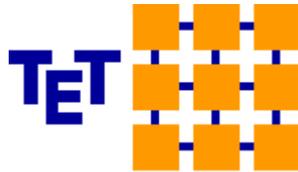


# CONCEPT-II Version 12.0 User's Manual

Institut für  
Theoretische Elektrotechnik



[www.tet.tuhh.de](http://www.tet.tuhh.de)

Dr.-Ing. Heinz-D. Brüns, Dipl.-Ing. Angela Freiberg

Copyright: Institut für Theoretische Elektrotechnik, 2018

**TUHH**  
*Technische Universität Hamburg-Harburg*

- December 2018 -

# Manual of the Program System

|            |   |    |
|------------|---|----|
| 1          | General description .....   | 4  |
| 1.1        | Supported platforms and the code capabilities .....                         | 4  |
| 1.2        | Possible excitations .....  | 4  |
| 1.3        | Post processing .....   | 5  |
| 1.4        | Installation .....  | 5  |
| 1.5        | Covered frequency interval, limits .....                                    | 5  |
| 1.6        | Importing of external surface patch data .....                              | 6  |
| 1.7        | Rules for discretizing structures .....                                     | 7  |
| 1.8        | Validation .....  | 11 |
| 1.9        | Three essential steps for solving a problem .....                           | 12 |
| 1.10       | Some important file names .....   | 12 |
| 1.11       | Performing a typical simulation .....                                       | 13 |
| 1.11.1     | Overview (main window) .....  | 18 |
| 1.11.1.1   | Icons of the <i>tool bar</i> .....  | 19 |
| 1.11.2     | The simulation project tree .....   | 20 |
| 1.11.2.1   | Handling of the simulation project tree .....                               | 21 |
| 1.11.2.2   | Dielectric bodies .....   | 22 |
| 1.11.2.3   | Wires .....   | 27 |
| 1.11.2.4   | Loads .....   | 29 |
| 1.11.2.5   | Frequency sampling .....  | 31 |
| 1.11.2.6   | Excitation .....  | 32 |
| 1.11.2.6.1 | Voltage generators .....  | 32 |
| 1.11.2.6.2 | Ports .....   | 33 |
| 1.11.2.6.3 | Plane wave field excitation .....   | 36 |
| 1.11.2.6.4 | Impressed wire currents .....   | 39 |
| 1.11.2.6.5 | Application of the Surface Equivalence Theorem: Huygens Principle .....     | 40 |
| 1.11.2.7   | Solvers .....   | 42 |
| 1.11.2.8   | Precision of the numerical solution .....                                   | 46 |
| 1.11.2.9   | Post processing tools .....   | 48 |
| 2          | Example: Top-loaded wire antenna on a table structure .....                 | 61 |
| 2.1.1      | Post processing .....   | 68 |
| 2.1.1.1    | Representation of the current distribution .....                            | 68 |
| 2.1.1.2    | Representation of electromagnetic 2D field distributions .....              | 69 |
| 2.1.1.3    | EM fields along straight paths or at arbitrary locations in space .....     | 71 |
| 2.1.1.4    | Some features of the Gnuplot front end .....                                | 75 |
| 2.1.1.5    | Creating 2D radiation diagrams .....  | 75 |
| 2.1.1.6    | Creating 3D radiation diagrams .....  | 79 |
| 2.1.1.7    | Creating Smith diagrams .....   | 82 |
| 3          | Important topics .....  | 83 |
| 3.1        | Dielectric bodies, types of surfaces .....                                  | 83 |
| 3.1.1      | Bodies containing high conducting material .....                            | 84 |
| 3.1.2      | Multi-dielectric structures with symmetry .....                             | 86 |
| 3.1.3      | Bodies inside bodies .....  | 87 |
| 3.1.4      | Wires in dielectric bodies, patch-wire junctions .....                      | 89 |
| 3.1.5      | Layers on bodies .....  | 90 |
| 3.1.5.1    | Example of a cylinder with a lossy wall of high relative permeability ..... | 92 |
| 3.2        | N-ports, S-parameters .....   | 93 |
| 3.2.1      | Files .....   | 95 |
| 3.2.2      | Computation of S,Y,Z-parameters .....                                       | 96 |



|  |     |
|--|-----|
| 3.2.3 Using <i>red-y-matrix.h5</i> with the network tool <i>CONCIRC</i> .....                  | 97  |
| 3.2.4 EM fields due to a modified port excitation .....  | 99  |
| 3.3 RCS .....  | 101 |
| 3.4 Computation of line integrals of E and H .....   | 104 |
| 3.4.1 Example: Line integral to measure the total current .....                                | 107 |
| 3.5 Integrating the Poynting vector on a triangulated surface .....                            | 108 |
| 3.5.1 Example: Loaded circular wire loop .....   | 109 |
| 4 Discretization of structures .....   | 111 |
| 4.1 Spheres, plates, cones, cylinders etc. ....  | 111 |
| 4.2 Treatment of wires .....   | 119 |
| 4.3 Generating triangular and quadrangular plates .....  | 121 |
| 4.4 Using the boundary tool .....  | 124 |
| 4.5 Turn a wire contour model into a surface patch model .....                                 | 127 |
| 4.6 The point option for meshing .....   | 129 |
| 4.7 Joining surface parts separated by unwanted gaps .....                                     | 132 |
| 4.8 Importing and exporting .....  | 134 |
| 4.8.1 Basic example for data conversion and application of <i>gmsh</i> .....                   | 135 |
| 4.9 Converting the wire format into the <i>.geo</i> format of <i>gmsh</i> and vice versa ..... | 138 |
| 4.10 Using the CONCEPT-II patch format for mesh generation in <i>gmsh</i> .....                | 140 |
| 4.11 Visualizing field and current distributions by means of <i>gmsh</i> .....                 | 143 |
| 5 Time domain system responses by an IFT .....   | 143 |
| 5.1 The necessary steps in short .....   | 144 |
| 5.2 Key points for performing an IFT .....   | 146 |
| 5.3 Frequency sampling for an IFT .....  | 156 |
| 5.4 Special aspects: Impressed currents and lightning currents .....                           | 158 |
| 6 Frequently asked questions .....   | 161 |
| 7 Using CONCEPT-II for parallel computation .....  | 164 |
| 7.1 Linux .....  | 164 |
| 7.2 Windows .....  | 165 |
| 8 Format of surface patch files and wire files, special topics .....                           | 165 |
| 8.1 Arrow scaling, <i>xtrans</i> , <i>ytrans</i> .....   | 167 |
| 8.2 Backup of project directories .....  | 168 |
| 8.3 Known problems .....   | 168 |
| 9 Theoretical background (method of moments, EFIE, MFIE) .....                                 | 169 |
| 10 Licenses .....  | 177 |



## 1 General description

*CONCEPT-II* is under development at the Institut für Theoretische Elektrotechnik at the Technische Universität Hamburg (Hamburg University of Technology, TUHH). It is a program package for the numerical computation of electromagnetic radiation and scattering problems in the frequency domain. The code is based on the Method of Moments (MoM) for solving the electric integral equation (EFIE) in the case of metal structures and a set of coupled integral equations in the case of (lossy) dielectric bodies, see Sec. 9 . It is designed for the computation of currents, voltages and fields, which result from the electromagnetic excitation of wire or surface structures. The application of the package is limited to linear materials which have to be homogeneous and isotropic in regions.

### 1.1 Supported platforms and the code capabilities

PC running under Linux and Windows.

The code capabilities include the treatment of:

- Lumped circuit elements at any location on wires and metallic surfaces
- Arbitrary metallic surfaces, which are simulated by the electric field integral equation (EFIE, single-layer current distribution)
- Surface impedance
- Thin lossy sheets on ideally conducting surfaces (not yet described in this manual, thin sheet approximation, TSA)
- Anisotropic layer material (CFC, not yet described in this manual)
- Lossy dielectric bodies modeled by double layers of electric currents
- Bodies covered by thin layers of finite conductivity to analyze shielding problems
- Structures in free space and over an ideally conducting ground plane
- Various magnetic symmetry planes
- Back end and all post processing tools are parallelized
- Direct H matrix solver, iterative H matrix solver (multiprocessor usage)
- MLFMA for perfectly conducting (PEC) structures (multiprocessor usage possible)
- Running a frequency loop with multi-processor usage
- Application of Huygens principle

### 1.2 Possible excitations

- Voltage generators and power generators (ports) on wires, edge generators on metallic surfaces
- Impressed line current source (current generator)
- Plane-wave fields of arbitrary polarization and direction of incidence
- Impressed electric and magnetic currents (Huygens sources)



- Stochastic excitation

### 1.3 Post processing

Once the matrix equation resulting from the application of the MoM has been set up and solved, the current distribution is known. Based on this the most important features are:

- Computation of 3D and 2D radiation diagrams
- RCS (monostatic, bistatic)
- Display current distributions on surfaces and wires
- Display quantities such as I,U,E,H as a function of frequency
- Time domain system responses by an inverse Fourier transform (IFT)
- 2D and 1D field distributions
- Computation of electromagnetic fields at arbitrary positions in space

For a successful application of the code the user should be familiar with the basic theory. A short outline is given in Section 9.

### 1.4 Installation

A document available on the *CONCEPT-II* download web page contains details about the installation of the package.

### 1.5 Covered frequency interval, limits

The conventional MoM provides a dense complex system matrix. This requires  $O(N^2)$  to store the matrix elements and  $O(N^3)$  to solve the equation system by LU decomposition for example, with  $N$  as the number of basis functions used to approximate the physical current distribution. The investigation of electrically large structures may become uneconomic using the LU decomposition, at first depending on the available hardware resources.

Each common edge between two neighboring surface elements (patches, triangles or quadrangles) leads to one unknown in the case of the EFIE (metal, single-layer current distribution) or to two unknowns in the case of a dielectric body (double-layer current distribution), respectively. The total number of unknowns  $N$  depends on the electrical size and on the geometrical complexity ("overall grid") of the structure under investigation.

An important rule of thumb is that 8....10 segments or patches per wavelength are necessary on electrically large structure parts, leading to an increasing number of unknowns at higher frequencies. For electrically small structures this rule does of course not apply.

The core memory to hold the system matrix can be determined as follows:



$$N \cdot N * 16 / 1024^3 \quad (\text{LU decomposition})$$

Hence an important limit of the problem size to be treated is determined by the available core memory (RAM). Note that *CONCEPT-II* does not include an out-of-core solver anymore starting with version 12.0. The amount of in-core memory used by *CONCEPT-II* should stay smaller (70%,...,80%) than the memory being physically available. Additional memory is necessary, depending on the structure under investigation. For electrically large structures special solvers are available which have a reduced memory requirement (direct H matrix solver, iterative H matrix solver, MLFMA).

Examples (LU decomposition):

Number of unknowns  $N=1000$  (small problem) → 0.0149 Gbyte

Number of unknowns  $N=10.000$  (medium sized problem) → Memory necessary to hold the system matrix: 1.49 GByte

Number of unknowns  $N=100.000$  (large problem) → 149 GByte

The smallest frequency which is possible is in the range a 100 Hz, 1kHz, ..., 10 kHz depending on the geometrical size and the type of integral equation used. Metallic structures (EFIE) with a typical dimension in the range of 1 meter or larger can be computed even at lower frequencies.

Transient system responses can be gained by computing an appropriate number of samples in the frequency domain and performing a successive inverse Fourier transform, for details see Sec. "Time domain system responses by an IFT".

## 1.6 Importing of external surface patch data

*CONCEPT-II* includes tools for creating surface patch meshes and/or wire meshes. Surface elements may be planar triangles and/or (non-planar) arbitrary quadrangles.

For sophisticated meshing purposes external tools such as *gms* (<http://geuz.org/gmsh>) can be used for meshing of surface and wire structures.

Note: until Oct. 18 only the 'Version 2.x ASCII' of the mesh format (.msh) of *gms* was supported. Starting at Oct. 18 'Version 4.0 ASCII' is supported additionally (both triangular and quadrangular surface patch elements are possible).

*Gms* is distributed under the terms of the GPL. It is able to read CAD data formats such as IGES or STEP and is able to convert this data into the well known STL format or into the *gms*-type .msh format. The last mentioned files can be directly read by *CONCEPT-II*. STL files can be converted into the corresponding *CONCEPT-II* format. How to use *gms* in conjunction with *CONCEPT-II* is outlined in Sec. 4.8.



**Tip**

Formats like IGES (.iges files) or STEP (.step files) can be converted into the surface patch grids by *gmsh* (.msh files) and from the .msh format into the *CONCEPT-II* surface patch data format, see Section 8. .msh files can even be read directly into the simulation.

For usage of the *CONCEPT-II* import/export tool click on **Misc** → **Import/export** → the corresponding dialog window opens.

## 1.7 Rules for discretizing structures

**Important**

A proper mesh is essential for the success and the convergence of the numerical computation.

It is not sufficient that the structure under consideration is described 'somehow' by a surface patch pattern giving the right three-dimensional outline. Very important is a proper structure discretization on the basis of the electromagnetic requirements. In this context we have to distinguish between electrically large and electrically small objects.

Generally a wire is characterized by a straight line that extends between the start coordinates (Xstart, Ystart, Zstart) and the end coordinates (Xend, Yend, Zend).

This indicates the current reference direction, visualize by  (tool bar). Depending

on the number of basis functions a wire is subdivided into a number of segments.

Surfaces have to be subdivided into a number of surface elements ("patches").

- Wires and surfaces which are large in terms of the wavelength (electrically large) have to be discretized assuming 8,...,10 patches or wire segments per wavelength at least.
- The minimum number of basis functions should be three on electrically short wires especially when they are part of a wire grid. The minimum number of basis functions on electrically short monopole antennas should be 8,...,10.
- In case of electrically small objects or electrically small parts on objects the corresponding shape should be properly defined.

Example: sphere with 1 m in diameter,  $f = 1$  MHz →  $\lambda = 300$  m. Applying the high frequency rule gives a typical patch size of 30 m which is far too large for such a sphere, of course. For a rough model of the sphere we have to apply at least 10,...,12 patches around its center circumference to sufficiently keep the shape.



In the low frequency region the user should answer the following question:

“How many elements are necessary to get a good approximation of the charge distribution provided that only constant charge blocks are available on the elements (triangles, quadrangles or wire elements)”

- There is a rapid increase of surface charges when approaching (sharp) edges for example. Thus, if the electric field strength is of interest close to such locations an appropriate grid refinement has to be introduced there.
- For wires the well-known thin wire assumptions have to be fulfilled:

$$L \gg a, \quad a \ll \lambda$$

with “L” as the wire length, “a” as the radius and  $\lambda$  as the wavelength.

Parallel wires should not get too close to each other.

In Fig. 1.1 can be seen how the characteristic impedance of a single wire line is behaving, when approaching the ideal ground. The analytical solution refers to the red curve. The blue curves indicates the behavior of the characteristic impedance assuming a simple current filament on the wire axis. Starting at approximately  $r/h=0.3$  (radius over height) the error gets more and more pronounced. One can imagine that an image wire is running at below the ground plane.

Hence for an accurate solution the axial distance of two (or more) wires in parallel should not be closer than approximately 4 times the largest radius.

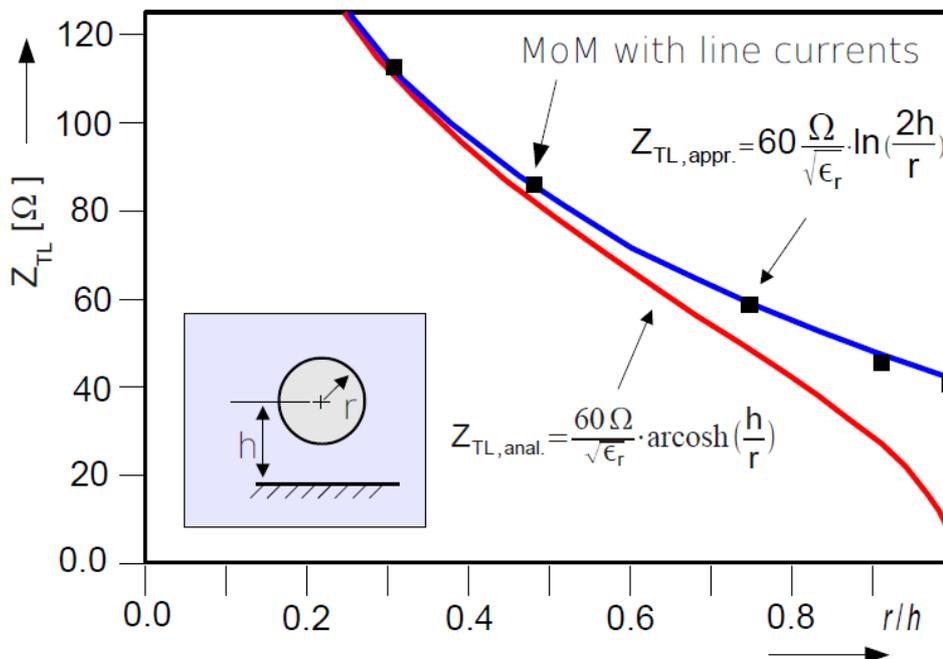


Fig 1.1: Characteristic impedance of a wire over a ground plane



- The ratio of the segment lengths of wires that are connected to each other should not exceed 10,...,15. Parallel wires should have the same number of segments.
- Tiny patches should not be attached to much larger ones. Patches with a big ratio of edge lengths should be avoided, see Fig 1.2 and Fig 1.4 demonstrating bad examples. 1 or 2 vertices are forming small angles.

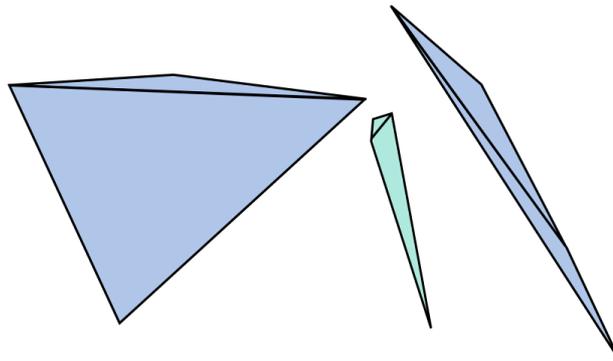


Fig 1.2: Examples of surface patch combinations that should be avoided

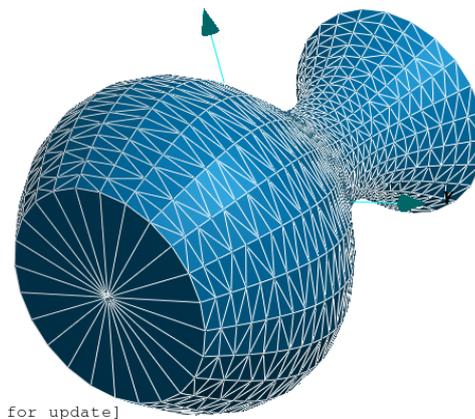


Fig 1.3: Example of a partly bad mesh. The discretization of this vase is good **except for the bottom**. Here we see individual triangles extending between the rim and the center of the bottom disk - at least 5 to 6 were necessary (also for very low frequencies)! Here the highest frequency possible is determined by the long edges of the triangular patches. Hence the grid at the bottom is insufficient for both high and for low frequencies.

In case of wires (lines, cables) running close to a discretized ground plate, the typical size of the patches underneath the wire should be smaller than one to two times the height of the wire. Otherwise the characteristic impedance of the wire cannot be captured.

In Fig 1.3 the bottom of the vase has been discretized according to pure visual



aspects – this an error frequently encountered in practice.

Fig 1.4 demonstrates a bad grid. We have:

- tiny triangles close to very large ones
- Triangles with the shape of needles
- An erroneous quadrangle (corner angle larger then 180 degree), quadrangles with one corner angle close to 180 degrees or a quadrangle with one corner angle of only a few degrees.



Fig 1.4: Example of a bad grid

## Note

A surface patch grid has to be generated always based on electromagnetic requirements not on visual aspects.

Simple examples:

- An electrically small disk with a wire antenna mounted in its center should have at least 5...6 patches between the center and the outer boundary. An isolated square plate should have at least 6 patches along each edge, even in case of a large wavelength.

- 
- Only triangulated surfaces should be used in case of patch antennas or in case of almost closed or closed cavities. For such structures best results are achieved under the following circumstances:

- Wires, especially monopole antennas, should be modeled as small triangulated patch strips (single sequence of triangles, width of a strip:  $4 \cdot (\text{wire radius})$ ). Hence the whole model should comprise a pure triangulated surface patch grid.

- Galerkin matching should be chosen

Example: `$CONCEPT/demo/example3-box-with-aperture`



Grid errors can be detected by a special tool: **CAD tools** → **CAD tools 2** → **Check surface structure** .

It is essential that the grid is connected as otherwise artificial gaps appear, surface currents cannot flow across these locations. Fig 1.5 shows an example of an erroneous grid.



Fig 1.5: The center nodes of both grids are not in contact with each other. This is a grid error as the current cannot pass such a line. Of course this is an unsuitable method for modeling small physically existing apertures! In such a case the typical size of patches close to a long narrow opening should not get larger than the width of that opening or aperture respectively.

## 1.8 Validation

Once the solution of the problem has been computed, the question arises whether the results are convergent, stable and reliable. Are the results confirming to the physical requirements? **Checking the credibility of numerical results is strongly recommended.**

The most important techniques:

- The inspection of the power budget (input power = radiated power + losses)
- Physical credibility of results (is the solution at least basically sound)
- Model variation and parameter variation and checking the results against each other
- Check of the boundary conditions (best done by creating vector plots of 2D field distributions. At dielectric interfaces we have  $H_{t1}=H_{t2}$ ,  $E_{t1}=E_{t2}$  , PEC:  $E_t=0$  )
- Convergence of computed quantities (no significant change in magnitude and phase when increasing the number of patches and wire segments, respectively)
- Comparison with analytical solutions
- Comparison with measured data



- Comparison with the results of other numerical methods

## 1.9 Three essential steps for solving a problem

The following steps are necessary for a numerical analysis:

1. Preprocessing (Discretization, frequency sampling, excitation, type of matrix solver...)

Set up a suitably discretized model of the structure under investigation. Use the tools of the *CONCEPT-II* CAD section: click on the **Cad tools** tab (see Fig 1.7, GUI, main window, left side). Once this has been accomplished additional data is needed: loads, type of excitation, kind of symmetry, field probes, frequency steps etc.. Click the **Simulation** tab and go through the entries of the simulation project tree.

2. Numerical computation (build up equation system and solve it, icon )

Setting up the equation system is proportional to  $N^2$  and solving it by LU decomposition (standard solver) is proportional to  $N^3$  with  $N$  as the number of unknowns.

3. Post processing (display and analyze the numerical results)  
Tab **Post processing**. Electromagnetic (EM) fields can be computed at arbitrary positions in space. Surface current distributions can be visualized, 2D or 3D radiation diagrams can be computed etc. .

## 1.10 Some important file names

The *CONCEPT-II* software package creates a number of ASCII and binary files in the current working directory. This following list gives a survey on the most important ones.

*concept.out* → Created by the back end. Contains a lot of results if the **Print** option (see simulation project tree) is active: current amplitudes, wire identifiers, number of unknowns, calculated system responses etc.. *concept.out* is an ASCII file (tab **Log data** → **CONCEPT-II simulation**)

*conerr.out* → Contains error messages and warnings.

*.in* files → Almost all executables of the *CONCEPT-II* package require input data files. Examples: *fieldcomp.in*, *radiate3D.in*, *freqloop.in* . When existing, the contents files is automatically loaded by the GUI.



*.out* files → Many executables of the *CONCEPT-II* package produce result files in ASCII format. Examples: *concept.out*, *radiate3D.out*

*surf.0* → Contains the overall surface patch data. Different surface patch files are combined. Do not use this name otherwise

*wire.0* → Contains the overall wire structure, do not use this name otherwise

### Tip

Simulation project tree: It is not allowed to use the file names *surf.0* and *wire.0* here

*confreq.in* → Contains the frequency samples (steps)

-----  
*surf.eh2d* → 2D EM field distribution, surface patch data format (see Sec. 8)

Post processing, click on , **Field probes** → **Field in rectangular 2D region**

-----  
*co\_cmovie.asc* → current vectors to be displayed on *surf.0*

*rad.n* → Data of 3D radiation diagrams (surface patch data format)

*red-y-mat.h5* → Reduced Y matrix / matrices in the case of multi-port excitation.

*freqi.asc*, *frequ.asc*, *freqzin.asc*, *freqe.asc*, *freqh.asc* → Frequency domain system responses as a function of frequency,  $I(f)$ ,  $U(F)$ ,  $Z_{in}(f)$ ,  $E(f)$ ,  $H(f)$

*contour.asc* → contour lines in wire format

.h5 files: many *CONCEPT-II* related data is stored in the HDF5 format. for more information see <https://support.hdfgroup.org/HDF5/>

## 1.11 Performing a typical simulation

For a quick start it is recommended to study the examples of the *\$CONCEPT/demo/* directory starting with *example1-wire-loop*. Each example is described step-by-step. The corresponding PDF files are available on the download web page.

Important under Windows: Use only file names without blanks!

Some important key and mouse actions: **Help** → **Navigation**

A right-handed xyz coordinate system is assumed (Cartesian coordinates). The plane  $z=0$  marks the ideal ground plane when specified.

1. Start the *CONCEPT-II* graphical user interface



This can be done by clicking the *CONCEPT-II* icon or by typing

a) Linux: *concept.sh* <Enter> at the command line in a terminal

b) Windows: *concept.exe* (*Concept-II-12.0.bat* in case of problems)

2. For a new simulation: click **File** → **New simulation**

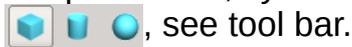
A dialogue opens with input fields for the *Project name*

and the *Project directory*. Click **File** → **New simulation** → **Browse...** to choose an existing directory or to create a new directory.

In case of a new structure open the CAD tools section by clicking the **CAD tools** tab (sub-window *Project View*, see Fig 1.7, left side of main window).

Discretize the structure to be analyzed by using the tools under **Cad tools 2**.

Simple bodies, cylinders, plates etc. can easily be discretized by clicking on



, see tool bar.

A structure already loaded under **Simulation** (see Fig. 1.6 ) can easily be transferred into the CAD tools section: tab **CAD Tools** → right click or double click on the top entry of the CAD tools tree **Geometry from simulation** → click on **load simulation file(s)**.

3. For an existing simulation: Click **File** → **Open simulation...** .

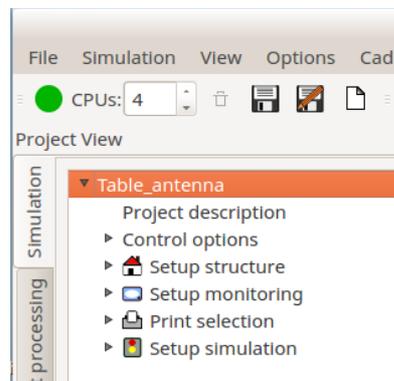


Fig 1.6: Example of the head section of the *CONCEPT-II* simulation project tree in the 'Project View' section. Here we have the project *Table\_antenna* (input file *Table\_antenna.sim*)

Input control files for *CONCEPT-II* end with the suffix *.sim*. The **Simulation** tab is activated in Fig 1.6.

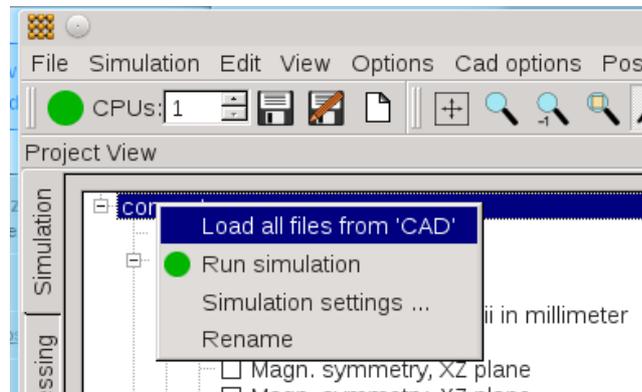
It is recommended to create a hierarchy of working directories, making it easy to access computed results.

Important for users who are familiar with older versions of *CONCEPT-II*: you can select *concept.in* which was the former name of the input file. The contents of this file will be converted into the *.sim* file format

Discretized structures can be easily loaded from the **CAD tools** section into



the **simulation** section by right clicking or double clicking on the root tree entry of the simulation project tree → select **load all files from 'CAD'**, see next figure)



- Carefully check all items of the simulation project tree. A double click or a right click opens the context menu which is not available for all items.

### Tip

Important for handling of the simulation project tree (section 'Project View')

- Right click or a double click on an entry of the project tree. This opens a list box for further selection

Display area: surfaces, nodes, wires, edges etc. are always selected by right click! Rotate mouse wheel: zoom in, zoom out. Left click and drag: rotate the structure. Please notice **Help** → **Navigation**

Names of wire files (examples):

*antenna.wire* or *wire.loop*

Names of surface files:

*plate.surf* or *surf.plate*

Examples: *surf.1*, *box.surf*, *surf.box*

*wire.1*, *grid.wire*, *wire.grid*

In principle arbitrary names are possible, however, only file names according to the given rules are automatically recognized and listed.

- Choose the number of CPUs to be used (see spin box right to the green



circular start icon, tool bar, see Fig 1.7).

6. Start the simulation by clicking on the green icon  on the left side of the tool bar, notice Section 1.5.

The number of unknowns can be determined before starting the simulation: **Simulation** (here item of menu bar, see Fig. 1.7) → **Run simulation front-end** → The number of unknowns appears in the project status view section directly under the display area. A new or an update of the current simulation control file is written into the working directory, default name is *concept.sim* an existing file of the same name will be overwritten.

See also tab **Log data** → **CONCEPT-II monitor**

7. After the simulation has finished start the post processing by clicking the **Post processing** tab (left vertical side of the main window).

Important buttons:

-  Displays the current (or charge) distribution on wires and patches.

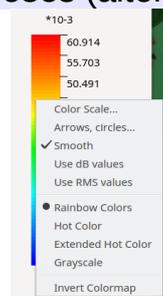
Simple rules for a first validation:

1. The vectors of the surface current distribution should follow imaginary smooth curves, they should look “beautiful”. Easy scaling: Right click into the 'Scale widget' section right to the display area → enable **Use dB values**
2. Jumps of neighboring current vectors should be carefully investigated.
3. “Hot spots” or randomly distributed current vectors are bad signs in general. In some cases this might be an indication of an insufficient grid (maybe too rough for a certain wavelength) or a non-convergent result.

Sixteen phase intervals are a good choice for creating movies. Press **Animate** to start the movie.

- Right click into the *Scale widget* → a drop down list opens, select *Arrows, circles...* for scaling purposes (alter length of arrows, change

colors, 3D arrows etc.)



-  Field computation E or H (SAR, energy flux density)
-  2D radiation diagrams
- ...

For ASCII output data of the back end run (for example contents of *concept.out*) click on tab **Log data** → **CONCEPT-II Simulation**

### Note

Validation of numerical results is a must-do!

The whole package can still be operated at the command line (terminal), examples:

Start of the front end:

Linux: *concept.fe filename.sim* <Enter>

Windows: *concept.fe.exe filename.sim* <Enter>

Start of the back end:

Linux *concept.be* <Enter> → Only a single CPU will be used

Windows *concept.be.exe* <Enter> → Only a single CPU will be used

Multi-processor usage:

OpenMP (see for example <https://computing.llnl.gov/tutorials/openMP>):  
*concept.be(.exe) -n no\_of\_processors*

Linux (MPI):

*mpirun -np no\_of\_processors concept.be* <Enter>

Windows (MPI):

*mpiexec -np no\_of\_processors concept.be.exe* <Enter>



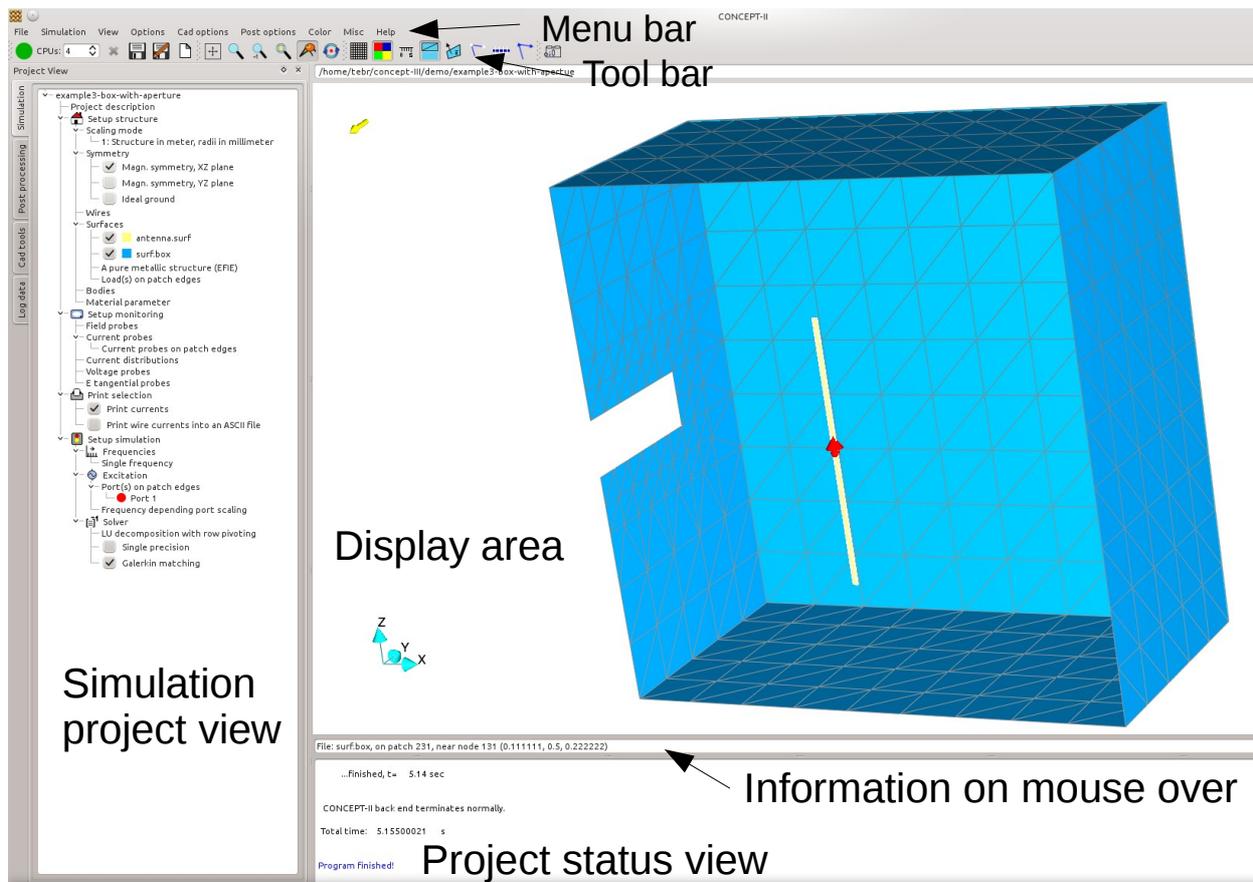


Fig 1.7: The *CONCEPT-II* main window

The *CONCEPT-II* GUI (main window) is shown in Fig 1.7.

