



EmCAD User Guide

Walter Steffè



*Hierarchical
Electromagnetics*

COPYRIGHT

This document is Copyright © 2020 Walter Steffè.

Permission is granted to make and distribute verbatim copies of this manual provided the copyright notice and this permission notice are preserved on all copies.

FEEDBACK

Please direct any comments or suggestions about this document to:
walter.steffe@hierarchical-electromagnetics.com

PUBLICATION DATE

First printing, July 2020

The image features a blue background with numerous water droplets of varying sizes, some in sharp focus and others blurred. A white rounded rectangle with an orange border is positioned in the lower-left quadrant, containing the word "Contents" in a bold, black, sans-serif font.

Contents

1	Introduction	5
1.1	What is EmCAD	6
1.2	The Design Flow	8
2	General Usage	13
2.1	The Graphical User Interface	14
2.1.1	The Main Window Structure	15
2.2	Component Definition	18
2.2.1	Importing the Component Geometry	18
2.2.2	Assignment of Materials and Boundary Conditions	20
2.2.3	Waveguide Ports	21
2.3	Material Definition	22
2.3.1	Non dispersive materials	22
2.3.2	Dispersive materials	23
2.3.3	Constant Loss Tangent	24
2.4	Definition of Boundary Condition Properties	25
2.4.1	Surface Impedance Model	25
2.4.2	Lossy Metal Model	26
2.5	Assembly Definition	28
2.5.1	Components Properties	29
2.6	Project Definition	30
2.7	Domain Decomposition	32
2.7.1	Decomposition Strategy	32
2.7.2	The Decompose Command	34
2.8	Meshing	36
2.9	Modelization	38
2.10	Frequency Domain Analysis	39
3	Application Cases	41
3.1	Filter Design	42
3.1.1	Ideal Filter Design	43
3.1.2	An example of Filter Implementation	49
3.1.3	Zero Pole Analysis	51
3.1.4	Filter Mapping	53
3.1.5	Filter Tuning	54
	Bibliography	60



1. Introduction

1.1 What is EmCAD

EmCAD is the graphical front-end of a client-server application for the numerical modelization of electromagnetic problems. *EmCAD* is Copyright © 2015 Walter Steffè and is distributed under the terms of the GNU General Public License, Version 2 or later.

The actual modelization work is performed in the *Hierarchical Electromagnetic Modeler* which is a proprietary code (owned by Walter Steffè) running on a cloud platform and delivered as a service on demand. *EmCAD* provides the graphical environment for the definition of the problem data structure which is sent to a remote server and processed by the *Hierarchical Electromagnetic Modeler*.

The *Hierarchical Electromagnetic Modeler* is based on a new numerical method for the generation of an equivalent circuit that represents the electrical response of a generic electromagnetic problem (microwave, optical ..). The circuit is generated in a direct path without going through an electromagnetic simulation and a subsequent circuit extraction process as it happens with the macro-modeling techniques based on the zero-pole fitting of response curves. The new method is more robust than the zero-pole fitting and it scales better making it possible to handle in an efficient way large multi-scale electromagnetic problems.

The electrical model preserves the fundamental properties (such as passivity, causality, reciprocity..) of the actual device and it is well suited for frequency and time domain simulations. It may be exported as a Spice circuit assuring the interoperability with most commercial circuit simulators. The electromagnetic problem may include dispersive materials defined by linear models (Lorentz, Debye, Drude . . .).

The *Hierarchical Electromagnetic Modeler* is designed to achieve the maximum efficiency when executed in a distributed environment exploiting the full power offered by the cloud computing. This computing architecture offers an efficient and economical pathway to tackle the complex electromagnetic problems arising in the design flow of modern microwave/electronic industry.

The need of accurate modeling of electromagnetic interactions occurring inside the electronic circuits is becoming always more important due to the steady trend towards higher system performance, higher operating frequencies and miniaturization. Higher frequencies and higher component density lead indeed to an increase of the electromagnetic interactions causing a degradation of the quality of electrical signals (noise effects, crosstalk..).

These adverse effects need to be properly simulated at system level in order to appreciate their global effect on the system performances. The *Hierarchical Electromagnetic Modeler* responds to these needs through the generation of circuit models that capture all the relevant electromagnetic effects and are well suited to be exploited in any kind of simulation in the time or in the frequency domain.

The equivalent model can be easily inserted into complex system (which may also include non linear devices) providing a better interoperability than the data generated by traditional Electromagnetic Solvers. The typical output data provided by those solvers

consist of Touchstone files describing the electrical response in the frequency domain. The utilization of this kind of data in time domain simulations (which are typically required at system level) is based on heavy numerical algorithms (Fourier transforms and convolutions) and it is not straightforward as with the EmCAD models.

The circuit generated by the *Hierarchical Electromagnetic Modeler* represents the electrical response of the actual device over a large frequency band which starts at the zero frequency and goes up to a user defined maximum frequency. The capability of handling, at the same time, the high frequency response and the DC response is very useful to analyze, in example, the noise effect, such as the simultaneous switching noise, which arise in digital circuits due to the interaction between the “Signal Delivery Network” and the “Power Delivery Network”. This is a challenging Signal Integrity problem with a strong impact on the performances of a digital circuits operating at high frequencies.

The most distinguishing property which puts the *Hierarchical Electromagnetic Modeler* aside of competition is nevertheless a very efficient domain decomposition. This technique allows to overcome the fundamental drawback of all other current general purpose full-wave simulators which, by going against the hierarchical (top-down) design methodologies that the electronics industry is accustomed to, treat the electromagnetic device as an unstructured black box.

The *Hierarchical Electromagnetic Modeler* makes it possible to decompose a large electromagnetic problem in its constituent parts (sub-domains) which are subjected to independent modelizations producing a set of sub-circuits. As a result there is a one to one relationship between the hierarchical structure of the sub-components and the hierarchical structure of the related equivalent circuits. This property allows a dramatic improvement in the computational speed of the repetitive runs which are typically required in the design/optimization phase. In fact, thanks to the preservation of the hierarchical structure of the EM problem, a local change in the geometrical or physical parameters of a given subcomponent affects only the related sub-circuit and there is no need to recompute the sub-circuits associated with the other (unchanged) parts.

Actually it is felt that the inability of the present EM tools to cope with a hierarchical design methodology has been, up to know, the principal barrier for their wider adoption. The utilization of a traditional EM solver for the analysis of a complex circuit is a heavy task which requires large computational resources and long run times. Because of that, in many practical situation, this kind of tool is relegated into the final post layout verification phase and can not be fully exploited in the design/optimization phase. We believe that the introduction of the *Hierarchical Electromagnetic Modeler* will fill this gap and will allow a proper handling of the electromagnetic effects since the initial phases of the design flow.

1.2 The Design Flow

In a typical design flow the geometrical structure associated with the electromagnetic problem is defined using a separate CAD system and transferred to the EmCAD environment using the STEP file format¹. The *EmCAD* tool is then used to define the electrical properties associated with the CAD objects (material properties, boundary conditions, waveguide ports ...).

The user names associated to those objects within the CAD systems play an important role to assure a stable association between the electrical properties and CAD entities and it is very important that their names are included in the exported step file. In a typical design/optimization activity the design flow has to be iterated several times changing the geometrical parameters defined inside the CAD environment. In this situation EmCAD assures that all the electrical properties associated to the initial geometry are preserved and automatically re-associated to the updated geometry.

Unfortunately not all commercial CAD systems are suitable for the definition of the problem geometry because many of them do not export the names of some CAD objects into the step file. Up to now only a few CAD systems have been successfully used in the definition of an EmCAD problems. These are CATIA V5, ZW3D and the RT branch of FreeCAD. My first EmCAD designs were based on the Catia CAD system which was available in the company where I was employed. Since a few years I have switched to FreeCAD (RT branch) which has improved a lot up to a point that I would recommend it for most EmCAD problems. The examples given in this document was designed with FreeCAD.

Depending on the particular application, the design flow may involve an external simulator used for certain kind of (application dependent) circuitual analyses. A simpler design flow relying entirely on EmCAD and *Hierarchical Electromagnetic Modeler (HEM)* is illustrated (without involvement of external tools) is shown in the Table 1.2. This example deals with the modelization and subsequent frequency domain simulation of a single electromagnetic component defined in a FreeCAD Part.

The hierarchical structure of the electromagnetic problem may be created in two ways. It can be done using the Assembly structure of the CAD systems. In this case the name of the Assembly must start with "NET_" while the part names must start with "CMP_". A lower level in the hierarchical structure may be created also using a set of splitting surfaces which are defined inside a CAD Part. To be effectively recognized as splitting surfaces these surfaces must have names starting with "SPLIT_".

The data structure based on the CAD Assembly may be considered as the result of a "bottom-up" decomposition strategy because the CAD Parts are designed before being inserted into an assembly structure. The data structure created by the splitting surfaces may instead be considered as the result of a "top down" decomposition strategy because the CAD Part is designed before being partitioned into its subdomains. The two decomposition strategies ("bottom-up" and "top down") may coexist in the same project. In fact an Assembly of parts may be used to represent a network of components and each component may be decomposed into a set of subdomains.

¹STEP-File is a widely used data exchange format. It is an ISO standard defined in the document ISO 10303-21 "Industrial automation systems and integration – Product data representation and exchange – Part 21: Clear Text Encoding of the Exchange Structure".

Name Prefix	Object Type
DIEL_	Dielectric
BC_	Boundary Condition
WGP_	Waveguide Port
LP_	Line Port
SPLIT_	Partitioning surface
GRID_	Grid of partitioning surfaces
CMP_	Component
NET_	Assembly of Components

Table 1.1: CAD Naming Convention

Tool	Tasks
CAD	<p>Component Geometrical Definition</p> <ul style="list-style-type: none"> -The geometrical structure associated with an electromagnetic component is described in a CAD Part which may include non manifold objects (surfaces and/or curves). -Solid bodies are used to define dielectrics. Surfaces are used to define boundary conditions, waveguide ports and splitting surfaces.
<i>EmCAD</i>	<p>Component Electrical Definition</p> <ul style="list-style-type: none"> -The step file associated with the CAD Part is imported in EmCAD. -Different objects of the CAD Part are automatically assigned an electromagnetic type (Dielectric, Boundary Condition, Waveguide Port, ...) using the naming convention reported in table 1.1 -Materials and Boundary Conditions may be imported from a library or created by the user which may assign a name and a set of electric properties to each new material. -Materials and Boundary Conditions (selected from the list) may be assigned to Dielectrics and to Boundary Condition Surfaces included in the structure tree. -Number of TE and TM port modes may be specified for the waveguide ports. If not specified by user those numbers are set to 1 for TE and to 0 for TM modes. Number of TEM ports is automatically computed from surface topology (simply or multiply connected). -Other general parameters (not related to a specific object) such as frequency band, mesh-size, ... are specified in the <i>Project Properties</i> window. -The data structure must be processed by the <i>decompose</i> and <i>Mesh</i> commands. The <i>Decompose</i> command splits the objects included in each Component (derived from a CAD Part) using the splitting surfaces defined in the same Part. The <i>Decompose</i> command performs also an "imprinting" of adjacent faces associated with different Parts of an Assembly. The common faces belonging to adjacent parts are grouped to form a sort of super-faces which define the electrical interfaces between adjacent components. The <i>Mesh</i> command discretizes these interfaces generating triangular meshes. The meshed interfaces are required for the definition of a common set of electrical ports on adjacent component when they are processed in parallel by separate instances of the <i>Hierarchical Electromagnetic Modeler</i>. -The generated data structures may be saved into the EmCAD file at any stage of the design flow using the <i>Save As</i> or the <i>Save</i> command.

<i>HEM</i>	<p>Modeling</p> <ul style="list-style-type: none"> -The "Modelize" command sends a request to a remote server² for the modelization of the subcomponents defined by the domain decomposition. -The Modelization tasks associated with different subcomponents are executed in parallel. When a task is completed the <i>EmCAD</i> frontend receives an electrical circuit representing the electrical response of that subcomponent. -The subcomponent circuits are inserted into a global circuit which represents the electrical response of the whole electromagnetic structure.
<i>EmCAD</i>	<p>Frequency Domain Analysis Definition</p> <ul style="list-style-type: none"> -The "Frequency Domain Analysis" command opens a new window aimed to the definition of the relevant analysis parameters such as the frequency band, number of sample points used in the computation of the frequency response and number of frequency point used in a model order reduction of the global circuit.
<i>HEM</i>	<p>Frequency Domain Analysis Computation</p> <ul style="list-style-type: none"> -The "Start" command inside of the "Frequency Domain Analysis" window sends a request to the remote servers for the processing (Model Order Reduction and subsequent Frequency Domain Analysis) of the global circuit. -The Frequency Response is computed on a Reduced Model which is much more compact (several order of magnitude) than the original global circuit and can be analysed in a very short time. -When the Frequency Domain Analysis is completed, the <i>EmCAD</i> frontend receives the reduced circuit associated with the electromagnetic structure and the related frequency response.
<i>EmCAD</i>	<p>Local Processing of Results</p> <ul style="list-style-type: none"> -The frequency response can be plotted in a graphic window or can be exported in a Touchstone file. -The reduced circuit can be exported in a Spice file.

Table 1.2: A typical design flow

²The access to the modelization service is restricted to authorized users. The *Hierarchical Electromagnetic Modeler* is still in an experimental phase and is not yet available for general usage. A user account may be granted only to a small number of partners which are actively contributing to the development of the *EmCAD* project. Companies interested on this collaboration may send an email to request@hierarchical-electromagnetics.com for further details.



2. General Usage

2.1 The Graphical User Interface

The figure 2.1 shows the geometrical an helical filter designed with the FreeCAD code while The figure 2.2 shows the main EmCAD window as it appears after having imported the step file and having defined the electrical properties of the dielectrics and the boundary conditions. It should be noticed that any void cavity must be regarded as a special dielectric (Vacuum) which is characterized by unitary relative dielectric constants ($\epsilon_r = 1$, $\mu_r = 1$).

