* axis. The polygon is automatically closed if you select the option *project to rotation axis for* the *Start / End point*. The

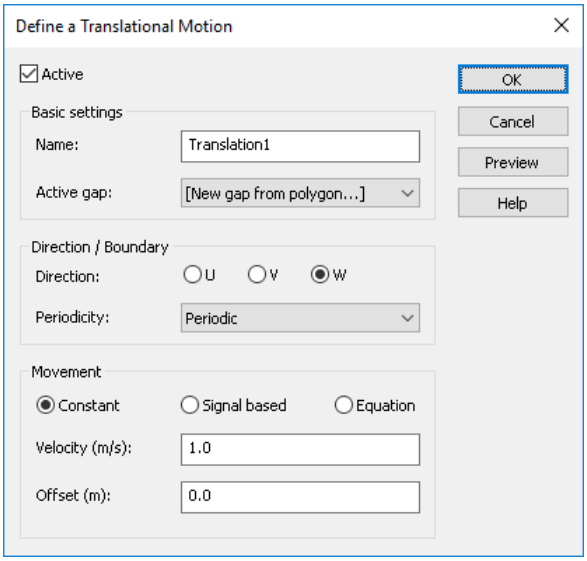
coordinates of the polygon points can be reviewed and edited in the *Points* table of the *Define Rotation Gap Profile* dialog box:



It is possible to create more than one gap for a motion, but only one of them can be active before starting the simulation. To activate a gap please use the context menu option *Select as Active Gap* in *NT* ⇨ *Motion* ⇨ *Motion name* ⇨ *Gap name*.

Translation definition

A new translational motion is defined by opening the *Define a Translational Motion* via *Simulation: Motion* ⇨ *Motion* ⇨ *New Translation*.



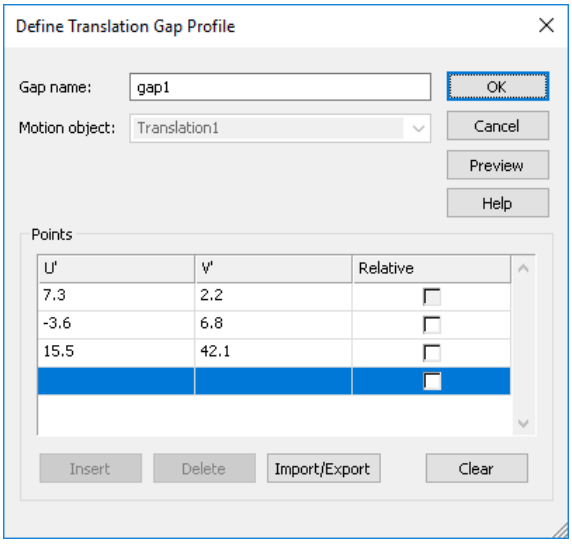
The translation *Direction* is defined as one of the axes (*U*, *V* or *W*) in the active working coordinate system with the translation direction being aligned with one of the axes of the global coordinate system. Furthermore, the translation direction should be found in the planar mesh plane for a translational planar mesh and parallel to the axis for a rotational planar mesh. The *Periodicity* of the boundaries normal to the translation direction can be *Periodic* or *Antiperiodic*.
The *Movement* can be defined as one of the following:

- *Constant* defined by the *Velocity* (m/s) and the *Offset* (m)
- *Signal based,* which allows the selection of a previously defined signal (with the *y*-component in m and the time axis in user units)

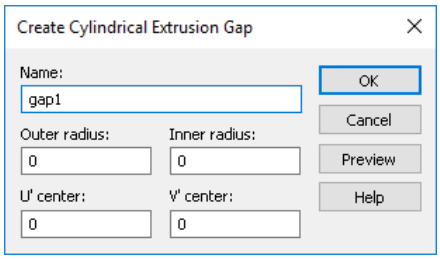
- *Equation of motion* defined by the solid parameters *Mass* (kg), *Damping constant* (N·s/m), *Spring constant* (N/m), *External force* (N), *Initial position* (m) and *Initial velocity* (m/s). At least the *Mass* must be non-zero in order to allow the calculation of motion.