### 1.11.1 Overview (main window)

- On the left side we have the 'Project View' section. In Fig 1.7 the **Simulation** tab is active, representing the simulation project tree. All input data for the numerical simulation has to be specified here.
- Display area
  - Left click, hold and drag causes the structure to rotate
  - Press keys <Ctrl>+<Shift>, left click, hold and drag → the selected sub-region will be enlarged.

Alternative: activate , left click and drag the mouse pointer. If several steps of enlargement have been applied get back to the last step by



clicking on . Show the initial representation by clicking on .

- Press <Ctrl>, left click and drag → the structure is shifted in the display area (the same action is initiated by clicking on , left click and drag)
  - Rotate the mouse wheel → zoom in or out, depending on the direction of rotation; center is the position of the mouse pointer.
  - All shortcuts: **Help** → **Navigation**
- Directly below the display area there is a single line section where patch numbers, coordinates, physical quantities etc. are displayed as soon as the mouse pointer touches a patch, a node and so forth.
  - At the bottom of the main window we have the project status view section (summarized content after job has finished, see also **Log data** → **CONCEPT-II monitor**)

### 1.11.1.1 Icons of the tool bar



Start button

1) Tab **Simulation** is active → the back end starts, setting up the equation system and solving it.

2) **CAD tools** is active → start a grid refinement or the generation of a surface or a similar action (the same action is initiated by typing on the <e> key).



The dialogue window for curve representation opens; an example application is

described in Sec. 2.1.1.3. Graphical output is possible in the formats png, svg and eps (CONCEPT-II gnuplot front end).



Indicates the x-y plane ( $z=0$ )



Tool for measuring distances between nodes. Select two different nodes by right clicks.



Loaded surfaces appear in different colors.



Switch the grid on/off. Especially useful in case of large patch numbers.



Show normal vectors of all patches.





The end of wires are marked. Modify markers by **Cad options** → **Dot size**

### (marker, wire ends)...



Show the segmentation of wires.



("Picker" icon) By default patch numbers, node numbers, coordinates etc. are

shown in special info line below the display area on mouse over. In case of very large grids this might lead to an annoying delay when attempting to rotate the structure in the display area. Disabling provides a smooth rotation again, of course depending on the performance the graphics card. It might be useful to deactivate this feature temporarily on a notebook or a computer with a less powerful graphics card.



: Enables the rotation of the structure around a selectable point . Right-click a

node and drag the mouse. Known bug: after deactivating this feature the structure might appear somewhat deformed (Jan. 18). Click on  to get the original representation again.



: The end of wires is indicated by arrows. The reference direction is from the beginning to the end.

## 1.11.2 The simulation project tree

See GUI section 'Project view', main window. The simulation tree consists of 4 major sections as illustrated in Fig. 1.6.

The root entry is the project name to be specified when opening a new project directory. The root entry is followed by these subsections:

- **Project description**
- **Setup structure** → Specify here wire coordinates, wire files, files with surface patch data, kind of symmetry, types of boundary conditions or surface properties for example.
- **Setup monitoring** → Set probes for the computation of EM fields or probes for currents and/or voltages to be displayed as a function of frequency after the back end has finished.
- **Setup simulation** → The frequency sampling, type of excitation and the type of solver have to be entered here.



### 1.11.2.1 Handling of the simulation project tree

Right-clicking or double-clicking on an entry of the project tree opens a drop-down list in most cases. Continue by selecting the required menu item.

Example:

A right click on **Surfaces** provides a drop-down list containing the entries

- **Load surface file(s)**

- **Load all surface(s) from CAD**

- All surface and wire files are loaded from the **Cad tools** section into the **Simulation** section. The corresponding file names appear as (additional) entries

- **Unload all surface files**

- **A pure metallic structure (EFIE)**

- All surfaces of type 'dielectric boundary' are turned into metallic surfaces

- **Surface impedance**

- A conductivity and a thickness can be assigned to metallic surfaces

- **Thin sheet approximation**

- An electrically thin and geometrically thin lossy sheet can be placed on one side of a PEC surface

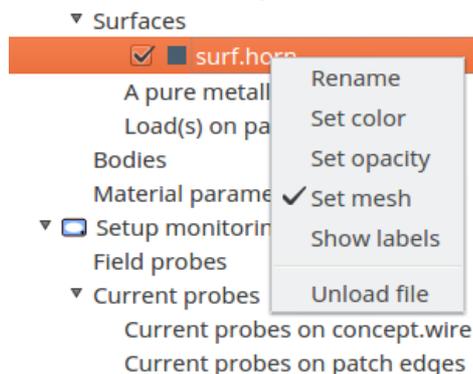


Fig 1.8: Drop-down list for modifying the properties of a surface, here surf.horn

Once surface patch files have been loaded a right click or a double click on such an entry gives additional possibilities as demonstrated in Fig 1.8 , for example:

**Show labels:** display node numbers or patch numbers.



### 1.11.2.2 Dielectric bodies

Starting with the example `$CONCEPT/demo/example4-dielectric-sphere` the following examples should be carefully studied:

`$CONCEPT/examples/Three_cubes,`  
`$CONCEPT/examples/Parallel_resistors,`  
`$CONCEPT/examples/Bodies_in_bodies`

- “Bodies” are defined as finite regions of space with homogeneous material parameters, namely permittivity  $\epsilon_r$  and conductivity  $\sigma$  ( imagine a sphere of lossy dielectric material “plastic”, for example). For a short recap of theory refer to Section 9.) A body has to be surrounded by a surface or several sub surfaces forming a completely closed envelope. An overlapping of such surfaces is not allowed.
- On the surface of a pure dielectric body the following boundary conditions are matched:

$$\begin{aligned} E_{t,internal} &= E_{t,external} \\ H_{t,internal} &= H_{t,external} \end{aligned} \quad (1)$$

Parts of a dielectric body may be covered by a metallic surface. In this case the boundary condition  $E_{\tan} = 0$  is separately matched on each side.

Each surface of a body carries two different electric current distributions in *CONCEPT-II*. One current distribution is source of the internal EM field distribution and the other one for the external EM field distribution. Both can be visualized and both are equivalent (non-physical) current distributions.

Note that each edge between two patches leads to one unknown for simple metal surfaces. In the case of a surface forming the boundary of a body we always have two unknowns per (internal) edge.

- The overall surface bounding a body (a dielectric region) has to be completely closed. Dielectric boundary surfaces are not possible in planes of symmetry when exploiting symmetry.
- Wires can be placed inside or outside bodies. If a wire is penetrating a surface of a body, it has to be split into two parts: each section must be attached to the same surface patch node. All patches sharing such a node will be metalized automatically (introduce local mesh refinement!).
- Structure parts (wires or surfaces) inside dielectric bodies have to be discretized considering the internal wavelength!
- A metal cavity with one or more apertures and filled with air can be modeled as a body: the apertures are considered by surfaces with boundary conditions according to the above Equation (1). The surface forming the cavity walls is described by the boundary condition  $E_{\tan} = 0$  which is then



matched separately at the internal and external side, leading to two current distributions. Such a proceeding may be advantageous for the accurate analysis of coupled-in EM fields. Note: Defining a completely closed metallic cavity with ideal conducting walls as a body makes no sense as the whole internal current distribution “sees” not excitation in case of an external source.

Completely closed bodies (also cavities with pure single-layer current distribution, EFIE) with internal sources and without any loss mechanism are critical from a numerical point of view.

- Surfaces where the boundary conditions (1) are specified can be layered by thin lossy dielectric or lossy magnetic sheets. The thickness of such sheets should not be larger than 5...6 times the penetration depths. Examples can be found here: *\$CONCEPT/examples/Layer\_examples*
- The typical dimension a body that can be computed depends mainly on
  - the electrical size of the bounding surface(s) and on
  - on the internal body conductivity. If this value is too high the solution may become numerically impossible.
  - In case of very high values for the rel. permittivity or permeability the curvature of the body surface should be carefully considered in relation to the internal wavelength and/or field variation.
- Rule: All patches of a (sub)surface bounding a body must have the same two regions on both sides, the corresponding normal vectors are not allowed to point into a third region...

More details concerning the modeling of bodies are given in Sections 3.1 and 9.

### Example

The procedure for specifying bodies shall be explained by an example: *\$CONCEPT/examples/Parallel\_resistors*.

We have two half cylinders of different lossy dielectric materials with an ideal metal plate at the top. The structure is forming a resistor consisting of two sections switched in parallel. The resistor is connected to a rectangular wire loop over an ideal ground plane. There is a voltage generator along the one of the wires.

As can be seen in Fig 1.9 there are two bodies attached to each other forming the overall cylinder. The symmetry of the problem shall be exploited. The cylindrical sections include homogeneous materials of different conductivities.



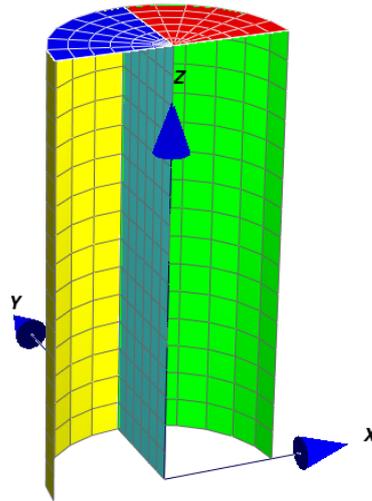


Fig 1.9: Example of two bodies attaches to each other on ideal ground (x-y plane). The assumed magnetic symmetry plane is the x-z plane. We need 5 surfaces to specify the problem. The top red and blue surfaces are assumed to be PEC. By specifying appropriate body conductivities two parallel resistors are simulated. The example is taken from *\$CONCEPT/examples/Parallel\_resistors*, where a wire loop including a voltage source is attached to the top plates in the plane of magnetic symmetry.

1. Load the surfaces describing the structure: **Surfaces** → **Load surface file(s)**. Five surfaces are necessary.
2. Right click **Bodies** → **Add body** → a new entry **body 1** appears.
3. Right click **body 1** → the drop-down list shown in Fig 1.10 appears. Now the surfaces covering **body 1** can be selected either by mouse over (right-click onto the corresponding surface) or by editing a list that appears when

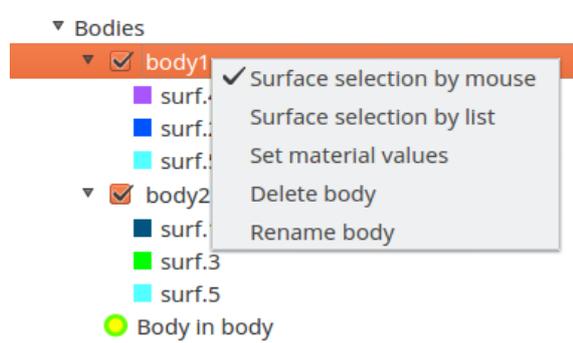


Fig 1.10: Defining the bodies from the surfaces

choosing **Surface selection by list**. Continue in the same way defining **body 2**

4. Next the material parameters for the bodies have to be entered. Select for example **body 1** → select **Set material values**, see Fig 1.10 → A dialogue



opens where the parameters  $\epsilon_r$  and  $\sigma$  have to be entered.

- The remaining step to be done is to specify the **type of the each surface** (Fig 1.11):

Select **Surfaces** → **Surface types (bodies) (bodies)** → A window as shown in Fig 1.12 opens where to enter the boundary conditions for all surfaces:

- *Metallic boundary*

Surface(s) related to the PEC part of a body boundary.

- *Dielectric boundary*

Both tangential components of the EM fields are matched here.

- *Free space*

If there is a surface outside the specified bodies, it has to be given this property, meaning metal in free space (EFIE problem with a single layer current distribution). Such a surface cannot cover a body or part of a body.

- *In body 1, In body 2,...*

The user could think of a ideal conducting metal disk for example completely immersed into the material of body 1 or body 2, respectively. Again, such a surface is not allowed to be part of the boundary of a body.

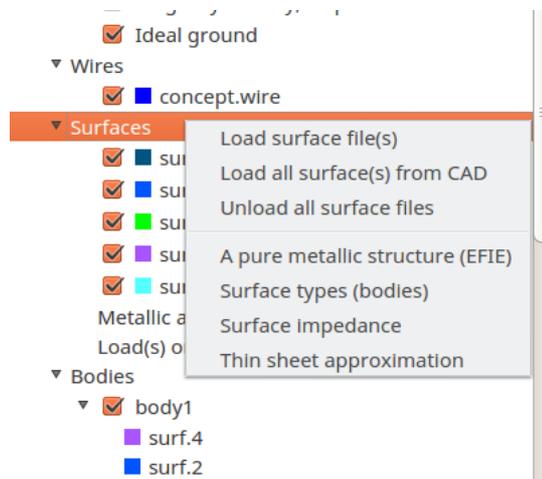


Fig 1.11: Surfaces forming the two bodies of the example structure

In Fig 1.12 the blue and red surfaces forming the top of the resistor structure are assumed to be ideal conducting metal plates of type *Metallic boundary* and all other ones are of type *Dielectric boundary*.

Surfaces can be layered by geometrically thin sheets, see Fig 1.13. Selecting *Add layer* enables to enter the material properties.



| All metall surfaces are ideally conducting |         |                    |
|--|---------|--------------------|
|  | Surface | Surface type       |
| 1  | surf.1  | Metallic boundary  |
| 2  | surf.2  | Metallic boundary  |
| 3  | surf.3  | Dielectric boundai |
| 4  | surf.4  | Dielectric boundai |
| 5  | surf.5  | Dielectric boundai |

Fig 1.12: In the example case the top plates are of metal whereas all other surfaces are dielectric interfaces.

### Note

Layers are only possible on surfaces of bodies. Layered surfaces must be of type *Dielectric boundary*. For details see Secs. 3.1, 3.1.5,4

In case of a body that is positioned completely inside another body with no common boundary (the user could think of two spheres with the same center but different radii) further information needs to be entered:

(Right click) **Body in body**  **Body in body** → **Assign** → a window opens

where to enter which body is inside another body. In the example [\\$CONCEPT/examples/Bodies\\_in\\_bodies](#) a structure is treated where we have a spherical body inside a cylindrical body which again is completely inside a cube. In addition we have an open PEC cone inside the cube, not in touch with the other bodies.

The user must be very careful specifying the “body in body” information: CONCEPT-II does not check if the data is correct.

By default all bodies are immersed in free space, which is body 0. The topic is further explained under Section 3.1.3.



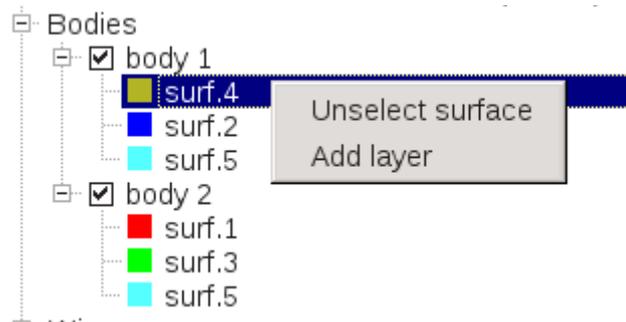


Fig 1.13: Adding a layer to surf.4. Clicking on *Add layer* opens a dialogue window where to enter the material parameters and the thickness.

### 1.11.2.3 Wires

Various features exist for wire treatment, see Fig 1.14. Similar to surfaces, wires may be grouped into different files the name of which should begin with “wire.” or end with the suffix “.wire”; *wire.antenna* or *wire.dipole* are examples for valid names.

Selection of **New wire file** opens a dialogue window where to enter file name, number of wires, coordinates, radii and number of basis functions (number of wire segments).

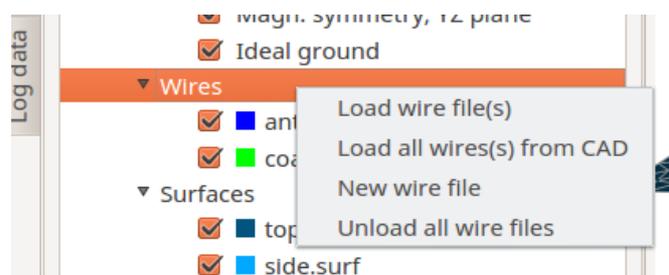


Fig 1.14: The drop-down list related to wires

## Note

Deactivation of a check mark in front of a wire or surface entry sets that structure part to invisible. It is not deleted from the simulation.

It is important that the thin-wire assumptions are fulfilled, notice the corresponding remarks under Sec. 1.7. Faulty wires are marked red, wires with warnings appear in pink.



The number of segments (almost the same as number of basis functions) has to be entered separately for each wire. This is necessary in case of a single frequency or for the starting frequency in a frequency loop. Once the length of a wire is getting into the range of the wavelength the wire is discretized automatically according to the '*Number of basis functions (bf) per wavelength*' to be set in the 'CONCEPT-II Wire Description' window (item New wire file, Fig 1.14). A number less than 8 is not possible.

- It is recommended that 3 segments are specified on electrically short sections in case of:
  - Wire-grid modeling where surfaces are simulated by corresponding wire grids; each wire being electrically short in general.
  - Short wires which connect different surface parts. Such wires are important from a network point of view ensuring current flow because of an existing galvanic connection. The physical connection might be more complicated
  - Short pieces of vertical wires, connecting lines to ground.
- Rules for discretization have already been given in Sec. 1.7.
- The initial discretization of a wire might not be sufficient when increasing the frequency in a frequency loop. The number of basis functions will be increased automatically.

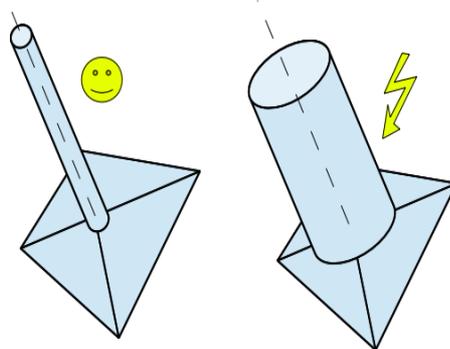


Fig 1.15: Wires attached to a surface patch node with four triangles.

Left: the ratio of radius/(edge length) is OK. Right: the radius of the wire is far too large compared to the edge length of the triangles. This geometry would require a full 3D surface patch modeling of a cylinder attached to a finite plate. Note that in case of the thin-wire approach there is only a filamentary current flowing along the wire axis. Testing the boundary condition around the whole circumference is not done.

Wires can only be connected to surfaces patch nodes and to other wires.



Connections are recognized if the difference in coordinates is smaller than  $10^{-5}$  of the applied unit (default is meter).

A triangular patch can only have wires being attached to one of its three nodes, hence there must be always 2 edges between different points of attachment in a triangular grid. In case of quadrangles at the point of attachment, these quadrangles are automatically turned into triangles.

Wire identification numbers are written into *concept.out*. By these numbers can be found out if a galvanic connection has been recognized or not.

As explained by Fig 1.16 it is not allowed that wires intersect with each other somewhere along their extension. In case of a physical point of attachment a straight wire section must start or end at that point.

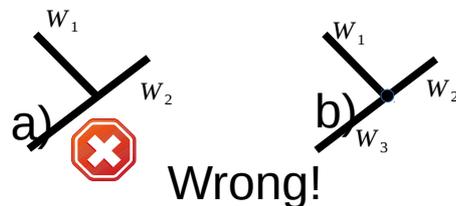


Fig 1.16: In a wire junction wires must start or end.

a) Wire 1 intersects with wire 2, a junction is not recognized. An error message is given.

b) If wire 1 shall be galvanically attached, three wires are necessary meeting in the common point of attachment.

The direction of a wire (interchange of beginning and end coordinates) can simply be modified by: Right click on wire file entry → **Edit wires** → Right click on the wire number (first entry of each line) → **Change direction**. Show all wire directions:   
(tool bar).

### 1.11.2.4 Loads

Both surfaces and wires can be loaded by lumped elements R, L, C in various combinations.

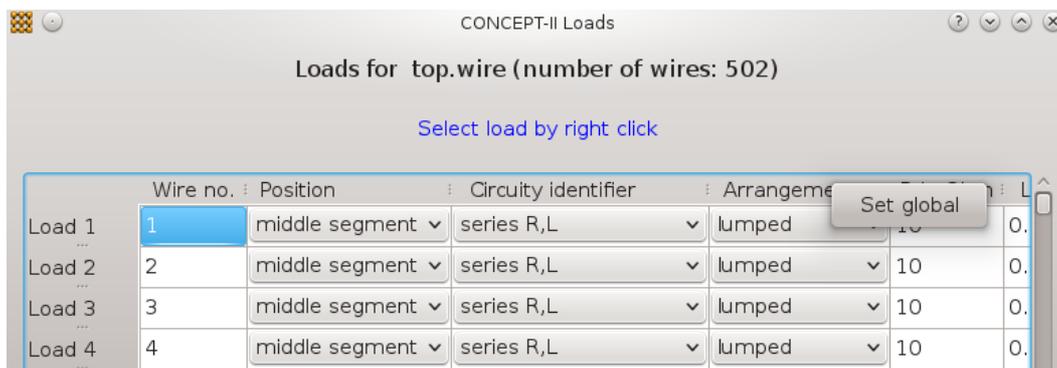
Select a wire file in the simulation project tree → (right click) → **Set loads** → a dialogue window opens for the specification of loads (possible positions: beginning, middle and end of a wire), various predefined circuit elements are available. An



example is demonstrated in Fig. 1.17: we have series connections of a resistors and an inductors positioned in the center of wire 1, 2, ....

In case of a large number of loads:

- Select **Append** and enter the corresponding number of lines (see '*Total loads*'), click **Do**
- Right clicking on 'Wire no.' provides the possibility to set all wire numbers at once.
- Right clicking on 'Circuit identifier' provides **Set global**. The chosen identifier can be assigned to all wires.
- Loads can be taken from files. Format:  
*Frequency Real part Imaginary part* (three real numbers in each line)
- For inserting, deleting or copying of a line ('Load 1', 'Load 2',..., see Fig 1.17) right click on 'Load 1' or 'Load 2' or whatever load number shall be selected



For placing lumped loads on surfaces (at least one surface files must be loaded):

right-click the project tree entry: **Load(s) on patch edges** → **Set edge loads** → a dialogue window opens. Right click on the edge of the surface patch grid, where a lumped shall be placed.

For an easy change of an existing load (or generator) right click on the symbol in the display area → the selected position is indicated in the corresponding dialogue window.

To delete a load right click onto an entry *Load1, Load 2, Load3 ...* → **Delete**

It is only allowed to place lumped loads on metal plates not covering a body (plates with single layer current distribution). Note that the vector sum of the surface current distribution is always computed by the EFIE!

The default reference direction of voltages at lumped loads on patch edges is related to the current reference direction, indicated by a black mark. Right clicking on the load symbol changes this, if desired.



For voltage probes: select **Voltage probes on patch edges (Voltage probes on *wire.file*)** by a right click → **Set probes** → select the probe at the required position.

For displaying voltages as a function of frequency: Post processing → .

### 1.11.2.5 Frequency sampling

In Fig 1.18 the entries of the **Setup simulation** section of the simulation project tree are illustrated: **Frequencies, Excitation, Solver**. Right-click on **Frequencies** gives the shown drop-down list shown in Fig 1.18:

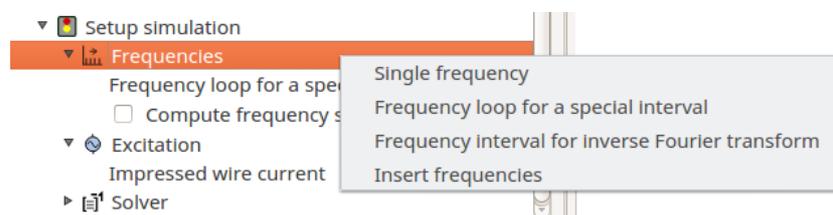


Fig 1.18: The **Setup simulation** section of the simulation project tree

Note that a special frequency sampling is required if an inverse Fourier transform shall be applied for time-domain system responses.

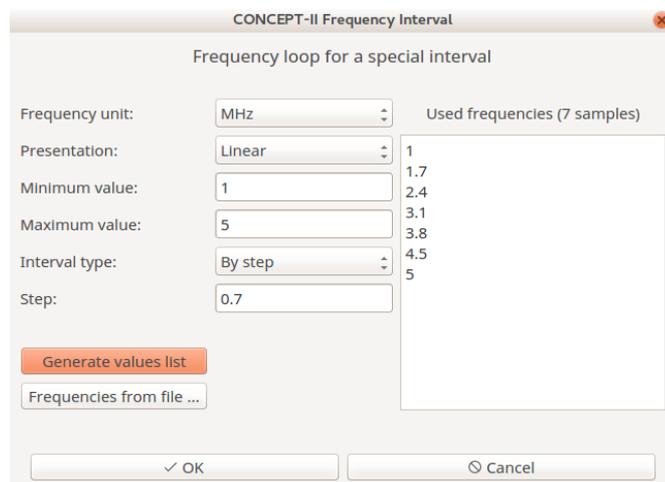


Fig 1.19: Window for setting the frequency sampling

Click on **Frequency loop for a special interval** → right click **Set frequencies** → window opens, see Fig 1.19. Clicking on **Generate value list** gives the list in the 'Used frequencies' area. It is possible to add values "by hand" here or to edit an existing list of frequencies. In the example the interval between 1 MHz and 5 MHz is sampled by 0.7 MHz steps. Click on **OK** → all steps as seen under *Used frequencies* will be taken in the simulation.



Activation of **Compute frequency steps parallel** means that all frequency samples are distributed according to the chosen number of CPUs (see tool bar). In principle the back end is started  $n$  times on the computer with  $n$  as the number of CPUs. Each CPU is setting up and solving its assigned number of frequencies.

Furthermore *CONCEPT-II* can compute time-domain system responses based on an inverse Fourier transform IFT. A careful selection of the frequency interval and of the frequency sampling is required; details can be found in Sec. 5

### 1.11.2.6 Excitation

The next logical step setting up a simulation is to select the type of excitation, see Fig 1.20.

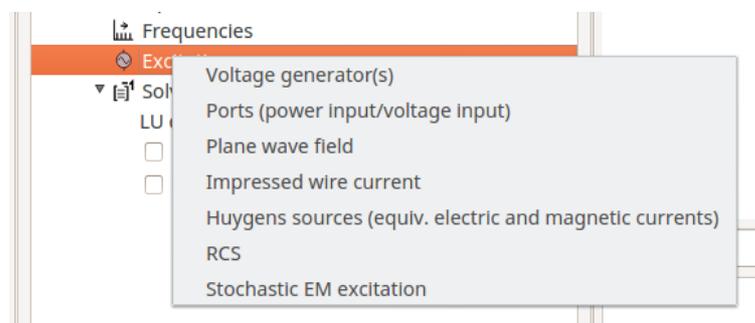


Fig 1.20: Types of excitation

How to use **Stochastic EM excitation** is not yet described in this manual. The results of an application have already been published:

M. Magdowski, A. Schröder, H.-D. Brüns, R. Vick: Effiziente Simulation der Einkopplung statistischer Felder in Leitungsstrukturen mit der MoM, EMV Düsseldorf, 2014, Seiten 238-244.

Simulation of coupled-in quantities of a structure in a mode stirred chamber.

#### 1.11.2.6.1 Voltage generators

Voltage generators can be located both on wires and surfaces (edge generators).

To specify one or more voltage generators the following steps are necessary:

##### Generators on wires

**Excitation** → **Voltage generator(s)** → **Generator(s) on wire.filename** (example *antenna.wire*) → **Set generator(s)** → a window according Fig 1.21 opens.

Position the mouse pointer on the wire where a generator shall be located. A right click selects the wire and a new line appears in the 'CONCEPT-II Voltage Generators' window. Three 'Generator positions' are possible: **1st segment**,



**middle segment, last segment.** The reference direction is indicated by the generator symbol. For changing the reference direction interchange the start and end coordinates of that wire (see Sec. 1.11.2.3).

A voltage generator can also be entered by appending a line (click the **OK** button,

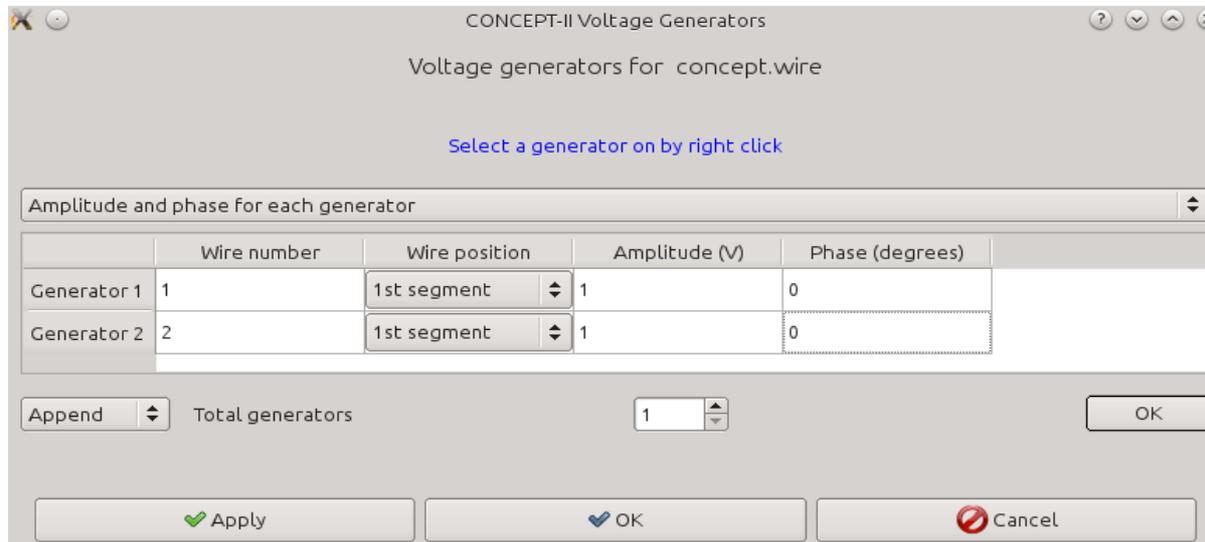


Fig 1.21: Window for the specification of voltage generators on wires

Fig 1.21) and entering the wire number by hand.

Delete a generator: Right click on the corresponding generator number at the beginning of the corresponding line → the line is marked → **Delete**

### Voltage generator on an edge between adjacent patches (edge generator)

Select the item **Generator(s) on patch edges** of the simulation project tree → right click → **Set edge generator(s)** → click on the desired edge of the surface grid in the display area. For a change of polarity right-click on the same edge again. Specify voltage magnitude and phase in the window that has been opened.

### 1.11.2.6.2 Ports

In practice it is frequently only of interest to investigate the mutual EM coupling with respect to varying generator positions, one could imagine a ship carrying a number of monopole antennas with feed positions close to the superstructure. Such feed points can be defined as ports. As demonstrated in Sect. 3.2 the mutual interaction between ports can be described by a reduced admittance matrix of order  $K$ , where  $K$  is the number of ports, which have been defined. In general  $K$  is much smaller than the order  $N$  of the system matrix.

If multiple ports were defined the corresponding current distributions are computed



one after another (**i.e. not at the same time**), i.e. the first generator is active and all other generators are short-circuited. Then the second generator is excited and all other generators are short-circuited and so on.

Right mouse click on **Excitation** in the project tree → **Ports (power input/ voltage input)**

If both wires and surfaces are in the simulation the following project tree entries appear:

- Port(s) on wire.file\_1
- Port(s) on wire.file\_2

...

----- Ports on patch edges

Right-Clicking on **Port(s) on wire.file\_1** → **Set port(s)** opens a dialogue window as depicted in Fig 1.22.

A power generator (excitation by means of an impressed power) can only be realized by assuming a port. Note that more than a single power generator feeding the system 'at the same time' is not possible with the exception of the presence of a symmetry plane: in such a case a second power generator is automatically set in the range of the symmetric part.

## Note

### Post processing

Note that generators defined as ports do not excite the structure simultaneously. After the CONCEPT-II back end has finished each port leads to its port related current distribution. All tools provide the possibility to select which port is the active one. Non-active ports are short circuited. Loads which may be defined at the position of ports are not effected.

EM field distributions and radiation patterns (2D, 3D) can also be computed with voltage generators being active "at the same time", i.e. simultaneously exciting all port positions with arbitrary voltage magnitudes and phases.



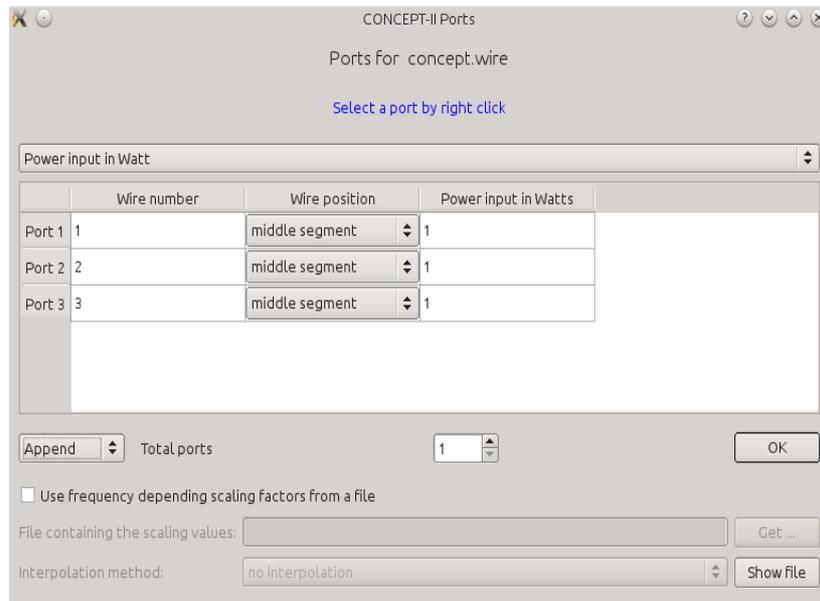


Fig 1.22: Example of three ports. At each port we have a power generator, located in the center of the wire (wire file name here: *concept.wire*)

Wires to be excited by ports can be selected by right-clicking on them in the display area or by directly entering the wire number. Then choose the required location (1<sup>st</sup> segment, middle, last).