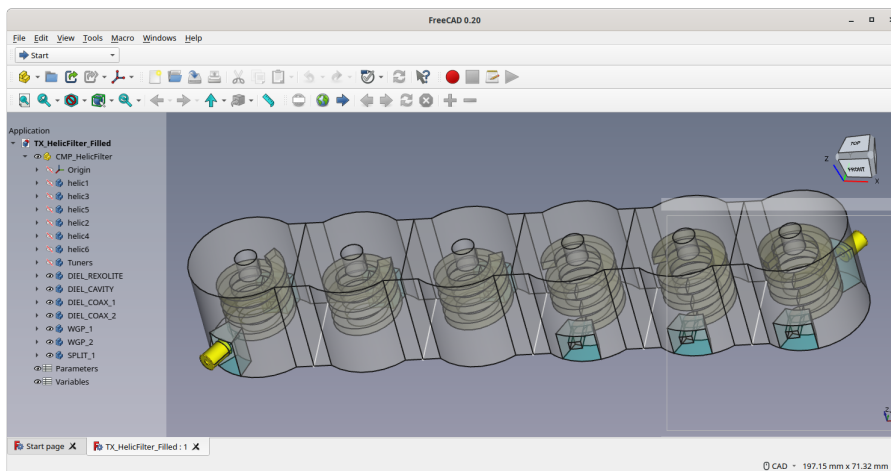


Figure 2.1: FreeCAD Part defining the geometry of an helical filter

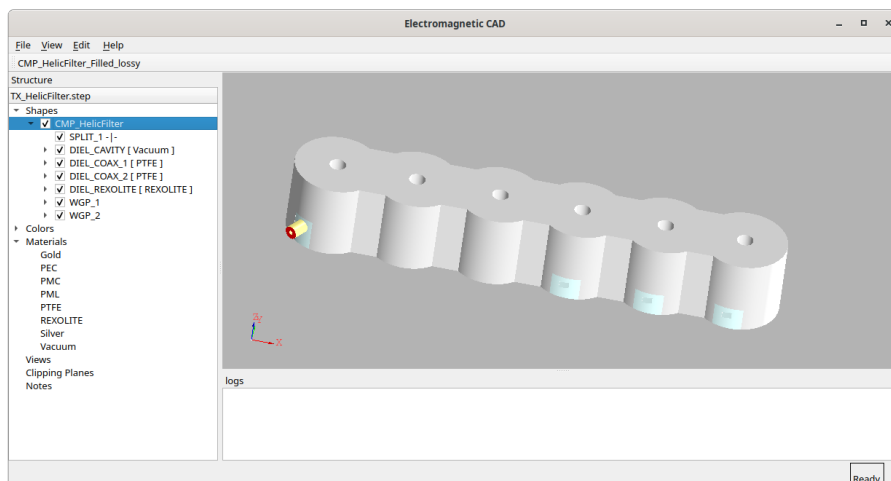


Figure 2.2: The main window after assignment of electric properties

2.1.1 The Main Window Structure

As shown in Figure 2.2 the main window of EmCAD comprises the following items:

The Menu-bar

The menu-bar, which is located across the top of the main window, provides several commands distributed across a set of drop-down menus. These commands are explained in full detail in the following sections.

The Geometry Viewer

The geometry viewer is located at the right side area below the menu-bar. This tool displays a 3 dimensional perspective of the imported geometries. The position and the aspect of the displayed objects can be adjusted using the *rotate*, *pan* and *zoom* commands activated through the mouse and keyboard controls defined in table 2. The objects displayed in the geometry viewer are mouse sensitive so that the clicking of a given object activates and highlights the corresponding item in the Structure Tree (see below).

Controls	Effect
Mouse movement + LMB	Rotation
(Mouse movement + shift + LMB) or (Mouse movement + CMB)	Pan
Rotation of mouse wheel	Zoom

Table 2.1: Keyboard Controls of the Geometry View

The Message area

The Message area is located at the lower right side of the main window. It is used to display status or error messages generated by local processing or by remote services.

The Structure Tree

The structure tree, which is located at the left side of the main window, gives a hierarchical view of the objects displayed in the 3D viewer window. The tree items can be displayed or undisplayed (hidden) in the 3D viewer window by checking or unchecking a small box. Object hiding is a useful mean to make visible other objects in the background.

The structure tree includes a node named *Shapes* which is the root of sub-tree containing the geometrical structure defined in the CAD system. Only the CAD Assemblies whose names start with “NET_“ and the CAD parts whose name start with “CMP_“ are actually imported in this data structure.

The geometrical items of the structure tree are mouse sensitive, so that, in example, when an item is clicked, the corresponding object in the 3D Viewer is selected and highlighted.

The structure shown in the figures 2.1 and 2.2 is defined by a single Part and it is decomposed into 6 subdomains by five splitting surfaces. The splitting surfaces are defined in the FreeCAD Body named “SPLIT_1“.

A FreeCAD Body is not restricted to be a Solid Object. It is instead a sort of container which can be used to define any kind of geometry. The “SPLIT_1“ Body has indeed been used to define a set of surface objects.

The Structure Tree may be traversed (up and down) using a few commands located in the *File* submenu. The commands *Open Component* (or the CTRL-O shortcut) and *Close Component* (or the CTRL-U shortcut) are used to enter into a lower level and respectively into the higher level of the model hierarchy. The 3D view window is updated according to the current level which is identified by a name displayed in a text line just below the menu bar. In example in figure 2.2 the displayed name ("CMP_HelicalFilter_lossy") is the name of the CAD Part defining the filter component. This is the top level of the model hierarchy associated with this design.

After a first invocation of the *Open Component* command the graphical window depicted in figure 2.3 displays the partitioned geometry that was created by the *Decompose* command. The switch to a new view is made evident by the */Partition* suffix appended to name of the current viewed level.

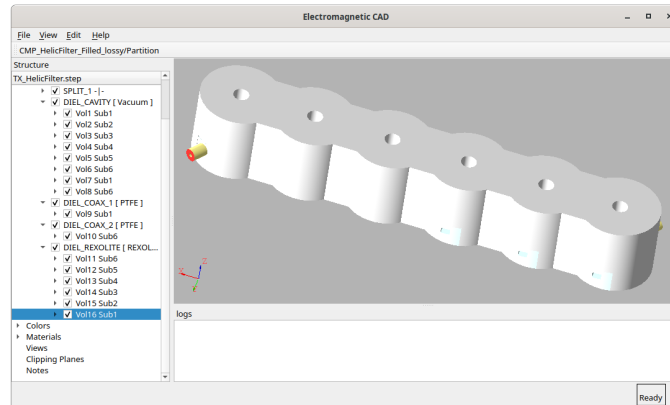


Figure 2.3: Partitioned Component View

The dielectric objects included in the structure tree of this Partition View are partitioned in a set of smaller volumes. A list of these volumes is displayed under the item associated with each dielectric object. In example the dielectric object named "DIEL_REXOLITE" is split in 6 volumes named "Vol_11" ... "Vol_16". Each volume is assigned to one of the 6 subdomains defined by the splitting surfaces. The subdomain associated with each volume is displayed after the volume name.

The internal geometry associated with a specific subdomain can be visualized using the command *Open Component* after having selected a volume that belongs to that subdomain. In example if the "Vol_6" is selected in the structure shown in figure 2.3, the command *Open Component* enters into the first subdomain as shown in figure 2.4. It may then be seen that the dielectrics included in this subdomains are two separate void cavities, a piece of rexolite and the PTFE dielectric associated with the input connector. The structure trees includes also a waveguide port which corresponds with the red face shown in the 3D window.

The window view can be reverted back to that one shown in figure 2.3 using the *Close Component* command. A second invocation of this command closes also the Partition level restoring the view shown in 2.2. The command named *Close Component AND Partition*

2.1 The Graphical User Interface

allow to jump directly from the subdomain to the component skipping the intermediate level associated with the Component Partition.

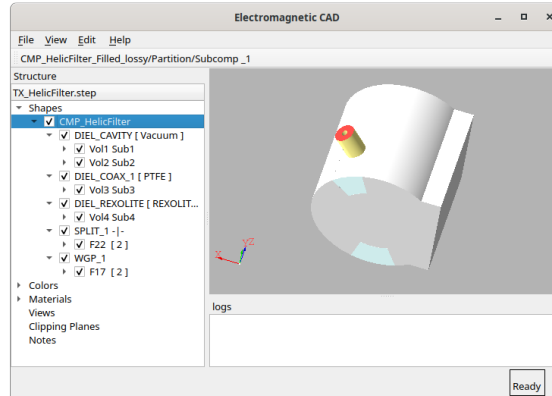


Figure 2.4: A Component Subdomain

The commands *Open Component* and *Close Component* can be used also to navigate in a more complex CAD structure consisting of an Assembly of Parts. After being imported into the EmCAD environment this kind of structure is translated into a Network of Components. The top level view is the Assembly (Net) view while a first invocation of the command *Open Component* allows to enter inside a Component (corresponding to a CAD Part). A second invocation of the command *Open Component* allows to enter into the partition of that component. The command *Open Component AND Partition* combines the two actions allowing to go directly from the assembly (Net) into the Partition of the selected component.

Besides the geometrical objects (placed under the *Shapes* node), the structure tree includes the materials and boundary condition properties which are placed under the *Materials* node. These data may be imported from a library or defined inside the project. The materials (or boundary conditions) may be assigned to the dielectrics (solids) or to the boundary conditions (surfaces) placed under the *Shapes* node. The assignment mechanism is explained in detail in section 2.2.2.

2.2 Component Definition

2.2.1 Importing the Component Geometry

After having created a new *EmCAD* project is created with the “File/New” command, the CAD geometry can be imported from a step file using the *File/Import Geometry* command.

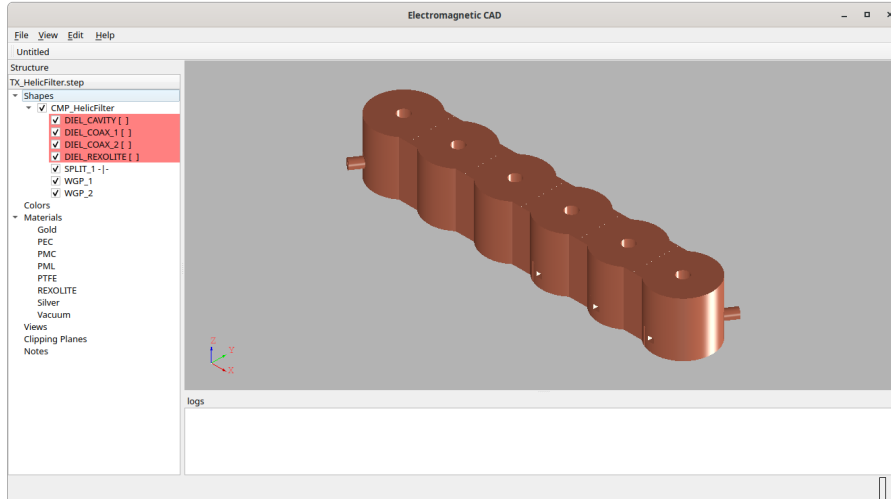


Figure 2.5: The main window after step file import

The electromagnetic types of the imported objects are defined through the naming convention defined in the table 1.1. The CAD objects whose names are not compliant with conventions listed in table 1.1 are not used by the EmCAD modeler. The imported Solid objects are typically used for the definition of dielectric elements while the imported Surfaces may be used for the definition of Boundary Conditions, Waveguide Ports and splitting tools. Imported Curves may be used for definition of Line Ports. The Boundary Conditions may be associated also with solid objects but in this case only their boundary surfaces are used.

The structure of figure 2.5 includes four dielectrics ("DIEL_CAVITY", "DIEL_COAX_1", "DIEL_COAX_2" and "DIEL_REXOLITE"), one set of splitting surfaces named "SPLIT_1" (which contains 5 splitting faces) and two Waveguide Ports ("WGP_1" and "WGP_1"). Inside of FreeCAD these ports were defined as two separate bodies each one of them contains a single face. The red color associated with the imported dielectrics is a warning of their undefined state. In fact, apart of the geometry, the definition of a dielectric object requires also an assignment of its material properties.

The CAD structure may include additional geometries used as construction elements such as, in example, a sketch used to define, through a solid extrusion, a dielectric object. These constructive elements are not used in the electromagnetic simulation and are not imported into the *EmCAD* environment. The naming convention based on the prefixes defined in the table 1.1 allows to distinguish the geometrical objects that are necessary for the definition of the electromagnetic problem and to identify their "Electromagnetic Types".

The metallic helics shown in fig 2.1 are used to cut (with a boolean operation) a few holes into the surrounding dielectrics. These objects could have been used to enforce a specific metal condition on their boundary. But this was not necessary for the definition of

2.2 Component Definition



the electromagnetic problem because these boundary condition was the default Boundary Condition. The default Boundary Condition, which can be set using a command placed in the “Edit/Materials“ submenu, is automatically applied to all external faces that are not subjected to any explicit boundary condition. Here "external face" means any boundary face of a dielectric object boundary which is not shared with any other dielectric object.

2.2.2 Assignment of Materials and Boundary Conditions

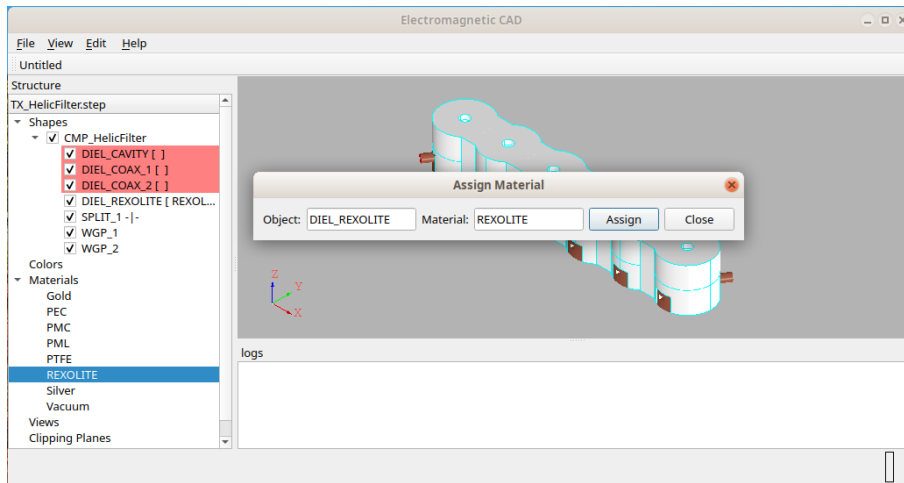


Figure 2.6: Material Assignment

The material association is done using the command *Edit/Materials/Assign Material* which opens a small window with two entry fields: the geometric object (to which the material is applied) and the material. The first field gets filled with the selection of an item in the Shapes list. The second one gets filled with the selection of an item in the Materials list. When the association is done by the set button, the related geometrical object displayed in the 3D viewer assumes the color of the material.

2.2.3 Waveguide Ports

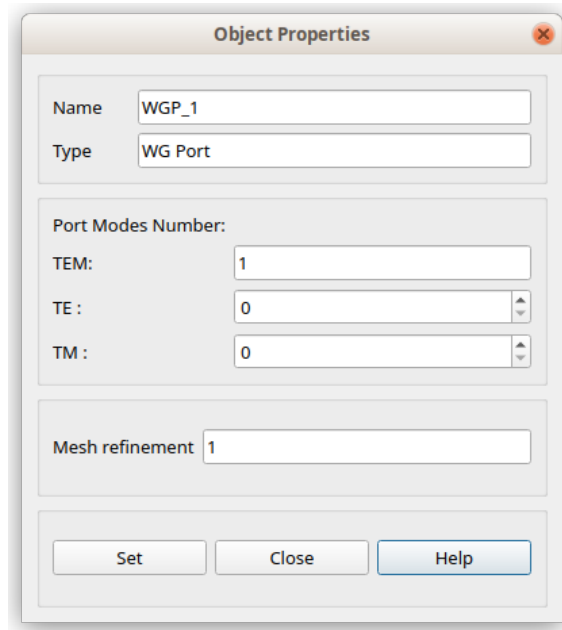


Figure 2.7: Waveguide part properties

The window shown in figure 2.7 is opened by the command *Set Object Properties* from a context menu that is activated by pressing the Right Mouse Button on the waveguide port named WGP_1 which is selected in the Structure Tree. In this window it is possible to set the numbers of TE and TM modes and the mesh refinement.