The translational motion gap is defined by clicking on one of the options on the *Active gap* drop-down menu ([*New gap from polygon]* or [*New gap from circle*]). If you already have closed the define motion dialog boxes without defining a gap, you can define a new gap by selecting the corresponding entry from the context menu *NT ⇨ Motion ⇨ Motion name ⇨ Gap name*.

The *Polygon* translational gap tool allows the definition of a closed polygon in a plane normal to the movement direction. The translation gap is defined by extrusion of the polygon between the two outer boundaries of the model. While the gap definition mode is active, a working coordinate system is shown with the *W'* axis aligned with the translation direction. Now you can define a (closed) polygon in the *U'-V'* plane. The coordinates of the polygon points can be reviewed and edited in the *Points* table of the *Create Translation Gap Profile* dialog box.



The *Circle* translational gap tool allows definition of the circular profile gap. This is useful if a rotational planar calculation is done, which allows motion only in the rotational axis direction. While the gap definition mode is active, a helper coordinate system is shown with the *W'* axis pointing toward the translation direction. Now you can define a center point and an inner and outer radius of the cylindrical extrusion gap in the *U'-V'* plane. All values can be reviewed and edited in the *Create Cylindrical Extrusion Gap* dialog box after closing the gap definition mode.



It is possible to create more than one gap for a motion but only one of them can be active before starting the simulation. To activate a gap, please use the context menu option Select as *Activate Gap* in *NT ⇨ Motion ⇨ Motion name ⇨ Gap name*.

Monitor Definition

In contrast to the static and time-harmonic solvers, no results will appear automatically in the navigation tree. It is not possible to store the fields and secondary results at every computed time step as this would require a tremendous amount of disk and memory space. You should,

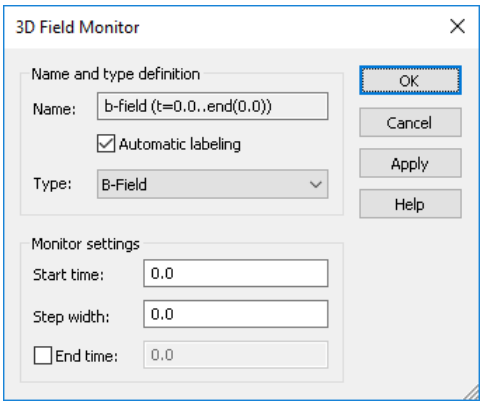
therefore, define certain results and time intervals at which the solver will record the desired data. These definitions are called *Monitors.*

Several different kinds of monitors are available in CST EM Studio: *3D Field Monitors*, *Monitors at Points, Monitors on Edges or Curves, Monitors on Faces* and *Monitors on Solids or Volumes.* The 3D Field Monitors yield field plots, which can be animated over the simulated time. The other monitors are classified by the objects on which appropriate integral functionals are defined. They yield 1D curves of scalar values versus the simulated time.

All defined monitors are listed in appropriate subfolders of the *Monitors* folder in the navigation tree. Within this folder, you may select a particular monitor to reveal its parameters in the main view.

3D Field Monitors

Several kinds of monitors record 3D vector or scalar fields (e.g. B-field, H-field, E-field, conductive current density, etc.). A 3D Field Monitor can be defined via *Simulation: Monitors ⇨ Field Monitor* . A dialog box opens where the type of the field, the start time and the sample step width can be defined:

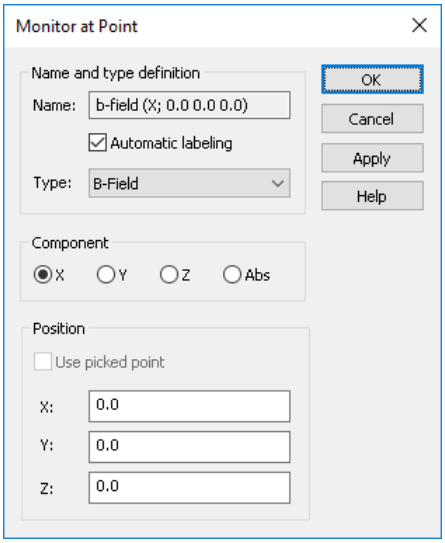


Available field types for the magnetoquasistatic simulator are *B-Field*, *H-Field*, *E-Field*, Cond. Current Densities, *Potential* (only for 2D models showing the magnetic vector potential), *Material* (showing the relative permeability), *Ohmic Losses*, *Averaged Ohmic Losses,* and *Magnetic Energy Density*.
Within the electroquasistatic simulator, the time evolution of the *E-field, D-field, Cond. and Displ. Current Densities* as well as *Potential* (showing the scalar electric potential) can be monitored.