In case of an edge generators choose **Port(s) on patch edges** → **Set edge port(s)** → Select the required edge for a voltage or power generator by a right mouse click. Change generator polarity by a second click on the same edge.

Defining ports enables the computation of S-parameters. In addition to this, a file named *red-y-mat.h5* (HDF5 format, see <https://support.hdfgroup.org/HDF5/>) is generated containing a reduced complex admittance matrix for each frequency step. Such an admittance matrix is describing the coupling between the ports. In the example case according to Fig 1.22 we have three ports, hence the admittance matrix will be of order 3x3.



**Note!**

If ports are set a reduced Y matrix is generated (data in *red-y-mat.h5*). In addition to this the port coupling is described by S, Z, or Y parameters (post processing tool: [N]). Corresponding matrices can be stored in Touchstone

format or in HDF5 format. Lumped elements defined directly at the position of ports are not contained in *red-y-mat.h5* and are also not contained in the S, Z, Y matrices.

The mentioned data can be imported using the network solver

CONCIRC ( element:  ), see

<http://www.tet.tuhh.de/concept/network-solver/>

For further details refer to Sect. 3.2 . After the back end run has finished, field distributions or radiation diagrams can be computed (tab **Post processing**)

Example: \$CONCEPT/examples/Coupled dipoles; notice the README file there. Further details can be found in Section 3.2.3 Using red-y-matrix.h5 with the network tool CONCIRC.

### 1.11.2.6.3 Plane wave field excitation

Setting the angles to  $\Theta=0, \Psi=0, \Psi=0$  provides a plane wave, propagating against the z axis, i.e. the wave vector  $\vec{k}$  is pointing in minus z direction with the E vector  $\vec{E}$  and the wave vector  $\vec{k}$  in the x-z plane.

The upper small sketch in Fig 1.23 demonstrates the situation with  $\Theta>0, \Phi=0, \Psi=0$  , the vectors of  $\vec{k}$  and  $\vec{E}$  are still in the x-z plane (right hand rotation about the y axis first). Bot now the wave is propagating in the direction of the “lower left side”.



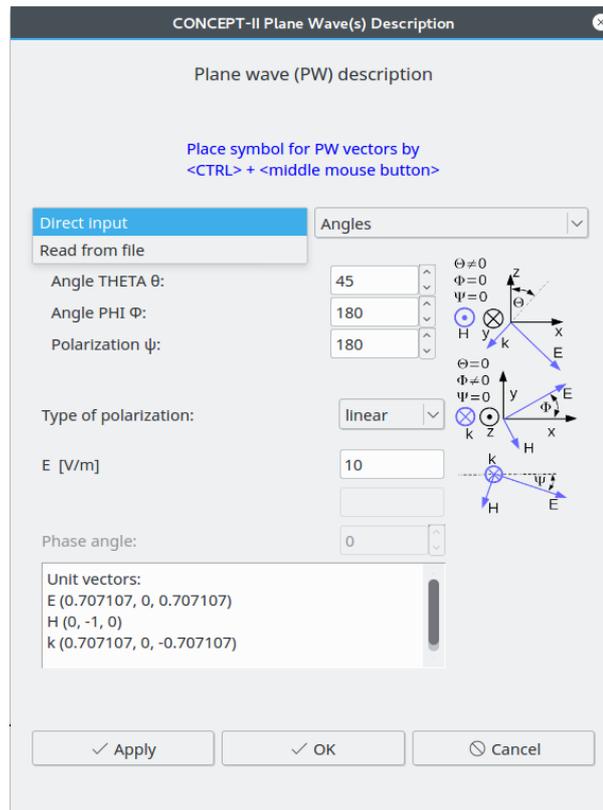


Fig 1.23: Input fields for defining a plane wave excitation.

The middle small sketch in Fig 1.23 demonstrates the situation with  $\Theta=0, \Phi>0, \Psi=0$ . Again, the wave is propagating in minus z direction but now  $\vec{E}$  includes an angle  $\Phi>0$  with the x axis (right hand rotation about the z axis).

Finally,  $\vec{E}$  can be rotated around  $\vec{k}$  by the polarization angle  $\Psi$ , this is demonstrated by the lower small sketch; the direction of  $\vec{k}$  is arbitrary.

The coordinates of the unit vectors E,H, k, resulting from the angles  $\Theta, \Phi, \Psi$  are shown in the message area in the lower part of the “Plane Wave(s) Description” window. A field excitation example is depicted in Fig 1.24.

Types of Polarization: Linear, elliptic

For elliptic polarization we have the following initial position  $\Theta=0, \Phi=0, \Psi=0$  :

1. The vector with amplitude E1 is pointing in positive x direction (all angles refer to this vector). At the same time the vector with amplitude E2 is pointing in the negative y direction. Both E vectors are perpendicular to each other. The phase angle determines the direction of rotation for both elliptical and circular polarization. For circular polarization we have

$$E1 = E2, \text{ phase angle: } \pm 90^\circ$$

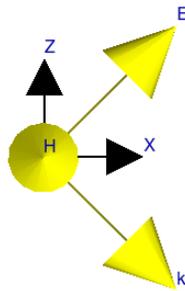


Fig 1.24: Orientation of all vectors based on the values as shown in Fig 1.23

Equations:  $E_x = E_1 \sin(\omega t + k z)$ ,  $E_y = E_2 \sin(\omega t + k z + \delta)$  with  $\delta$  as the *Phase angle* (propagation against the z axis).

The field rotation is observed when the wave travels away from the observer (looking into the direction of the wave vector  $\vec{k}$ ).

Positive phase angles  $\rightarrow$  RHC: right hand circular polarization, clockwise rotation.

Negative phase angles  $\rightarrow$  LHC: left hand polarization, rotation counter clockwise.

Direct input: Input directly using the GUI

- a) Enter angles and amplitude(s)
- b) Enter the Cartesian coordinates of the E vector  $\vec{E}$  and the wave vector  $\vec{k}$

Read from file: The required data format is shown when clicking onto **Show file format**. An arbitrary number of plane waves, all with different field magnitudes, orientations, and polarizations can be specified (ASCII data). After the back end has terminated field and current distributions, backscatter diagrams,  $I(f)$  and  $U(f)$  can be displayed for each incident wave.

Plot field distributions, **Post Processing**  $\rightarrow$   :

(PW. "Plane Waves")

Combination: 'All Frequencies' checked, 'All PW' unchecked  $\rightarrow$  Plot of field distribution as a function of frequency at the chosen phase (**Step** or **Animate**)

Combination: 'All Frequencies' checked, 'All PW' checked  $\rightarrow$  Plot of field distribution as a function of frequency at the chosen phase at a '*Serial number of PW*' to be selected (**Step** or **Animate**). If '*Show values for all frequencies*' is not active, the desired '*Frequency number*' has to be entered. Then it is possible to generate a movie with a '*Number of phase intervals*' (suggested 16) at the chosen frequency and specified field excitation.



### 1.11.2.6.4 Impressed wire currents

Wires carrying line currents with known amplitudes can serve as the structure excitation. The induced currents on other surfaces and/or wires that are present in the simulation have no impact back onto the exciting wire current distribution. Hence these currents are impressed ones.

A short section of a wire with known current may be used as a (ideal) current source.

Extended, long wires can be used to simulate a lightning strike, either as a nearby stroke or as a direct stroke, based on the transmission line model originally proposed by [Uman], see also end of Sec. 5.4. The channel is taken into account by a single wire or a spatial sequence of wires (connect the end of a wire to the beginning of the next one) carrying an impressed current distribution. Only a single current amplitude is possible.

Clicking on **Impressed wire current** (Fig 1.20) opens a dialogue window as depicted in Fig 1.25.

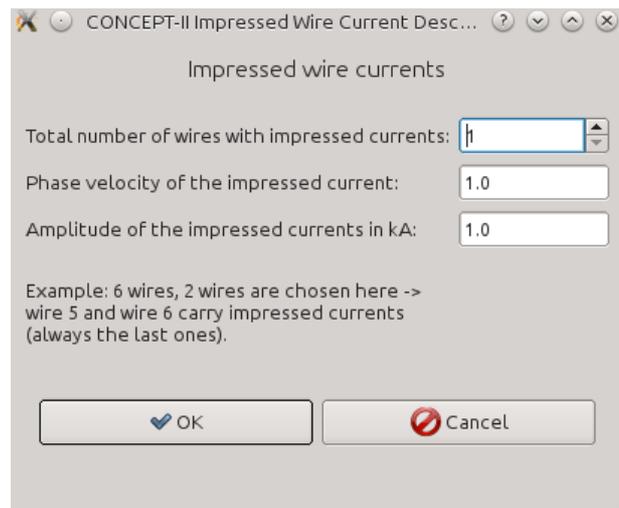


Fig 1.25: Dialogue for specifying impressed wire current

- *'Total number of wires with impressed currents'*

Entering a number  $n$  means that the last  $n$  wires carry an impressed current distribution. In the example we see a "1" in the top spin box. Hence the last wire has a known current distribution to be specified in the remaining two input fields.

If only wires carrying an impressed current have been entered the corresponding EM fields can directly be computed, see **Solver** in the simulation project tree, item **Compute only the impressed field**. Solving an equation system is not necessary.



If a short piece of wire with an impressed current is part of a loop it is possible to model a current generator (of negligible spatial extension): the specified current will flow no matter how large the load is.

Wires with impressed currents have to form a sequence (end of a wire is connected to the beginning of the next one).

- *'Phase velocity of the impressed current'*

In case of a lightning simulation this is the relative propagating speed of the time-domain current function (for example a double-exponential function) moving from the wire beginning to its end.

After the back end has finished: **Post processing** → , inverse Fourier transform, see Secs. 5, 5.4 .

0 → All basis functions have a constant amplitude all over the entire wire(s) with an impressed current distribution.

1 → Propagation with of speed of light. For lightning simulations a factor of 0.3 instead of 1.0 is frequently taken.

- *'Amplitude of impressed current in kA'*

For more information refer to Sec. 5.4

An example can be found under `%CONCEPT/examples/Lightning_overhead_line`.

### 1.11.2.6.5 Application of the Surface Equivalence Theorem: Huygens Principle

The theoretical background is explained in textbooks on antenna theory or electromagnetic theory, see for example [Balanis].

An actual source such as an antenna or an arbitrary transmitter is replaced by equivalent sources. This is accomplished by mapping the radiated tangential E and tangential H fields of the original source onto an equivalent surface (of arbitrary shape) that encloses the actual source of radiation (the aperture). Such an equivalent surface will be called "Huygens surface". A Huygens surface carries both an equivalent electric and an equivalent magnetic current distribution in the general case. Once these equivalent currents have been computed they can be used as impressed sources, exciting a secondary structure. Note that there is no mutual interaction of a structure illuminated by equivalent currents and the equivalent currents itself (and -of course- with the original sources).

#### General proceeding

An example can be found under `$CONCEPT/examples/Huygens/Horn-antenna`. Notice the README file there.



1. Compute the radiating or scattering structure as usual.

In the mentioned example a horn antenna is considered, excited by a vertical wire close to the base of the feeding waveguide section.

2. Generate a 'Huygens surface' which in the most general case is completely enclosing the radiator. This can be done by means of the tools of the **Cad tools** section. The surface discretization has carried out based on the well-known discretization rules.

In the horn antenna example there is no completely closed Huygens surface. To consider most of the radiation pattern it is sufficient that the Huygens surface is covering only the aperture of the antenna.

In the originally radiating structure symmetry can be exploited. Using the equivalent currents on a Huygens surface as excitation this is not possible any more.

3. Compute the equivalent sources on the Huygens surface (for example a discretized sphere enclosing the radiating structure or a certain region of interest). Go to **Post processing**. Click on  (tool tip: Generate equivalent

currents) . **Get file** → choose the file containing the surface patch model of the Huygens surface. Notice the direction of the surface normal vectors, set the check mark if the field of interest field is on the opposite side. For displaying the normal vectors: Tool bar, main window,  . Start the

computation of the equivalent currents by clicking on **OK**.

The window 'Plot equivalent currents' opens after the currents  $\vec{J}_{Equivalent}$  and  $\vec{M}_{Equivalent}$  have been computed (The current distributions are developed into basis functions with corresponding amplitudes).

*Selection* → Select the type of current. Additionally the original fields can be plotted which are the basis for the computation of the equivalent currents (only the tangential components matter).

For a repetitive representation right click on  **Equivalent currents** →

**show results** (*View results* section).

The results are stored into the file *[surfaceName]-JmequiCurrs.h5*. This file contains all equivalent current amplitudes and the grid data of the Huygens surface.

4. For applying the Huygens surface with its equivalent sources as excitation it is recommended to open a new working directory: Main window of the GUI → **File** → **New simulation** → **Browse ...** CONCEPT-II opens the new directory.

In the example this directory is called *Equiv*.



5. Basic steps to be done on the new working directory:

- **Excitation** → **Huygens sources (equiv. electric and magnetic currents)** → Load the file *[surfaceName]-JmequiCurrs.h5* from the parent working directory where the equivalent sources were computed.

Note that fields that shall be converted to equivalent surface currents can be loaded from external, for example in the form of measured data. For this purpose the fields have to be available at the centroids of an appropriate surface patch file serving as a Huygens surface. For more information click on **Show file formats**. Generate these field points by

**Cad tools** → **Cad tools 2** →  → **Create points at centroids**, load the corresponding surface file...

The Huygens surface used for excitation appears in color pink.

- Load the frequency file *confreq.in* from the parent directory:  
**Frequencies** → **Frequency loop for a special interval** → **Frequencies from file...**
- Specify a surface and/or wire structure to be illuminated by the equivalent and impressed current distributions on the Huygens surface. If such a structure does not exist for in a preliminary investigation select **Solver** → **Compute only the impressed field**. Click on  → the back end runs.
- Continue with the tools under **Post processing** as usual.

[Balanis] C. Balanis: "Advanced Engineering Electromagnetics", John Wiley & Sons, 1989

### 1.11.2.7 Solvers

Available solvers are indicated in Fig 1.26. The following solvers can be used with multiple processors:

- LU decomposition → All types of structures. Frequency range: Lower limit 100 Hz, ..., 10 kHz, upper limit → number of unknowns, depending on kind of structure
- Direct H-matrix solver → All types of structures, essentially only for high frequency problems, i.e. electrically large structures
- Iterative H-matrix solver → All types of structures, essentially only for high frequency problems
- MLFMA based solver → Only PEC, only high frequency problems, triangular meshes are preferred

### LU decomposition



Note: LU decomposition is the default solver. It is suitable for all kinds of problems.

The number of processors to be used can be entered in the upper left corner of the main window, right to the button with a green circle.

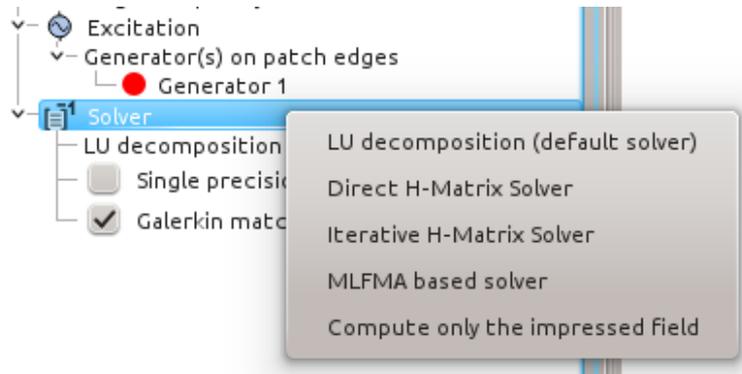


Fig 1.26: Solvers in the CONCEPT-II package

For *LU decomposition* it is advantageous to select the number of CPUs according to a quadratic number: 4, 9, 16, 25,... in case of using MPI (menu bar, **Simulation** → **Simulation Settings...** → **LU decomposition: Use distributed memory (MPI)** . Default is using OpenMP (shared memory), here no such rule exists. This holds also for MLFMA or for the H matrix solvers (direct, iterative).

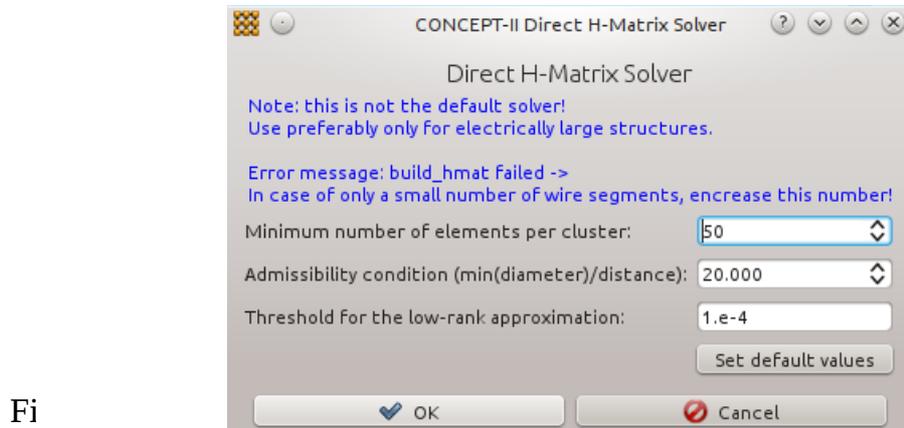
Never enter a number larger than the number of CPUs that are available on the computer!

A major limitation of the MoM in conjunction with LU decomposition and electrically large objects is the extensive computation time and the huge amount of memory that are necessary. The following solvers might reduce these requirements significantly. Unfortunately none of them is suitable for the low frequency range, where the dimension of the structure is smaller than the wavelength.

### H Matrix Solvers (direct, iterative)

Do not use for electrically small structures. The asymptotic complexity of the matrix setup could be reduced from  $O(N^2)$  to  $O(N^{4/3} \log N)$  . The complexity of the solution stage is reduced from the cubic behavior of the pure LU decomposition to a quadratic complexity. The solution process of the H matrix is essentially a block-wise LU decomposition with subsequent forward- and backward substitution. Details can be found in [1]. Although the pure numbers are not as good as for the MLFMA the H-matrix solvers are stable even at highly resonant structures and can be applied to all kinds of boundary conditions. Nevertheless bad results have been found for bodies of high-loss material (human tissue for example).





Fi

Fig 1.27: Carefully note highlighted text when using the H-matrix solver.

Note also the tool tips that appear on mouse over at the input fields. The same statement holds for the iterative H matrix solver which has its own dialogue window.

In practical cases the H-matrix solver needs 5%,...,40 % of the storage of the the full MoM matrix where smaller values are observed for problems which are getting increasingly electrically large. Relatively low compression rates are also observed for dielectric problems. The iterative H matrix solver is faster then the direct H matrix solver but needs more memory in general.

### MLFMA

The MLFMA reduces both storage and CPU time requirements significantly to  $N \log(N)$ . The MLFMA solver can only be applied to electrically large PEC structures including wires, surfaces and symmetry (*magnetic symmetry to both the XZ and the YZ plane does not work, Nov. 15*). Input parameters:

'Residual tolerance': Determines the precision of the iterative numerical solution. 0.01 could be sufficient in case of far field computations. 0.0001 could be necessary for almost closed cavities and complex structures.

'Max. number of iterations': If the *Residual tolerance* is not achieved after this number of iterations the computation stops.

'Drop tolerance for SPAI': Determines the sparsity of the preconditioner. The smaller the number the more dense the preconditioner and the faster the convergence. Disadvantage of a small number is a long set-up time.

### Compute only the impressed field

This not a solver as no unknowns appear. In the case of an incoming plane wave with no structure being present or in the case of an impressed current distribution with no further structure parts, this type has to be selected.

### General remarks on multiprocessor usage



Solvers are based on either MPI (Message Passing Interface) or OpenMP (Open Multi-Processing) . The next table gives shows the details:

| Solver              | MPI | OpenMP |
|---------------------|-----|--------|
| Lu decomposition    | x   | x      |
| H-matrix            |     | x      |
| H-matrix interative |     | x      |
| MLFMA               | x   |        |

Default is OpenMP for LU decomposition. In case of necessity of switching to MPI for LU decomposition: **Simulation** (menu bar) → **Simulation settings** → *Type of parallel Solver*

### Usage at the command line

MPI, Linux:

`mpirun -np number of processors concept.be`

MPI, Windows:

`mpiexec -np number of processors concept.be.exe`

OpenMP, Linux:

`concept.be -n number of processors`

OpenMP, Windows:

`concept.be.exe -n number of processors`

### Special case: Parallel computation of frequency steps

This special solution technique cannot be applied in case of MLFMA, only MPI is

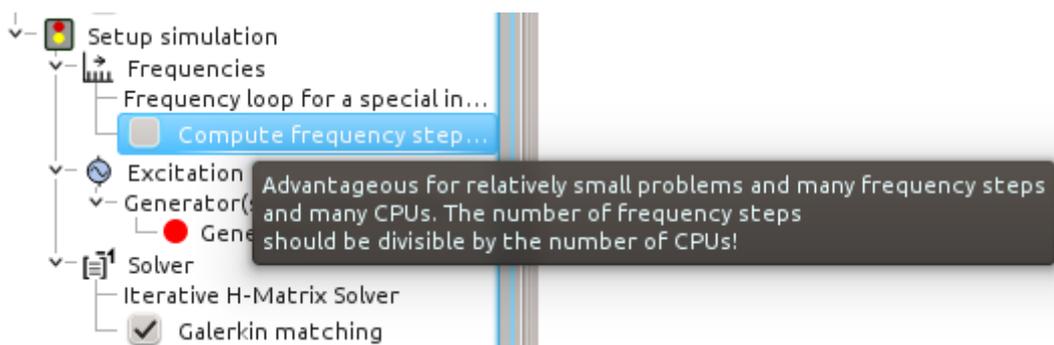


Fig 1.29: Each CPU is setting up a matrix equation and solving it

used for MLFMA. Parallel computation of frequency steps may have advantages in case of many frequency steps, small problems and the availability of a huge number of CPUs. Note that each CPU needs the whole executable including the corresponding memory. Quite a similar situation: Starting CONCEPT-II in different working directories assuming different frequencies. Advantage of selecting **Compute frequency steps parallel**: data collecting is done automatically.

Further remarks on parallel computing can be found in Section 7.

Literature:

[1] A. Vogt, T. Reuschel, H.-D. Brüns, S. Le Borne, and C. Schuster, "On the treatment of arbitrary boundary conditions for a fast direct H-matrix solver in MoM," Submitted to IEEE Trans. Antennas Propag., Jan. 2015.

### 1.11.2.8 Precision of the numerical solution

The default matching technique is line matching which is fast and gives good results for freely radiating antennas/structures. It is emphasized again that the MoM cannot be applied down to  $f=0$  ! The lowest possible frequency depends on the type and typical size of the structure under investigation and is in the range on 1 kHz to 100 kHz. Always choose a pure triangular surface patch grid in case of critical structures, such as parallel plates very close to each other, more or less closed cavities...

The following listing includes cases where a high degree of numerical precision is required.

- Generally at very **low frequencies** (electrically small structures).
  - Line matching causes no problems in general in case of pure wire structures, see `$CONCEPT/demo/example1-wire-loop` for example.
  - Conductor loops: it turns out to be difficult to compute the proper, very small real part of the input impedance in case of loops where **both wires and patches** are involved in the current path.

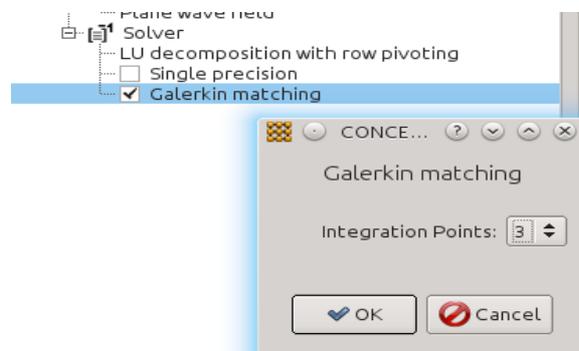


Fig 1.30: Galerkin matching can only be used in case of a pure triangular mesh

No problems with such loops at higher frequencies. Also no problems with



such loops at any frequency in case of pure patch loops (loops simulated by patch strips, discretized as a row of patches, choose triangles!). Hence wires that are connected to surfaces should be modeled by patch strips, the width of which should be chosen according to  $width = 4 * r$  with  $r$  as the radius of the wire.

- Monopole antennas that are attached to triangulated surfaces could be modeled by equivalent patch strips; width of a strip =  $4 * r$ .
- Cavities with small apertures, where coupled-in or coupled-out field quantities are of interest: use Galerkin matching (similar case: [\\$CONCEPT/demo/example3-box-with-aperture](#)); three integration points are normally sufficient (Fig 1.30). Galerkin matching can only be applied in case of triangulated surfaces.
  - Generally no problem with wire antennas inside or outside the cavity **and existing wires being not attached to patch nodes**.
  - Problems might occur in case of wire antennas that are attached to patch nodes. Workaround here: model monopole antennas by small patch strips and use edge generators. Set the width  $w$  of the strips:  $w = 4 * r$  with  $r$  as the radius of the original wire.
- Very flat structures (patch antennas): use only triangles here, use metal strips to model the antenna feed, use Galerkin matching.

Single precision: Experience has shown that the single precision mode only gives valid results for well-conditioned problems with external excitations in conjunction with iterative solvers (e.g. MLFMA). Advantage: half the matrix size. For direct solvers or problems involving cavities always use double-precision (default setting).

Summary: In most cases the highest precision is achieved by using pure triangular grids in conjunction with Galerkin matching.



### 1.11.2.9 Post processing tools

When clicking on the **Post processing** tab, the 'Project View' section looks as

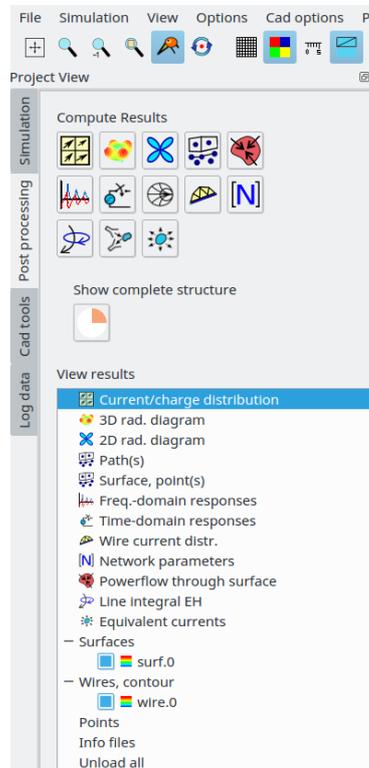


Fig 1.31: The post processing project view and project tree, respectively

depicted in Fig 1.31. All programs for post-processing purposes can be found here. For the application of these programs the *CONCEPT-II* back end run must have finished. Tool tips give information about the purpose of each tool.

#### Current distributions



→ Representation of current/charge distributions both on wires and surfaces.

Sixteen phase intervals between 0,...,360 degrees are recommended for a movie of the current distribution. If the movie is running too fast enter an integer value (10,20,100... depending on the graphics card) into the *Delay* box at the bottom of the post processing section.

Pay attention which surface patch file is active (pull-down menu in the top of the Scale widget section!

If *surf.0* is selected (see *Scale widget* section on the right side of the display area) → the shown color map refers to the surface current distribution. If *wire.0* is selected → the shown color map refers to the wire current distribution, ...

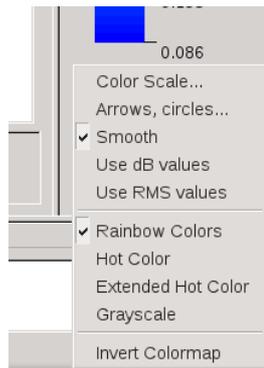
The default type of representation in case of a surface current distribution is current



vectors combined with colors on the patches. The colors are indicating the magnitude of the current vectors referring to the active phase.

Right-click somewhere on the *Scale widget* → A number of items can be selected for modification of the displayed data, see next figure (color map: Rainbow colors). Changing the length of vectors or the scaling of the colors has only an impact on those data that belong to the selected surface or wire part. Hence current distributions on surfaces and wires are scaled differently.

**Color scale** → The color distribution of the current scheme can be changed. Different color maps such as '*Rainbow colors*' (default) or '*Hot colors*' are available



**Arrows, circles...** → Changing vectors: log scale, global length, width, 3D type etc., see dialogue window according to Fig 1.32 .

**Smooth** deactivated → Show only a single color per patch. The numbers close to the vertical value list right to the color bar may change a little, depending on the mode of representation.

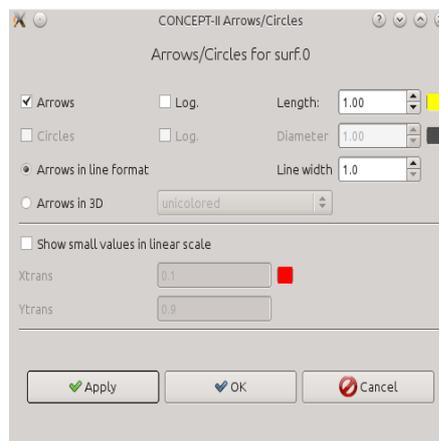


Fig 1.32: Dialog for modifying the vector representation



Current distribution on wires: A black arrowhead indicates the current direction at the beginning of the wire. In case of a change of sign in the current distribution an arrow with a different color appears at the corresponding location pointing in the reverse direction. Scaling (wire.0 is active): Right click on the scale widget → **Arrows circles** → Lengths, widths, and colors of the arrows can be changed for the wire-type representation.

For displaying the current distribution on wires as curves click on  in the tool bar.

The following window pops up:

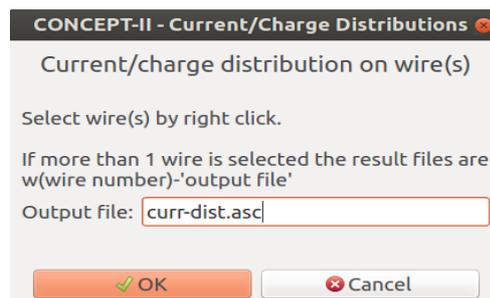


Fig 1.33: Wire current distributions as curves

Right click on the wires the current distribution of which are to be shown. ASCII data is available, see header of the respective files, for example *w1-curr-dist.asc*, *w2-curr-dist.asc*... Clicking on **OK** provides the curve(s) in a gnuplot-type preview window.

For enhanced display possibilities right click on the '*Wire current wire distr.*' entry in the '*View results*' tree. The *CONCEPT-II gnuplot front end* opens.

### 3D radiation diagrams



→ Dialog window for configuring 3D radiation diagrams opens

- Electric far field, EM near field distributions, max. magnitude, theta and phi components, gain in case of power input
- The fields are computed on a spherical grid with step numbers to be specified in directions  $\Theta$  and  $\Phi$  .

The geometrical size of a computed 3D diagram always fits into a bounding sphere of radius 1 .

- Unit of radius according to the chosen scaling system
- Diagram structure is build according to the *CONCEPT-II* patch data format, see Sec. 8.
- Filename: *rad.1* . In case of more than one frequency: *rad.1*, *rad.2*, *rad.3* ... The corresponding data files: *radvs1*, *radv2.2*, *radvs.3*, ...



This information is useful for the following situation: The radiating structure shall be displayed together with its 3D radiation pattern.

- Load structure under investigation in addition to the 3D pattern: press the  button.
- Right click on the surface patch file or the wire file to be scaled (typically *rad.1*, *wire.0*, *surf.0*) → **Move/scale...** (post processing tree, *View results* section) → reduce or enlarge or shift the desired structure part or the 3D radiation pattern, respectively.

Note that in case of a small antenna its structure might be completely inside the radiation pattern. To display this: right click on the entry *rad.1* (*rad.\** in case of multiple frequencies) → **Set opacity** (reduce)

- Graphic file: **File** → **Save image as**

*High precision integration for far field* is checked by default. In most cases it is not necessary to activate this. Use only for structures requiring a high numerical precision, for example patch antennas or for structures with sources inside almost closed cavities. For relatively small structures the computation time does not matter anyway.

- Complex far field data for for E are stored in a file, default name *3DEthetaEphi.asc*, see header of the file for format information.
- Movies of 3D radiation diagrams, the shape is changing with frequency
- Port excitation:
  - The port number (active generator) can be selected, the corresponding 3D diagram will be displayed. Only the chosen port is active.
  - *Modify port voltages* can be can be checked. In such a case the impact of all ports will be computed in a first step (linear system), one after another. Of course this might take a while depending on the size of the radiating structure and the number of ports, computer power, number of chosen and available CPUs and the number of frequencies, if **All frequencies** has been selected.

After the program has finished the generator voltages at all ports can be specified, now feeding the structure simultaneously. This takes only a relatively small time now and can be repeated quickly with different voltages. An example involving three ports is shown in Fig 1.34. Again, when *Show values for all frequencies* is set, movies showing the variation of the diagram shape as a function of frequency can be displayed.

- The port number (active generator) can be selected, the corresponding 3D diagram will be displayed. Only the chosen port is active.
- *Modify port voltages* can be can be checked. In such a case the impact of all ports will be computed in a first step (linear system), one after another.



Of course this might take a while depending on the size of the radiating structure and the number of ports (and the number of frequencies, if **All frequencies** has been selected).

After the program has finished the generator voltages at all ports can be

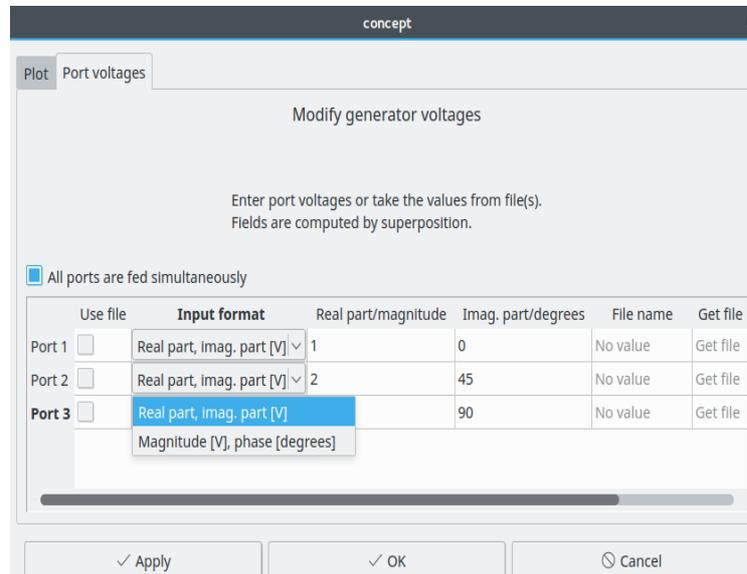


Fig 1.34: A 3D radiation diagram shall be computed with all ports being fed by active generators. Notice that *All ports are fed simultaneously* is checked.