The number of TEM modes is automatically evaluated from the surface topology and is not editable. The total number of modes (TEM+TE+TM) must be greater than 0. The default number of TM modes is 0. The default number of TE modes is 1 or 0 depending on whether the number of TEM modes is null or a positive number.

The mesh refinement is defined as the ratio between the number of mesh elements per wavelength associated with the selected object and the global value of this parameter which is defined in the *Project Options*. Mesh Refinement=1 means no local refinement.

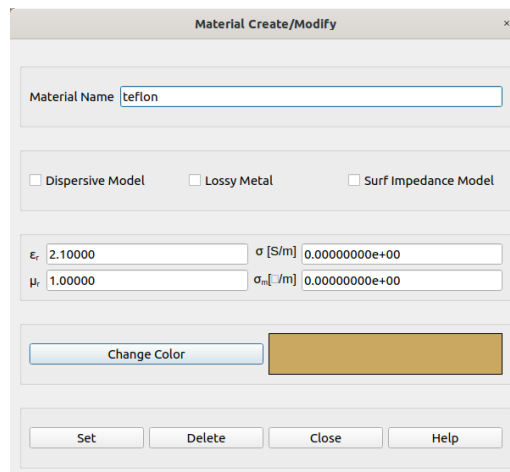
The *Set Object Properties* command may be applied also to other kind of objects such as, in example, Dielectrics or Boundary Conditions. In this case the windows 2.7 allows only the setting of the mesh refinement.

2.3 Material Definition

New materials are created or edited using the *Material/Create Modify* command. This action opens the window shown in fig 2.8 where it is possible to specify the material name, its color and the main electrical properties.

2.3.1 Non dispersive materials

The fig 2.8 shows the definition of a non dispersive material which is fully defined by the relative permittivity and permeability constants.



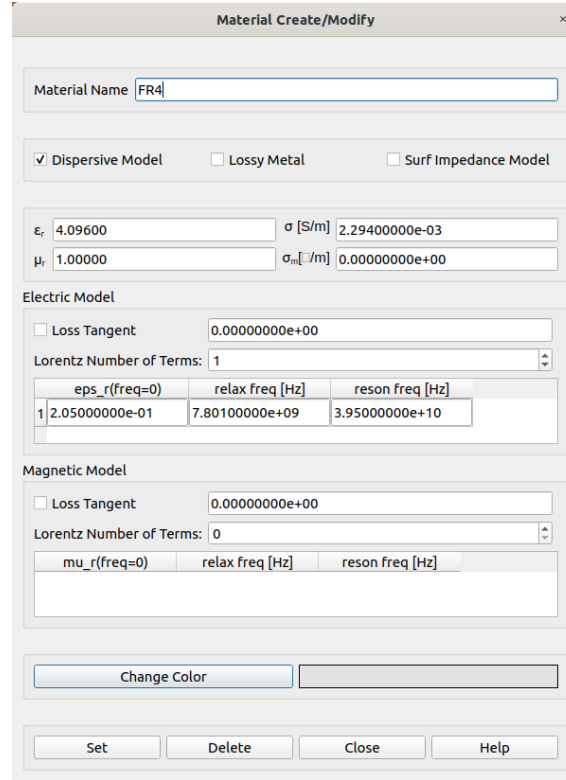
The image shows a software dialog box titled "Material Create/Modify". It contains the following elements:

- A text input field for "Material Name" containing the text "teflon".
- Three unchecked checkboxes: "Dispersive Model", "Lossy Metal", and "Surf Impedance Model".
- Two rows of input fields for electrical properties:
 - Row 1: ϵ_r (value: 2.10000) and σ [S/m] (value: 0.00000000e+00)
 - Row 2: μ_r (value: 1.00000) and σ_m [A/m] (value: 0.00000000e+00)
- A "Change Color" button next to a color selection box showing a brownish-gold color.
- Four buttons at the bottom: "Set", "Delete", "Close", and "Help".

Figure 2.8: Non dispersive material

2.3.2 Dispersive materials

When the flag named *Dispersive Model* is checked the *Material Create/Modify* window is expanded and a few additional fields are shown (see fig 2.9).



Material Name: FR4

Dispersive Model Lossy Metal Surf Impedance Model

ϵ_r : 4.09600 σ [S/m]: 2.29400000e-03
 μ_r : 1.00000 σ_m [A/m]: 0.00000000e+00

Electric Model

Loss Tangent: 0.00000000e+00

Lorentz Number of Terms: 1

	eps_r(freq=0)	relax freq [Hz]	reson freq [Hz]
1	2.05000000e-01	7.80100000e+09	3.95000000e+10

Magnetic Model

Loss Tangent: 0.00000000e+00

Lorentz Number of Terms: 0

	mu_r(freq=0)	relax freq [Hz]	reson freq [Hz]

Change Color

Set Delete Close Help

Figure 2.9: Dispersive material

The *Hierarchical Electromagnetic Modeler* can deal with a generic dispersive materials which, according to equation 2.1, may be expressed as a superposition of Lorentz and Debye terms, plus a conductive term. This kind of representation allows an accurate modelization of all practical linear materials.

$$\epsilon(\omega) = \epsilon_0 \epsilon_{r\infty} - j\sigma/\omega + \sum_i \epsilon_{Di}(\omega) + \sum_i \epsilon_{Li}(\omega) \quad (2.1)$$

$$\epsilon_{Di}(\omega) = \frac{\epsilon_0 \Delta \epsilon_{ri}}{1 + j\omega/\omega_{ri}} \quad \omega_{ri} = \text{Relaxation Frequency} \quad (2.2)$$

$$\epsilon_{Li}(\omega) = \frac{\epsilon_0 \Delta \epsilon_{ri}}{1 + j\omega/\omega_{ri} - (\omega/\omega_{oi})^2} \quad \omega_{oi} = \text{Resonant Frequency} \quad (2.3)$$

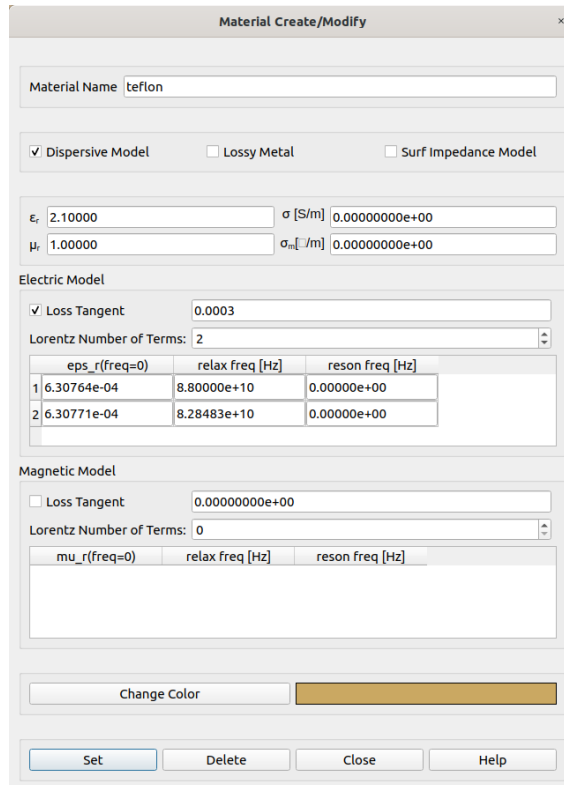
The parameter *Terms Number* allows to specify an arbitrary number of Lorentz/Debye terms. The Debye terms are (conventionally) defined by entering a “0” value in the last column (resonant frequency).

In a similar way it is possible to specify the dispersion law of the magnetic constant $\mu(\omega)$. The Lorentz and Debye models associated with $\mu(\omega)$ are defined by a couple of equations which are formally identical to equations 2.3 and 2.2.

An expansion term of the form 2.2 is characterized by a simple (real) pole and known as a Debye model. An expansion term of the form 2.3 is characterized by a couple of conjugate poles and is known as a Lorentz model.

The example shown in fig 2.9 refers to the FR4 material which (see [3]) can be well approximated by the superposition of a conductive term and a single Lorentz term with a relaxation frequency of 7.801 GHz and a resonant frequency of 39.5 GHz.

2.3.3 Constant Loss Tangent



Material Name: teflon

Dispersive Model Lossy Metal Surf Impedance Model

ϵ_r : 2.10000 σ [S/m]: 0.00000000e+00
 μ_r : 1.00000 σ_m [A/m]: 0.00000000e+00

Electric Model

Loss Tangent: 0.0003

Lorentz Number of Terms: 2

	eps_r(freq=0)	relax freq [Hz]	reson freq [Hz]
1	6.30764e-04	8.80000e+10	0.00000e+00
2	6.30771e-04	8.28483e+10	0.00000e+00

Magnetic Model

Loss Tangent: 0.00000000e+00

Lorentz Number of Terms: 0

	mu_r(freq=0)	relax freq [Hz]	reson freq [Hz]

Change Color

Set Delete Close Help

Figure 2.10: Constant Loss Tangent

Quite often in the practical applications the electrical losses of a material are specified by a loss tangent ($\tan \delta = -Im(\epsilon)/Re(\epsilon)$) which is assumed to be constant over the frequency band of interest. A strictly constant loss tangent is not physically realizable but can be approximated over a finite bandwidth by a superposition of Debye terms. With EmCAD this is achieved by checking the Loss Tangent box inside the “Material Create/Modify” window and then by specifying the number of terms used in the approximation. The frequency band coincides with that one specified in the "Project Options". The approximate lossy model is generated upon pressing the set button.

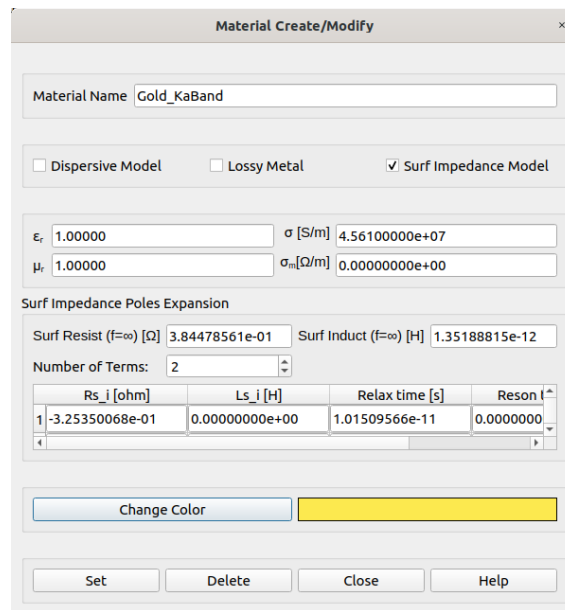
The example shown in fig 2.10 shows a couple of Debye Terms generated for a Teflon material with a constant loss tangent of 3×10^{-4} over the frequency band from 78 to 88 GHz.

2.4 Definition of Boundary Condition Properties

Besides bulk materials, the “Material Create/Modify” window can also be used to define the electrical properties of Boundary Conditions. These properties are displayed in the material list together with the properties of ordinary materials but are meant to be assigned to Boundary Surfaces with names prefixed by the string "BC_". Ordinary materials are instead meant to be assigned to Dielectrics which are Solids Objects with names prefixed by the string "DIEL_".

Boundary condition properties may be created by checking the *Surf Impedance Model* flag or the *Lossy Metal* flag inside of the window opened by the *Material Create/Modify* command. The window is then expanded showing a number of additional fields that are explained in the following subsections. Two boundary condition properties are already predefined. These are the *PEC* (Perfect Electric Conductor) and the *PMC* (Perfect Magnetic Conductor) boundary conditions.

2.4.1 Surface Impedance Model



The screenshot shows the "Material Create/Modify" dialog box. The "Material Name" field is set to "Gold_KaBand". Under the "Model" section, the "Surf Impedance Model" checkbox is checked, while "Dispersive Model" and "Lossy Metal" are unchecked. The "Properties" section shows $\epsilon_r = 1.00000$, $\mu_r = 1.00000$, σ [S/m] = $4.56100000e+07$, and σ_m [Ω/m] = $0.00000000e+00$. The "Surf Impedance Poles Expansion" section includes "Surf Resist (f=∞) [Ω]" = $3.84478561e-01$, "Surf Induct (f=∞) [H]" = $1.35188815e-12$, and "Number of Terms" = 2. A table below shows the expansion parameters:

	Rs_i [ohm]	Ls_i [H]	Relax time [s]	Reson t
1	-3.25350068e-01	0.00000000e+00	1.01509566e-11	0.00000000

At the bottom, there is a "Change Color" button with a yellow color swatch, and "Set", "Delete", "Close", and "Help" buttons.

Figure 2.11: A Surface Impedance Model

When the flag named *Surf Impedance Model* is checked, the *Material Create/Modify* window is expanded as shown in fig 2.11. The new field allows to specify a rational pole/residue expansion of the surface impedance $\mathcal{Z}_S(\omega)$ which defined according to the equation 2.4.

$$\mathcal{Z}_S(\omega) = R_S + j\omega L_S + \sum_p \frac{R_{Sp}}{1 + j\omega/\omega_{rp}} + \sum_q \frac{R_{Sq} + j\omega L_{Sq}}{1 + j\omega/\omega_{rq} - (\omega/\omega_{0q})^2} \quad (2.4)$$

$$\omega_r = \frac{2\pi}{\text{Relaxation Time}} \quad \omega_0 = \frac{2\pi}{\text{Resonant Time}}$$

2.4.2 Lossy Metal Model

When the flag named *Lossy Metal* is checked, the *Material Create/Modify* window is expanded as shown in fig 2.12. The new field allows to set a couple of parameters which uniquely define a Causal Huray Model [1] of the metallic surface surface. As shown in

The screenshot shows the 'Material Create/Modify' dialog box. The 'Material Name' is 'Gold_KaBand'. The 'Dispersive Model' checkbox is unchecked, while 'Lossy Metal' and 'Surf Impedance Model' are checked. The material properties are: $\epsilon_r = 1.00000$, $\mu_r = 1.00000$, σ [S/m] = $4.56100000e+07$, and σ_m [Ω m] = $0.00000000e+00$. The 'Lossy Metal Model' section includes: Frequency [GHz] = 28.70000, Number of Fitting Poles = 1, Loss Factor = 1.51670712, and Q Factor = 4.60234630. The 'Surf Impedance Poles Expansion' section shows: Surf Resist ($f=\infty$) [Ω] = $3.84478561e-01$, Surf Induct ($f=\infty$) [H] = $1.35188815e-12$, and Number of Terms = 2. A table below lists the terms:

	Rs_i [ohm]	Ls_i [H]	Relax time [s]	Reson time [s]
1	-3.25350068e-01	0.00000000e+00	1.01509566e-11	0.00000000e+00
2	-4.02351622e-02	0.00000000e+00	6.49158631e-11	0.00000000e+00

At the bottom, there is a 'Change Color' button with a yellow color swatch, and 'Set', 'Delete', 'Close', and 'Help' buttons.