After the solver run, the recorded result can be accessed via the *NT: 2D/3D Results* folder in the navigation tree. The scalar or vector field can be animated over the defined time period.

Monitors at Points

These kinds of monitors record scalar values that are defined at a point (previously picked or entered numerically), e.g. the x-component of the magnetic flux density at a fixed position. Such a monitor can be created via *Simulation: Monitors ⇨ Monitor on Entity ⇨ Monitor at Point* .



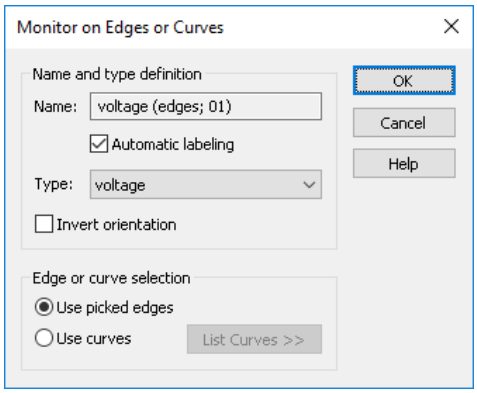
The magnetoquasistatic solver supports following monitor types: *B-Field*, *H-Field*, *E-Field*, *Cond. Current Density*, *Material*, *Potential* (magnetic vector potential, only available for 2D simulations), and *Ohmic Losses*.
For the electroquasistatic solver, available monitor types are *E-field*, *D-field*, *Cond*. and *Displ*. *Current Densities* and *Potential* (the scalar electric potential).

The monitor generates a 1D-plot over time during the solver run. The result plot can be accessed in the *NT: 1D Results ⇨ LT Solver* folder.

Please note that this kind of monitor is similar, although not identical, to *Probes* available within CST Microwave Studio.

Monitors on Edges or Curves

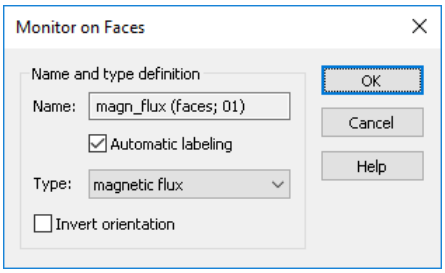
These kinds of monitors record scalar values that are defined for (previously picked via (*Simulation: Picks ⇨ Picks*) model edges or on curve items. Currently available are the *voltage* and the *source current* along a path. You can create it via *Simulation: Monitors ⇨ Monitor on Entity ⇨ Monitor on Edge*



Again, the monitor generates a 1D-plot over time during the solver run and the result plot can be accessed in the *NT: 1D Results ⇨ LT Solver* folder.

Monitors on Faces

These kinds of monitors record scalar values that are defined for (connected set of) model faces, which have to be picked (*Simulation: Picks ⇨ Picks*) before the monitor definition. For the 3D magnetoquasistatic simulations, magnetic flux and conduction current monitors are supported. You can create them via *Simulation: Monitors ⇨ Monitor on Entity ⇨ Monitor on Face*.

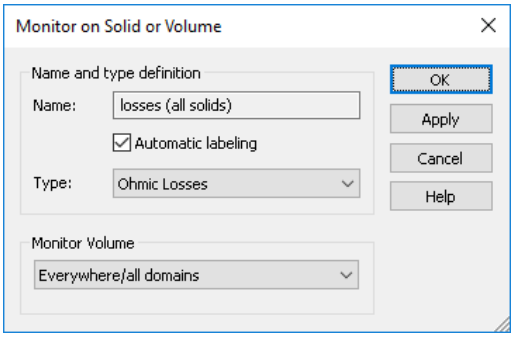


Again, the monitor generates a 1D-plot over time during the solver run and the result plot can be accessed in the *NT: 1D Results ⇨ LT Solver* folder.