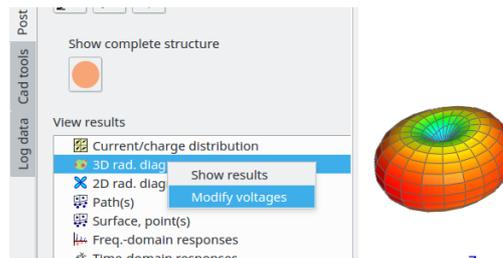
specified, now feeding the structure simultaneously. This takes only a relatively small time now and can be repeated quickly with different voltages. An example involving three ports is shown in Fig 1.34. Again, when *Show values for all frequencies* is set, movies showing the variation of the diagram shape as a function of frequency can be displayed.

Checking *Use file* the corresponding ASCII file has to be loaded. Such a file has to contain the frequency (see *confreq.in*) the real part and the imaginary part in each line (alternatively: frequency magnitude phase).

In case of a further run being necessary with different input data right click onto the *3D rad. diagram* entry (  3D rad. diagram ) of the post

processing tree and select *Modify voltages* (*View results* section as demonstrated in the following figure). The window according to Fig 1.34 opens again enabling the change of input data.





## 2D radiation diagrams



→ Display 2D radiation diagrams

- Electric far field in orbit plane of arbitrary orientation.
- The diagram plane can be displayed together with the radiating structure
- Magnitude, theta and phi components, LHC and RHC, gain in the case of port excitation (power input).
- Port excitation

In principle the same is valid as has been said already, see above, item **3D radiation diagrams**, “Port excitation”. Exception: considering of multiple frequencies is not possible. Only a single diagram per frequency step can be displayed.

## E and H field computation (near field)



→ Computation of EM fields in 3D regions, on arbitrary surfaces, along paths in space or at arbitrary points in space, see Fig 1.35.

**CPUs:** Parallel computation of EM fields; specify the number of CPUs.

### **Method Conventional**

OpenMP is used, in this case “CPUs” designates the number of threads.

### **Method ACA (only PEC, elec. large structures!)**

The ACA technique is based on MPI and cannot be applied if dielectric bodies are present. Only possible in case of electrically large radiating PEC structures.

**Accuracy,, Box size:** Values for fast evaluation by ACA. It is generally not necessary to change the default values. For more information study:

*A. Schröder, H.-D. Brüns, C. Schuster: Fast Evaluation of Electromagnetic Fields using a Parallelized Adaptive Cross Approximation. IEEE Transactions*



*on Antennas and Propagation, vol. 60, no. 12, pp 6058-6061, Dec. 2012*

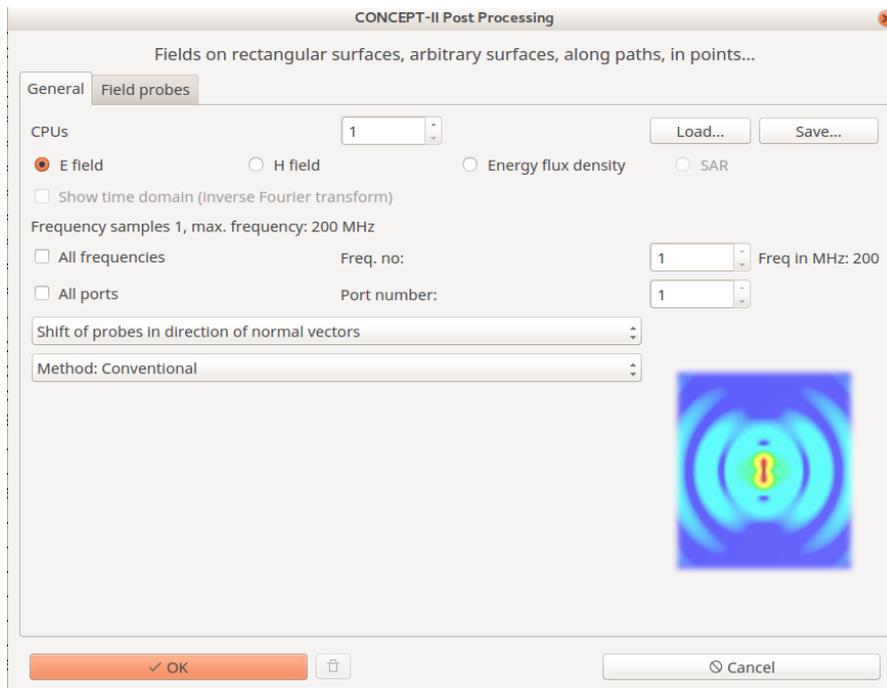


Fig 1.35: Dialogue window for computing EM fields at arbitrary positions

### Method H matrix (electrically large structures!)

This technique for a fast field computation can be applied for both PEC and dielectric structures (bodies). Preconditions: 1) Electrically large radiating structure. 2) Electrically (very) large field region.

### No shift of field probes

Probe positions are directly taken according to the specified node coordinates, which are the centroids of patches in case of surfaces defining the field region.

Sometimes it is advantageous to set the field probes a little bit differently:

### Shift of probes in direction of normal vectors (or against direction of normal vectors)

This is of importance for example when the whole surface patch structure itself is taken as the probe file. The MoM is a boundary element method and therefore the equivalent surface currents are located on infinitely thin sheets. Hence it is necessary to decide on which side of a current carrying surface a probe shall be located, for example on the interior side or on the exterior side in case of a metal box. Find out the direction of the normal vectors by clicking on  (tool bar).

Note that when a shift option is active then all field probes are shifted by a small distance, no matter if they are located on a structure surface or not!



Not setting a shift option may result in a field computation error (singularity problem). The described situation might also happen in case of a rectangular 2D field region intersecting the surface patch structure under investigation.

**All frequencies** is checked.

The selected electric or magnetic field will be computed for all frequencies and all probes.

**All ports** is checked.

The field distributions due to all ports will be computed, one after another, starting with the first port and ending at the last port. This might take a while depending on the size of the structure, the number of probes and the number of frequency steps (if *All frequencies* is checked), the number of available CPUs and, of course, the computer power. Investigation of EM field distributions with all ports being active simultaneous (notice the remarks given for multipoint excitation under **3D radiation diagrams** a few pages above) can be carried out.

In the “**Field representation**” window which opens after completing the field computation the following possibilities are given (valid for generator or plane wave illumination or a single port excitation):

a) Only a single frequency step is considered

Setting a certain number of phase intervals (16 is suggested!) provides the possibility of watching the animated field distribution in the display area. The user will see a repetitive sequence of pictures between 0° and 360°.

b) Check *Show values for all frequencies*

In this case only a single phase can be chosen.

But now it is possible to create a movie of the field distribution for a constant phase as a function of frequency.

Selection of field probes (*tab **Field probes***, see Fig. 1.36).

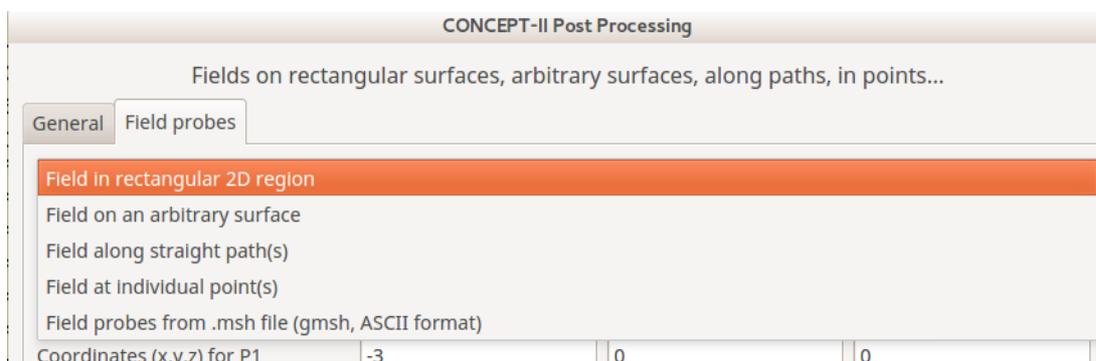


Fig 1.36: Field Probes: several possibilities are available from individual points to arbitrarily shape surfaces.



The drop down list has 5 entries:

- **Field in rectangular 2D region**

Select the desired plane (X-Y, X-Z, Y-Z) and click on **Compute P1,P2,P3**

Choose a suitable mesh. Rectangular planes can be positioned arbitrarily in space. Generated file *surf.eh2d* (surface patch format)

- **Field probes on an arbitrary surface (CONCEPT-II surface format)**

Any *CONCEPT-II* surface patch file may serve as input file. The probe positions correspond to the centroids of patches. Once the 'Name of the surface file' has been specified the 'Name of the result file' is automatically suggested.

An example of the computed normal electric field distribution on a metallic surface is illustrated in Fig 1.37 . Vectors and colors can be arbitrarily scaled and animations are possible.

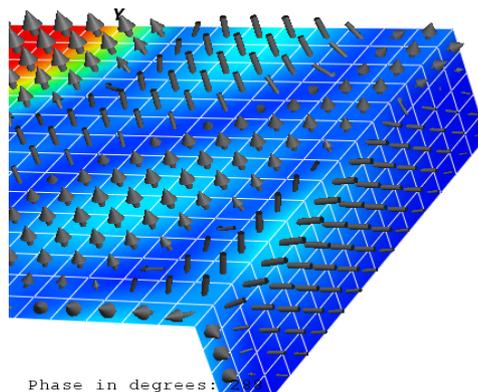


Fig 1.37: E field distribution with 3D vectors computed by the  tool.

- **Fields along straight path(s)**

- **Field at individual points**

If the probe positions are taken from file the format has to be as follows:

first line:        x1 y1 z1

second line:     x2 y2 z2

....

Only in case of check mark 'Only curves as a function of frequency' not set:

Clicking on  (see *Tool bar*) opens a dialog where to specify a file name.

Set the mouse pointer on a field point and right click; the selected point is



marked. The data belonging to the selected point are written into the specified file. The gnuplot front end is used to show the result.

Pay attention that the check marks 'All frequencies' (CONCEPT-II Processing window) and 'Show values for all frequencies' (Field representation window) are enabled!

### Plot system responses as a function of frequency



→ A drop-down list shows the system responses that are available.

Examples: **E(f) (Probes from monitoring)** → Field probes have already been set in the simulation project tree before the back end has been started

**E(f) (Post processing)** → Fields have been computed after the back end run, tool: 

### Plot system responses as a function of time



→ Enables the analysis of a transient system excitation, i.e., computing time-

domain system responses based on an IFT. Precondition is that a special frequency sampling has been selected: **Simulation** → **Frequency** → **Frequency interval for inverse Fourier** transform.

#### Note

In case of an IFT: Both the limits and the proper sampling of the covered frequency interval are important for a successful application. The computation may be time-consuming as a large number of frequency steps might be necessary, depending on various parameters such as typical dimension of the structure, signal duration and rise-time. Details are explained in Section 5 .

### Smith charts



→ Allows the representation of an antenna input impedance in the form of a

Smith chart (program *smith*). Also representation of port impedances.



→ The current distribution on wires can be displayed. Here the corresponding



wires must have been selected under **Simulation** (front end run). Of importance for a sequence of time steps in case of an inverse Fourier transform.

## Network parameters

 → Plot of S,Z,Y parameters in case of (a) port excitation(s), see Fig 1.38. For the definition of ports: **Simulation** → **Excitation** → **Ports (power input/ voltage input)**

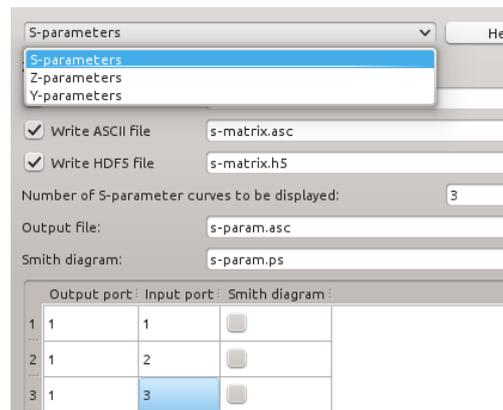


Fig 1.38: Network parameters can be stored in various formats (Touchstone, ASCII, HDF5). Selected input/output combinations are displayed as a function of frequency.

## Power flow

 → Computation of the power flow through a surface (numerical integration of the Poynting vector over a surface). The simplest way to create such a surface is to click **Cad tools** → icon **Sphere** . Only triangles are allowed for the integration surface! Especially useful for validation purposes: input power must be equal to the output power for lossless structures for example.

## Line Integrals of E and H

 → Computation of line integrals of E or H, see Sec. 3.4.

## RCS (monostatic and bistatic radar cross section)



→ The results of an RCS calculation can be displayed. Precondition for

application: back end run finished for this situation:

**Simulation** → **Excitation** → **RCS**

For more information refer to Sec. 3.3 .

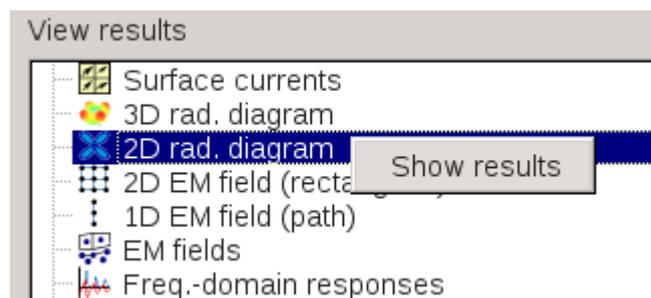


Fig 1.39: Handling of the post processing project tree

To display the whole, completed structure under investigation click on the **Build-geometry** icon . Depending on the kind of magnetic symmetry the icon appears differently.

### Handling of the post processing tree

Right-click on an item, then click on **Show results**. This will load previously computed data into the display area, see Fig 1.39.

Examples:

a) Right click on the entry **Freq.-domain responses** → **Show results** → The most recently generated curves are displayed again by the *CONCEPT-II* Gnuplot front end.



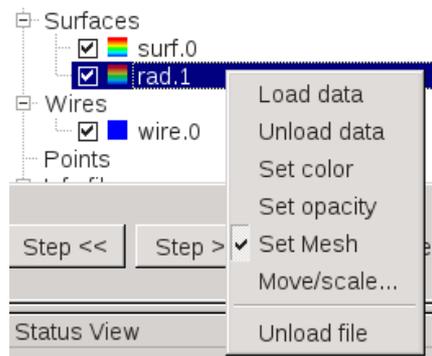


Fig 1.40: Drop-down list that opens on right mouse click

b) A 3D radiation diagram, which has previously been computed can quickly be loaded again and scaled for a proper representation together with the structure including its currents. Once a 3D radiation diagram has been loaded, the entry **rad.1** (file with surface patch format) appears under **Surfaces** in the post processing tree (Fig 1.40).

**Surf.0** and **wire.0** are default names which are not allowed to be used as input files as has been mentioned earlier. *Surf.0* and *wire.0* comprise the whole structure with symmetrical parts; they are automatically build from the input files specified the simulation project tree.

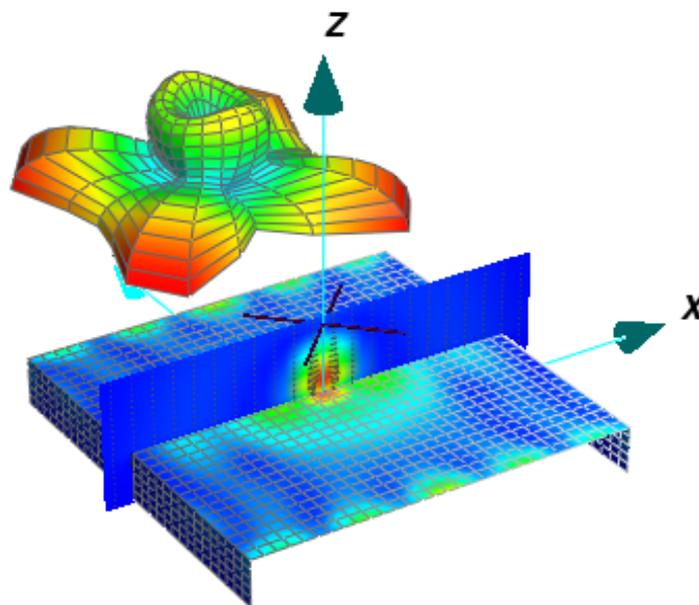


Fig 1.41: Table antenna example with current distribution, 3D radiation diagram and 2D field distribution. Radiation pattern and radiating structure can be scaled and separately shifted for a suitable representation.

Clicking on **Move/scale...** opens a dialogue by means of which structure parts or



3D radiation patterns can be scaled or shifted. Important: this does not change the structure itself; it is only for visualization purposes. Notice further that a 3D radiation diagram is scaled into sphere of radius 1 (center corresponding to the origin of the coordinate system, unit according to the scaling system). Therefore the size and the position of a 3D radiation pattern need to be adapted for the purpose of a good representation in most practical cases. An example is shown in Fig 1.41 .

Right clicking onto the **Surfaces** entry of the post processing tree, see Fig 1.40, offers the possibility to load additional surfaces to be displayed. By means of **Load data** corresponding field or current distributions can be visualized (see Section 1.10).

## 2 Example: Top-loaded wire antenna on a table structure

The handling of the GUI shall be demonstrated by an example.

The input files of the structure under discussion can be found here:  
*\$CONCEPT/examples/Table\_antenna.*

Further examples are in the *\$CONCEPT/demo* directory; pdf files can be downloaded from here: <http://www.tet.tuhh.de/concept/demo-version/>.

A top-loaded monopole antenna which is mounted in the center of a metallic platform shall be analyzed. The whole structure is placed on an ideal ground plane. The antenna platform has a height of 0.5 m and consists of a top plate and two side plates being connected to the ground. The size of the top plate is 2m x 2m. The length of each wire is 0.75 m. There is an additional wire between the top plate and ground modeling a coaxial feed cable. The electrical and geometrical symmetries involved in the configuration shall be exploited and the frequency interval between 1MHz and 100 MHz is of interest. The structure including computed results has been demonstrated already in Fig 1.41.

The whole arrangement could be discretized by means of the *CONCEPT-II* CAD tools (notice that the **CAD tools** tab is active according to Fig. 2.1, we have *top.surf*, *side.surf*, *antenna.wire* and *coax.wire* loaded into the CAD section). As the vertical antenna wire shall be fed at its base (*antenna.wire*), magnetic symmetry can be exploited with respect to the x-z plane and the y-z plane. Hence there is only need to specify the geometry which is located in the first quadrant of the Cartesian coordinate system as illustrated.

### Discretization of surfaces by gmsh

In the example directory the ASCII file *Table\_antenna.geo* is available for usage in conjunction with the *gmsh* tool (<http://geuz.org/gmsh>), see also Secs. 3.7 and 1.6.

Command line, use *gmsh* as follows: `gmsh -2 Table_antenna.geo <CR>`



The resulting surface patch file with suffix `.msh` can be converted into the *CONCEPT-II* surface patch data format (*Menu bar, Import/export*) or can be directly loaded, see entry **Surfaces** of the simulation project tree.

**Simulation** → **Surfaces** → **Load surface file(s)** → select **Gmsh files** → select *Table\_antenna.msh*

Loading the `.msh` file two surfaces appear in the simulation project tree: *Table\_antenna-msh1.surf* and *Table\_antenna-msh2.surf* automatically created from the `.msh` file; renamed here as *top.surf* and *side.surf*; right click on the surface file name, select **Rename** (this step is not necessary). We get two surface files as in *gms*

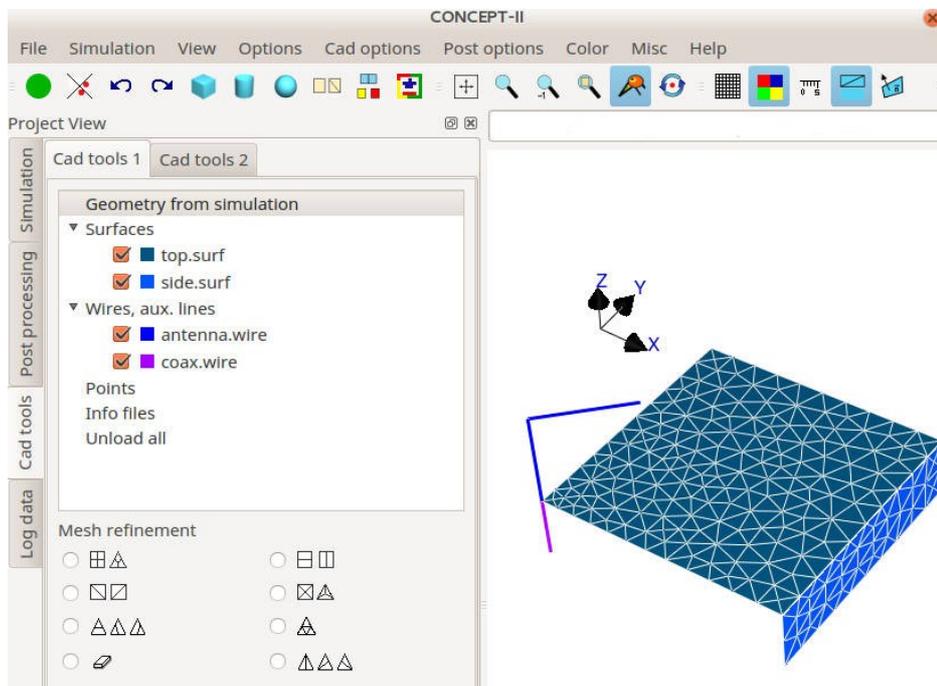


Fig 2.1: The top loaded antenna with supporting structure. Mesh created by *gms*. Fig 1.41 shows a grid of quadrangles, created by a *CONCEPT-II* tool (  ).

Use an editor and change the frequency in the head of the `.geo` file, run *gms* again to get the new `.msh` files with the adapted grid.



## Note

General rule: in the case of magnetic symmetry only the symmetric part of the structure needs to be discretized.

The ideal ground plane always corresponds to the x-y plane.

1. A magnetic symmetry plane is characterized by E field vectors which are completely tangential in the plane of symmetry. On the other hand the H field vectors have only perpendicular vector components there. Visualize the ground plane by clicking on  (tool bar).

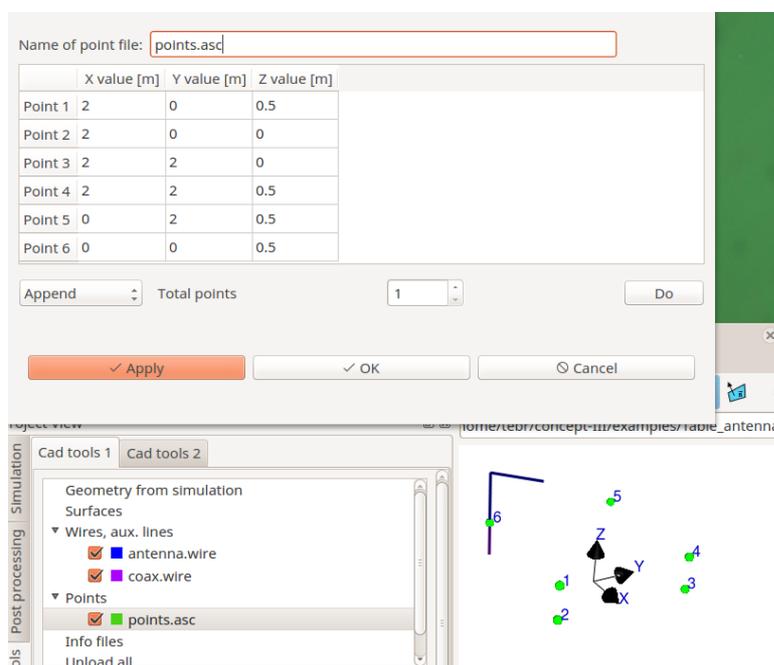


Fig 2.2: Corner nodes of the antenna table

## Discretization of surfaces using CONCEPT-II tools

Click on tab **Cad tools**. There are several possibilities for discretizing the surface parts of the example under discussion, for example:

- **Points** (cad tree) → **Edit a new point file** and enter the coordinates of all corners as demonstrated in Fig 2.2.
- Switch on the radio button '*Plate, arbitrary quadrangles*' and select the four points (1-4-5-6 or 5-6-1-4 ..., clockwise or counterclockwise selection, the starting number is arbitrary) by right clicks.  → the window '*Plate,*



*arbitrary quadrangles*' opens. Set the frequency to 200 MHz (8 basis functions per wavelength) and click **Compute**. Enter the file name (here *top.surf*) as suggested in Fig 2.3, then **OK**. If it should be necessary to deselect a selection click on  (tool bar). Meshing in the described way is based on the plate tool.

Note: The plate tool  can discretize arbitrary planar or non-planar quadrangular areas, also triangular areas. Steps: tab **Cad tools** → tab **Cad tools 2** → 

- Next discretize the side plate in the same way.
- Click on  → a triangular grid appears on both surfaces.

-----  
Hint: The meshing can also be done by the external tool *gmsh* (see for example Sec. 4.8)

> *gmsh -2 Table\_antenna.geo (enter at the prompt)*

→ The file *Table\_antenna.msh* is generated containing the surface patch grid. This file can be directly loaded into the simulation (... **Surfaces** → **Load surface files(s)** → 'Files of type:' **Gmsh files (\*.msh)** ...

→ For a higher frequency edit the ASCII file *Table\_antenna.geo* and change the frequency accordingly.

Starting with Oct. 18 the ASCII versions 2.0 and 4.0 of the .msh format are supported.

- For introducing the wires the following section under the tab **Cad tools 1** can be used:



If the end and the start coordinates of wires are sufficiently close to each other a galvanic connection is recognized ( $10^{-5}$  of the used unit). The node coordinates can be entered directly or by clicking on existing nodes (click on **New node** → dialogue window opens, notice that at least two nodes have to be defined before a new wire is created by clicking on **Apply**). It is also possible to enter wires directly under **Simulation** → **Wires** → **New wire file ...**



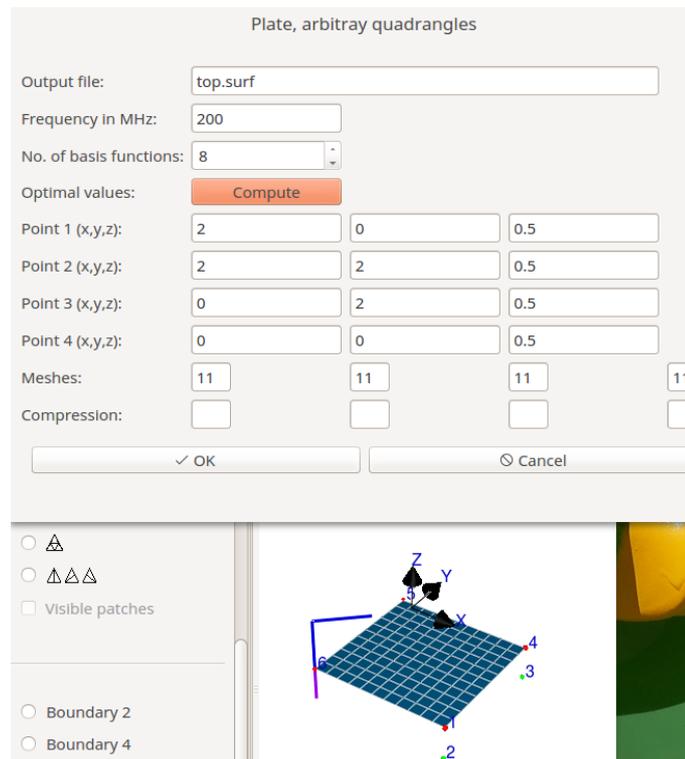


Fig 2.3: Meshing of the top plate.

Next click onto the **Simulation** tab (*Project view*) -> right click **Table\_antenna** (top entry) → **Load all files from 'CAD'**.

Continue the input process by moving through all menu items of the simulation project tree. According to Fig 2.4 magnetic symmetry planes and the option 'ideal ground' have been set (see check marks under **Symmetry**). If a magnetic symmetry plane has been selected this is indicated by a yellow arrow, appearing in the top left corner of the display area. If **Ideal ground** is active (an ideal ground can only be located at  $z=0$ ) this is indicated by the following symbol in the display area:



(change of position: <Shift> + click on the mouse wheel, middle mouse button).



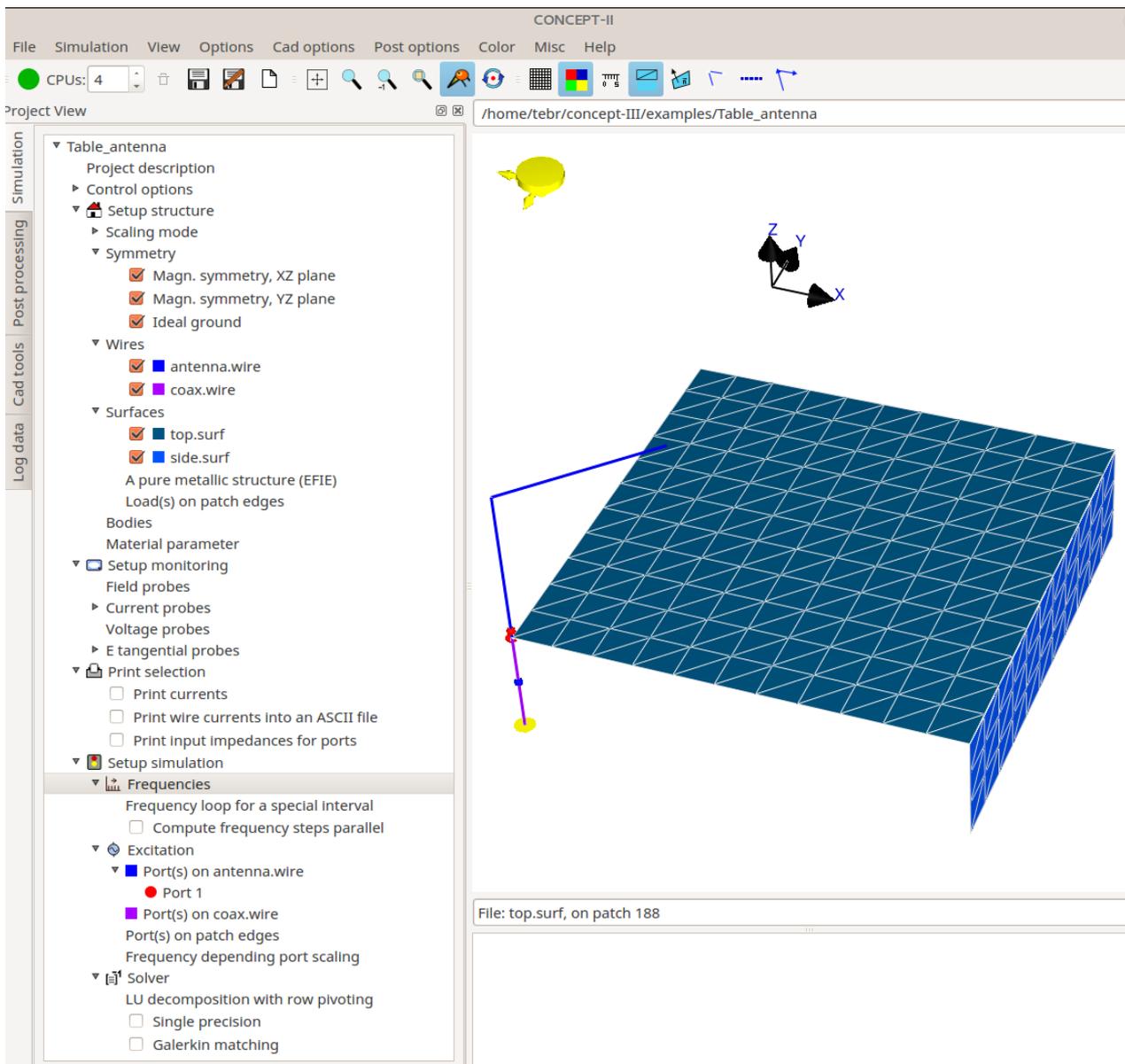


Fig 2.4: All entries of the simulation project tree for the table antenna example

Specify the wire segmentation: Select a loaded wire file, for example **coax.wire** ( Fig 2.4 ) → **Edit wires** → **No. of bf**.

A frequency dependent automatic wire segmentation is based on the '*Number of basis functions (bf) per wavelength*', where 8 is the smallest possible value, see also see Sec. 1.11.2.3.

It is essential to notice that for **electrically short** wires a wavelength dependent segmentation is not applicable. Thinking in terms of electrostatics is useful. The user should answer the following question: How many charge blocks (approx. equal to the number of basis functions) are required for a good approximation of the normal electric field strength distribution along a wire? The charge distribution is approximated by a stair-case function in the numerical simulation. From



electromagnetic theory is known that we have a rapid increase and variation of the charge density distribution at corners and edges. For a convergent current distribution we have to capture the impact of an integrated charge distribution in a sufficient manner. For electrically short monopole antennas select 10,...,12 basis functions at least, for wire grids consisting of numerous electrically small sections three basis functions per wire are sufficient. The same holds for electrically small wires connecting metal plates and so forth.

The simulation project tree with all entries completed is illustrated in Fig 2.4.

### Setting current probes on wires

Later these currents are available to be displayed as a function of frequency.

Notice the small blue dot in the middle of the vertical wire below the antenna platform

( Fig 2.4) . **+Current probes** → **Current probes on coax.wire** → right click or double click → **Set probes** → dialogue window → **Append** → '*Wire number*' 1, **middle segment**.

Alternative method: Select the wire by mouse and right-click on it.

In general three locations are possible for probes on wires: first segment, middle segment and last segment (position has to be chosen in drop-down list of the '*Position*' column).

#### Note

Wires for the placement of probes can be selected by right mouse clicks. The specific location (beginning, middle, end) has to be entered in the in the input window.