Figure 2.12: A Surface Impedance Model

the following formulas, which are taken from [1], a Causal Huray Model is the product of the surface impedance $Z_S(j\omega)$ of a smooth metallic surface multiplied by a complex correction factor $H_C(j\omega)$ that depends on the parameters K and ω_c .

$$Z_{\text{Rough}}(j\omega) = Z_S(j\omega) H_C(j\omega) \quad (2.5)$$

$$Z_S(j\omega) = (1+j) \sqrt{\frac{\omega\mu}{2\sigma}} \quad H_C(j\omega) = 1 + \frac{K}{1 + (j2\omega/\omega_c)^{-1/2}}$$

The two parameters may be expressed as a function of the loss factor and Q factor evaluated at a frequency ω_0 :

$$\text{Loss_factor} = \left. \frac{\text{Re}(Z_{\text{Rough}})}{\text{Re}(Z_S)} \right|_{\omega=\omega_0} \quad Q_factor = \left. \frac{\text{Im}(Z_{\text{Rough}})}{\text{Re}(Z_{\text{Rough}})} \right|_{\omega=\omega_0} \quad (2.6)$$

The Loss factor and the Q factor should never be less than 1. The values Loss factor = 1, Q factor = 1 lead to the surface impedance of a smooth metallic surface $Z_{\text{Rough}} = Z_S$.

The Huray model need to be converted into a rational model which is done trough a pole zero fitting. This operation is triggered by pressing the set button. The field named "Number of fitting Poles" is used to specify the maximum number of terms used in the fitting algorithm. I have observed that a single pole is usually sufficient to assure a good rational fitting of the Causal Huray Model. The generated rational model is described by

2.4 Definition of Boundary Condition Properties



the equation 2.4 and it may be visualized upon checking the flag named "Surf Impedance Model".

For a more practical usage of the Causal Huray Model it would be better if the parameters K and ω_c were expressed in terms of some physical property of the metallic surface such as in example the Ra value associated with the roughness profile. Unfortunately I have not yet found a formula which is able to describe the functional relationship between the physical parameters and the Huray model.

2.5 Assembly Definition

While a simple electromagnetic problem may be defined using a single CAD part, in many applications it is preferable to use an assembly with a name prefixed with “*NET_*“. The assembly enforces a first level of domain decomposition in which the sub-domains are identified with its parts. If needed, a lower level of decomposition may be enforced using a set of splitting surfaces defined inside of each part. The step file exported from the CAD assembly is imported into the *EmCAD* environment using the command *File/Import Geometry*.

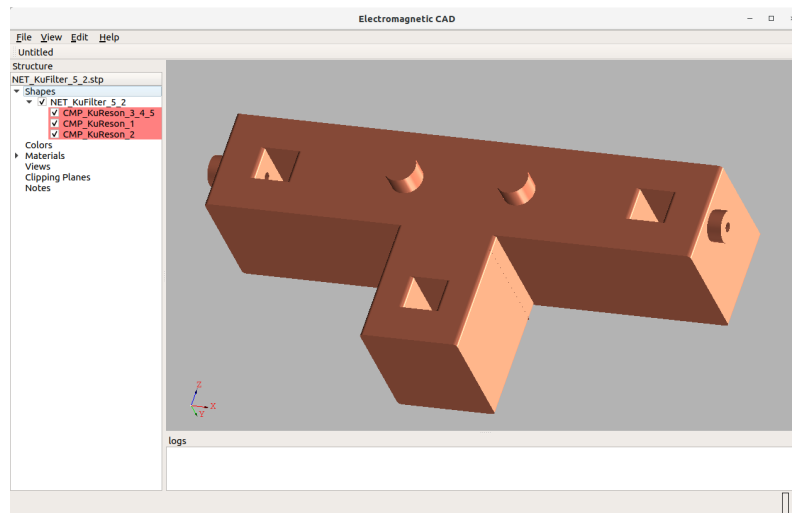


Figure 2.13: Imported Assembly

The fig 2.13 shows a filter which has been imported into the *EmCAD* environment from a CAD assembly composed of three parts. These parts are recognized as electrical components and are therefore included in Shapes sub-tree. The imported components are marked with a red box as a warning that they are not yet fully defined (some electrical properties are still undefined).

2.5.1 Components Properties

The electrical properties of each component can be specified after having imported the assembly by descending in the data structure with the “Open Component” command but this is not the recommended way to proceed.

In most cases it is more convenient to define the electrical properties of each component in a separate *EmCAD* document. In this process, for each component, the related step file is imported in a new document and the electrical properties are defined for all the component objects. Then the component is decomposed and meshed to validate the data structure and it is saved in a new *EmCAD* file.

The component data which was previously defined in separate *EmCAD* files may be easily re-associated to the components imported from the CAD assembly. This is done using the command named *Import Component Properties* from a context menu (activated by the Right Mouse Button) associated with a selected component. This command opens a window which allows to associate the component the electromagnetic data stored in an *EmCAD* file selected with a file browser.

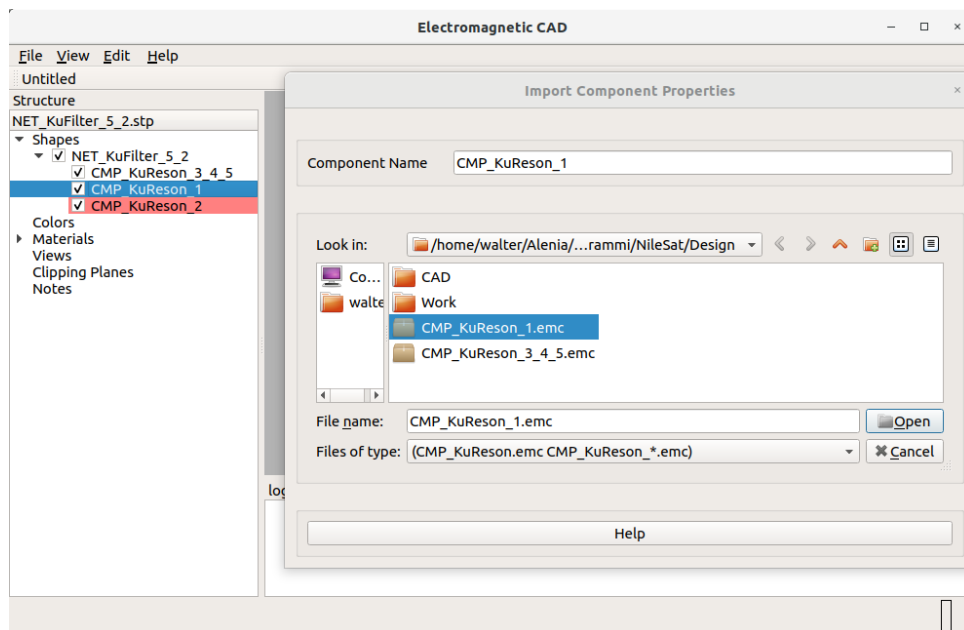


Figure 2.14: Importing component properties

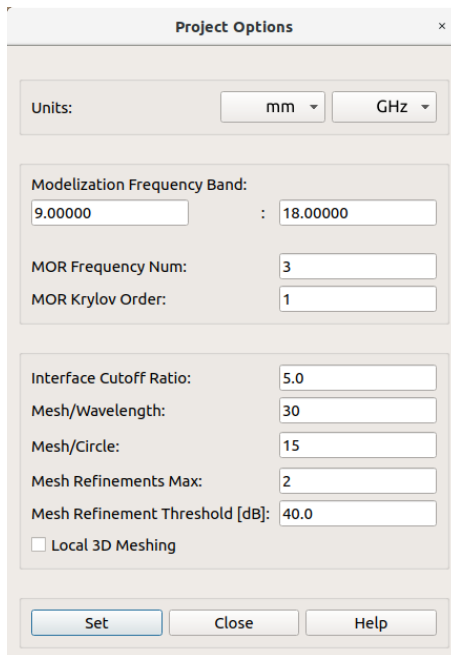
The selections shown in figure 2.14 make an association between the tree item named "COMP_RESON_1" and the data file "CMP_RESON_1.emc".

An assembly may include several identical or similar components. In this situation it is not necessary to define a separate data file for each one of these components. The electromagnetic properties defined inside of a representative electrical component may in fact be reused, through the *Import Component Properties* command, for all the other instances (or variations) included in the assembly. This is possible because the “Import Component Properties” action affects only the non-geometrical data (material properties, number of ports..) associated with the selected component. These data are assumed to be identical in all the instances/variations of the given component.

2.6 Project Definition

The command named the *Edit/Project Options* opens the graphical window shown in fig 2.15 which is aimed to the setting of the following general project properties:

- *Length and Frequency units*
- *Modelization Frequency Band*. This is the frequency band used in the modelization of each component or sub-component.
- *MOR Frequency Num*. This is the number of discrete frequencies used in the Model Order Reduction of each component. The MOR procedure computes the electric voltages associated with the circuit nodes, for all the given frequencies and all the ports excitations. The circuit reduction is defined by a projection on the subspace spanned by these solutions.
- *MOR Krylov Order*. The Krylov order specifies the number of derivatives versus frequency of the electrical response that are matched at each modelitazion frequency. A higher Krylov order improves the approximation but the simulation time is also increased. A Krylov order equal to 1 means that the only the value of the frequency response (which may be interpreted as a derivative of order 0) is matched in the Reduced Model. This is the most convenient choice for most practical cases.
- *Interface Cutoff Ratio*. This parameter defines a threshold for the number of modes used to represent the tangential field at the interfaces between adjacent components or adjacent sub-components. The used modes are those characterized by a cut-off frequency lower than the upper frequency specified in the *Modelization Frequency Band* multiplied by the *Interface Cutoff Ratio*.
- *Mesh per Wavelength*. This number specifies the initial mesh density before any refinement. The wavelength parameter is that one associated with the upper frequency specified in the *Modelization Frequency Band*.
- *Mesh per Circle*. This parameter (number of mesh elements in a full circle) controls the mesh density on curved surfaces.
- *Mesh Refinements Max*. This number specifies the maximum number of mesh refinements which may be applied in the modelization of each component.
- *Mesh Refinements Threshold*. The mesh refinement is driven by the ratio between electromagnetic energies associated with quadratic and linear basis functions. The Mesh refinement process is stopped when this ratio is lower than the specified threshold or the maximum number of refinements is reached.
- *Local 3D meshing*. This flag enforces a local 3D meshing which makes it possible the inspection of the initial mesh before proceeding with a modelization request. To save the transmission bandwidth the local 3D mesh is not sent to the remote server. On the server side the meshing is performed before starting the first simulation and it is repeated at each step of the mesh refinement.



The screenshot shows a dialog box titled "Project Options" with a close button (X) in the top right corner. The dialog is organized into several sections:

- Units:** Two dropdown menus, one for "mm" and one for "GHz".
- Modelization Frequency Band:** Two input fields with values "9.00000" and "18.00000" separated by a colon.
- MOR Frequency Num:** An input field with the value "3".
- MOR Krylov Order:** An input field with the value "1".
- Interface Cutoff Ratio:** An input field with the value "5.0".
- Mesh/Wavelength:** An input field with the value "30".
- Mesh/Circle:** An input field with the value "15".
- Mesh Refinements Max:** An input field with the value "2".
- Mesh Refinement Threshold [dB]:** An input field with the value "40.0".
- Local 3D Meshing:** A checkbox that is currently unchecked.

At the bottom of the dialog, there are three buttons: "Set", "Close", and "Help".

Figure 2.15: Project Options

2.7 Domain Decomposition

2.7.1 Decomposition Strategy

The *Hierarchical Electromagnetic Modeler* achieves the maximum efficiency when the electromagnetic problem is decomposed in a set of domains that are not too large in terms of a wavelength. A large electromagnetic problem should be decomposed into many small domains that may be processed in a relatively short time by several instances of the *Hierarchical Electromagnetic Modeler*.

The numerical complexity of the global circuit is usually much lower than that one associated with the original electromagnetic problem solved with of a FEM method. The lower numerical complexity is brought by a Model Order Reduction algorithm that is applied to each separate sub-component. The tetrahedral mesh associated with a typical electromagnetic component with a size in the order of a wavelength may contain several tens of thousands of mesh elements. The Model Order Reduction Algorithm replaces this relatively complex structure with a compact circuit that represents the electrical response of the electromagnetic problem with a high accuracy.

There are many ways to decompose a given electromagnetic problem and not all decompositions are equivalent in terms of numerical efficiency. Besides the number of sub-domains and the related sizes, the numerical efficiency does also depend on the positioning of the interfaces between adjacent subdomains. For a better efficiency the interfaces should non be placed too close to sharp edges or sharp corners of a conducting boundary because in these zones the electrical field may present very fast spatial variations. An accurate representation of such a singular field would require a high number of modes leading to a strong penalization on the computational time because the numerical complexity of the MOR algorithm is proportional to the square of the number of ports.

To be more precise the numerical complexity of the projection task (which is a part of the MOR algorithm) is proportional to the square of the number of field solutions multiplied by the mesh size (or the number of mesh edges). The number of field solution is equal to the product of the number of ports and the number MOR frequencies:

$$\begin{aligned}
 N_{\text{complexity}} &\propto N_{\text{Solutions}}^2 \times N_{\text{Edges}} \\
 N_{\text{Solutions}} &= N_{\text{Frequencies}} \times N_{\text{Ports}}
 \end{aligned}
 \tag{2.7}$$

It should be noted that an increasing of the component size leads to an increase of all the numbers involved in equation 2.7. The number of mesh edges is clearly increased. The number of modal ports is also increased because of the increased size of the component interfaces. Finally the number of sample frequencies should also be increased to maintain the same accuracy because a larger component typically is characterized by faster variations of its frequency response.

It may then be concluded that the computational time required for the modelization of a component increases very fast with the size of the related sub-domains. This fast increase is not compensated by the reduction of the number of subdomains so that, a decomposition with many small components is usually more convenient than a decomposition with few larger components. The improvement of the computational efficiency with the decreasing of the sub-domain sizes may not proceed indefinitely and it encounters a limit when the time requested for the simulation of the global circuit starts to dominate over the time associated with the modelization of the components.

Up to now the *EmCAD* code was used and tested almost exclusively in the design of microwave filters. In this application it is quite reasonable to identify the subdomains with the individual resonators as done with the coaxial filter shown in fig 2.2.

In the filter design the time spent in the simulation of the global circuit is usually in the order of a second or less and is much lower than the time (in the order of several minutes) requested for the modelization of the components. It is nevertheless believed that in other applications (to be investigated in the near future) the simulation of the global circuit may play a more significant role in the whole process.

2.7.2 The Decompose Command

The *Edit/Decompose* command starts a processing of the CAD geometry which involves:

- Decomposition of each component in a set of subdomains which are defined by the splitting surfaces included in the related CAD part.
- The “imprinting” and “sharing” of all adjacent geometrical entities. Touching faces (or touching edges edges) are split with the separation of their the common part (their intersection). If the two faces (edges) belong to adjacent solids the common part becomes a shared face (edge) of the two solids.

The *Decompose* command is disabled until all the components properties have been defined. The imprinting and sharing is applied to geometrical entities belonging to the same component (in example to adjacent dielectric of a component) and also to adjacent components. A face which is shared between two components defines a multiport (multimodal) electrical interface.