For the electroquasistatic simulator, this monitor type is not supported.

Monitors on Solids or Volumes

Within the magnetoquasistatic simulator, these kinds of monitors record values that are defined for a solid, volume or a group of solids (the force on a solid etc.). You can create it via *Simulation: Monitors ⇨ Monitor on Entity ⇨ Monitors on Volume* .



Available monitor types for the magnetoquasistatic simulator are: *Ohmic Losses, Force and Torque, Iron Losses*, *Apparent* and *Incremental Inductance Matrices* as well as *Demagnetization*.

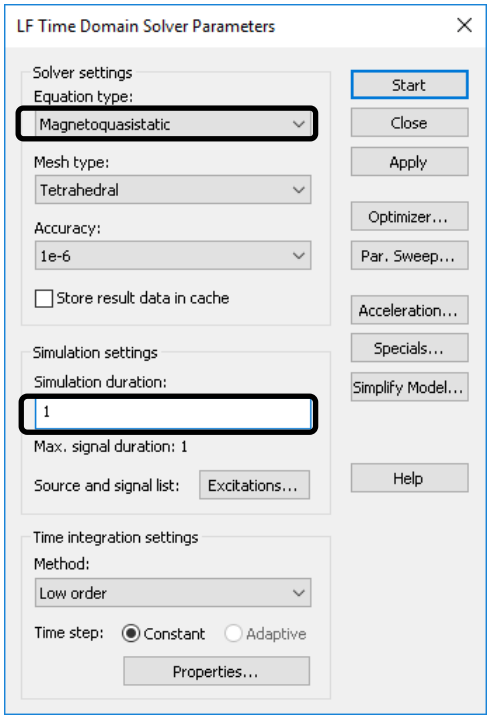
Again, the monitor generates a 1D-plot over the time during the solver run (or in case of *Force* monitors one 1D-plot per component) and the result plot can be accessed in the *NT: 1D Results ⇨ LT Solver* folder. The demagnetization monitor does not generate a 1D-plot but only gives a warning if the maximum demagnetizing field strength is higher than the set value and generates a 3D Field monitor similar plot with the distribution of the maximum demagnetization field strength in solids. The iron loss monitor also generates a plot with the distribution of the iron losses in solids along with the calculated loss in the *NT: 1D Results ⇨ LT Solver* folder.

The monitors can be defined everywhere, on a certain solid or on groups of solids. The groups of solids are defined in *NT: Groups* as Normal Groups and are populated with solids via Drag&Drop.

This monitor type is not supported for the electroquasistatic simulator.

Starting the Simulation

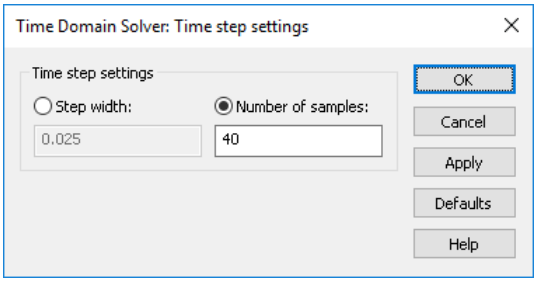
As already mentioned, the solver dialog box can be opened via *Home: Simulation ⇨ Setup Solver* . Firstly, define the *Equation type* you are going to employ. Secondly, before starting the simulation, the *Simulation duration* must be entered. This value defines the length of the simulated time interval in the currently active time unit. Note that every simulation starts at time zero.