### Frequency sampling

A frequency loop shall be specified: 10,20,...,200 MHz. **Frequencies** → right click or double click → **Frequency loop for a special interval** → a dialogue window opens → '*Frequency unit*' **MHz**, '*Minimum value*': 10, '*Maximum value*': 200, '*Interval type*' **By step** → *Step* 10 → **Generate list** → All frequency samples appear under '*Used frequencies*'.

Notice that only frequency samples explicitly listed are considered in the simulation after having clicked on **OK**. Samples can be added manually.

### Excitation

A power generator providing 1 W forward power shall be placed at the beginning of the vertical antenna wire close to the antenna platform.



Double click **Excitation** → **Ports (power input/voltage input)** → double click **Port(s) on antenna.wire** → **Set port(s)** → **Append** → 'Wire number' → 1, 'Position' → **1st segment** → 'Power input [W]' → 1 (Again wire selection for positioning of a power generator can be done by mouse).

Finally the number of CPUs (see left side of the *Tool bar*) has to be entered. Then click on the run back end icon .

Before starting the simulation the structure should be carefully inspected!

## Note

Modification of various elements (dots size for probe positions, colors, length of axes etc.):  
*Menu bar: Options* → **Modify elements**

## 2.1.1 Post processing

The system current distribution is known once the back end has finished. It is a good idea to start the post processing by displaying the structure current distribution for a first validation. Enable **Post processing** tab.

### 2.1.1.1 Representation of the current distribution

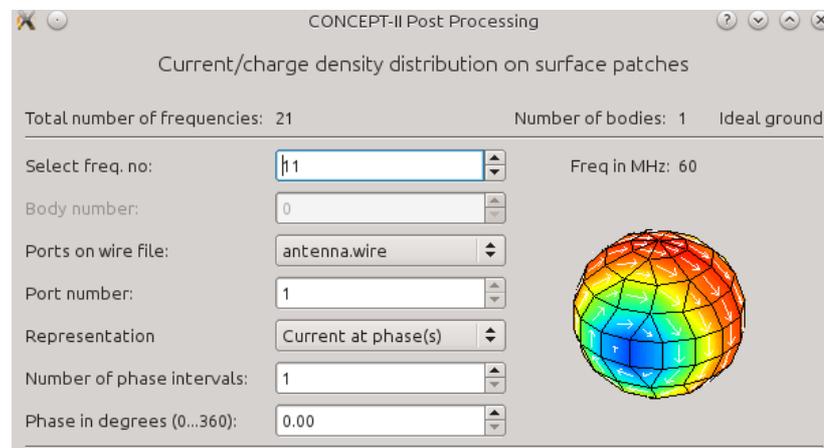


Fig 2.5: Currents on surfaces and wires shall be displayed.

Click on , a dialogue window as depicted in Fig. 2.5 appears:

'Number of phase intervals'

For creating a movie 16 phase intervals are a good choice in general.

'Phase in degrees (0...360)' → Only active in case of a single phase.

An example current distribution is shown in Fig 2.6. Moving the mouse pointer over patches provides the corresponding values of the current density in the message line below the display area (  enabled, *tool bar*). The colors are representing the magnitude of the current distribution for the chosen phase.

### Some features:

*Menu bar*: **Options** → **Draw wires as...** → **Cylinders**

Wires are shown as 3D cylinders (for a better control of the thin-wire assumptions which may be violated...) . Depending on the radii it might be necessary to turn the mouse wheel for zooming in.

**File** → **Save image as** → The contents of the display area can be stored in a number of different graphics formats, an example is shown in Fig 2.6.

Right click on the '*Scale widget*' section right to the display area → a drop down list appears. Various color maps are available, default is **Rainbow Colors**. For vector scaling click on **Arrows, circles...**

Fig 2.6 is drawn using the **Hot Color** map with '*Arrows in 3D*', '*unicolored*'. Hot colors are suitable for black and white printing, the default setting is **Rainbow colors** .

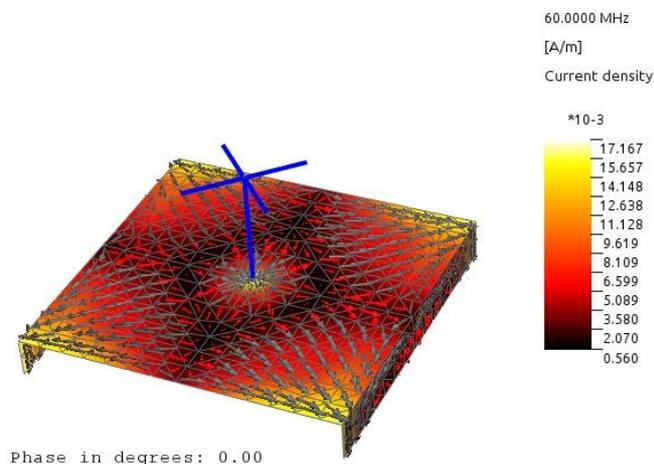


Fig 2.6: Current distribution on the example structure at a certain phase.

### 2.1.1.2 Representation of electromagnetic 2D field distributions

Click on  'Compute Results' Section: → a window according to (Fig. 2.7) opens.

First enter the data required under the tab **General**: type of field, number of CPUs



and so forth.

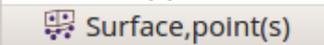
For the quick creation of a 2D field region: **Field probes** → **Field in rectangular 2D region** → A rectangular plane can be defined by points P1,P2,P3 (notice the small drawing in input window).

Select a suitable coordinate system plane (**XZ plane** for example, then click **Compute P1, P2, P3**) or enter the corner points by hand, then fill in the number of meshes, for example by setting an appropriate number between P1 and P2 and between P1 and P3. It is recommended to check the spatial position of the probe plane by clicking on **Show plane/Create surf.eh2d** before starting the field computation. The number of meshes that are necessary for a suitable field representation depends on the frequency, the kind of representation (color map, vectors), the complexity (variation) of the field distribution, and the size of the area to be displayed. For electrically large regions it might be advantageous to first specify the '*No. of meshes/wavelength*'.

Clicking on **OK** causes the selected field type to be calculated by integrating over the whole structure current distribution for each field point. Each probe location corresponds to the center of a surface element of the file *surf.eh2d* (case of rectangular field region). This special surface patch file is generated automatically.

It is emphasized here that EM fields can be computed on surfaces of any shape containing both quadrangles and triangles.

Once the field computation has terminated the type of graphic representation can be specified ('*Field representation*' window opens). Automatically enabled is '*Draw contour for rectangular 2D regions*' for displaying contour lines in case of an intersection of the 2D field display area with the radiating or scattering structure, file: *contour.asc* (wire format, only for planar rectangular field regions). Fill in all entries that are necessary and click on **OK** → the computed field distribution will be shown in the display area based on some default representation settings. For an improved vector and color scaling right click onto the *Scale widget* (right side of the CONCEPT-II main window) → **Arrows, circles...** For details about '*Show small values in linear scale*' see Sec. 8.1.

If there is need to (later) repeat the representation select the following menu item of the post processing tree (*View results* section):  → right click

→ **Show results.** The stored complex field data (default name for the E field: *E-Field-surf\_eh2d.h5*) is used without running a new, probably time-consuming field computation.



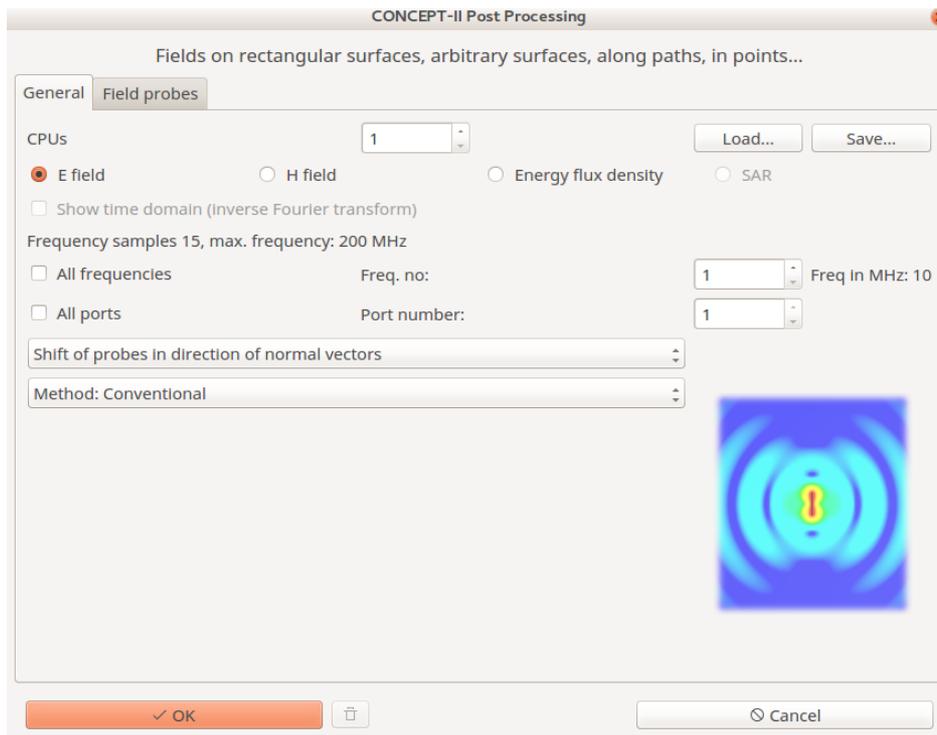


Fig 2.7: Compute electromagnetic fields in 2D regions. Under **Field probes** there are several possibilities for setting field probes: 2D regions, on arbitrary surfaces, along straight paths, at arbitrary positions in space and so forth.

The file name for saving the field data can be specified (hdf5 format).

Comment on validation: Randomly distributed arrows may be observed at very low frequencies (electrically very small structure). Example: H field distribution of a short dipole antenna, phase: 0 degree. CONCEPT-II might not be able to compute the very small radiation resistance properly. Note that the code is working in the frequency domain. Setting the phase to one or a few degrees gives a physical meaningful field distribution already.

### 2.1.1.3 EM fields along straight paths or at arbitrary locations in space

#### Fields along straight paths



has been selected, notice all input fields under **General**. Click on **Field probes**

→ **Field along straight path(s)** → the spatial dependence of electromagnetic fields along straight lines in space shall be computed. It is suggested not to set paths or field points too close to a surface – the minimum distance should be half the typical



patch size of that surface. Otherwise the piecewise constant surface charges may become visible in the corresponding curve leading to kind of a step function. For reliable EM fields very close to surfaces a local mesh refinement might be required!

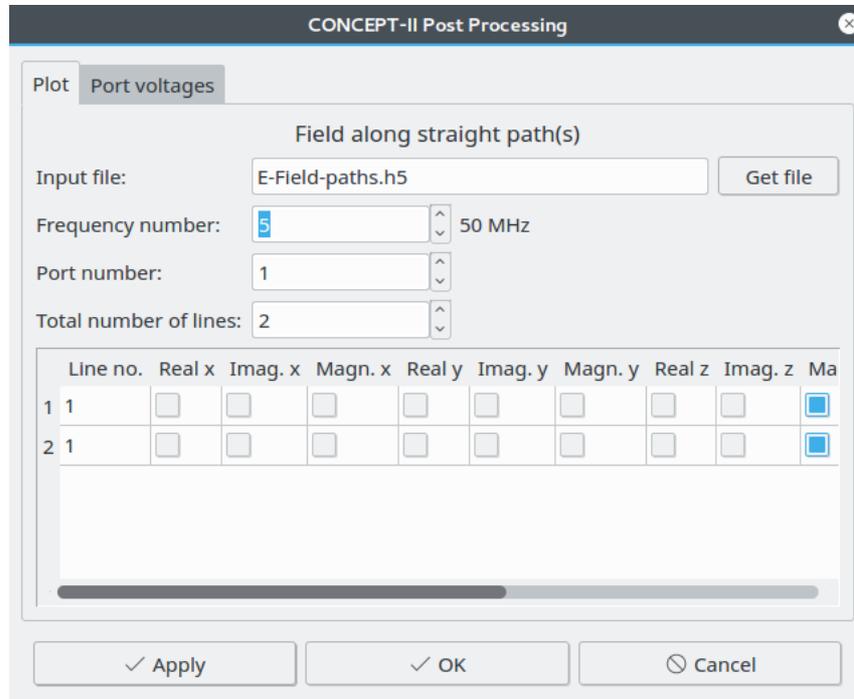
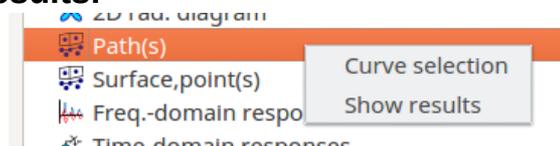


Fig 2.8: The magnitude of the E field shall be displayed along two paths in space for 'Frequency number' 5 (Tab **General** of , check mark 'All frequencies' has been enabled before

Once the field computation is done the user can select the field component(s) to be displayed (see Fig 2.8 ). Let us assume that a curve is shown. Clicking into the display area opens a dialogue windows for saving the data. Default file name for E: *e-path(n).asc* with n as the path number.

In a later step the curves can easily be replotted (see the figure below): select **Path(s)** → **Show results**.



**Path(s)** → **Curve selection**: A window according to Fig 2.8 opens, further field components can be selected to be displayed.

## Fields at probe locations (points in space)



When activating **Field at individual point(s)** (📍, tab **Field probes**) the user can enter points manually or load a file with probe coordinates. The data format is:

1st line      x y z (*coordinates of the first field point*)  
 2nd line      x y z (*coordinates of the second field point*)  
 3rd line ...

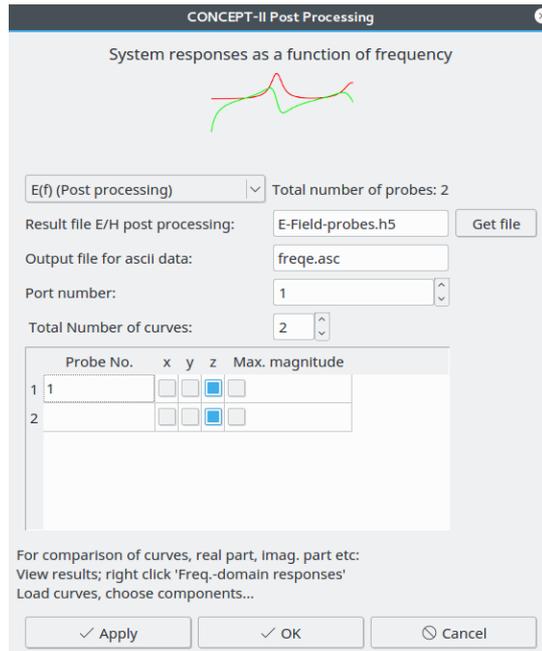


Fig 2.9: Dialogue window for the specification of system responses as a function of frequency. Here the z components of the E field at two probe positions are selected. In the display area the magnitudes of the z components are shown as a function of frequency once the OK button has been pressed. Note that the output file is *freqe.asc* (e for E field) is stored in the current working directory. For the data organization see header of this ASCII file.

Click **Show probe(s)** for displaying the probe location(s).

📍, tab **General**, assumption 'All frequencies' enabled:

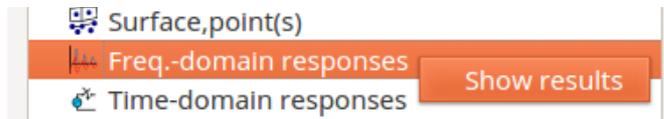
Case 'Only curves as a function of frequency' is checked, click on OK

A window according to Fig 2.9 opens after field program execution, only magnitudes will be displayed.

For an extended curve representation (function of frequency) use the post



processing tree. Right click onto the entry **Freq.-domain responses** (*View results* section) as shown here:



Now real part, imaginary part, magnitude, phase can be chosen. Of course, if 'Max. magnitude' was selected only a single curve is possible.

Case 'Only curves as function of frequency' is not checked

Small spheres are placed at the probe locations. The color is representing the scalar field value, see corresponding map in 'Scale widget' section. **Animate** → the color of the spheres is changing as a function of frequency or phase, depending on what has selected.

In case of a frequency loop, fields can be displayed as a function of frequency as follows: click on  (tool bar) → dialogue window opens →

enter file name → select probes by right clicks → **OK** → the magnitude(s) of the field(s) at the chosen phase or the max. magnitude is (are) displayed

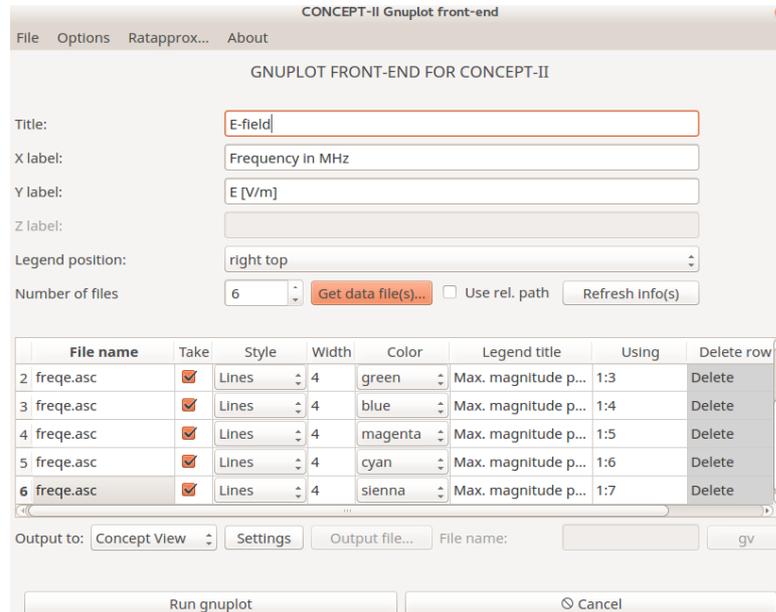
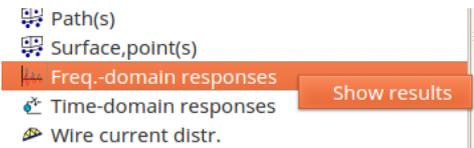


Fig 2.10: The Gnuplot front end of *CONCEPT-II* as a function of frequency.

In this case the  tool can be used directly, but then it is necessary to specify the



file name where the E field data were collected (default is *E-Field-probes.h5*).

Repeat curve representation:  . The curves are

shown by the CONCEPT-II gnuplot front end, see Fig 2.10 as an example. Desired curves ('Take' is active) are displayed. Images of the curves can be stored (**File** → **Save image as ...**)

### 2.1.1.4 Some features of the Gnuplot front end

Several curves in a single plot are possible, for example: column 4 (magnitude) vs. column 1 (frequency) → enter 1:4). Load the same file several times and choose different columns to be displayed in each case. Settings → Modify '*Font size*' and '*Font name*'. **Options** → activate '*Log y axis*' for showing the magnitude...

A curve is not shown if the '*Take*' check box is deactivated.

Data can be modified for representation: Example: 1:(\$2\*1000) means that all data of the second column is multiplied by a factor of 1000. 1:(20\*log10(\$2)) means that the values of the second column are displayed in dB. For further Information see the Gnuplot documentation.

Setting the mouse pointer over the *Using* input field displays the head section of the corresponding ASCII data file, indicating the contents of all columns.

### 2.1.1.5 Creating 2D radiation diagrams

For 2D radiation diagrams click on  (tab **Post processing**)

The diagram plane can be modified by two angles for tilt and rotation as indicated by the small drawing in Fig 2.11. The default number of CPUs to be used is preset according to the number as chosen under **Simulation**.

The required '*No. observation points per 180 degrees*' for a good curve presentation depends on the complexity of the radiated field distribution along a full circle or semicircle, which might include a number of minor lobes. Field data (Maximum magnitude, gain,  $E_{\theta}$ ,  $E_{\phi}$  ...) is stored in files, the names of which can be chosen arbitrarily. For the data format notice the head sections of the corresponding ASCII files.

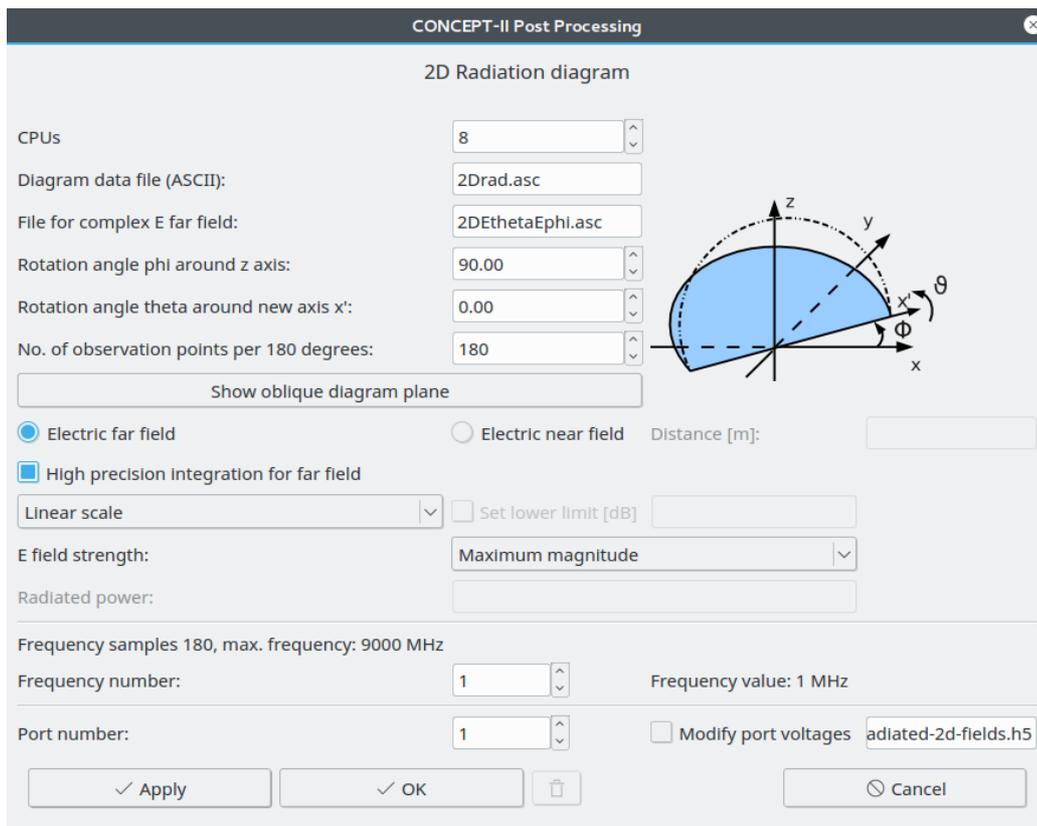


Fig 2.11: Dialogue window for 2D radiation diagrams

Clicking on **Show oblique diagram plane** depicts the diagram plane together with the structure under study. A rotation of  $\Phi = 90^\circ$  around the z axis leads to a 2D radiation diagram in the y-z plane (for explanation please refer to the drawing in Fig 2.11)

In case of *Maximum magnitude*  $|E|$  radiation pattern data is given in [V]. To get the E field strength value in a certain direction simply divide by the distance of the far-field probe.

For a near field computation the radius of the circular diagram plane must be entered. The center of the diagram plane refers to the center of the structure bounding box.

In case of 'Electric far field' being active the following possibilities exist (see drop-down list 'E field strength'):

- Maximum magnitude (of the polarization ellipse)
- Theta component
- Phi component
- LHC (left-hand circular polarization)
- RHC (right-hand circular polarization)



- Axial ratio
- Power gain
- Directive gain

Data representation: **Linear scale, dB scale, Normalized dB scale**. When a dB scale is active an artificial lower limit can be set.

When a diagram is displayed, a mouse click into the display area enables the possibility to save the diagram data. To compare the curves of different investigations: Open the gnuplot front end (  ) and 'Get data files(s)...'.

Already existing results can be displayed by right clicking or double clicking on  2D rad. diagram ('View results' section). Hence for a later, repeated

representation there is no need to do the computation again. This time the CONCEPT-II gnuplot front end opens enabling a modified curve representation.

Case '*High precision integration for far field*' is active: The contribution of each patch is computed based a 9-point-numerical integration (if not active only a single point per patch is taken → faster computation). It turned out that in the vast majority of cases a single-point integration is sufficient, notice tooltip.

LHC, RHC: A plane wave can be represented as a linear combination of a right circularly polarized wave and a left circularly polarized wave as known from theory. In case of a linear polarization the amplitudes of the LHC and RHC components are equal (curves lie on top of each other).

For linear polarization the maximum value for the axial ratio is set to 1000!



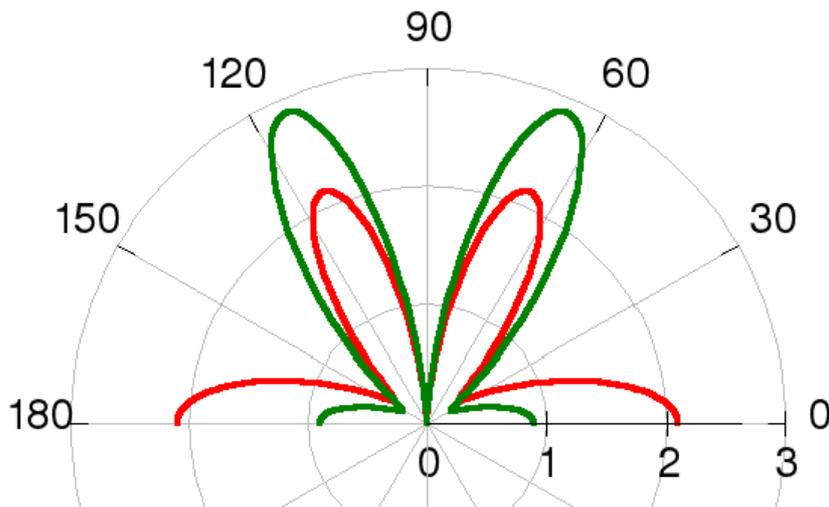


Fig 2.12: Power gain of the table antenna at 100 MHz. Green curve: y-z plane, red curve: x-z plane. The max. gain is 2.13

Fig 2.12 illustrates the power gain of the top-loaded table antenna according to Fig 2.6 at 100 MHz.

The **Power gain** is computed according to [Stutzman]:

$$G = 4\pi \frac{U}{P_{\text{in}}} \quad \text{with}$$

$P_{\text{in}}$  as the input power,

$$U(\Theta, \Phi) = \frac{r^2}{2\eta} |\vec{E}(r, \Theta, \Phi)|^2 \quad \text{as the radiation intensity, } \eta: \text{ impedance of free space,}$$

and  $\vec{E}(r, \Theta, \Phi)$  as the far-zone electric field intensity. Input power can be specified in conjunction with port excitation (**Simulation** → **Excitation** → **Ports (power input/voltage input)**)

**Directive gain** is computed according to  $D = \frac{U(\Theta, \Phi)}{U_{\text{ave}}}$  with

$$U_{\text{ave}} = \frac{P_{\text{radiated}}}{4\pi} \quad \text{and } P_{\text{radiated}} \text{ as the radiated power. Note that in the case of losses}$$

the radiated power is not equal to the input power. The radiated power can easily be determined by computing 3D radiation diagrams (see next chapter). Note further that **directivity** (diagram output) is defined as the maximum of the directive gain.

In case of a multi-port excitation 2D diagrams can be computed as the result of the superposition of the impact of all generators. Remember that at multiple port

excitation each port is considered separately (see Sec. 1.11.2.6.2 Ports) , leading to its own port-specific current distribution, all other ports are short circuited. Click on  **Modify port voltages** and then on **OK** (notice the result file name). A new window

opens: specify here the frequency number, type of curve representation (linear, dB), **Maximum magnitude**,...,**Directive gain**. In case of directive gain the '*Radiated power*' is necessary, see equations above. This value can be determined by a preceding 3D diagram computation including the same set of voltage generators. Tab **Port voltages**: The voltage of each generator can be entered here in various input formats. Set '*All ports are fed simultaneously*' to active!

[Stutzman] Warren L. Stutzman, G. A. Thiele: Antenna Theory and Design, John Wiley & Sons, 1981

### 2.1.1.6 Creating 3D radiation diagrams

A lot of remarks were quite the same as already given for the case of 2D diagrams in the last chapter. So they are not repeated here.

For 3D radiation diagrams it is necessary to specify a uniform grid of field observation points on a fictitious sphere of unit radius. An appropriate spherical section can be defined by the angles "THETA" and "PHI", referring to a spherical coordinate system (Fig 2.13).

In general, with increasing electrical size of the radiating structure a larger number of grid points is necessary to have a diagram including all minor lobes which might appear. A simple rule for an appropriate number of points cannot be given.

Approximate rule: 20 X 20 elements for electrical small structures, 60 X 60 elements and more for electrical large structures.

The structure of a 3D radiation diagram is based on surface patch format and its default size fits into a fictitious sphere of radius  $R=1$  m. This information is helpful for rescaling when the radiating structure and the diagram shall be depicted in the same figure (see below).



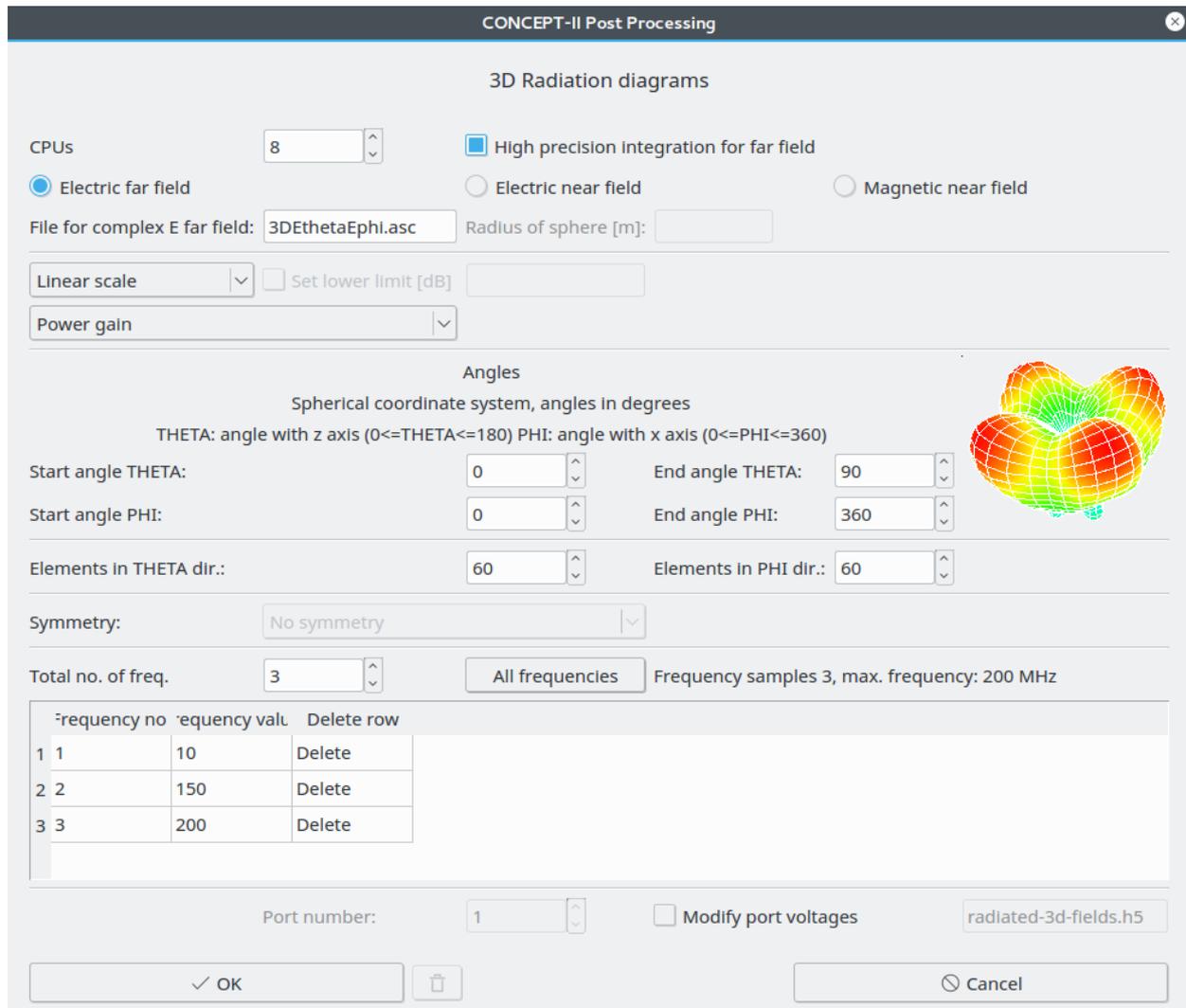


Fig 2.13: Dialogue window for creating 3D radiation diagrams

Enabling **All frequencies** the shape of the 3D diagram can be animated as a function of frequency after all diagrams have been computed.

The computations are carried out based on the following equations:

$$P_{rad} = \frac{1}{2\gamma} \iint (|\hat{E}_\theta|^2 + |\hat{E}_\phi|^2) r^2 \sin \Theta d\Theta d\Phi, \quad D = \frac{4\pi U}{P_{rad}}, \quad U = \frac{r^2}{2\eta} |\vec{E}(r, \Theta, \Phi)|^2$$

D: Directive gain, U: radiation intensity,  $\eta$ : impedance of free space.  $P_{rad}$ : radiated power.

As a numerical simulation is carried out, a small deviation has to be accepted comparing input power and radiated power in case of lossless structures. This

deviation should not be larger than 5% ... 10 %, very good are values of about 1 % and even smaller. From experience is known that the following ratio should hold in case of a far field based analysis:

$$\frac{\text{Radiated power}}{\text{Input power (ideal conducting structure)}} \approx 0.90, \dots, 1.1$$

The radiated power is listed here: Tab **Log data** →  [3D rad. diagram](#) . ASCII file *radiate.out*.