A generic decomposition is defined by a set of surfaces with names starting with the string "SPLIT_". Equispaced splitting planes may be more conveniently defined by means of a “Planar Grid“. This is defined by a curve segment with a name starting with the string "GRID_". The curve segment can be linear or circular and the splitting planes are orthogonal to it. A couple of splitting planes (the first plane and the last plane in an oriented sequence) pass through the extreme vertices of the curve segment. The total number of planes is specified using the command *Object Properties* in the *EmCAD* environment after having selected the imported GRID object.

An equispaced cylindrical grid is specified by a linear segment and a circle (or a circular arc). These two entities must have an equal name starting with the string "GRID_". The circle defines the common axis of the cylindrical surfaces while the linear segment, which must be directed along a radius, is used in the same way as the segment of the planar grid. The two segment vertices define the first and last cylindrical surfaces while the number of intermediate cylindrical surfaces is specified inside of the *EmCAD* environment.

The face sharing assured by the "Decompose" action may be inspected by looking inside of the components data structure. So in example the fig 2.16 shows the internal data structure of the component named "CMP_KuReson_1". From the figure 2.16 it may be seen that the face number 26, which is selected in the Tree Data Structure, is shared by the dielectric objects named "DIEL_COAX" and "DIEL_Cavity". The face number 19 which belongs to the dielectric object named "DIEL_COAX" is hidden in the 3D Viewer because it was unchecked in the Tree Data Structure.

2.7 Domain Decomposition

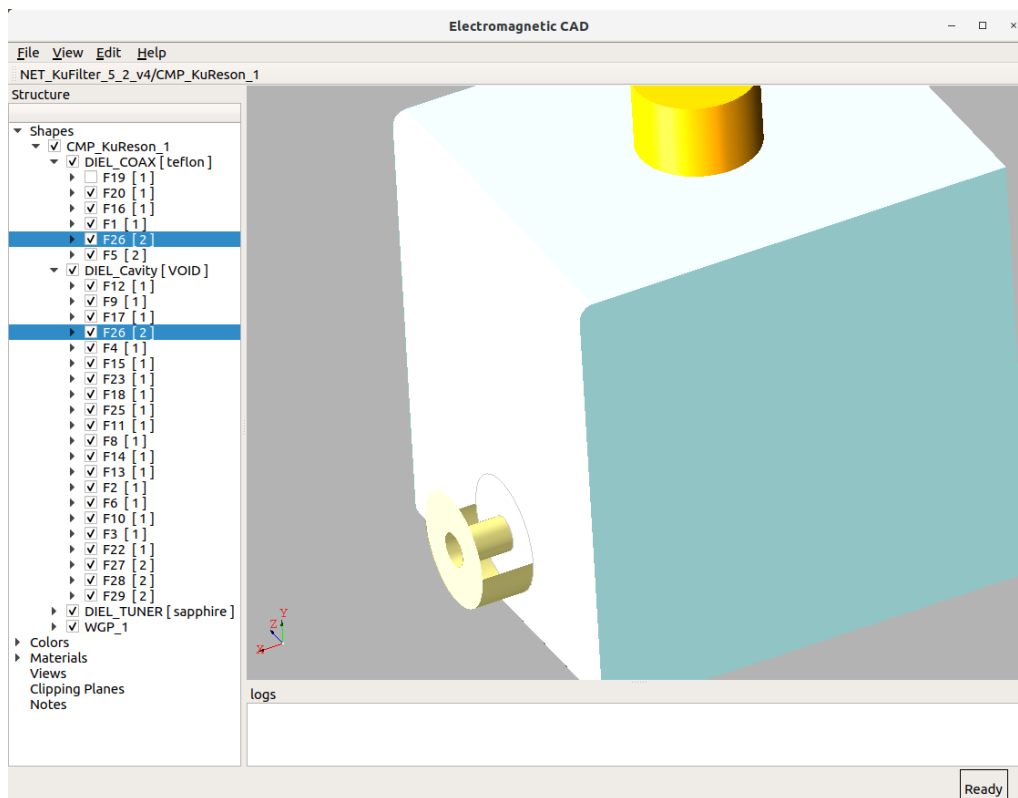


Figure 2.16: Component Decomposition

2.8 Meshing

The *Mesh* command is disabled until the user has defined all the component objects, the frequency band and the mesh size. The meshing action should be preceded also by the decomposition but this operation is automatically executed, if not yet done, when the *Mesh* command is invoked.

If the flag named *Local 3D meshing* is not checked the *Mesh* command generates only the surfaces meshes associated with the interfaces shared by adjacent components. The tetrahedral meshing of the internal solids is left to the remote modelization service. The triangulated interfaces are sent to the modelization service to assure a congruent definition of the modal fields over the common interfaces.

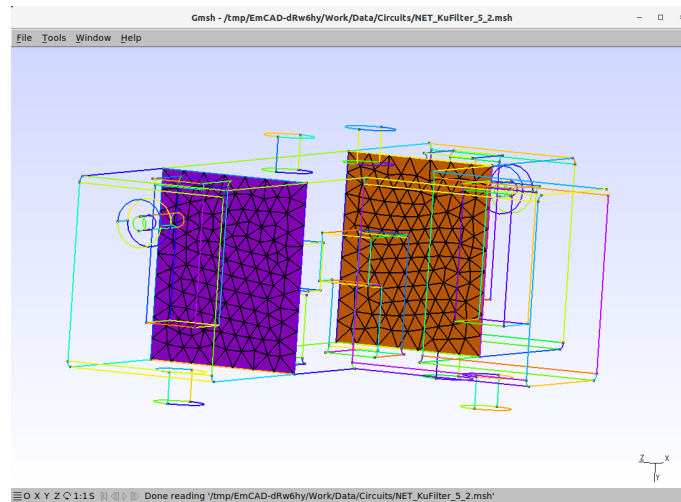


Figure 2.17: Visualization of meshed interfaces defined in the main assembly

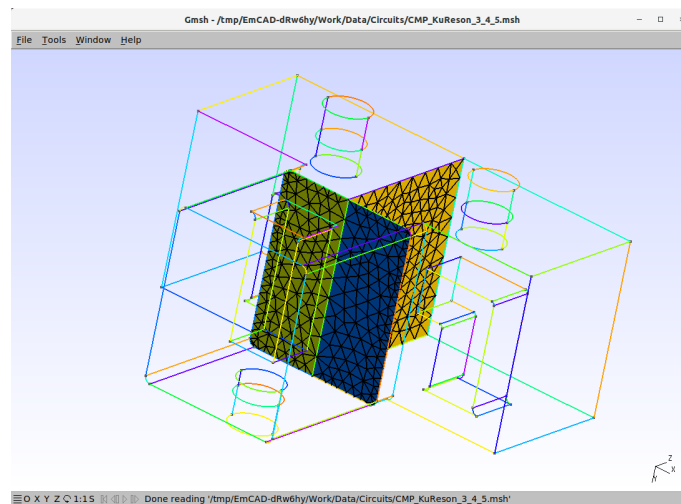


Figure 2.18: Visualization of meshed interfaces defined in a Component

The generated mesh may be inspected by entering the command named *Mesh View* which makes a call to the Gmsh If the *Mesh View* command is entered when the *EmCAD* window is positioned as shown in figure 2.2, it [2] software which should be installed on the same machine in which *EmCAD* resides.

If the *Mesh View* command is entered when the *EmCAD* window displays the main assembly (as shown in figure 2.2) it displays the triangulation associated with the internal interfaces of that assembly (as shown in fig 2.17).

If the *Mesh View* command is entered when the *EmCAD* has entered into a component which is split in subcomponents (such as the component named “CMP_KuReson_3_4_5” shown in figure ??) it displays the triangulation associated with the internal interfaces of that component as shown in fig 2.18.

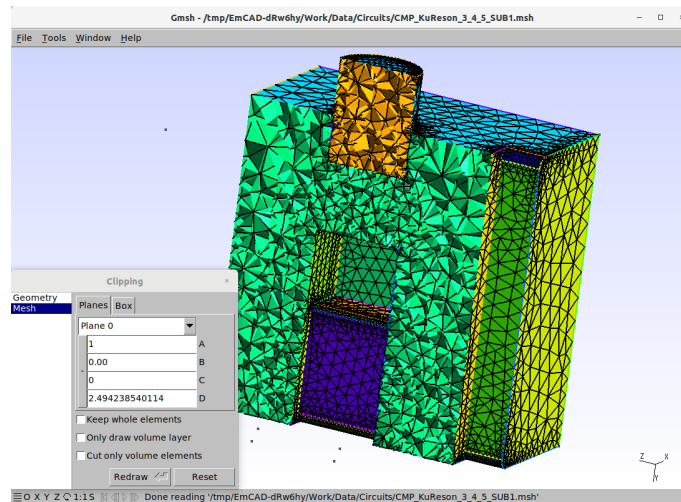


Figure 2.19: Visualization of volume mesh of a Sub Component

If flag named *Local 3D meshing* is checked the *Mesh* command generates also the tetrahedral meshes. Such a tetrahedral mesh may also be inspected with the *Mesh View* when the *EmCAD* is positioned inside of the related component (or subcomponent).

In example, in order to inspect the tetrahedral mesh associated with the first subcomponent defined by the splitting surfaces shown in fig 2.18, it is necessary to navigate inside of that subcomponent as explained in section 2.1.1. If the clipping tool, which is provided by the Gmsh code, is activated, the view of the tetrahedral mesh appears as shown in fig 2.19

2.9 Modelization

The *Modelize* command sends a request to the remote service asking for the modelization of each component included in the project. This command is disabled until the local meshing is completed.

The modelization requests associated with different components are processed in parallel. Once the modelization of a component is completed, the *EmCAD* frontend downloads the related equivalent circuit and stores it on the local machine. A message appears in the user window to inform that the task is completed (see figure 2.20).

The *Modelize* command terminates (with a ready message) when all the circuits have been retrieved and downloaded into the local machine.

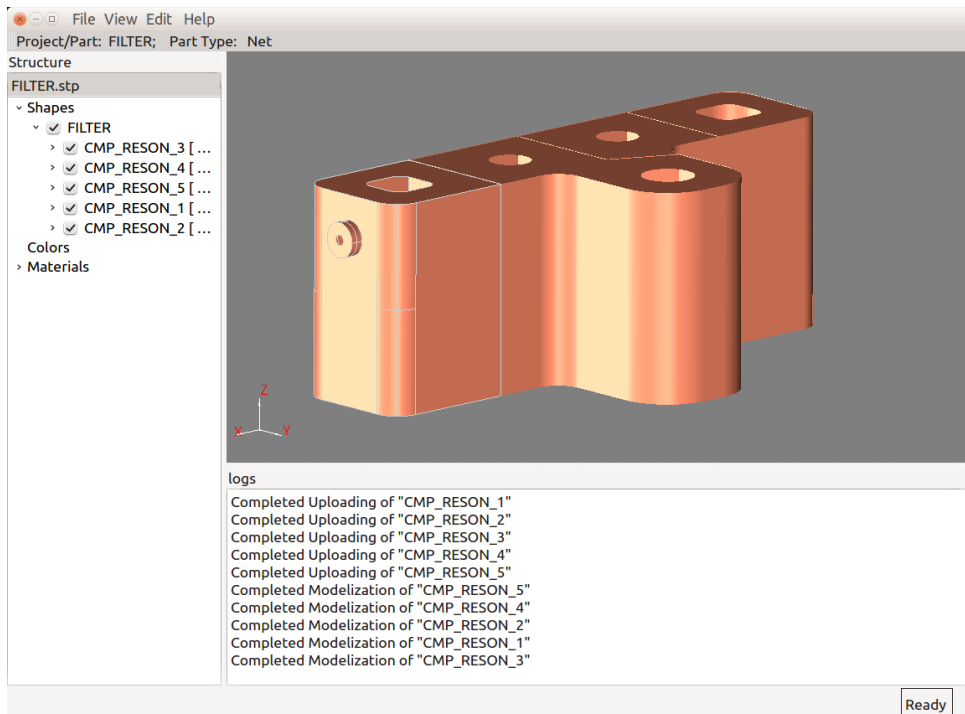


Figure 2.20: Modeling

2.10 Frequency Domain Analysis

The *Frequency Domain Analysis* command is disabled until all the components circuits have been generated. When started this command asks the remote service for the computation of the frequency response associated with the entire electromagnetic structure (which may be composed of several components and sub-components).

The computation of the frequency response is done on a reduced model which is computed in advance (on the server side) starting from the global circuit composed of the component circuits.

The *Frequency Domain Analysis* command opens the window shown in figure 2.21 which allows the specification of the following parameters.

- *Frequency Band*. This is the frequency band used in the Model Order Reduction of the global circuit and in the subsequent frequency domain analysis.
- *Frequency Point Number* This is the number frequency points included in the frequency response.
- *Parameter Type* This is the type of network parameter for which the frequency response is requested. Admissible parameters are *S* (Scattering Parameters), *Z* (Impedance Parameters) and *Y* (Admittance Parameters).
- *MOR Frequency Num.* This is the number of frequency points used in the generation of the Reduced Global Circuit. The default value of this number is 10 but it should set accordingly to the problem size (in terms of wavelengths) and the frequency bandwidth. The number of points required for a good interpolation of the frequency response do indeed increase as the problem size and frequency bandwidth increase.
- *Circuit* In the normal usage this entry is set to *Electromagnetic Model* which means that the requested frequency response is that one associated with the electromagnetic structure. Other admissible entries are *Ideal Filter* and *Mapped Circuit*. These alternative options are meant to compute the responses of other circuits which are generated in the design and tuning of microwave filters (see section 3.1).

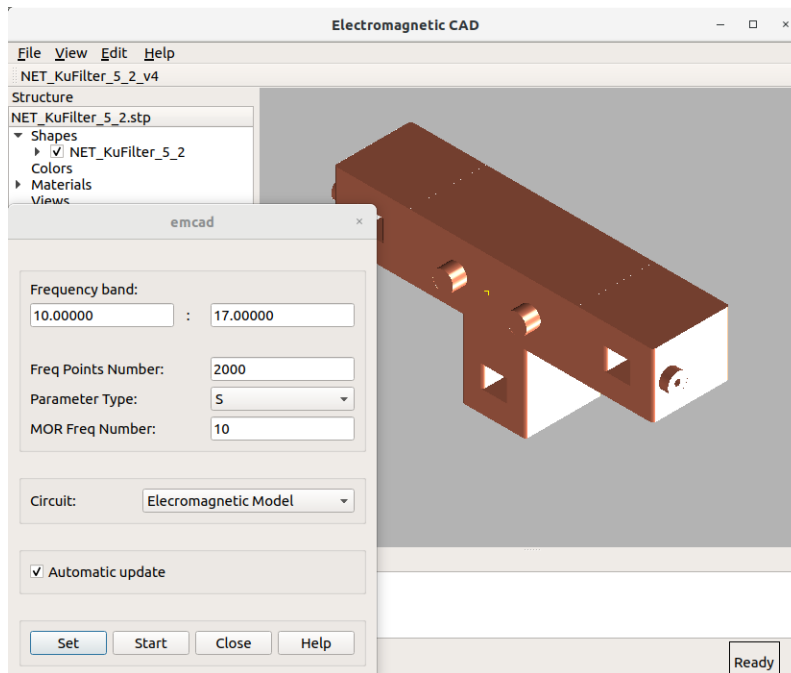


Figure 2.21: Frequency Domain Analysis

The frequency response associated with the electromagnetic model may be visualized using the command *View/Plots/Frequency Response/Electromagnetic Model*.