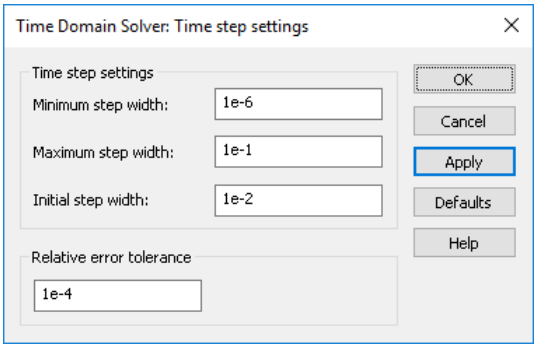
If at least one non-constant signal is in use, the maximum over all assigned time signal is displayed below the duration entry field (taking possible time shifts into account). This information gives some hint for a reasonable simulation duration and can be used for cross-checking, e.g. to ensure that signals and simulation duration are defined for a similar time period and scale.

Two different time-stepping strategies are available for the solver: Constant and adaptive time-stepping. By default, the constant time-stepping is enabled, which should always be used for simulations that contain rigid body motion. The adaptive time-stepping may be used for simulation without motion, especially for calculation with a fading transient component since the adaptive strategy may be more efficient in this case.

The default settings for the constant time-step algorithm are accessible in the *Time step settings* dialog after selecting *Properties* and are set to 40 steps for the simulation duration. This default setting should be changed to allow a sufficient discretization of the time axis considering the expected signals variation in the simulation time.



If adaptive time-stepping is preferred, you need to switch the *Method* to *High Order* and then select the *Adaptive time step* radio button. It is a good idea to have a look at the parameters of the adaptive time-stepping scheme before starting the simulation. The parameters can be displayed and modified in the *Time step settings* sub-dialog, which can be activated by pressing the *Properties* button:



The most important value is the *Relative error tolerance*. The smaller this value the more rigorous is the behavior of the adaptive scheme, leading to smaller time steps and smaller time-discretization errors. On the other hand, smaller time steps will increase the simulation time. Furthermore, you can define upper and lower bounds for the size of a time step and set the size of the initial time step. If you have some knowledge about typical time scales of your model, it might be meaningful to modify the default settings.

Note that for some problems it may be also necessary to increase the accuracy for the solution of the linear (or respectively nonlinear) systems of equations that are solved for each time step. This can be done by choosing the necessary Accuracy in the solver start up dialog box. However, in most cases, the default-settings can be left unchanged.

Finally, the LF Time Domain solver can be started by pressing the Start button and the results can be analyzed.

Coupled Simulations with CST MPhysics Studio

Ohmic and iron losses from the solvers of CST EM Studio can be used for thermal simulations in CST MPhysics Studio. Based on these results, temperature dependent material properties can be updated in the stationary current solver or the LF frequency domain solver. The Mechanical Solver allows performing a continuative stress simulation on the temperature distribution. Moreover, force density distributions from magnetostatic or electrostatic simulations can be fed into the mechanical solver of CST MPhysics Studio.

Please refer to the *CST MPhysics Studio Workflow* document for more detailed information about these multi-physics workflows.

Equivalent Circuit EMS/DS Co-Simulation

Equivalent circuit parameters describing the physical behavior of the field part of a CST EM Studio model can be used for co-simulations within CST Design Studio. The extraction of the lumped parameters from the field model is supported by the following tetrahedral mesh based solvers:

- Electrostatic Solver
- Magnetostatic Solver (linear and nonlinear problems)
- Stationary Current Solver

Please note: For nonlinear problems, the equivalent circuit parameters are calculated in the working point determined by the excitation sources defined within CST EM Studio. To cover the full parameter space of a nonlinear model, a parameter sweep can be used to retrieve the required data in a convenient way.