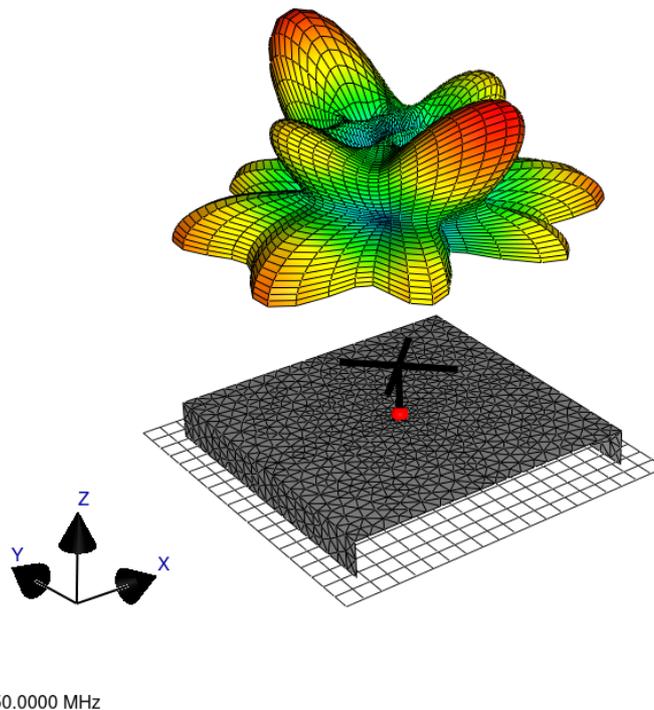


Fig 2.14: Example of a 3D radiation diagram (top-loaded antenna on a table, 150 MHz). For a representation as shown (the rectangular grid at the bottom is indicating the ideal ground and not part of the structure, click on icon in tool bar):

Click on ‘*Show complete structure*’  → surf.0 and wire.0 are loaded, representing the complete structure. Double click on  **rad.1** (surface file of the 3D diagram) →

**Move/scale...** → Enter: ‘*Scaling factor*’ = 3, ‘*Move z direction*’: 4

If the ratio is not within the indicated limits, the mesh (the discretization) should be investigated carefully and improved at critical locations. Perhaps the grid might be bad or too rough near the generator, the structure is excited close to or at a

resonance frequency... (investigate the current distribution, where are randomly distributed currents etc. ) .

In case of a near field computation (only magnitude of E or H) the '*Radius of sphere [m]*' has to be entered. This radius refers to a sphere with its center corresponding to the center of the structure bounding box. Note in case of a near field computation: the value under '*The total radiated power*' in *radiate.out* may be wrong if the radius does not correspond to a typical far field distance. Reason for this: Although a near field equation is used to compute the 3D E field distribution, far field expressions (E,H) are still used for the determination of the radiated power (status Jan. 18). If interested in the near field power flow refer to  (power flow tool).

An example of a 3D radiation diagram is depicted in Fig 2.14, together with the radiating structure.

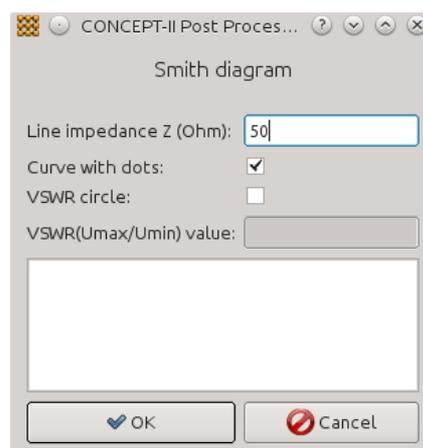
The format of the '*File for complex E far field*' (default name *3DEthetaEphi.asc*) can be taken from the head section of the file.

### 2.1.1.7 Creating Smith diagrams

Click on ; a window as illustrated in Fig. 2.15 opens.

The characteristic impedance of the voltage generator feed transmission line has to be entered. Clicking on 'Run smith' automatically generates the PostScript file *smith.ps*, which is displayed in an extra window.

Fig 2.15: Window to create Smith diagrams



## 3 Important topics

### 3.1 Dielectric bodies, types of surfaces

The following surface types are available:

#### Surface type 'Dielectric boundary'

This is a surface separating two regions, the boundary conditions  $E_{t1}=E_{t2}, H_{t1}=H_{t2}$  are enforced, leading to a double-layer current distribution on that surface, see *surf.2* in Fig 3.1 as an example ( $\vec{J}_{internal}, \vec{J}_{external}$ ). Each current layer has an impact only to the assigned side.

The number of unknowns doubles compared to a single-layer current distribution ( $\vec{J}$ ) which is computed on simple metallic surfaces where only  $E_t=0$  is matched (EFIE problem), see surface type 'PEC in free space' (*surf.3*) and 'PEC in body 1' (*surf.4*) in Fig 3.1 .

#### Surface type 'Metallic boundary'

This is an ideally conducting surface separating two media. The boundary condition  $E_{tan}=0$  is matched separately on both sides as well leading to a double-layer current distribution. An example is *surf.1* in Fig 3.1, again note the current distributions  $\vec{J}_{internal}, \vec{J}_{external}$ , each one is 'responsible' for the assigned region.

#### Surface type 'Free space'

This is a surface in free space, outside a body; an example is *surf.3* in Fig 3.1. Such a surface carries a single-layer current. The boundary condition  $E_{tan}=0$  is applied. This type of surface can be galvanically connected to a surface of type 'Metallic boundary', which partially covers a dielectric body.

#### Surface types 'In body 1' ('In body 2', ...)

Metallic surfaces that are located inside body 1, body 2, body 3 etc.. Such surfaces can be connected to 'Metallic boundary' type surfaces and are treated by the electric field integral equation only (single-layer current distributions) similar the 'Free space' type.

---

In the case of an internal surface forming the common boundary between two bodies which are attached to each other, the necessary discretization depends on the smallest internal wavelength which, in general, is smaller than the free space wavelength due to the internal material properties (rule: 8...10 patches per internal wavelength). The same is true for all other surfaces forming body-internal scatterers or radiators, also wires.



**Note**

A dielectric body must always be **completely covered** by surfaces of type '*Metallic boundary*' in conjunction with the type '*Dielectric boundary*' or of type '*Dielectric boundary*' alone.

Again: Surfaces forming the boundaries of bodies carry a double-layer current distribution, i.e., there are two unknowns per common edge between two patches. Other surfaces inside or outside bodies carry single current layers and are PEC in most cases.

### 3.1.1 Bodies containing high conducting material

The ratio of maximum body size (typical dimension, largest extension) and skin depth should not become too large. A large conductivity might lead to a numerical instability. If a high-loss structure shall be investigated please refer to Section 3.1.5. One reason for a possible failure is that the computation of the mutual interaction between matching paths (test or weighting functions) and sources (basis functions) cannot be computed any more with the necessary precision. CONCEPT-II will stop then with a warning and an error message, respectively. Field-coupling through apertures into metal cavities

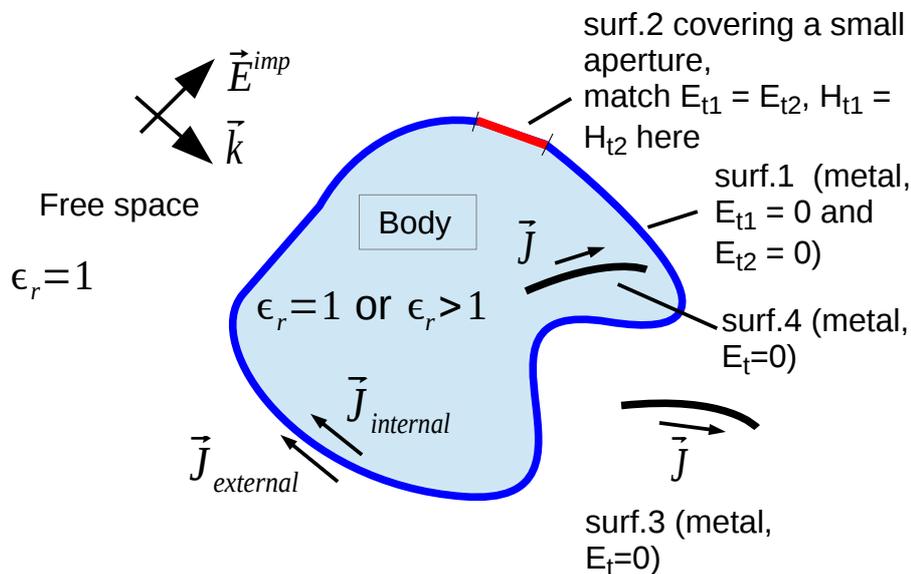


Fig 3.1: Cross section of a cavity in free space illuminated by a plane wave. The red marked line section (surf.2) marks the contour of an aperture where EM fields are able to penetrate into the inside. **A body must be covered by a completely closed grid.**

Fig. 3.1 shows the cross section of a body with an aperture including additional external and internal metal surfaces (surf.3, surf.4). The whole structure is



illuminated by a plane wave field field.

*surf.2* which covers a **small** aperture is of type '*Dielectric boundary*' and *surf.1* is a metallic surface (type '*Metallic boundary*'). The dielectric constant of the material inside the body can be set to one so that air is both inside and outside. For the sake of simplicity only names like *surf.1*, *surf.2* are used, names such as *aperture.surf* or *Wall.surf* would also be accepted.

### Case: simple cavity with air *inside* and *outside* according to Fig 3.1, model with *surf.1* and *surf.2*

In principle an analysis of coupled-in fields based on *surf.1* only with no body formulation, i.e. solely based on a single-layer current distribution would be possible, but may lead to inaccurate internal fields especially for small apertures, assuming line matching which is the default matching technique. For not too small aperture sizes Galerkin matching in conjunction with a single-layer current distribution (pure EFIE solution) may already be sufficient for a good numerical solution.

It turns out that the analysis based on a completely closed grid with *surf.1* (metal walls of the cavity, '*Metallic boundary*') and *surf.2* ('*Dielectric boundary*') covering a small aperture, each one carrying a double-layer current distribution provides more reliable results. This especially refers to the coupled-in or coupled-out fields.

Unfortunately the number of unknowns becomes more than twice as large compared to the case of the standard single-current layer EFIE solution. The body formulation with two different current layers located on *surf.1* and *surf.2* provides a powerful means for the investigation of field coupling through small apertures. Disadvantages of this method are the increased computation time and the increased modeling effort.

Details about the theoretical back ground can be found here:

A. Schröder, G. A. Rasek, H.-D. Brüns, Z. Reznicek, J. Kucera, J. E. Loos, C. Schuster: Analysis of High Intensity Radiated Field Coupling into Aircraft Using the Method of Moments. IB. 2014EEE Trans. on EMC, vol. 56, no. 1, FE

To achieve a good numerical result in the case of aperture coupling it is important that the surface elements along the aperture boundaries are smaller than the radius or width of the aperture, respectively.



### 3.1.2 Multi-dielectric structures with symmetry

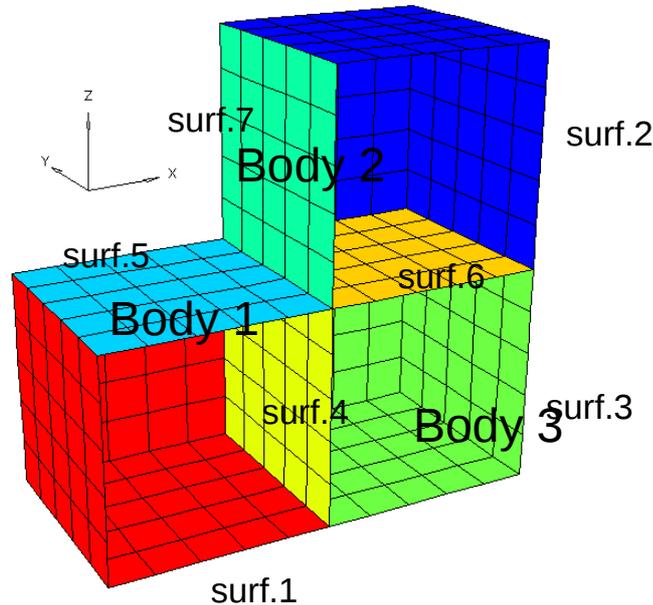


Fig 3.2: Three different dielectric blocks attached to each other

The general proceeding shall be explained by an example. It is a dielectric structure (see `$CONCEPT/examples/Three_cubes`) consisting of three different regions (bodies, here cubical blocks) as shown in Fig. 3.2. Body 1 is attached to body 3 and body 2 is attached to body 3. The whole structure is located in free space and is illuminated by a plane wave with both the E vector and the k vector lying in the x-z plane.

All three cubes enclose different dielectric materials:

$\epsilon_{r, \text{body } 1} = 3$ ;  $\epsilon_{r, \text{body } 2} = 4$ ;  $\epsilon_{r, \text{body } 3} = 5$  and all surfaces are of type '*Dielectric boundary*'. The open front corresponds to the x-z plane which is a magnetic symmetry plane.

It is not allowed (and would make no sense) to place a surface of type '*Dielectric boundary*' into a symmetry plane!

Only the symmetric part of the complete block is depicted and needs to be discretized (half of the structure). In summary we see seven surfaces:

body 1 consists of *surf.1*, *surf.4* and *surf.5*,

body 2 consists of *surf.2*, *surf.6* and *surf.7*,

body 3 consists of *surf.3*, *surf.4* and *surf.6*

*surf.1* and *surf.5* could have been combined to a single surface. The same holds for *surf.2* and *surf.7*, hence the minimum number of surface patch files would be five



here.

*Surf.4* separates the bodies 1 and 3. It is important that **each patch** of *surf.4* has body 1 on one side and body 3 on the other side (a third body number would be wrong).

Important: a body must always be completely bounded by surfaces of type '*Metallic boundary*' and/or by surfaces of type '*Dielectric boundary*', which has been said already. So one or more surfaces of the example under discussion could be of type '*Metallic boundary*'. It should be avoided to cover a region (a body) completely by surfaces of type '*Metallic boundary*': all amplitudes of the internal current basis functions would be zero because of a missing internal excitation.

### 3.1.3 Bodies inside bodies

Fig 3.3 illustrates another possible situation: we see the cross section of a body inside another body (here body 2 is inside body 1). The metal surface *surf.6* (curved magenta line) is also inside body 1. The attached metal plate *surf.5* (light green line) is located in free space. Free space is considered as a "special body" carrying number 0 (body 0). The structure without body 2, and no surfaces *surf.5* and *surf.6* could represent a microstrip line on a finite plate.

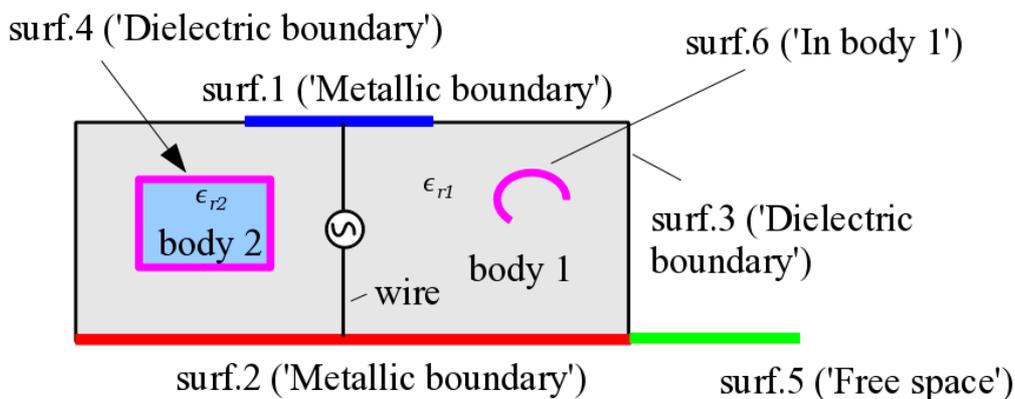


Fig 3.3: A more sophisticated example

Body 1 is bounded by *surf.1* (metal, '*Metallic boundary*', blue), *surf.2* (metal, '*Metallic Boundary*', red), and *surf.3* ('*Dielectric boundary*')

Body 2 is located completely inside body 1 and only bounded by *surf.4* ('*Dielectric boundary*'). The information that body 2 is not in touch with any boundary of body 1 has to be specified under the entry  Body in body, see the simulation project tree.

*Surf.5* is a finite metallic extension of the metallic bottom plate *surf.2* and is of type

'PEC in space'.

*Surf.6* is a metallic sheet in body 1 (surface type: 'PEC in body 1').

It is **not** necessary to specify the direction of normal vector on surfaces or patches, respectively.

A three-body problem is illustrated in Fig 3.4 (see also [\\$CONCEPT/examples/Bodies\\_in\\_bodies](#)). Two bodies, i.e. a cylinder and a sphere are freely floating inside a dielectric cube without any connection. The sphere is characterized by  $\epsilon_r = 8$ . For the cylinder we have  $\epsilon_r = 4$  and for the cube we have  $\epsilon_r = 2$ . Validating the solution we expect the normal components of the electric field strength values to jump at each boundary by a factor of 2 ( $D_{n1} = D_{n2}$ ).

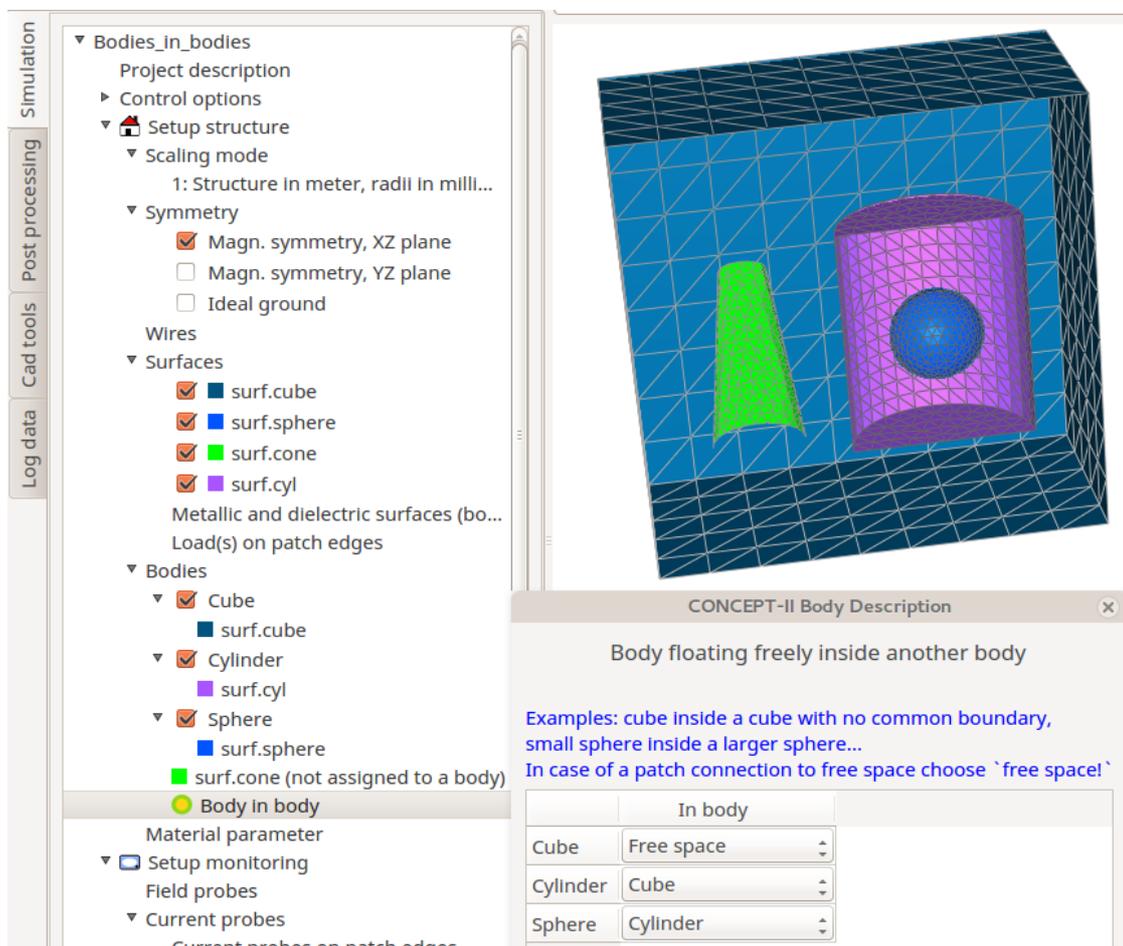


Fig 3.4: Structure according to the example under [\\$CONCEPT/examples/Bodies\\_in\\_bodies](#).

The cone (*surf. cone*) is of type 'PEC in body 1', the other surfaces are forming the boundaries of the bodies "Cube", "Cylinder" and "Sphere".

Notice the entries in the simulation project tree and also the entries in the sub-



window: body 1 “Cube” is in free space (default); body 2 “Cylinder” is located completely inside the cube and body 3 “Sphere” is completely located inside the cylinder. This information has to be entered carefully by hand – it is not automatically detected! **Possible input errors might not be detected.**

### 3.1.4 Wires in dielectric bodies, patch-wire junctions

Wires can be located inside and/or outside a dielectric body. This is automatically detected. When a wire is attached to the node of a surface of type '*Dielectric boundary*', all patches attached to that node are “metalized”. Take care of a fine mesh at such points! Note that wires can only be attached to patch nodes, not to the edges of patches.

Note again: wires can only be attached to a single vertex per triangle.

A wire is not allowed to directly penetrate through surface patches somewhere without fragmentation. If such a wire continues from the outside to the inside of a body, such a wire has to be split into two parts: one wire section must end on a node and the next wire section must start on the same node into the neighboring body.

It is forbidden to place a wire directly, i.e. in parallel along a dielectric boundary with zero height or directly in parallel on a metallic surface without any height!

#### Note

When placing wires over a discretized ground to model a transmission line the edge length of patches perpendicular to the wire axis should at be least 1,...,2 times the height of the wire. If the mesh is too rough the characteristic impedance of the wire cannot be captured numerically. This is an additional requirement to the one already explained in Fig 1.1.

If a wire is attached to a surface node involving quadrangles all these quadrangles are automatically turned into triangles!

It is important that the wire radius remains significantly smaller than the shortest length of the patch edges that are connected to the node. If the ratio edge/radius is too large an error message is given and the program stops. The same statement holds if the wire segments become too short. If for -some reason- the default junction patch-wire checking shall not be carried out, set the following check mark:

#### ▼ Control options

**No check: radius/edge length** (see simulation project tree).

If no wire checking shall be carried out at all (test if the thin-wire assumptions are fulfilled) enable **No check: wires** , also under **Control options** .



It is a good idea visualizing the situation at a patch-wire junction or the mutual wire distances by

**Options** → **Draw wires as...** → **Cylinders** (3D mode)

Then: **Options** → **Setting for wires in cylinder mode ...** (The wire radius can be scaled for visualization)

Further possibilities: **Options** → **Draw wires as...** → **Lines**. Then **Options** → **Setting for wires in line mode...** → Enter the line width in Pixel. Click on  to display the wire segmentation according to the chosen number of basis functions.

### 3.1.5 Layers on bodies

The full-wave computation of the shielding effectiveness (SE) of completely closed cavities may become difficult in general. One reason for this are the geometrically thin conducting walls enclosing a large volume of space. Inside the bounding walls a rapid decay of the local current density happens (keyword: penetration depth). This means for volume-based methods such as the finite element technique, that a very fine discretization of thin sheets has to be introduced which cannot be continued over the whole volume of the shielded cavity. The MoM has severe difficulties as well, referring to the body formulation:

- Surface patches of large extension on opposite sides of wall would be very close to each other (see chapters above).
- Evaluation of Greens function is practically not possible any more, hence the mutual interaction of basis and test functions fails.

To overcome the described difficulties a special hybrid approach combining the MoM with an analytical solution has been included in *CONCEPT-II* which is able to handle the following cases:

- The total volume of the wall itself must be much smaller than the volume enclosed by the wall
- The thickness of the wall is not too large in terms of the penetration depth

Under these circumstances material walls (surfaces) that are covering bodies can be considered as thin layers, the material of which is assumed to be homogeneous and isotropic, characterized by a conductivity, a relative permittivity  $\epsilon_r$ , and a relative permeability  $\mu_r$ . Practical examples are conducting foils, protective metalized clothing, or glass that has a thin metal film, vaporized onto its surface. As has been said already: from numerical reasons (high conductivity, see Sec. 3.1.1, very close vicinity of surface patches) such structures cannot be treated as a multi-body problem with one body being formed by the wall itself and another body by the interior of the cavity.

**Important: Only surfaces of type 'Dielectric boundary' can be layered.**



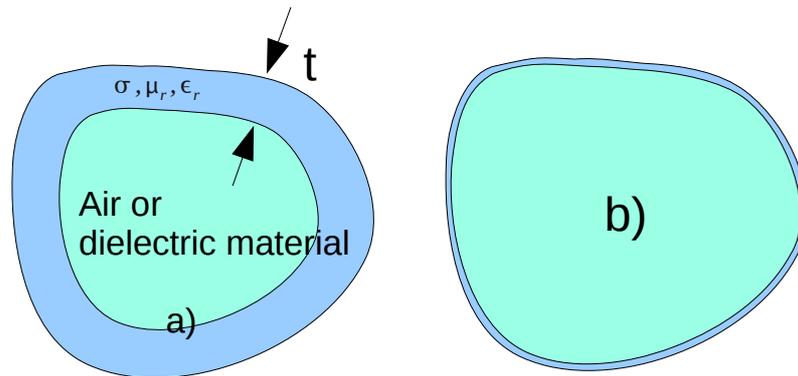


Fig 3.5: Body structures for the explanation of the layer technique. a) Such a problem could be treated as a two-body problem. b) In this case the wall thickness is too small to consider it as a two-body problem. Here the layer technique can be applied.

The layer technique is suitable for the computation of the shielding effectiveness of enclosures. These enclosures are considered as bodies filled with air ( $\epsilon_r=1, \mu_r=1, \sigma=0.0$ ). As has been pointed out already, the approach can be applied under the following conditions:

- **The thickness of the layer has to be geometrically small**, i.e., small compared with the dimension of the body that is enclosed (see Fig. 3.5b). A thickness  $t$  of some micrometers or millimeters on a body of a typical volume of several cubic meters may not be a challenge. A “layer” of 0.5 cm thickness on a sphere with 10 cm radius marks already the limit for the applicability of the hybrid technique (the MoM is combined with an analytical solution concerning the wave propagation inside the layer, see Sec. 9 for details).

A structure as depicted in Fig. 3.5a) can be treated as a multi-body problem provided. Of course, the conductivity of the blue region has to stay small such  $t$  does not get too large compared to the penetration depth.

- **The layer should have a certain conductivity.** The implemented method works best if the wave propagation inside the layer is essentially perpendicular to the layer walls, i.e. a certain conductivity in conjunction with high values for the relative permittivity or permeability are advantageous.
- **The thickness of the layer  $t$  should not exceed 5,...,6 times the penetration depth**  $t < (5, \dots, 6) \cdot \delta$ , with  $\delta = \sqrt{2 / (\omega \mu \sigma)}$ . Violation of this rule may lead to instable current distributions.

EM fields inside the layers cannot be displayed. These fields and especially their tangential components are analytical solutions of the one-dimensional Helmholtz equations for  $E$  and  $H$ , respectively, see also Sec. 9. They are not taken explicitly taken into account.



### 3.1.5.1 Example of a cylinder with a lossy wall of high relative permeability

A simple numerical example can be found in `$CONCEPT/examples/Cylinder_with_layer` and is illustrated in Fig. 3.6 . Only a quarter of the overall cylinder which is illuminated by a plane (1V/m), is shown. The height of the depicted cylinder is 1.8 m and the radius is 0.3 m.

The wall of the cylinder including the top and bottom plate is characterized by  $\sigma=1824000 \text{ S/m}$  ,  $\mu_r=11000$  and the wall thickness is 0.8 mm. The frequency is 1 kHz. Note that very high values for the relative permeability cause no difficulties.

The advantage of this example is that reference results can be computed by an analytical expression, to be found in the book of Heinrich Kaden [Ka-59]. Kadens formula is valid for infinitely long cylinders. Of course, such a structure cannot be modeled in *CONCEPT-II*. In order to get comparable results the length of the cylinder has been chosen 6 times the diameter including its image below the infinite ground plane at  $z=0$ . . For further details see the *README* file in the mentioned directory.

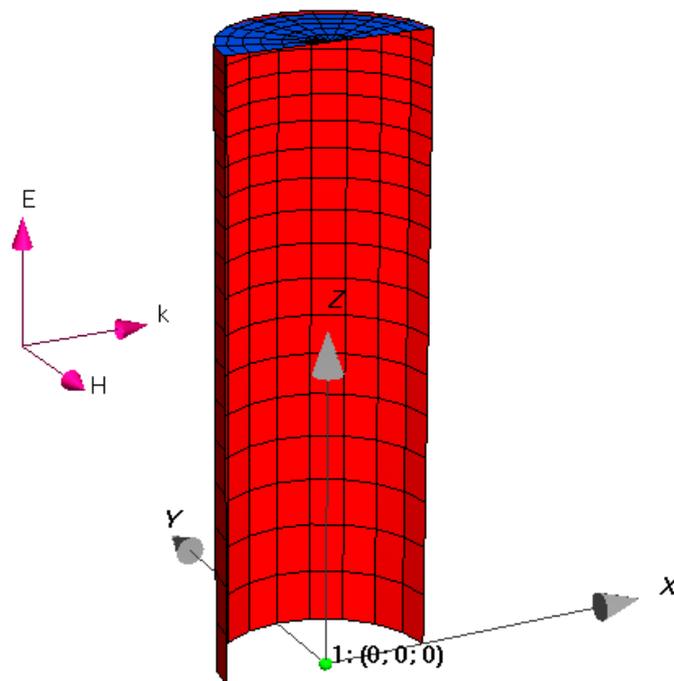


Fig 3.6: A cylinder with a lossy magnetic wall, illuminated by a plane wave field.



Symmetry has been used with respect to the x-z plane. A field probe has been placed at (0., 0., 0.) m. Note that the probe location is marked by a green dot right in the center of the coordinate system as illustrated in Fig. 3.6. The cylinder is modeled as an air-filled body by *tube.surf* (wall of cylinder, color red) and *top.surf* (top plate, blue blue).

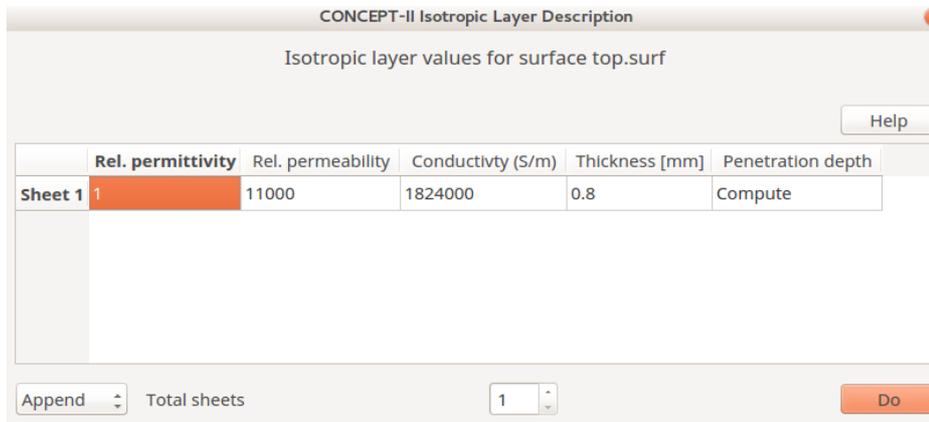


Fig 3.7: Specifying a layer on a surface of typ 'Dielectric boundary'. Right click on the surface as part of a body boundary (simulation project tree), then **Add isotropic layer**. By clicking on **Compute** ('Penetration depth') the user can determine the ratio of wall thickness to penetration depth with should not exceed a value of approx. 5...6

It is possible to assume more than one layer on a surface: Increase the number of 'Total sheets', see Fig 3.7. Note: In the case of a multi-layered sheet the material parameters of the layer which is on the side of the body with the lower number, has to be specified first.

Another area of application is the inclusion of losses in metallic blocks. Imagine for example a monopole antenna mounted on a lossy metallic block. Although the block is solid and cannot be treated by the body approach (see Sec. 3.1.1), a finite wall thickness of 5...6 times the penetration depth can be assumed. So the effect of the losses on the antenna input impedance can be computed quite precisely.

[Ka-59]: H. Kaden. Wirbelstroeme und Schirmung in der Nachrichtentechnik.

Springer Verlag 1959

## 3.2 N-ports, S-parameters

An introduction to the topic has already been given in In Sec. 1.11.2.6.2 .

The mutual interaction between ports can be described by a reduced admittance matrix  $[Y]_{KK}$  , where  $K$  is the number of ports. In general  $K$  is much smaller than the order  $N$  of the system matrix.

With the *CONCEPT-II* network tool *CONCIRC*

(download from here: → <http://www.tet.tuhh.de/concept/network-solver/>)



the impact of loads on the mutual coupling of ports can easily be investigated after the back-end run. This can be accomplished fast as there is no need for solving the large system of equations of the MoM part again and as only the relatively small admittance matrix is involved now (one Y matrix per frequency step!).

In the method of moments the system of linear equations can be written as follows:

$$[U]_N = ([Z]_{NN} + [Z_L]_{NN}) \cdot [I]_N, \text{ where}$$

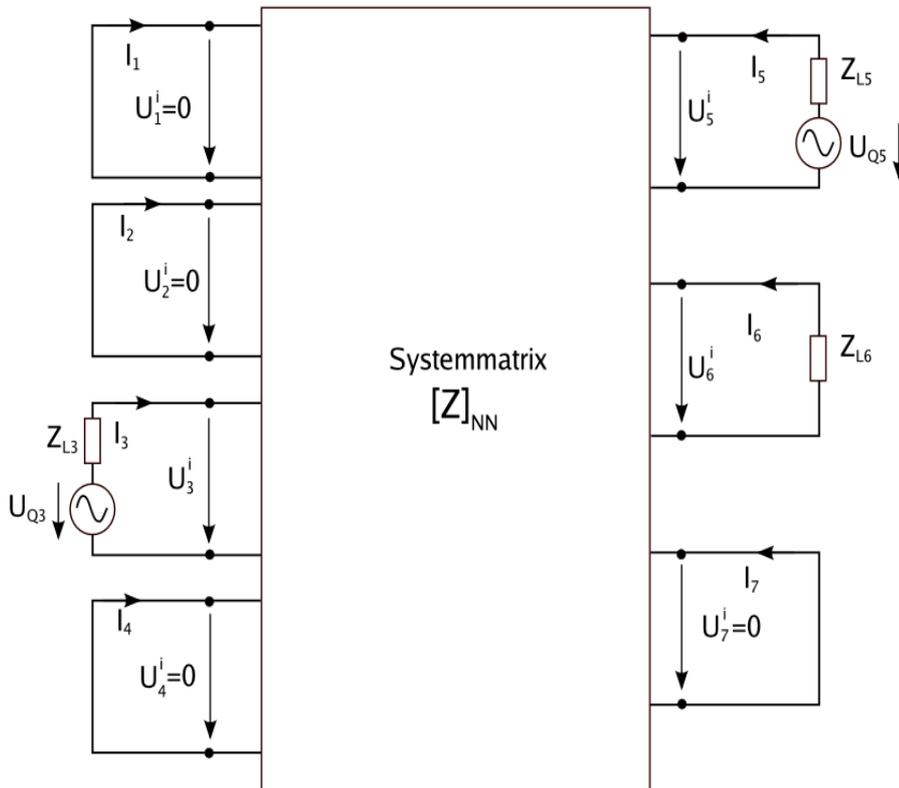


Fig 3.8: A small small example structure with N=7. Ports are defined at positions 3 and 5, hence K=2

$[Z]_{NN}$  is the system matrix,  $[U]_N$  is the voltage vector (excitation),  $[Z_L]_{NN}$  is a diagonal matrix containing lumped loads with all off-diagonal elements being zero,  $[I]_N$  is the current vector (complex amplitudes of the basis functions).