The figure 2.22 shows the S parameter plot of the filter shown in figure 2.21. The axis scale can be changed with the *Edit/Set Scale* command from the plot window.

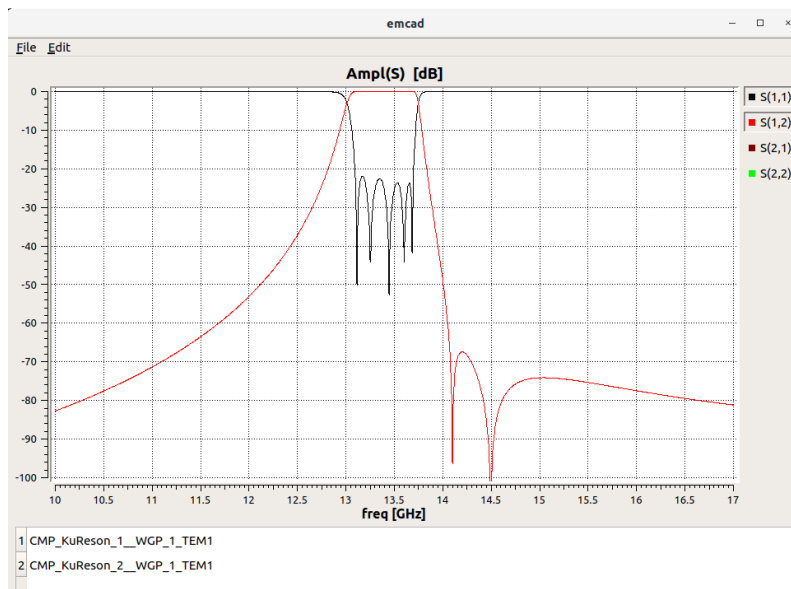


Figure 2.22: Frequency Response Plot

The frequency response may also be exported in a *Touchstone* file using the command *File/Export/Frequency Response*. The reduced global circuit may be exported into a *Spice* netlist using the command *File/Export/Spice Circuit*.



3. Application Cases

3.1 Filter Design

The *EmCAD* design environment is very well suited for the design of microwave filters. The equivalent circuit generated by the *Hierarchical Electromagnetic Modeler* may be exploited for the computation of the complex pattern (in the Laplace plane) of transmission and reflection zeros associated with the physical structure. The computation of these data is a unique capability of the *EmCAD* software which is not matched but any other electromagnetic solver available on the market.

The pattern of transmission and reflection zeros is a very useful information because the comparison of the patterns associated with the electromagnetic model and the ideal filter is a good measure of the error associated with the microwave implementation. This kind of comparison is much more helpful than the comparison between the frequency responses because the zeros patterns have a finite size and are in a one to one relationship with the electrical parameters associated with the ideal filter circuit.

In fact from these patterns (plus an additional constant) it is possible to compute an equivalent circuit which has the same topology as the ideal circuit and the electric response of the microwave structure. In the following this circuit will be denoted as the "mapped circuit".

The *EmCAD* design environment provides a set of tools which, for a limited class of filter topologies, allows to perform the design and tuning of a microwave structure in a short time and in an efficient manner.

A first tool is aimed to the generation of an ideal circuit that meets the electrical specifications of the filter. Other tools address the computation of the Transmission and Reflection Zeros and, from these data, the computation of the "mapped circuit".

The comparison of the numerical parameters associated with the mapped and ideal circuits is very powerful diagnostic tool which may be exploited in the design a tuning of the electromagnetic structure. The *EmCAD* environment offers a quasi-automatic iterative tuning procedure which is driven by this comparison. At each step of the iteration the *EmCAD* tuner applies a variation to the geometrical parameters which are stored in a text file associated with the CAD document ¹. The procedure is not yet fully automated because the updating and exporting of the CAD geometry into the step file has to be done by the user at each iteration. The effort left to the designer is not very daunting because the filter tuner usually is able to reach the design goal in a small number of steps.

¹In the current *EmCAD* release the link with the CAD data works only with Catia v5

3.1.1 Ideal Filter Design

The *EmCAD* environment provides a tool aimed to the generation of an ideal circuit which meets the given filter specifications. These specifications are defined through the graphical window shown in fig. 3.1 which is activated with the command *Edit/Analyses/Design/Ideal Filter*.

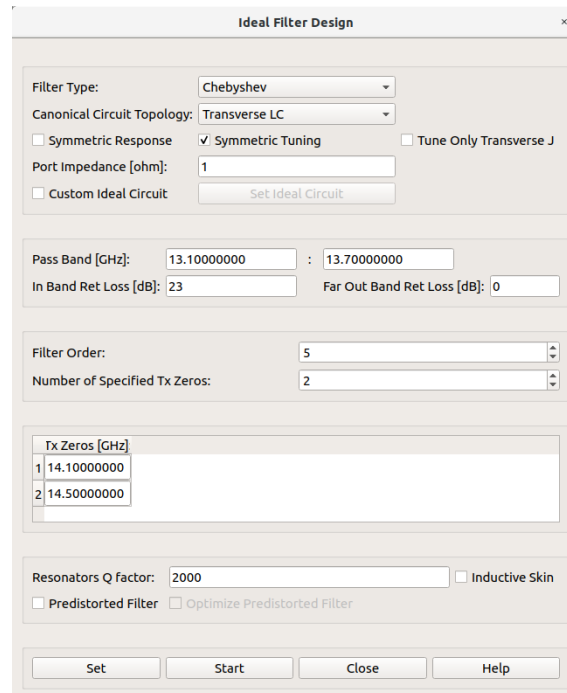


Figure 3.1: Input Parameters

The most relevant design parameters are:

- *The filter Type.* The available filter types are *Chebyshev* and *Maximally Flat*. The *Chebyshev* type is characterized by an equi-ripple response of the return loss parameter. This is the most usual choice because it guarantees the maximum selectivity for a fixed value of the maximum return loss. In the *Maximally Flat* filter all the reflection zeros are placed at the center frequency. It has been observed that this choice gives better results pre-distorted designs characterized by a low unloaded Q factor.
- *The Canonical filter Topology.* There are a few canonical circuit topologies which may be used to implement a given frequency response. The most interesting and better tested is the default one which is named "Transverse LC".
- *Custom Ideal Circuit.* When checked this flag activates the "Set Ideal Circuit" button. This button starts a new window aimed to the definition of a Custom Circuit Topology which may be better suited (than the canonical one) to implement the filter response in the selected resonator technology.
- *The filter passband.*
- *The maximum return loss*
- *The filter order (number of resonators)*
- *The number of transmission zeros*
This number may be zero (which leads to an all inline filter topology) or greater than

zero but it must always be lesser than the filter order.

- *The frequencies of the transmission zeros*
- *The Q factor*

The Q factor is assumed to be identical for all resonators (uniform Q) and a null value of Q is interpreted as an infinite value. In a normal (not predistorted) design a positive value of Q does not change the values of the reactive elements but it is used, in a post-processing phase, to define the resistive part of the resonators.

- *The predistorted flag*

The checking of this flag leads to a pre-distorted design which is characterized by a insertion loss response that is flattened over the passband. This flattening is obtained at the expense of the return loss and insertion loss figures. The degradation of these parameters becomes more severe as the Q factor decreases.

In the pre-distorted design the specification of the return loss and of the transmission zeros are used to define a lossless filter that represents a starting point of the pre-distorted design. A perturbation is then applied to this initial design in order to compensate for the degradation of S21 flatness caused by the finite Q value. This perturbation decreases the S21 amplitude in the proximity of the center frequency so that it becomes lower than the S21 amplitude near the band edges. This kind of deformation counterbalances that one caused by the finite Q value.

If the specified Q value is not too low, the predistorted filter should be characterized by a frequency response of the S21 parameter (expressed in dB) that, apart of a vertical translation, is equal to that one of the lossless filter defined by the return loss and by the transmission zeros.

The Canonical Filter Topology

The figure 3.2 shows a schematic of a canonical filter circuit composed of 5 resonators which implements a couple of transmission zeros. A circuit with this topology (named *Transverse LC*) is generated when the Ideal Filter Design tool is started with the settings shown in figure 3.1. The canonical filter topology consists of a folded circuit in which the longitudinal couplings are realized by J inverters while the transversal couplings are realized by a superposition of mutual capacitances and mutual inductances. The mutual capacitances may also be replaced by a shunt capacitors with opposite sign.

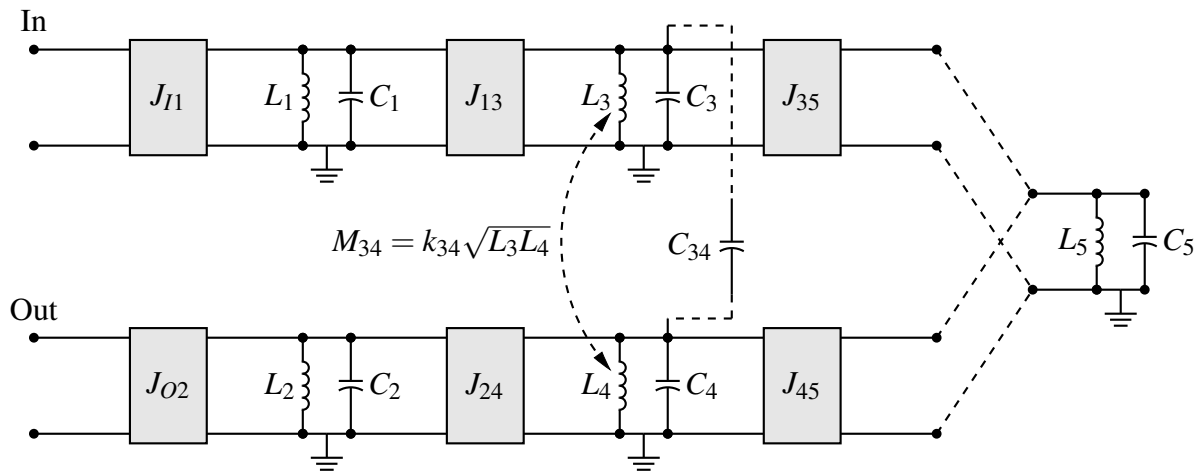


Figure 3.2: A Canonical Filter Circuit with 5 resonators and 2 transmission zeros

The canonical circuit shown in fig. 3.2 is related to a filter of order 5 with a couple of transmission zeros but the underlying topology is much more general. In fact it may be used to design a filter of any order and with any number of transmission zeros (lower than the filter order) that are arbitrarily placed on the frequency axis. This topology is characterized by a perfect symmetry with respect to a reflection about an horizontal axis.

Plotting and Exporting Ideal Circuit Data

The frequency response of the ideal circuit can be computed and plotted using the commands *Edit/Analyses/Freq Domain Analysis* and *View/Plot/Freq Response/Ideal Filter*. The parameters which control the frequency domain analysis are similar to those shown in fig 2.21 with the only difference that the circuit selector must be switched to the entry named *Ideal Filter*.

The figure 3.3 shows the frequency response of the ideal circuit defined by the input parameters reported in figure 3.1.

The ideal circuit may be exported in the *Spice* format or in a native *EmCAD* format (named *JC* format) by entering the commands *File/Export/Ideal Spice Circuit* and *File/Export/Ideal JC Circuit*.

The *Spice* circuit is slightly different than that one depicted in the figure 3.2 because the J inverters and the mutual capacitances are not available in a *Spice* model. The first and are replaced by *Girators* while the latter are replaced by shunt capacitors with a correction of the resonator capacitors. The electrical response is not affected by these changes.

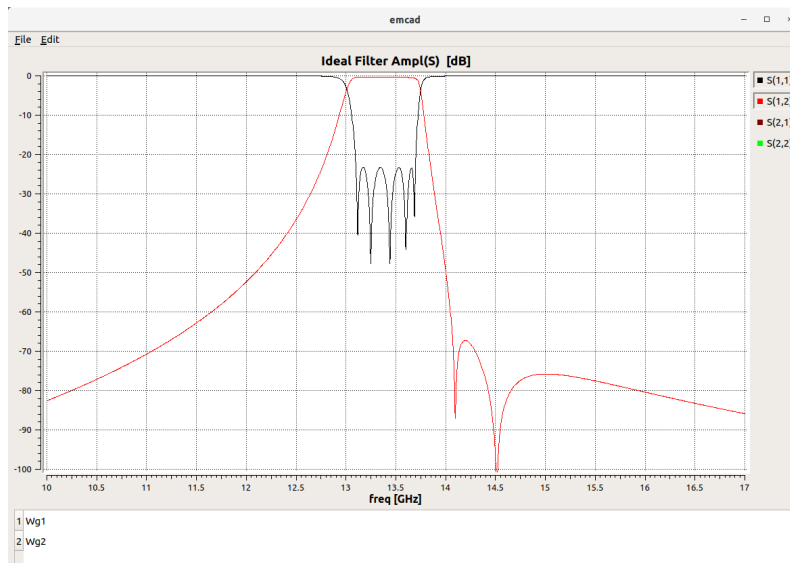
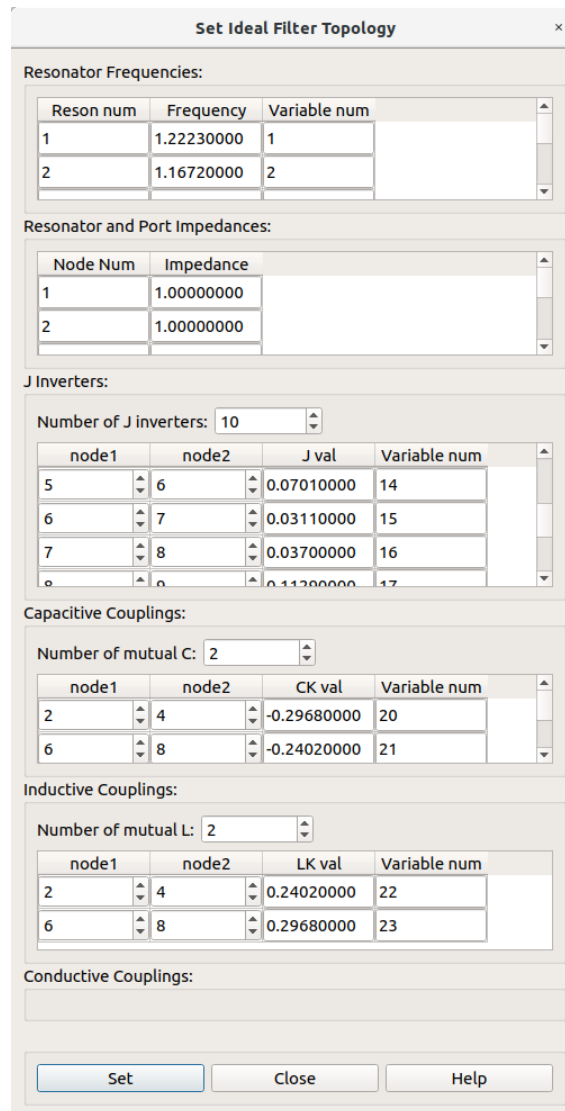


Figure 3.3: Frequency Response of the Ideal Circuit

The JC circuit also differs from the circuit depicted in the figure 3.2 because the inductances and mutual inductances are not available in the JC format which is composed of J inverters, discrete capacitors, mutual capacitances and conductances. In the JC format the inductances of the circuit 3.2 are realized by a capacitors interfaced by unitary J inverters. The mutual inductances are replaced by mutual capacitances acting across the inverted capacitors.