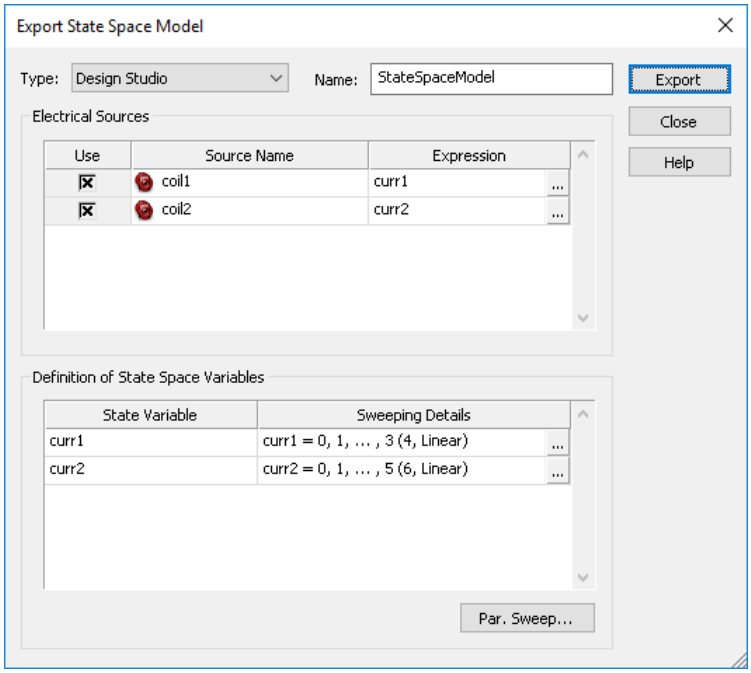
For further information, please refer to the examples within the *Equivalent Circuit EMS/DS Co-Simulation* section contained in the *CST EMS Examples* of the online help system.

State Space Model

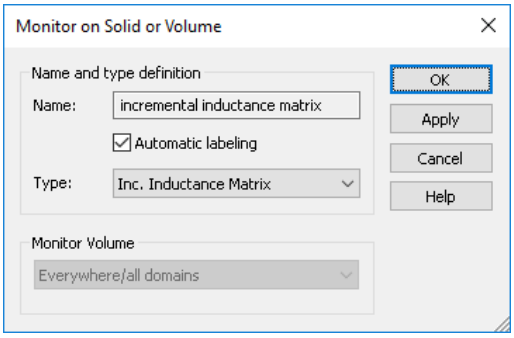
The 2D/3D magnetostatic and 2D LF Time Domain simulators offer a possibility to compress the equivalent circuit parameters describing the physical behavior of the field part of the model into

a so-called *state space model*. This feature is extremely useful for exporting accurate reduced order models of electrical machines and actuators to system simulators. For models composed of nonlinear materials, the lumped parameters are calculated in the working point determined by the excitation sources. If, in addition, a motion is defined for the field model part, the description of every working point also includes the mechanical position of a rigid body subjected to motion.

The extraction of a state space model is realized through the embedded mechanism called "Export State Space Model" available via *Simulation: Solver ⇨ State Space Model*:



In this dialog box, the user can define the name of the state-space model, the type (name) of the simulator for which the extracted model will be exported and also specify which sources have to be involved. Within the magnetostatic solver, the export to *CST Design Studio* and *Modelica*-based simulators is possible. For the transient case, the export is supported only to *Modelica*–based simulators.
Since the export is realized on the basis of the lumped parameters, a sufficient amount of data has to be prepared in advance. This ensures the availability of the required values to interpolate the state space model during a system simulator run. For this purpose, the *Parameter Sweep* option is available directly from the *State Space Model* dialog box. The lumped parameters employed in the presence of both electromagnetic and mechanical excitation sources are incremental inductances and the torque/force values. If no mechanical motion is defined for a system, only the incremental inductances are used for a coupling. For the magnetostatic solver, the calculation of the incremental inductance has to be activated within the *solver dialog Parameters* dialog box before the collection of data for the state-space model starts; for the time domain case, a definition of the *Incremental Inductance Matrix* monitor is mandatory (*NT: Monitors ⇨ Monitors on Solids or Volumes)*:



The preparation of the numerical data for the state-space model is launched with the *Start* button of the embedded *Parameter Sweep* dialog. During this process, a large number of working points is calculated and stored parametrically within the corresponding CST EM Studio project.

After the calculation of all working points is finished, the state space model of a system is extracted into the binary file specified by the *Export State Space Model* dialog box. This file contains the serialization of the calculated working points, which is used as a basis for the calculation of the lumped parameters during the system simulation. The serialized data can then be imported either by CST Design Studio (magnetostatic case) or by Modelica-based system simulators (magnetostatic and time domain cases).