Ports are defined by setting K voltage or power generators at K different positions at wires or on patches. As a simple example demonstrating the principle consider Fig. 3.8 where a structure with N = 7 is depicted; we see generators at segments 3 and 5.

N=7 shall be reduced to K=2. For this purpose the equation system of order N has to be solved K times to compute K corresponding global current distributions, which can easily be carried out by means of the LU decomposition type of solver. From this data it is possible to extract the following reduced equation system:

$$[I]_K = [Y]_{KK} \cdot [U^i]_K .$$

From the viewpoint of the ports the impact of the whole structure, one can think about an electrically large ship or aircraft is contained in  $[Y]_{KK}$ . The reduced admittance matrix is stored in the file *red-y-mat.h5*.

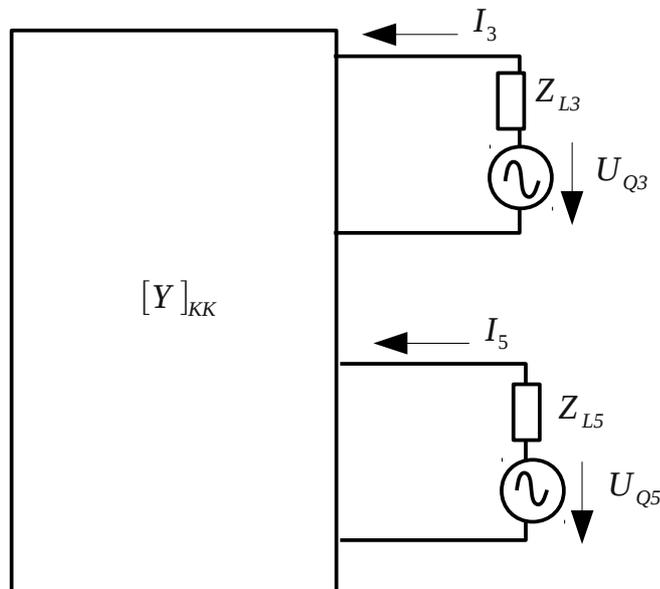


Fig 3.9: The system according to Fig. 3.8 has been reduced to a two-port system. These ports can be loaded in an arbitrary manner after the back end has finished. Code *CONCIRC*

S-parameters can be computed from the reduced admittance matrix by means of

$$[S] = (Y_0 \cdot [I] + [Y])^{-1} \cdot (Y_0 \cdot [I] - [Y])$$

with

$Y_0 = 1/Z_0$  and  $Z_0 = 50$  Ohms in many cases.  $[I]$  is the identity matrix and  $[Y]$  is the reduced admittance matrix as described above.

### 3.2.1 Files

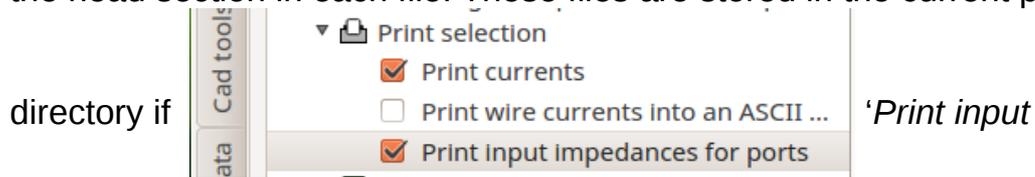
- *red-y-matrix.h5*

The Hierarchical Data Format (HDF) is a set of file formats designed to store and organize large amounts of data, see [https://en.wikipedia.org/wiki/Hierarchical\\_Data\\_Format](https://en.wikipedia.org/wiki/Hierarchical_Data_Format) .

- *port-zin(n).asc*



$n$  is the port number. Input impedance of the  $n$ th port. For the file format see the head section in each file. These files are stored in the current project



'impedances for ports' had been activated (**Simulation**, simulation project tree). Generator loads have been subtracted, we have the pure antenna input impedance(s) under the condition that all other ports are short circuited. For the format see header of files.

- *freqzin.asc*

Post processing →  → window opens → **Input impedance** → select

number of curves and the corresponding port number(s). We get the input impedance(s) under the condition that the loads at all other (non active) ports are still present but the generators at these ports are non-active (short-circuited). File format: see header of file.

## 3.2.2 Computation of S,Y,Z-parameters

Click on the button [\[N\]](#), Tab **Post processing**. Data is only available if ports were defined in the front-end run. Desired network parameters (S,Z, Y) are evaluated from the reduced Y matrix (see last chapter). Lumped loads directly at port positions which might have been set in the original data according to the simulation project tree were removed (example: lumped load at the center of wire  $m$ , port at the center of wire  $m$  → load is not contained in network parameters. Lumped load at the beginning of wire  $m$ , port at the center of wire  $m$  → impact of this load is contained in the network parameters).

According to the example depicted in Fig 3.10 various formats are can be chosen for each type of network parameter.



**CONCEPT-II Post Processing**  
Create Network Parameters

S-parameters Help

Z (Ohm)

Write Touchstone file

Write ASCII file

Write HDF5 file

Number of S-parameter curves to be displayed:

Output file:

Smith diagram:

|   | Output port | Input port | Smith diagram            |
|---|-------------|------------|--------------------------|
| 1 | 1           | 1          | <input type="checkbox"/> |
| 2 | 1           | 2          | <input type="checkbox"/> |
| 3 | 1           | 3          | <input type="checkbox"/> |

Fig 3.10: Representation of S-parameters has been selected

Select the '*Number of S-parameter curves to be displayed:*' and choose '*Output port*' and '*Input port*' for each curve as a function of frequency.

For displaying curves again use :  Network parameters → **Show results** (View

results section). The *CONCEPT-II* Gnuplot front end dialogue window opens. Leave the mouse pointer on a '*Using*' input field for opening a tool tip → information about the various columns.

Data is available in the working directory depending on the chosen type of format. Each output file contains the parameter matrices for all frequency steps.

*s-matrix.s(number of ports)p*, *y-matrix.s(number of ports)p*, *z-matrix.s(number of ports)p* → Touchstone format

*s-matrix.asc*, *y-matrix.asc*, *z-matrix.asc* → see head section for format

*s-matrix.h5*, *y-matrix.h5*, *z-matrix.h5* → HDF5 format

Postscript files are generated in case of '*Smith diagram*' being selected.

### 3.2.3 Using *red-y-matrix.h5* with the network tool *CONCIRC*

Using the data file *red-y-matrix.h5* (see previous sections) shall be demonstrated by a simple example, see *\$CONCEPT/examples/Couples\_dipoles*. Three dipole antennas are considered and



port excitation has been chosen, see Fig. 3.11.

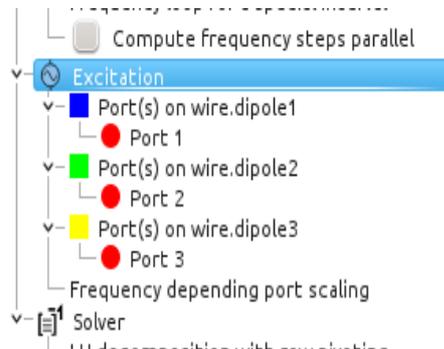


Fig 3.11: Part of the simulation project tree (**Simulation** tab active)

The file *red-y-matrix.h5* is stored automatically once the back end run has terminated. *Red-y-matrix.h5* contains a 3x3 matrix for each frequency in this example.

Now start *CONCIRC* and draw a network as depicted in Fig 3.12.

Windows: click on icon; at the command line enter *concirc1.exe* <CR>

Linux: at the command line: *concirc1* <CR>

As shown in the figure all ports (circuit points at the large rectangular center block) have been connected to transmission lines, each line is loaded by a 50-Ohm resistor on the opposite end. Such a load situation can be introduced quickly and be changed with minimum effort, if desired.

Download *CONCIRC* and its manual *from here*:

<http://www.tet.tuhh.de/concept/network-solver/> .

For creating the large center block, click on the 'Network' symbol in the left section of the *CONCIRC* GUI and draw it into the work area (display area). Then right click onto the block → **Edit properties** → **MTL: \*.h5-file, CONCEPT-II: red-y-mat.h5** → **Get port/MTL file ...** select *red-y-mat.h5* .

It is important to enter the correct '*number of ports*' now (for the number see simulation project tree, tab **Simulation** in the *CONCEPT-II* GUI).

Now each port can be connected to a transmission line or other network elements as shown in the example. In the example we have:

Port 1 → Active generator, 1V, transmission line of length 1m; Port 2 → transmission line of length 1 m; Port 3 → transmission line of length 0.5m.

All transmission lines have a wave impedance of 275,09 Ohms and are terminated by 50 Ohms. Of interest are the port currents as a function of frequency.

The results of the computation can be seen in *current.png*, This file is provided in the mentioned example folder. In addition *Current.png* contains curves resulting from a full wave computation, which can be carried out in the validation



subdirectory `$CONCEPT/examples/Coupled_dipoles/validation`. Here the transmission lines were modeled as real two-wire lines with the same wave impedance as has been assumed in the network computation, see Fig 3.12 .

To get the currents in `$CONCEPT/examples/Coupled_dipoles/validation` proceed as follows:

Tab **Post processing** → click  → **I(f)** set probes 1,2 3 → file `freq1.asc` is generated.

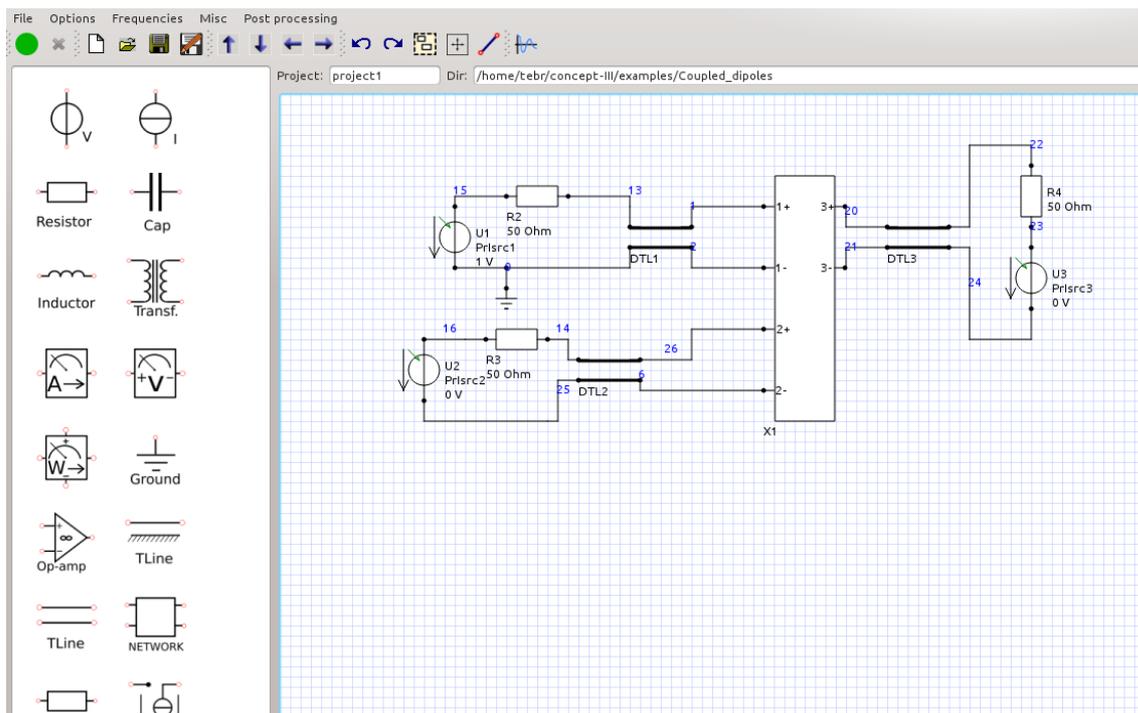


Fig 3.12: Graphical user interface of the network tool *CONCIRC* . Please study the *CONCIRC* users manual. For the right frequency sampling: **Frequencies** → **Frequencies from file** → `confreq.in`

The results of the full-wave computation and the two-step approach are in a good agreement. In another case let us assume that we have a simple monopole antenna. The input impedance as a function of frequency is provided by *CONCEPT-II*. By using *CONCIRC* it is easy to compute the change in input impedance caused by a feed line (transmission line) for example. Use the element 'impedance' and load the input impedance from file (`freqzin.asc`, `port.zin1` in case of a port excitation) and compute the modified input impedance by *CONCIRC*.

### 3.2.4 EM fields due to a modified port excitation

Precondition for the following steps: port excitation has been chosen.



In practical cases it can be of interest computing electromagnetic fields in observation points due to changes of port voltages. Enable tab **Post processing** >

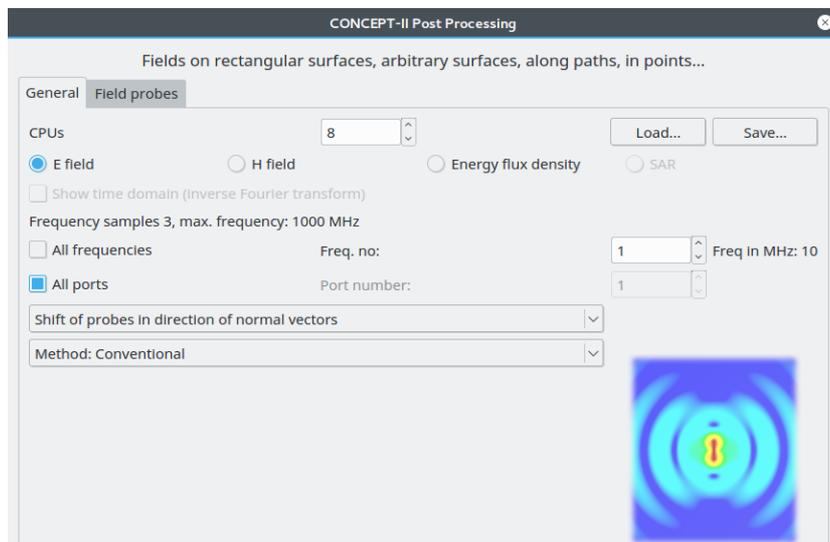


Fig 3.13: Enabling ‘All ports’ provides the possibility to compute EM fields at arbitrary positions due to modified port voltages (application of the superposition principle).

Click  → enable ‘All ports’ → Continue with the selection of the **Field probes** .

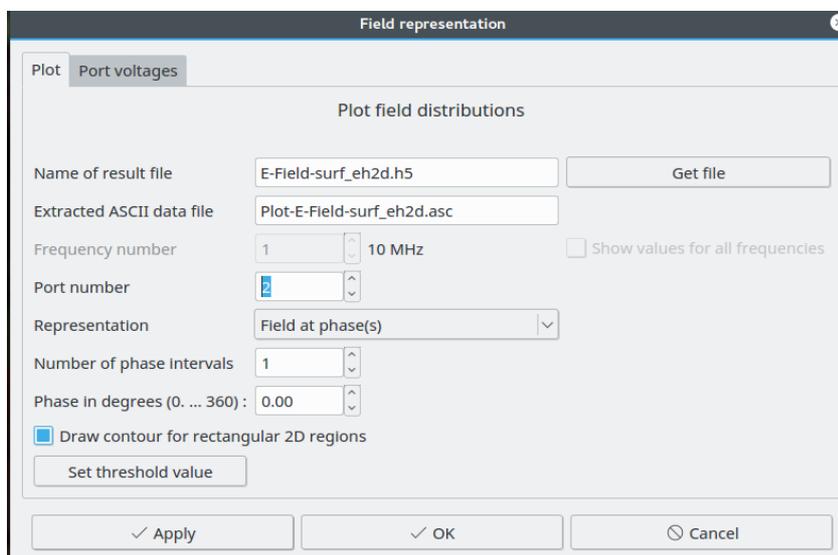


Fig 3.14: Here a specific port can be specified as excitation (port 2 has been chosen) with an amplitude as entered under **Simulation**



Once the field computation has finished a window according to Fig 3.14 opens. Note the tab **Port voltages**. Fig 3.15 demonstrates how to proceed: enable 'All ports are fed simultaneously'. Then the table of available ports will be loaded and enabled. Enter the port voltages (note that input from ASCII files is possible).

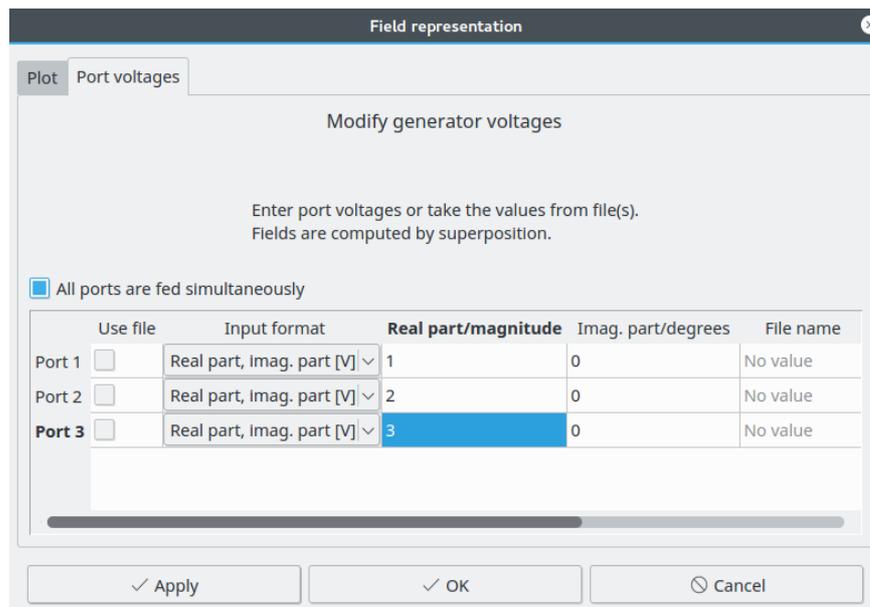


Fig 3.15: In the example three ports are available which are fed by voltage generators with 1 V, 2 V, and 3 V, respectively, at the same time

### 3.3 RCS

The RCS (radar cross-section) is a measure of how detectable an object is when illuminated with a radar. “The RCS of a radar target is the hypothetical area required to intercept the transmitted power density at the target such that if the total intercepted power were re-radiated isotropically, the power density actually observed at the receiver is produced” (<http://en.wikipedia.org> -> radar cross-section).

Generally one has to distinguish between the so-called **bistatic** and the **monostatic** radar cross section.

In the case of the monostatic RCS receiver and transmitter are always located at the same position in space while moving around the radar target (orbit plane). In the bistatic case the source of excitation is fixed in space whereas the receiver rotates around the target on a circular path.



Any RCS calculation is restricted to an excitation of the radar target by a plane wave of linear polarization  $\vec{E}_{inc}, \vec{H}_{inc}, \vec{k}_{inc}$ . These vectors are moved with fixed mutual position (all angle 90 degrees) on the orbit plane (index *inc* for “incoming”)

The calculated RCS values  $\sigma$  are given in  $dB\lambda^2$  or  $dBm^2$ . The definition is as follows (index *scat* for “scattered”):

$$\sigma_{rCS} = \lim_{R \rightarrow \infty} 4\pi R^2 \frac{|E_{scat}|^2}{|E_{inc}|^2} = \sigma_{rCS} = \lim_{R \rightarrow \infty} 4\pi \frac{|RE_{scat}|^2}{|E_{inc}|^2}$$

$$\sigma_{dB} \lambda^2 = 10 \log_{10} \frac{\sigma_{rCS}}{\lambda^2}, \quad \sigma_{dBm}^2 = 10 \log_{10} \frac{\sigma_{rCS}}{m^2}$$

For electrically large structures application of the H-matrix solver is strongly recommended and should be used in conjunction with the 'Conventional' option !

RCS → **Simulation** → **Excitation** → RCS → window according to Fig 3.16 .

'Under treatment of RCS excitation' the following items are available:

- *Conventional* (use with H matrix solver or with LU decomposition, multiple processor application possible)
- *SVD* or *ACA* (only for the monostatic case)

Speed-up of the solution by compressing the excitation matrix formed by multiple right-hand sides. Applicable only if the number of unknowns is larger than the number of right-hand sides, only a fraction of the coefficients forming the matrix of the right-hand side has to be computed explicitly. Only for single-processor usage! All solvers.

*Accuracy of compression*: Default is 1.e-5. Is a measure for the matrix approximation (all right-hand sides are forming a matrix).

Details concerning the SVD (Singular Value Decomposition) or ACA (Adaptive Cross Approximation) usage in the context of RCS computations can be found in the following paper:

*A. Schröder, Heinz-D. Brüns, C. Schuster: "A Hybrid Approach for Rapid Computation of Two-dimensional Monostatic Radar Cross Section Problems With the Multilevel Fast Multipole Algorithm", IEEE AP, vol. 60, no.12, December 2012*

*Polarization*: The vector of the electrical field strength is rotated around the k vector by the angle to be entered (right-hand rule applies).



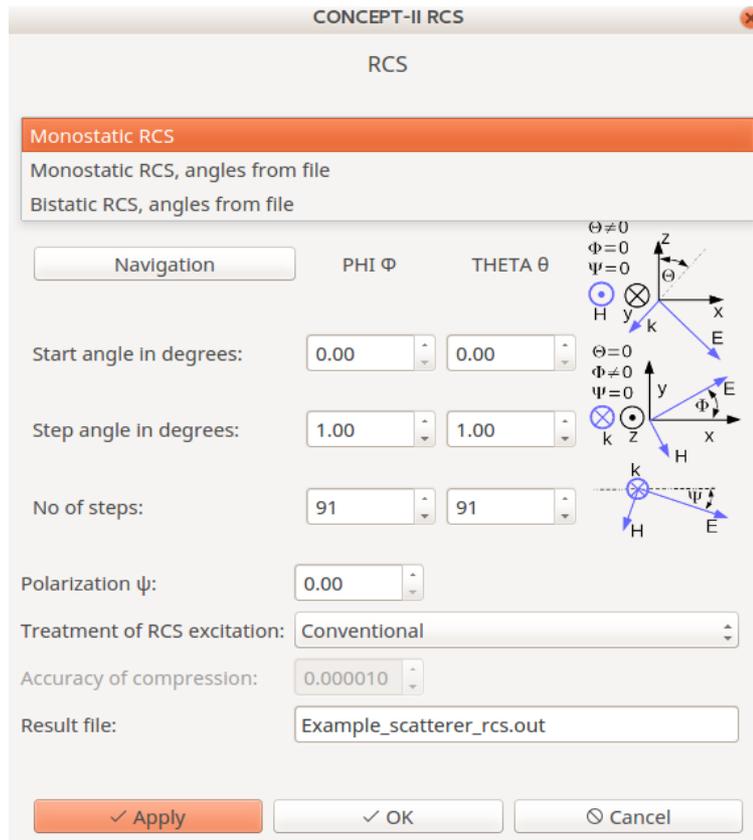


Fig 3.16: Input fields for an RCS computation (**Simulation** → **Excitation** → **RCS**)

Note: In case of 'Conventional' a very long solution time has to be expected in conjunction with MLFMA. Hence use MLFMA (single CPU!) only with SVD or ACA! For very large problems ACA is faster than SVD and should be preferred.

The RCS can be computed in spherical sectors defined by the intervals  $\Delta\Phi$  and  $\Delta\Theta$  and along orbits around the object. Information on how to set the angles: click on **Navigation** in the RCS dialogue window.

As can be seen in Fig 3.16 (arbitrary) angles can be read from ASCII files for both bistatic and monostatic RCS computations – for the required format click on **Show file format** in the corresponding window that opens. The default name of the 'Result file' is *concept\_rcs.out*, the contents of which may be displayed by external software for example or by  (tool bar) in case of a suitable sequence of data. In

the monostatic case the head section of *concept\_rcs.out* looks as follows:

```
#####
# RCS Results
#
# Column 1: Frequency [MHz]
# Column 2: Phi_inc [degrees]
```

```
# Column 3: Theta_inc [degrees]
# Column 4: Sigma [dB lambda**2]
# Column 5: Sigma [dB m**2]
# Column 6: E [V]
# Column 7: Real{E_Phi} [V]
# Column 8: Imag{E_Phi} [V]
# Column 9: Real{E_Theta} [V]
# Column 10: Imag{E_Theta} [V]
...
```

Notice that various E field components are available in addition to the RCS data.

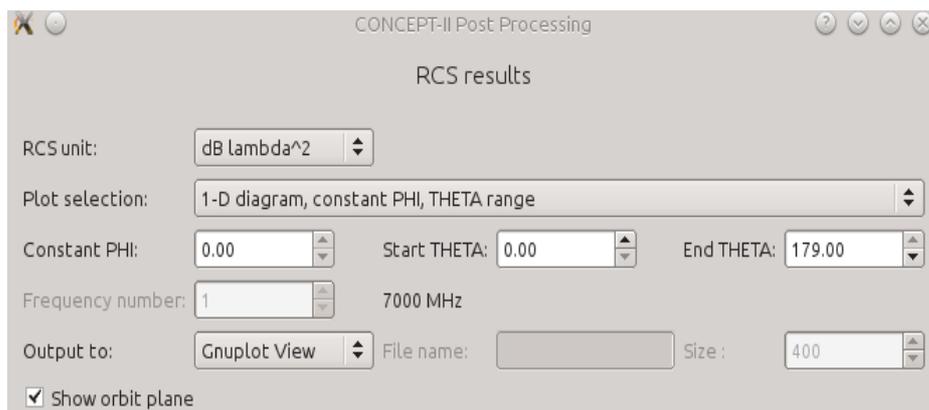


Fig 3.17: Window for plotting RCS curves (**Post processing**)

Click on  (**Post-processing**) → the *CONCEPT-II* window for plotting RCS

results opens ( Fig 3.17); both curves and 2D color maps are possible. The RCS data can be represented in  $dB\lambda^2$  or in  $dBm^2$  (see definitions above). In case of “RCS angles from file” the contents of the result file is shown. RCS examples can be found under  $\$CONCEPT/examples/RCS\_corner\_reflector$  and  $\$CONCEPT/examples/NASA\_almond$ .

Output file in case of a curve: *1Dracs.asc*.

Curve is shown in the display area → right click into display area → file can easily be saved to a file with different name.

In case of displaying currents the excitation according to the last right-hand side, is applied corresponding to the end position of the field excitation vectors  $\vec{E}, \vec{H}, \vec{k}$ , (last pair of angles). Note that any near field can currently not be computed! (Nov. 15.)

### 3.4 Computation of line integrals of E and H

The following line integrals of E or H can be numerically computed:



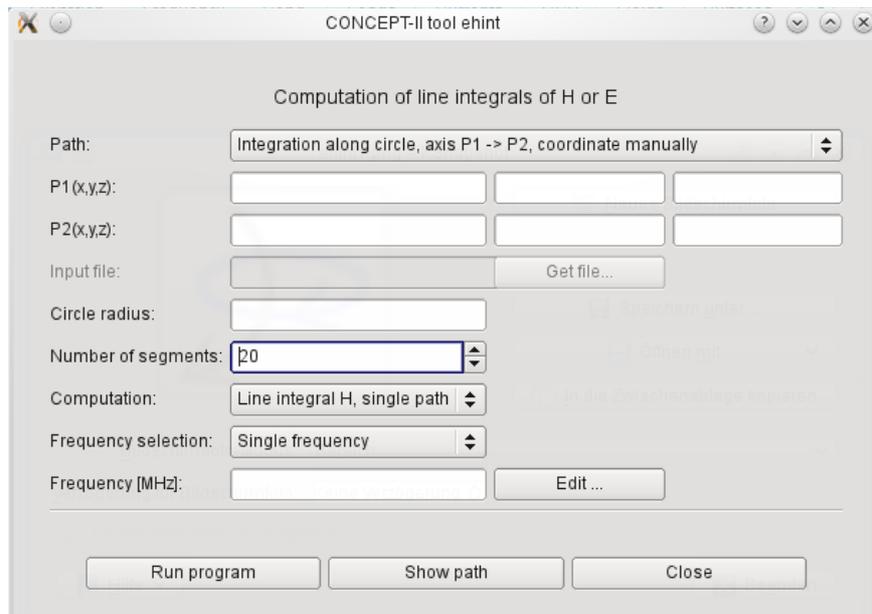


Fig 3.18: Window for the specification of paths for line integrals of E and H

$$\oint_{\partial A} \vec{E} \cdot d\vec{s} = -j\omega \iint_A \vec{B} \cdot d\vec{A} \quad , \quad \oint_{\partial A} \vec{H} \cdot d\vec{s} = I_{\text{wire}} + \iint_A (j\omega \epsilon \vec{E} + \vec{J}_{\text{Surface}}) \cdot d\vec{A}$$

Clicking on  opens a window according to Fig.3.18.

### Path

#### 1. Integration on a circle, axis P1 -> P2, coordinates manually

For the axis definition, see Fig. 3.19.

#### 2. Integration on a circle, axis P1 -> P2, coordinates from file

P1 and P2 (see 3.19.) of the axis are taken from a file:

```
Line 1  Xstart  Ystart  Zstart
Line 2  Xend   Ystart  Zstart
```

A convenient way to define the points P1 and P2 Fig 3.19 is to open the structure under **Cad tools** → **Cad tools 1** → activate *points into file* → the “Create Points” window opens. Right click on two nodes at the center distance of which the line integral shall be computed. The chosen nodes mark the ends of the axis according to Fig 3.19. Save the node coordinates into a file.

*Circle radius* and the *Number of segments* (integration intervals) have to be specified

#### 3. Integration path from file (wire format)

The file must have the *CONCEPT-II* wire format: the first line contains the



number of straight sections. Each following line has the following entries.

Xstart Ystart Zstart Xend Yend Zend *Dummy* *Number of integration points*

*Dummy* has no meaning here, 0 is suggested.

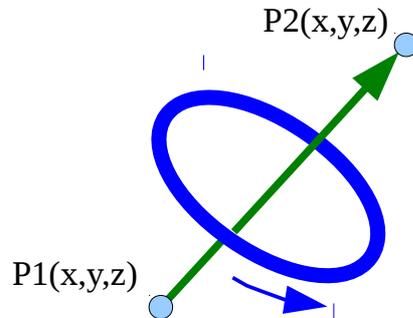


Fig 3.19: Direction of the line integration and axis of the circle (the right-hand rule applies)

The following options are available under “Computation”:

- Line integral of the E field
- Line integral of the H field-coupling
- Under the condition that **Integration path from wire file (wire format)** has been selected, voltages as a result of integrating the E field along straight paths can be computed.

Notice that we have a behavior different from static fields. In the high frequency case the integration result depends on the shape of the path. The lowest frequency which can be computed depends on the type and geometrical size of the structure and the discretization and may vary between 100 Hz (large structure) and 10 kHz (dielectric structure).

*Frequency selection: selecting All frequencies*

The results are written into the ASCII file *EH\_int\_result.asc*. In case of a frequency loop the *CONCEPT-II* Gnuplot front end  can be used to display curves as a function

of frequency. The result of the first 4 frequency steps are shown on the screen in a special window. The corresponding file is: *EH\_int\_screen.asc*.



### 3.4.1 Example: Line integral to measure the total current

Fig. 3.20 shows a torus, illuminated by a plane wave field ( $E = 1V/m$ ) .

Torus:            Radius of center line: 0.3 m  
                      Radius of conductor (tube) cross section: 1 cm  
                      Frequency 100 MHz

The axis of the circular integration path can be determined as described in the previous chapter, notice the blue dots. It is interesting that the surface current distribution looks rather complicated, see the detailed view in Fig. 3.21. Here we can see that the center of the integration circle does not correspond to the axis of the torus. The thin-wire assumptions are valid for the structure and therefore the torus can be approximated by a sequence of straight wire sections. One finds that the value of the “measured “ current and the line current of the simple wire model are in a very good agreement. A wire circle can easily be created by means of the helix tool  (**Cad tools** → **Cad tools 2**, set “Pitch angle of helix in degrees” to 0

Radius ( $\phi_{min}$ )=0.3, Radius ( $\phi_{max}$ )=0.3,  $0^\circ, 360^\circ$  ).

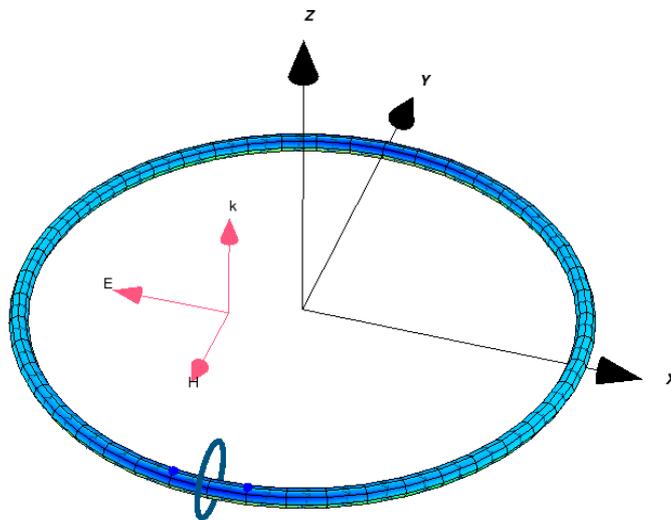


Fig 3.20: Metallic torus in a plane wave field.