Custom Filter Topology

While being quite general, in many applications the canonical ideal circuit is not a good reference model for the microwave implementation. The most convenient circuit topology depends on the selected resonator technology and it is not possible to base all filter designs on a set of predefined circuit topologies. The EmCAD environment gives to the designer the possibility to specify a custom circuit topology which is composed of parallel LC resonators interconnected with user defined coupling elements. The designer may choose among three types of coupling elements: J inverters, mutual inductance and mutual capacitances.



Set Ideal Filter Topology

Resonator Frequencies:

Reson num	Frequency	Variable num
1	1.22230000	1
2	1.16720000	2

Resonator and Port Impedances:

Node Num	Impedance
1	1.00000000
2	1.00000000

J Inverters:

Number of J inverters: 10

node1	node2	J val	Variable num
5	6	0.07010000	14
6	7	0.03110000	15
7	8	0.03700000	16
8	9	0.11200000	17

Capacitive Couplings:

Number of mutual C: 2

node1	node2	CK val	Variable num
2	4	-0.29680000	20
6	8	-0.24020000	21

Inductive Couplings:

Number of mutual L: 2

node1	node2	LK val	Variable num
2	4	0.24020000	22
6	8	0.29680000	23

Conductive Couplings:

Buttons: Set, Close, Help

Figure 3.4: A custom filter of order 9 with 2 capacitive and 2 inductive extra couplings

The graphic window shown in figure 3.4 is activated by the “Set Ideal Circuit“ button shown in figure 3.1. In this windows it is possible to define the resonant frequencies and impedances associated with all resonators and the number and values of coupling elements acting among them.

The numerical values of J inverters entered in window 3.1 are expressed in unit $[1/\Omega]$.

The inductive and capacitive couplings are specified by the adimensional inductive and capacitive coupling coefficients. These are expressed in terms of mutual inductance and mutual capacitances through following formulas:

$$K_{ij}^h = \frac{L_{ij}}{\sqrt{L_i} \sqrt{L_j}} \quad K_{ij}^e = \frac{C_{ij}}{\sqrt{C_i} \sqrt{C_j}}$$

where L_i , C_i are the self inductance and capacity and of the i th resonator and L_{ij} , C_{ij} are the mutual inductance and capacitance acting between the L_i , L_j and, respectively between C_i , C_j .

The values of the resonator frequencies and of the coupling parameters entered in window 3.1 are just the starting point for a fast optimization procedure which is executed when the user presses the “Start“ button shown in figure 3.1. The exact values of these parameters may not be known in advance but the starting values should not be too far from the solution otherwise the optimization procedure may fail to converge. It is also up to the designer to ascertain that the selected circuit topology has the capability to implement the prescribed filtering function (and the related zero/pole pattern).

A good starting point of the custom ideal circuit can be defined by assembling parts extracted from canonical circuits of the type shown in fig 3.2. To better clarify this point we may refer to the 9th order circuit topology specified in figure 3.4. This circuit includes the cascade of two triplets which are responsible, respectively, for the generation of a couple of transmission zeros placed at the right side and a couple of transmission zeros placed at the left of the filter passband.

An initial estimation of the coupling parameters of the first triplet may be computed through the design of a canonical filter of order 5 which is characterized by a couple of zeros at the right side of the filter passband. Such a filter has the circuit topology shown in figure 3.2 and its electrical response is similar to that one reported in 3.3. The central frequency and the two transmission zeros specified in the design of this smaller filter should coincide with the central frequency and the two right hand side transmission zeros specified in the design of the 9 order filter. Under this hypothesis the triplet we are looking for can be identified with the triplet composed of the resonators 3, 4 and 5 included in circuit 3.2.

The same procedure can be used also for the computation of the initial parameters of the other triplet (that one responsible for the generation of the two transmission zeros placed at the left side of the filter passband).

It should be noted that there are other possible ways to distribute the four transmission zeros between the two triplets included in the custom circuit topology. In example it could have also been possible to assign to the first triplet two zeros symmetrically placed at both sides of the filter passband. Unfortunately the transversal mutual capacitance and inductance associated with this kind of triplet are not easy to implement in the coaxial technology because of their “wrong“ signs. The transversal couplings associated with a triplet characterized by two transmission zeros placed at the same side of the filter passband are instead easy to implement when the associated coaxial resonators are equally-oriented, a configuration that is typical of comb-line filters.

3.1.2 An example of Filter Implementation

The objective of the filter design is the definition of a microwave structure which implements the ideal filter in the selected resonator technology (waveguide, coaxial, stripline, ...).

The ideal circuit shown in figure 3.2 is very well suited to be implemented in coaxial technology with the layout shown in figure 3.5. Two adjacent resonators which, in the ideal circuit, are connected by a J inverter correspond, in the physical implementation, with two adjacent coaxial resonators characterized by an opposite (up-down) orientation. The electromagnetic coupling associated with this “interdigitated“ configuration is well approximated by an ideal J inverter.

Two adjacent resonators which, in the ideal circuit, are subjected to a mixed inductive/capacitive coupling correspond, in the physical implementation, with two adjacent coaxial resonators characterized by an equal orientation. The electromagnetic coupling associated with this “comblined“ configuration is well approximated by a “positive“ mutual inductance and a negative “mutual capacitance“ (which becomes a positive shunt capacitor in the *Spice* model).

The experimental results have shown that the sign of transversal couplings associated with the ideal circuit shown in figure 3.2 correspond with those associated with the physical implementation shown in figure 3.5 when the two transmission zeros are set on the same side with respect to the passband. The physical model shown in figure 3.5 may therefore be used to implement a filter with a couple of transmission zeros placed at the left side of the passband or a filter with a couple of transmission zeros placed at the right side of the passband. It may not be used to implement a filter with a couple of transmission zeros symmetrically placed below and above the passband.

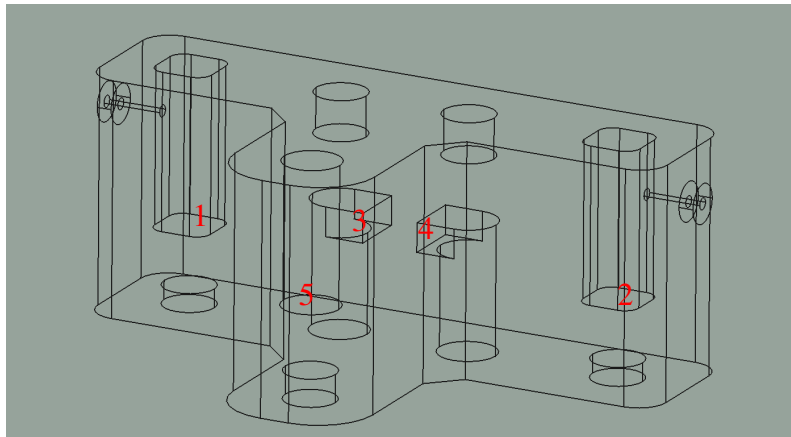


Figure 3.5: Coaxial Realization

In order to allow a complete control of the electrical parameters associated with the ideal circuit shown in figure 3.2 the physical implementation must allow a separate regulation of the capacitive and of the inductive parameters associated with the transversal couplings. For the coaxial configuration shown in figure 3.5 these regulations are actuated by changing the distance between resonators 3 and 4 and the gap between the heads of the same resonators.

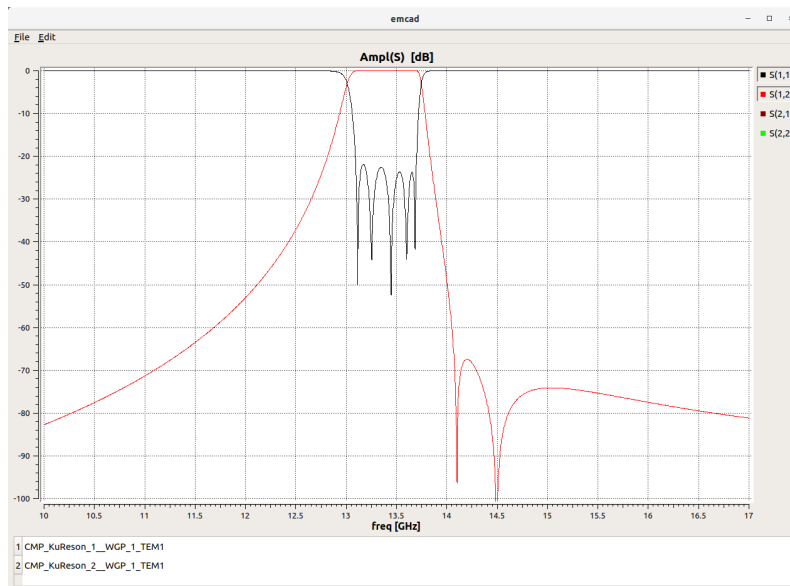


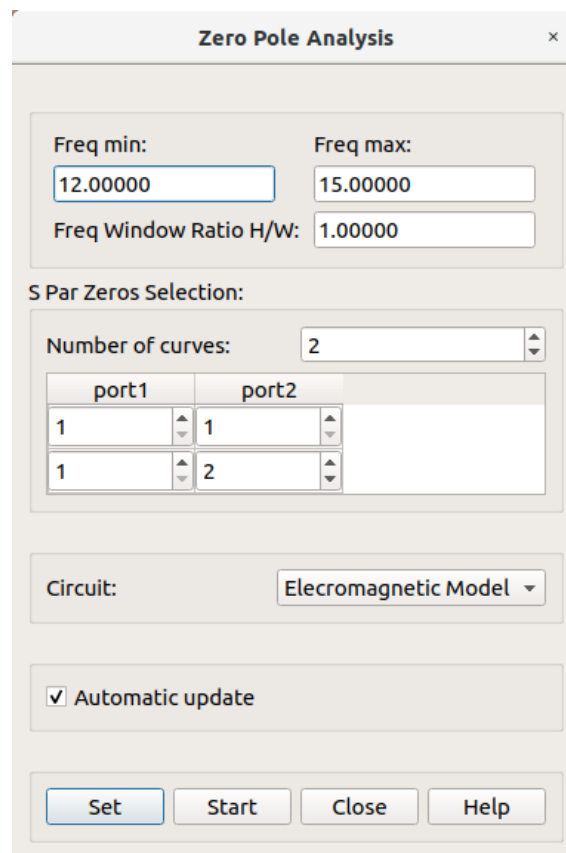
Figure 3.6: Frequency Response of the Coaxial Filter

This kind of filter may be optimized in an efficient way using the tuner tool provided by the *EmCAD* environment. The figure 3.6 shows the electromagnetic response of a coaxial filter which was designed and tuned to meet the requirements specified in figure 3.1.

3.1.3 Zero Pole Analysis

The Zero Pole Analysis is a unique capability of the *Hierarchical Electromagnetic Modeler* which differentiates it from all other full wave electromagnetic solvers. This function is activated by the command *Edit/Analyses/S Parameter Zero/Poles* which opens the window shown in figure 3.7 for the specification of the input parameters.

The computed Zeros and Poles pattern can then be visualized over the Laplace Plane using the command *View/Plots/Zeros and Poles*. The Laplace Plane is rotated by 90 degree with respect to its usual orientation so that the frequency axis is horizontal and the left side of Laplace plane ($Re(s) < 0$) is mapped to the lower side of the plot. The computed zeros and poles are filtered by selecting only those lying inside of a rectangular window in the plot plane.



Zero Pole Analysis ×

Freq min: Freq max:
 Freq Window Ratio H/W:

S Par Zeros Selection:

Number of curves:

port1	port2
<input type="text" value="1"/>	<input type="text" value="1"/>
<input type="text" value="1"/>	<input type="text" value="2"/>

Circuit:

Automatic update

Figure 3.7: Input Parameters

The input parameters which define the Zero/Pole analysis are:

- *Frequency Min and Frequency Max* These parameters specify the horizontal (frequency) span associated with the plot window.
- *Frequency Window Ratio H/W* These parameters specify the ratio between the height and width of the plot window. The vertical span (which corresponds to the horizontal span in the Laplace plane) is symmetrical about the frequency axis ($j\omega$ axis in the Laplace plane).
- *S Parameter Selection* This window section specifies the scattering parameters for which the zeros are requested. Each parameter is defined by a couple of indices

- (1, 1) for $S_{1,1}$ (1, 2) for $S_{1,2}$... The poles are the same for all S parameters and are always computed.
- *Circuit* This parameters specifies the circuit subjected to the Zero/Pole computation. Possible choices are "Electromagnetic Model", "Mapped Circuit", "Ideal Circuit".

The figure 3.8 shows the zero pole plot associated with a coaxial filter of order 5 which was designed according with electrical specifications reported in fig. 3.1. This plot includes five reflection zeros (zeros of S_{11} parameter), two transmission zeros (zeros of S_{12}) and the five poles of the scattering matrix. Being related with a passive component the poles must lay on the negative side of the Laplace plane which corresponds with the lower side of the plot plane.

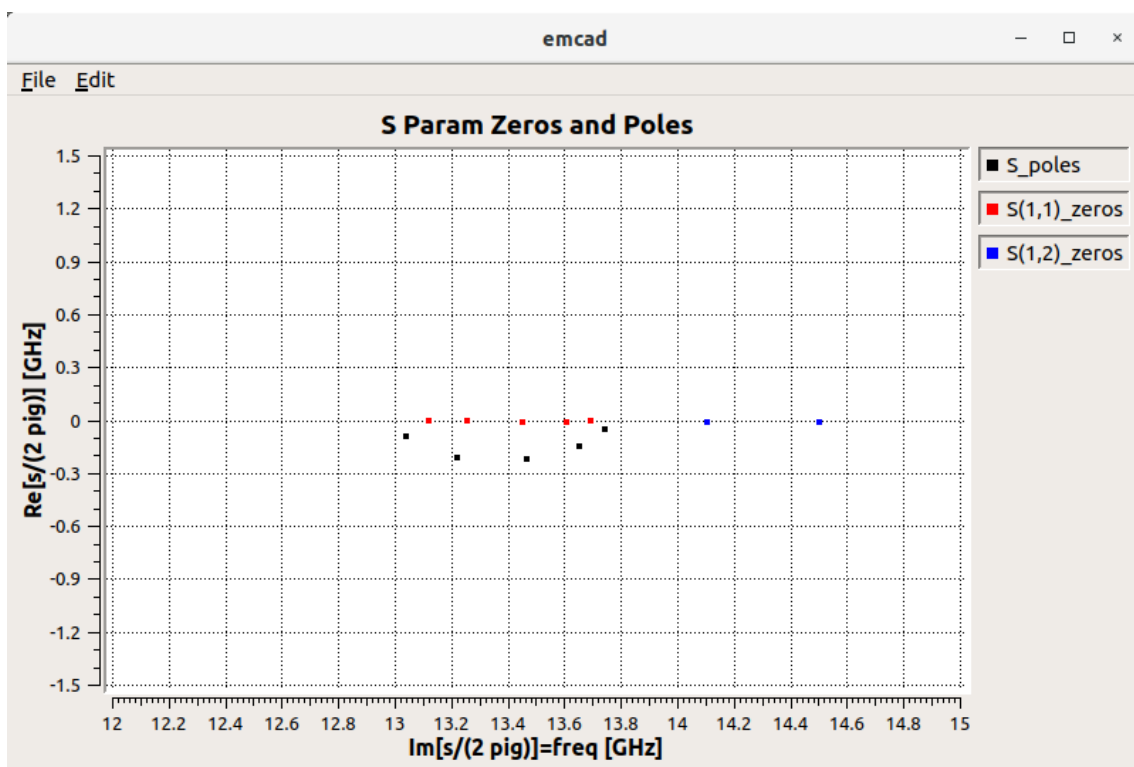


Figure 3.8: Zero Pole Analysis

The arrangement of the zeros and poles shown in figure 3.8 (with all zeros on the frequency axis) is characteristic of the well tuned filter. The Zero/Pole analysis is a very powerful tool which may be exploited since the initial phase of the filter design. In a typical design flow, after having selected and synthesized an ideal circuit, which meets the given requirement, the designer has to look for an electromagnetic structure which implements that circuit. Initially the resonator geometries and the inter-resonator distances (or the sizes of inter-resonator apertures) are dimensioned with approximate formulas in order to assure that the resonator frequencies and coupling coefficients are close to the ideal values. It is quite common that the electrical response associated with this initial geometry is quite distant from the design goal so that, in some cases, it may even be difficult to recognize the electromagnetic structure as a potential implementation of the ideal circuit.