For more information on the coupled simulation based on the state space model concept, please refer to the online help.

Electrical Machine Task

For fast and convenient configuration of electrical machine models, the electrical machine task is available in the system assembly modeler (SAM). The task allows the simulation of predefined typical drive scenarios for usual electrical machine types. For further details on the electrical machine task workflow, the component library includes preconfigured drive scenarios and electrical machines elaborating the workflow.

Chapter 4 – Finding Further Information

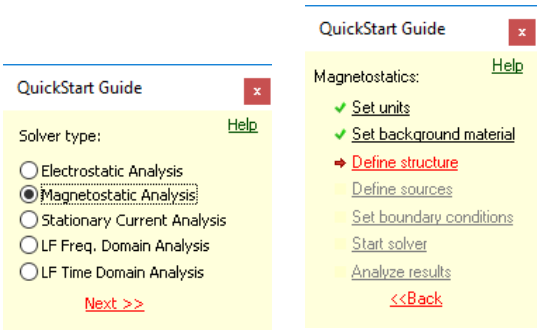
After having read this manual carefully, you should already have some idea of how to use CST EM Studio efficiently for your own problems. However, when you are creating your own first models, some questions will arise. In this chapter, we give you a short overview of the available documentation.

The QuickStart Guide

The main task of the QuickStart Guide is to remind you to complete all necessary steps in order to perform a simulation successfully. Especially for new users – or for those rarely using the software – it may be helpful to have some assistance.

The QuickStart Guide is opened automatically on each project start, when the checkbox *File: Options ⇨ Preferences ⇨ Open QuickStart Guide on project load* is checked. Alternatively, you may start this assistant at any time by selecting *QuickStart Guide* from the Help button in the upper right corner.

When the QuickStart Guide is launched, a dialog box opens showing a list of tasks, where each item represents a step in the model definition and simulation process. Usually, a project template will already set the problem type and initialize some basic settings like units and background properties. Otherwise, the QuickStart Guide will first open a dialog box in which you can specify the type of calculation you wish to analyze and proceed with the *Next* button:



As soon as you have successfully completed a step, the corresponding item will be checked and the next necessary step will be highlighted. You may, however, change any of your previous settings throughout the procedure.

In order to access information about the QuickStart Guide itself, click the *Help* button. To obtain more information about a particular operation, click on the appropriate item in the QuickStart Guide.

Online Documentation

The online help system is your primary source of information. You can access the help system's overview page at any time by choosing *File: Help ⇨ Help*. The online help system includes a powerful full text search engine.

In each of the dialog boxes, there is a specific *Help* button, which directly opens the corresponding manual page. Additionally, the *F1* key gives some context sensitive help when a particular mode is active. For instance, by pressing the *F1* key while a block is selected, you will obtain some information about the block's properties.

When no specific information is available, pressing the *F1* key will open an overview page from which you may navigate through the help system.

Please refer to the *CST Studio Suite - Getting Started* manual to find some more detailed explanations about the usage of the CST Studio Suite Online Documentation.

Tutorials and Examples

The component library provides tutorials and examples, which are generally your first source of information when trying to solve a particular problem. See also the explanation given when following the *Tutorials and Examples Overview* link on the online help system's start page. We recommend that you browse through the list of all available tutorials and examples and choose the one closest to your application.

Technical Support

Before contacting Technical Support, you should check the online help system. If this does not help to solve your problem, you find additional information in the Knowledge Base and obtain general product support at 3DS.com/Support.

Macro Language Documentation

More information concerning the built-in macro language for a particular module can be accessed from within the online help system's *VBA* book: Visual Basic (VBA) Language. The macro language's documentation consists of four parts:

- An overview and a general description of the macro language.
- A description of all specific macro language extensions.
- A syntax reference of the Visual Basic for Applications (VBA) compatible macro language.
- Some documented macro examples.

History of Changes

An overview of important changes in the latest version of the software can be obtained by following the *What's New in this Version* link on the help system's main page or from the *File: Help* backstage page. Since there are many new features in each new version, you should browse through these lists even if you are already familiar with one of the previous releases.