In this situation the zero/pole analysis may be very helpful because an important indication that the selected geometry is well suited to implement the ideal circuit may

come from the verification that the number of its reflection and transmission zeros match with those associated with the ideal circuit.

3.1.4 Filter Mapping

The pattern of zeros computed by the *Zero/Pole Analysis* is the input data for the synthesis of a new circuit (named *Mapped Circuit*) which is characterized by the electrical response of the electromagnetic model and the topology of the ideal circuit.

The *Mapped Circuit* is computed with the command *Edit/Analyses/Filter Mapping* which is inactive until the zero patterns have been computed. The *Mapped Circuit* may be exported in the *Spice* format or in the *EmCAD* format (which is named *JC* format) with the commands *File/Export/Mapped Spice Circuit* and respectively *File/Export/Mapped JC Circuit*.

3.1.5 Filter Tuning

The tuning procedure is based on the comparison between the numerical values of the electrical parameters associated with the *Ideal Circuit* and the *Mapped Circuit*.

The number of geometrical variables which are optimized by the tuning procedure must correspond with the number of degree of freedoms associated with the ideal circuit. In example the number of degree of freedom associated with a generic “in line“ filter of order N is equal to $2 * N + 1$. In fact this filter is uniquely defined by the N resonant frequencies and $N + 1$ coupling coefficients which include also the input/output couplings.

For a normal (not predistorted design), the *EmCAD* procedure reported in section 3.1 generates a symmetric circuit. This symmetry may be exploited to reduce dimension of the problem dealt by the tuning wizard. This reduction (by a factor of 2) is enforced by checking the flag named *Symmetric Tuning* in the window shown in figure 3.1. The same reduction should be applied also to In order to assure the balance between the number of independent parameters associated with the circuit and with the geometrical structure, the symmetry constraint must be applied also to the latter.

The objective of the optimization process is the minimization of the norm of the difference between the parameters associated with the mapped and ideal circuits. This process makes use of a Jacobian matrix which is defined by the derivatives of components of this error vector versus the variations of the geometrical variables. The Jacobian is a square matrix because the number of unknown circuit parameters is equal to the number of the geometrical variables. With a proper choice of the geometrical variables, the Jacobian matrix should be invertible so that it is possible to compute a variation of the geometrical variables which, in a perfectly linear model, should be able to nullify the error vector.

Due to the linear approximation involved in the Jacobian matrix, the correction of the geometrical variables is only approximate. Usually, in order to achieve an error that is small enough to meet the filter requirements, it is necessary to iterate this process a few times.

In each iteration the *tuner wizard* updates the error vector, computes a correction of the Jacobian matrix using the *Broyden method* and applies a new variation to the geometrical variables. At this point the user must perform an update of the CAD document, save the new geometry in a step file, reload this file in the *EmCAD* environment and invoke the *EmCAD* command *Edit/Update*. When this command is completed the wizard automatically recomputes the error vector starting a new iteration.

The user is informed that the geometrical variables are ready for a new iteration and has the possibility to inspect the error vector and the frequency response associated with the previous iteration. If these data are good enough the user may interrupt the iterative process pressing the "break" command. Otherwise he may proceed with a new update of the CAD document and a new electromagnetic modelization.

The optimization process starts with the computation of the initial error vector and the Jacobian matrix. Being N the number of variables, this initial phase requires the modelization of $N + 1$ different geometrical configurations (the initial configuration and the N configuration generated by a small variation of each geometrical variable). These geometrical variations are automatically applied to the file which define the geometrical variables but the actual update of the CAD document and of the *EmCAD* models is left to the user.

The parameters of the tuning process are defined in the window shown in figure 3.9

which is opened by the command *Edit/Design/Filter Tuning*.

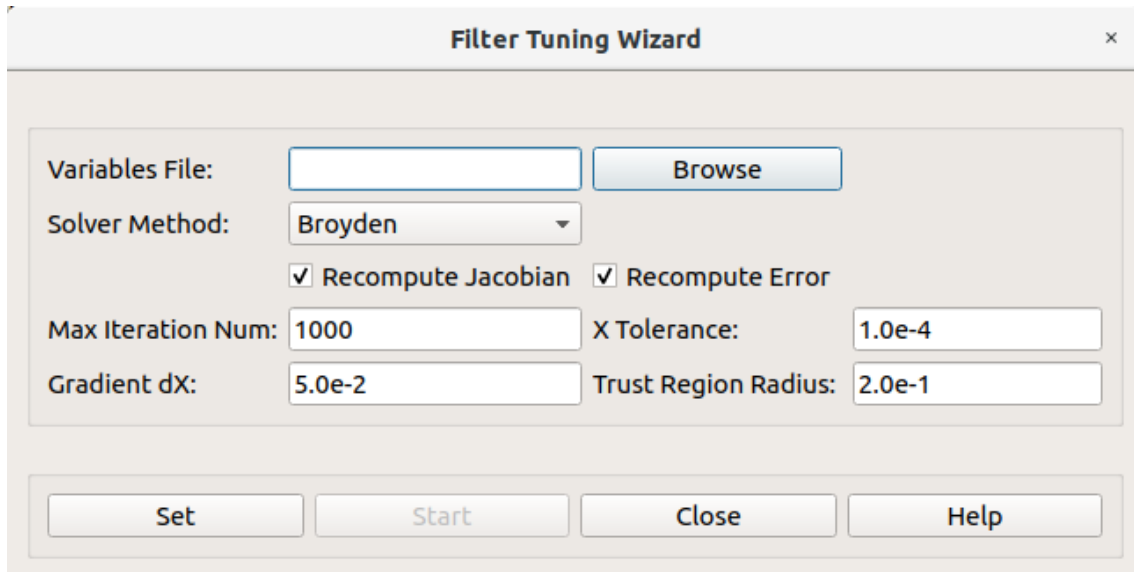


Figure 3.9: Filter Tuning Wizard Parameters

The input parameters are:

– *Variables File*

This text entry specifies the path of a file which controls the geometrical variables associated with the CAD document. Currently ZW3D and Catia V5 are the only CAD system supported by the *Tuning Wizard*. If the geometry was defined inside of ZW3D the geometrical variables should be stored in an Excel file (with extension “.xlsx”) that was initially exported by ZW3D. The “.xlsx” file exported by ZW3D includes all the parameters defined in the CAD system and, in order to distinguish the tunable parameters from other CAD parameters, the former must be enclosed in a special Equation Set named “Variables”. The tuning procedure rewrites the “.xlsx” file but only the tunable parameters are changed. In order to update the CAD structure the user needs to import the modified “.xlsx” file and explicitly accept the variation for all the reimported parameters. If the geometry was defined inside of Catia V5 the relevant variables may be defined in a text file according to the text format format exported by the Catia Design Table.

– *Solver Method*

The solver method may not be changed. The only option is "Broyden" which refers to a rank one update of the Jacobian based on the "Broyden Method". The selector of the Solver Method was introduced to experiment with other methods but it is deactivated in the current release.

– *Recompute Jacobian and Recompute Error*

These flags are useful when the tuner is started for a second time on a filter design that was partially tuned in a previous run. In this case the *Jacobian Matrix* and the *Error Vector* are already saved in the *EmCAD* document and, by checking these two flags, the user decides to use the stored data without recompute them. Some time it may happen that, after a certain number of iteration, the stored *Jacobian Matrix* becomes inaccurate because of the approximations associated with the *Broyden*

update. If this error is too large the iterative solver may lose its convergence and then it may be safe to restart the solver asking for a recomputation of the Jacobian Matrix.

– *Max Iteration Number*

This parameter sets the maximum number of iteration but it is not very useful in the current semi-automatic implementation of the Tuning Wizard. The default value is set to a high number (1000) because the tuning process should be terminated, through an explicit user request, before this number is reached. This parameter will play a more important role in a planned future evolution towards a fully automated tuning procedure.

– *X tolerance*

Acting as a secondary stopping criteria (beside of the Max Iteration Number), the tuning process is terminated when the X tolerance is greater than the norm of the next variation of the geometrical variables.

– *Gradient dX*

Amplitude of variables variations for the numerical computation of the Jacobian.

– *Trust Region Radius*

This is the radius of a trust region centered at the actual point in the space of geometrical variables. The variation of the geometrical variables is reduced by a numerical factor so that it preserves its direction but its norm does not exceed the specified *Trust Radius*. Due to the linear approximation expressed by the Jacobian Matrix, the computed correction of the geometrical variables may be effective, in that it brings an improvement of the error vector, only if it is not too big.

– *Set*

When the *Set* button is pressed the tuner wizard makes a few checks to verify the format of the control file, that the ideal circuit (target of the tuning process) is properly defined and that the number of its unknown parameters corresponds with the number of geometric variables. If all these checks are passed the *Start* button becomes active.

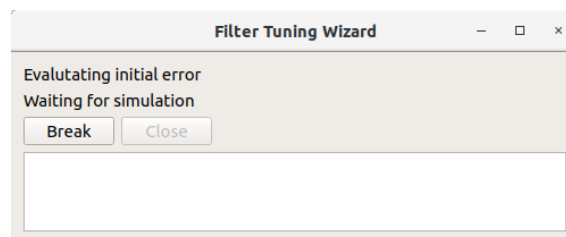


Figure 3.10: Filter Tuning Wizard Computing Initial Error

The actual tuning process is started by pressing the *Start* button in the window 3.9. This action opens the window shown in figure 3.10 which informs the user that the wizard is waiting for the computation of the initial error.

At this point the user should invoke an *EmCAD* update. When this operation is completed the *tuner wizard* evaluates and displays the components of the error vector as shown in figure 3.11. The first line of this window informs that the tuner is waiting for data related with the error derivatives versus the first geometrical variable. At this point the user should perform the following actions:

3.1 Filter Design

- Updating of the CAD geometry
- Exporting of the step file from CAD system
- Reloading the step file into the EmCAD environment
- Updating the CAD models

When the last operation is completed the tuner wizard displays the first row of the *Jacobian Matrix* as shown in figure 3.12. The first line of this window informs that the tuner is waiting for data related with the error derivatives versus the second geometrical variable. At this point the user should repeat the same actions already done for the computation of the derivatives versus the first variable.

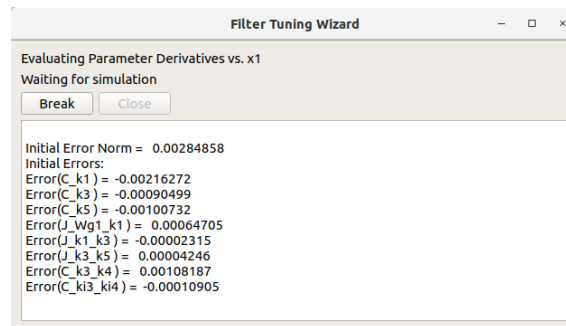


Figure 3.11: Filter Tuning Wizard Computing First Row of Jacobian Matrix

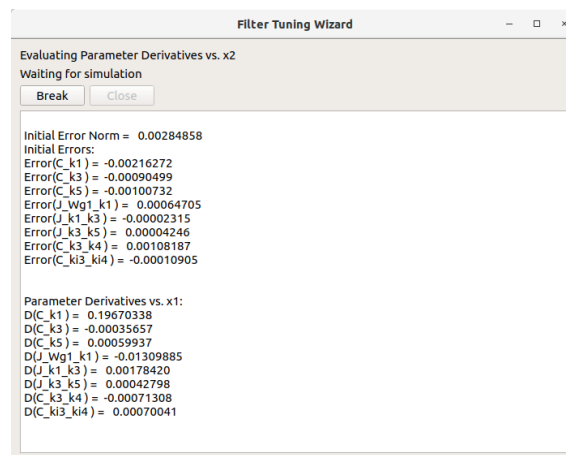


Figure 3.12: Filter Tuning Wizard Computing Second Row of Jacobian Matrix

When the wizard has completed the computation of all the Jacobian rows, it performs a first correction of the geometrical variables and the user is informed (with messages similar to those reported in figure 3.13) that the first iteration of the geometrical tuning has been started.

At this point the user should repeat the same actions already done for the computation of a jacobian column. The iterative geometrical tuning goes on until the user presses the *Break* button. When this action the geometrical variables, which were updated in view of a possible new iteration, are reverted back to realign them with the electrical model generated in the last run.

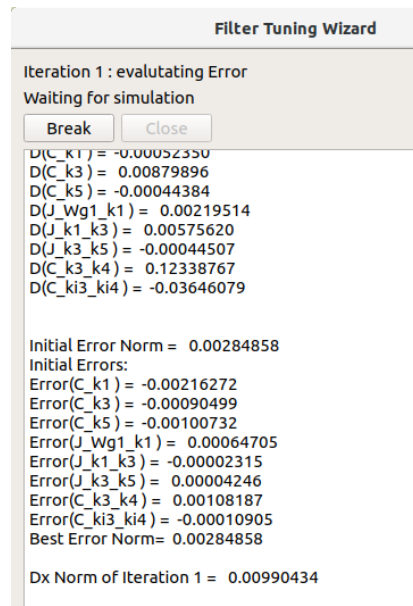


Figure 3.13: Filter Tuning Wizard Evaluation of errors in the first iteration



Bibliography

- [1] J. Eric Bracken. “A Causal Huray Model for Surface Roughness”. In: (2012) (cited on page 26).
- [2] Christophe Geuzaine and Jean-François Remacle. “Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities”. In: (2008) (cited on page 36).
- [3] Marina Y. Koledintseva, Konstantin N. Rozanov, Antonio Orlandi, and James L. Drewniak. “Extraction of Lorentzian and Debye parameters of dielectric and magnetic dispersive materials for FDTD modeling”. In: 53.9 (2002) (cited on page 24).