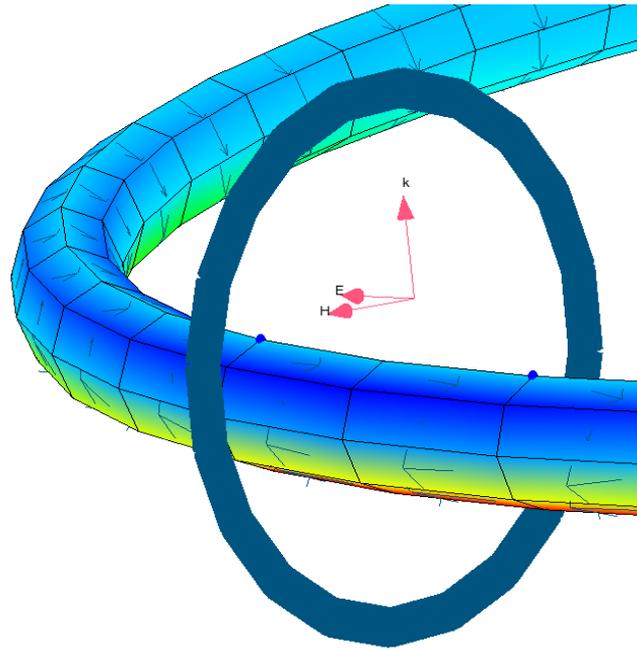


Fig 3.21: The induced surface current distribution in the region of the torus according to Fig. 3.20 where the line integral is performed. The vertical ring indicates the path of integration.

See `$CONCEPT/examples/Torus_plane_wave`

+

### 3.5 Integrating the Poynting vector on a triangulated surface

Based on the following expression the power flow through arbitrary surfaces can be computed:

$$P = \frac{1}{2} \operatorname{Re} \iint_A \vec{E} \times \vec{H}^* \cdot d\vec{A}' \quad (1)$$

Clicking on  the window according to Fig. 3.22 opens.

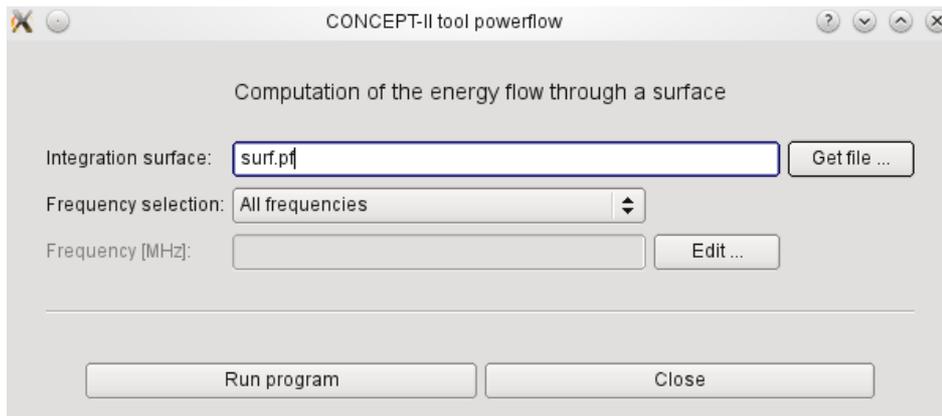


Fig 3.22: The input window for the computation of the power flow

### Integration surface

Here the integration surface (Eq. (1)) has to be specified. Required is the *CONCEPT-II* patch format and only triangles are allowed. Creating such an integration surface is possible by means of tools, which available under **Cad tools**, for example . The orientation of the normal vectors can easily be checked by clicking on .

Example: completely closed surface, all normals are pointing to the outside, an antenna is inside the integration surface → the resulting power value will be positive. If power is flowing from the exterior to the exterior, the power result will be negative (scattering case). The power flow density distribution is displayed on the integration surface.

### Frequency selection

If **All frequency steps** has been selected the result of the first 4 steps is displayed on the screen (special result file: *PowerScreen.asc*)

Result file: *PowerResult.asc* → ready to be displayed using .

## 3.5.1 Example: Loaded circular wire loop

Let us consider the same loop as shown in Fig. 3.20 as a pure wire loop. Here the loop is turned around the x axis by 90 degrees, the loop area lies in the x-z plane now. The loop is loaded by a resistor of  $50 \Omega$  at the center of the first wire which is beginning at  $z=0$  m. Thirty segments are forming the circumference ( $R_{\text{loop}} = 0.3$  m,  $R_{\text{wires}} = 5$  mm,  $f = 100$  MHz). The loop is illuminated by a plane wave field (1V/m) such that the wave is traveling in positive z direction and the H vector is perpendicularly oriented to the loop area as illustrated in Fig. 3.16.



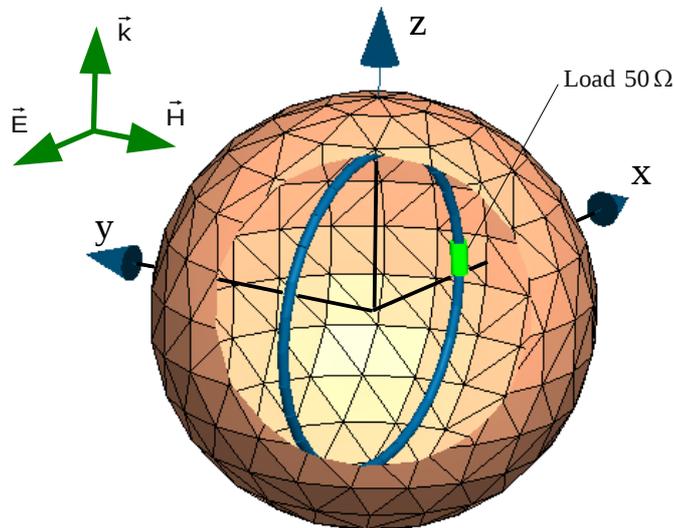


Fig 3.23: Circular loop loaded by a resistor in a plane wave field. The wire loop is inside a spherical integration surface

The patches of the integration surface should not get too close to the structure. At least 8 triangular patches per wavelength are necessary in case of an electrically large circumference (default surface name: *surf.pf*). In case of a short wavelength enough patches have to be set to approximate the surface of the sphere properly. There is no need to locate the integration surface in the far field.

This example is under `$CONCEPT/examples/Torus_plane_wave`

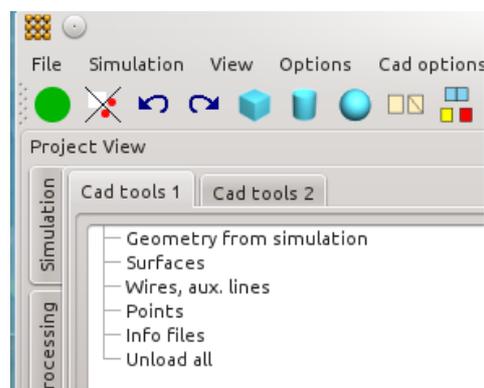


Fig 3.24: The items of the **CAD tools** tree

The power which is dissipated in the load can be taken from *concept.out*, provided

that the print option has been switched on and a voltage probe has been specified. This value has to be compared with the result of the power flow tool. Discretizing structures

## 4 Discretization of structures

Wires and surfaces can be meshed using the tools available after clicking the **CAD tools** tab → the corresponding tree appears in the 'Project view' section as depicted in Fig 3.24)

**Geometry from simulation** → **Load simulation files:** Load all wire files and surfaces files as specified under **Simulation**.

**Surfaces, Wires, aux. lines, Points** → load or unload files with corresponding data



: Deselect a selection of patches or nodes



: Turn a quadrangular mesh into a pure triangular mesh



: The “explode” tool. It decomposes a global surface into sub-surfaces. Details

are described in the head section of the window.



: Running the explode tool on large structures that have been converted from

external tools into the *CONCEPT-II* surface patch data format, a lot of files might appear. By this tool sub-surfaces can be quickly unloaded by mouse selection. The corresponding file names can be written into a list. This list can be used to create a single surface with a unique patch and node numbering by means of the combine tool (**Cad tools 2**, )

### 4.1 Spheres, plates, cones, cylinders etc.

For a quick generation of simple surfaces three basic tools are available, see tool bar. Each of these tools allows a frequency dependent discretization. Be careful with electrically small structures – in such cases the provided grids might be too rough. Simply enhance the frequency or the number of patches per wavelength to a fictitious values for a proper grid in the case of electrically small structures (refer to Sec. 1.7).



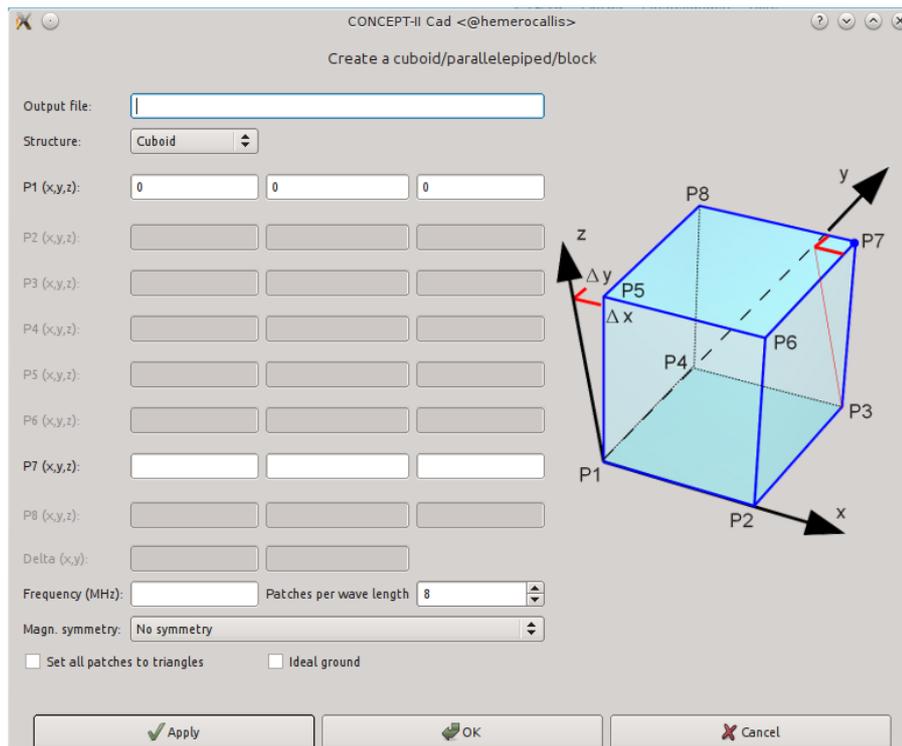


Fig 4.1: Tool for discretizing cuboids, blocks, pyramids etc.. Also plates are possible. In the most general case all eight points P1, ..., P8 have to be specified.

For a rectangular plate set point 7 ('P7') to x, y, 0, with x and y as the edge lengths of the plate.



→ Meshing of arbitrary circular cylinders., cones, pyramids



→ Discretization of spheres or ellipsoids

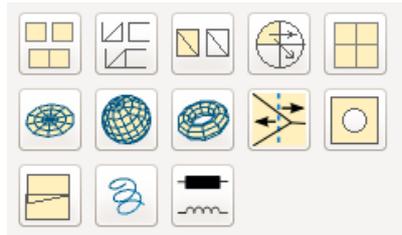


→ Cubes, rectangular, boxes, pyramids..., see Fig. 4.1.

In case of a parallelepiped the two points P1 and P7 have to be specified and in addition two shift values "Delta (x,y)" ( $\Delta x, \Delta y$ ). Setting for example P5=P6=P7=P8 gives a pyramidal structure, an example is shown in Fig 4.2. Setting the z coordinate of P7 to zero in case of a cuboid provides a plate in the y-z plane. Further important tools can be found in the bottom section of the **Cad**



tools2 window:



#### Plate tool

Discretization of arbitrary plates with a uniform or a non-uniform mesh. Quadrangular plates can be twisted but straight lines are between the corners. Triangular plates are possible. Mixed quadrangular – triangular patterns of patches are created. To convert to a mesh consisting only of triangular patches use  see tool bar.



Sphere tool with enhanced capabilities.  
Discretization of spheres, ellipses, etc..



#### Combine tool

Converts a number of individual surfaces into a single surface with a unique node and patch numbering.

Surface patch files contain the mesh data, each partial surface with a unique node and patch numbering (see Section 8 for the file format). The goal is to combine different surfaces to a single surface. At all common interfaces of the original files double nodes are sorted out automatically.



#### Combine wires tool

Combine different wire structures to a single wire structure



#### Disk tool

Discretization of

- circular and elliptical disks
- disk with a hole in the center
- cylinders (with different radii at bottom and top, specify '*Height*' and '*Size of hole*')
- cones (also tilted cones, specify '*Height*' and '*Projection length*')



Tool for making complete structures from symmetrical parts.

In case of electric and magnetic symmetry only the symmetric part of the overall structure has to be discretized. Applying this tool the complete structure can easily be generated.



Tool for detecting grid errors. Bad or tiny patches or very small patches or



unwanted caps (grid not closed) can be detected.



A surface patch file is converted into a wire grid. All patch edges are turned into connected wires.



Tool for creating rings, helices, spirals of wires.



Tool for reducing surface patch structures to the part which lies in the symmetry plane only.



Tool for quickly discretizing various types of basic coils.

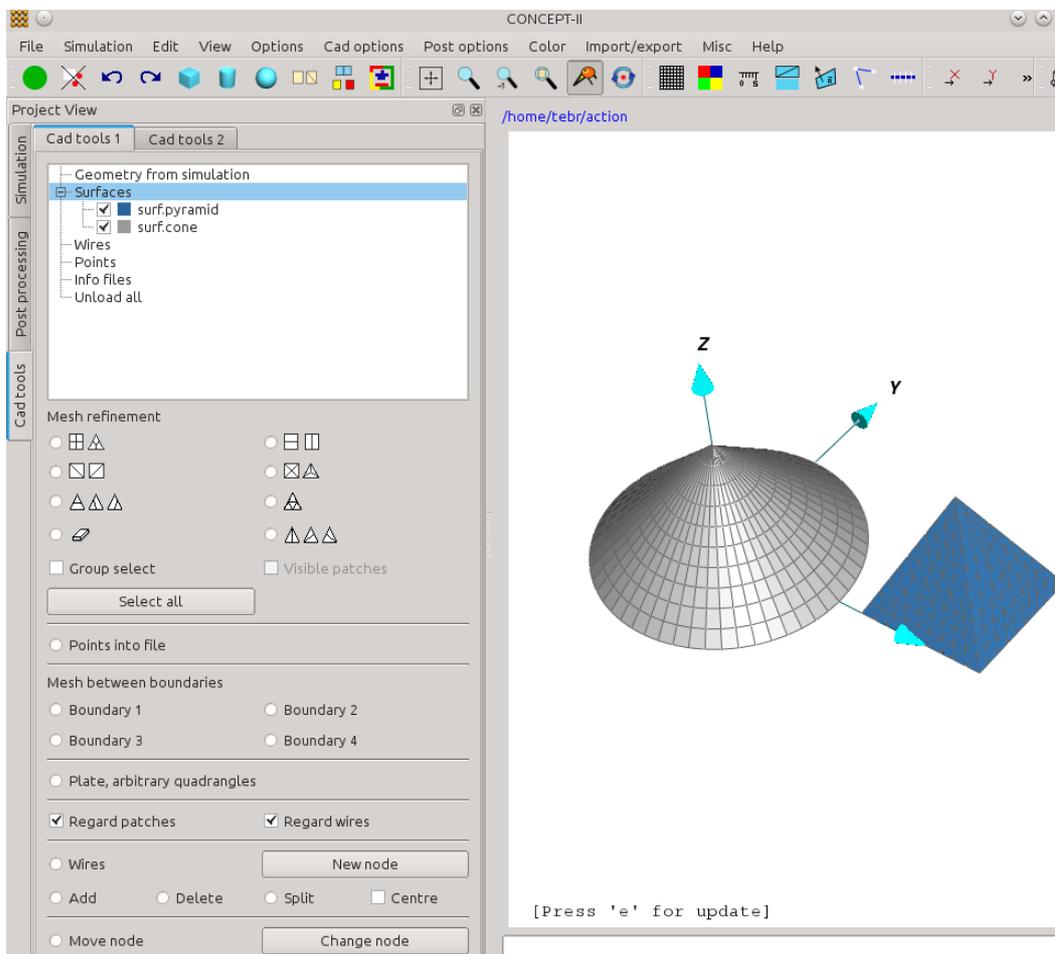


Fig 4.2: **Cad tools 1:** Tools for quick modification of grids. Create wires and surface parts. The cone has been created by the disk tool  and the pyramid with block tool 



### Tab Cad tools 1 (see Fig 4.2)

Select a preview pattern, for example  → move the mouse pointer over a triangle of the surface being displayed in the display area → a right click maps the preview pattern onto the corresponding patch. Typing <e> or clicking on  updates the grid.

Note that there are multiple pattern choices possible in some cases – right click the same patch for the desired one.

#### Application example 1

The quadrangles marked by black dots in Fig. Fig 4.3a) are unfavorably deformed. Select the quadrangle-to-triangle tool (preview pattern: ) and mark the

indicated quadrangles. Typing <e> generates the improved mesh as depicted in Fig 4.3 b).

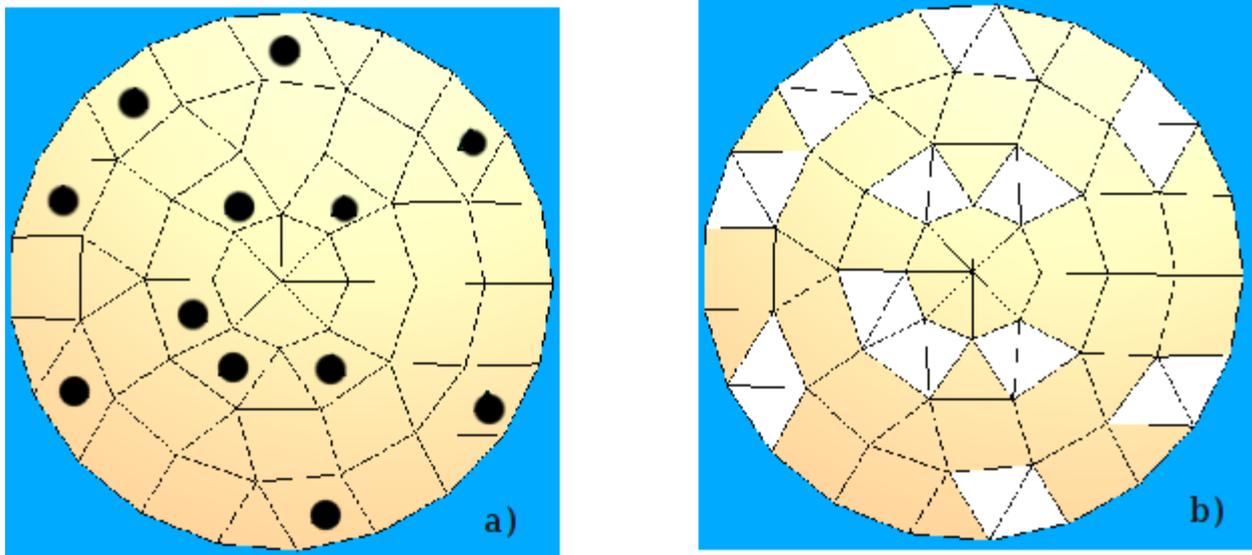


Fig 4.3: Improvement of an existing mesh. a) Original mesh with unfavorable quadrangles. b) Preview pattern (and final grid as no neighboring patches are effected)

Sometimes it is necessary to change or modify the mesh in large areas. To do this, choose the appropriate preview pattern and activate  Group select , then select the region to be modified: press and hold down the left mouse button and drag the mouse. All patches in the selected rectangular region are marked by the chosen preview pattern.



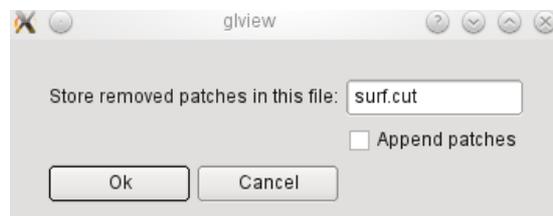
If  Visible patches is active only those patches are marked by a preview pattern which can be directly seen in the display area.

Click on  to rebuild the mesh.

### Application example 2

A cubical box according to Fig. 4.4 b) shall be discretized. An aperture has to be inserted into the top plate and the mesh has to be refined around the boundary of the aperture. The following steps are necessary for this task:

1. Create the cube, click on , edge length 1m, sub-figure a)
2. Select:   which divides quadrangles into 2 parts. As two symbols are depicted, a pattern can be mapped onto a quadrangle in both orientations.
3. Select the top patches according to sub-figure c).  
General rule: If there are several possibilities to place a preview pattern on a patch and the first choice does not have the desired orientation, click again: the next choice will be mapped onto the patch. This feature is also valid in the group selection mode.
4. Now we have to pay attention for the corners: place the pattern according to   as indicated in sub-figure d).
5. Click on , or type <e> → The mesh according to Fig. 4.4 e) is generated.
6. Select the rubber tool  and click on the corresponding patches of the top plate, see Fig. 4.4 f). Click on , or type <e> → the following window opens.



By default surface patches or surface parts that are selected for erasure will be written into the file *surf.cut* (surface patch format).

File names for storing deleted patches can be chosen as follows (Menu bar):  
**Cad options** → **Store removed patches...**

It is possible to reload the corresponding surfaces parts: **Cad tools 1** →



**Surfaces** → **Load surface file(s)**. It is not possible to take a file name of the list of loaded files for storing deleted patches again. Solution for this problem: **Cad tools 1** → **Surfaces** → **Rename**

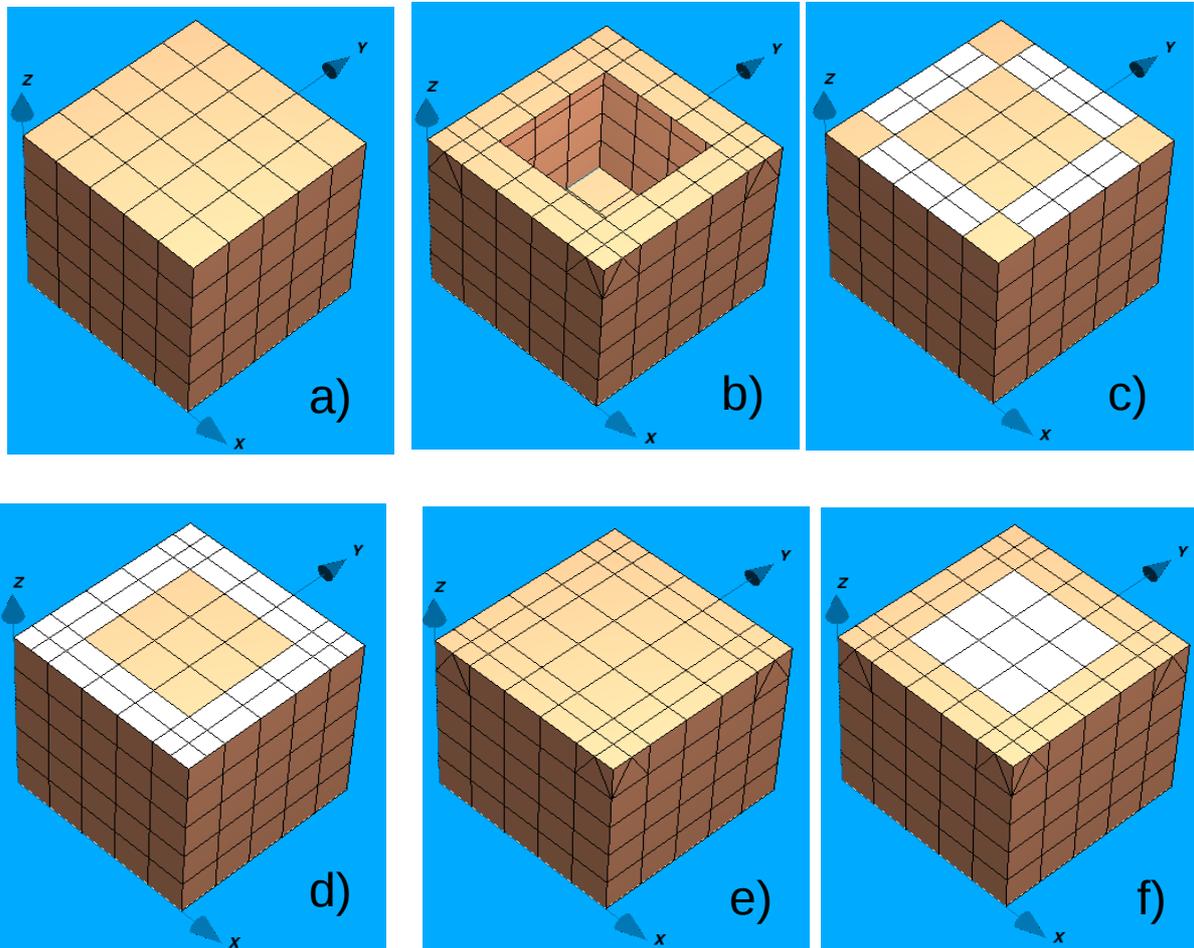


Fig 4.4: Cube with an aperture in the top plate. Sequence of modeling steps: a) c) d) e) f) b)

Finally the box according to Fig. 4.4 b) is generated.

### Application example 3

Let us assume that the erased aperture part (see previous chapter) has been stored under *surf.2* and that the box has been stored under *surf.1* (see previous chapter and Fig 4.5).

Load *surf.1* and *surf.2*: **Cad tools 1** → **Surfaces** → **Load surface file(s)**

Different colors of surfaces:



- (Menu bar) → **Color** → **Single structure** → select surface file → select color or
- Click on 

The mesh of *surf.2* (see Fig. 4.5 a) ) shall be refined as it is too coarse for capturing field coupling into the box improperly.

1. Activate this preview pattern:  and select all patches of the aperture as has been done in Fig. 4.5 b) for several patches already.
2. When all patches of the aperture file have been marked for refinement click on  → the mesh as illustrated in Fig. 4.5 c) is created, ready to be

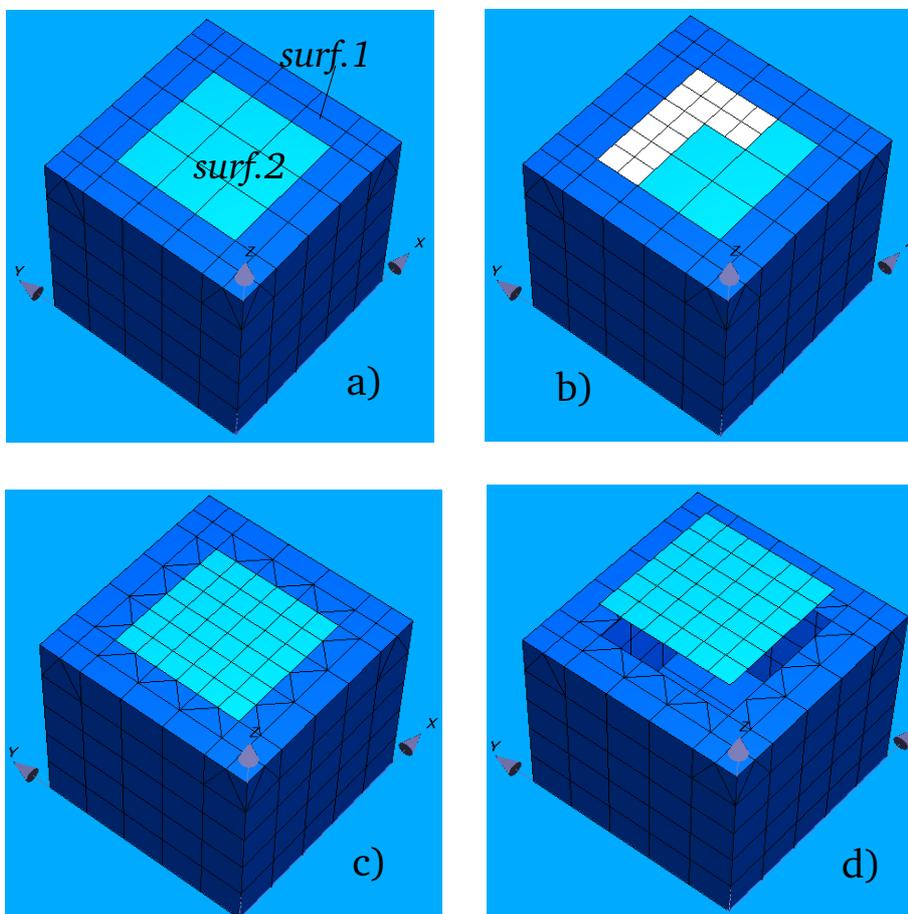


Fig 4.5: Example of a mesh refinement on a multiple surface structure with common boundary.

taken over into the **Simulation** section.

It is important to notice that although only patches of *surf.2* have been selected for refinement the patches at the rim of *surf.1* are automatically

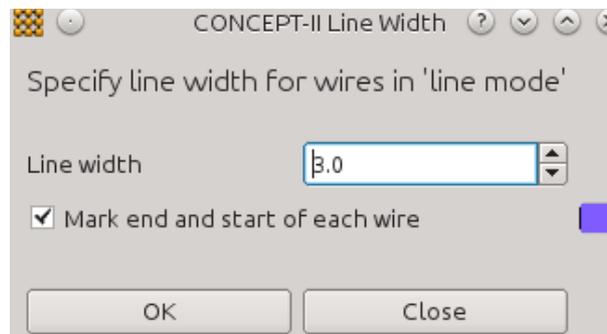


effected in order to keep the grid closed. Fig. 4.5 d) shows *surf.1* and *surf.2* when separated from each other. This has been accomplished by: **Cad tools 2** → activate **Move** → **Enter shift values** (0, 0, *z\_shift*) → select *surf.2* in the file list → type <e> or click on  → we will have the situation according to Fig 4.5 d), depending on the value of *z\_shift*.

## 4.2 Treatment of wires

Wires can be displayed in two ways

- **As straight lines** (lines drawn with pixel)  
This is the default mode when loading wires. Change the line width by
- **(Options** → **Draw wires as ...** → **Lines**), **Options** → **Settings for wires in line mode** → the following window appears:



When “Mark end and start of each wire” is active small dots are placed at the corresponding locations (color blue is chosen here). The representation of dots can be switched on by clicking on . The size of the dots can be

changed: **Options** → **Modify elements...** → ‘*Dot size for wire ends*’ → enter size.

Clicking on  indicates the wire segmentation according to the chosen

number of basis functions. Change color of wires: **Color** → **All wires** → select color

- **In 3D cylinder mode.** Switch on by: **Options** → **Draw wires as ...** → **Cylinders**.

Attention: in case of small radii the structure might become almost invisible.

Under the **Cad tools 1** tab the following section exists:





**Wires** → Once this button is activated wires can be added, deleted or subdivided. If the 3D wire representation mode is active it will automatically be switched to the line mode.

**Add** → Wires can be added: select at least two different nodes of the loaded structure by right mouse clicks and then . The user will be prompted to enter

the '*Name of wire output file*', '*No. of wires between the dots*' which have been marked, '*Radius in mm*', '*Number of basis functions on each wire*'.

If a new node shall be introduced click on **New node** → a window opens where new nodes can be entered in the top section (see Fig 4.6 ). Nodes that have been selected already appear under the '*List of last selected coordinates*'.

Clicking on **OK** adds the entered node with '*New coordinates x,y,z*' to the '*List of last selected coordinates*'. Clicking into the input fields '*X value*', '*Y value*', '*Z value*' puts the corresponding coordinates into the top line which can easily be modified.

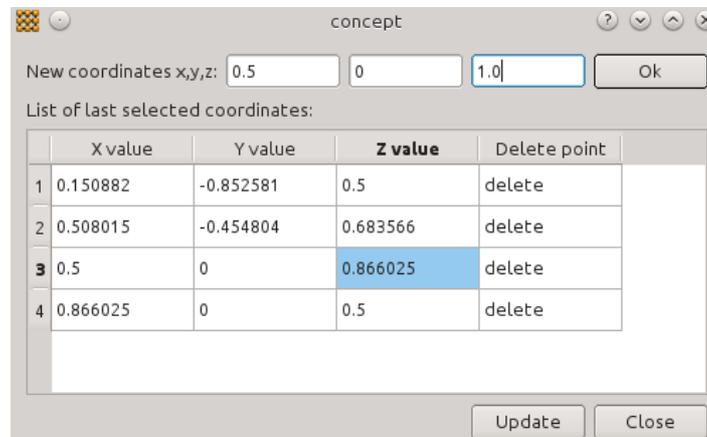


Fig 4.6: Setting of new nodes. Clicking on an existing coordinate value, for example on the high-lighted z value of node 3, causes this value appear in the list of *New coordinates x,y,z* where it can be edited. Editing is also possible by double clicking on the input fields. At least two points must have been specified before clicking on **Update**

At least two nodes are required in the '*List of last selected coordinates*'. Click on **Apply** → a sub-window opens → name of the wire output file, number of wire segments, radius and number of basis functions have to be specified → the new wire(s) are displayed. In case of a new file name the name is added to the list of



loaded wire files.

**Delete** → Select a wire by right click, then  → the wire(s) will be deleted. The default name of the file containing the deleted wires is *wire.cut*. The name can be changed: **Cad options** → **Store removed wires...**

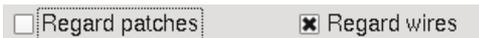
**Split** → Right-click on an arbitrary location along a wire → the position is marked. Click on  → a window opens allowing to set the precise position where to divide

the wire into two parts. It is also possible to subdivide a wire into several wires.

Enter the split factors in an increasing order, for three wire we have:  $0 < a < b < 1$ .

$a=0.25$ ,  $b=0.75$  will lead to three wires: the first one and the last one will be of equal length and the center one will be of double length.

Sometimes it is difficult to pick the right node, for example if many wires are in front of a rather fine surface patch grid. Then it can be helpful to switch on the following option:



Now surface patch nodes cannot be selected any more.

### 4.3 Generating triangular and quadrangular plates

When the radio button  Plate, arbitrary quadrangles is enabled (**Cad tools 1** tab) arbitrary

quadrangular plates can easily be discretized: select 4 nodes (wire end points, wire start points, patch nodes, points in space) in a clockwise or counterclockwise manner and click on .

The procedure shall be demonstrated in detail by an example shown in Fig. 4.7 . The top region of the structure (sub-figure a)) shall be closed, i.e., the final result should look as depicted in sub-figure f).

- Select the corner points of the top rectangular region as depicted in Fig. 4.7 b) in the order 1->2->3->4. In the general case marked nodes do not need to span a flat rectangular region. If only 3 corner points are selected a triangular surface or - in the simplest case - a single triangle is generated, depending on the number of meshes between the corner points being selected.



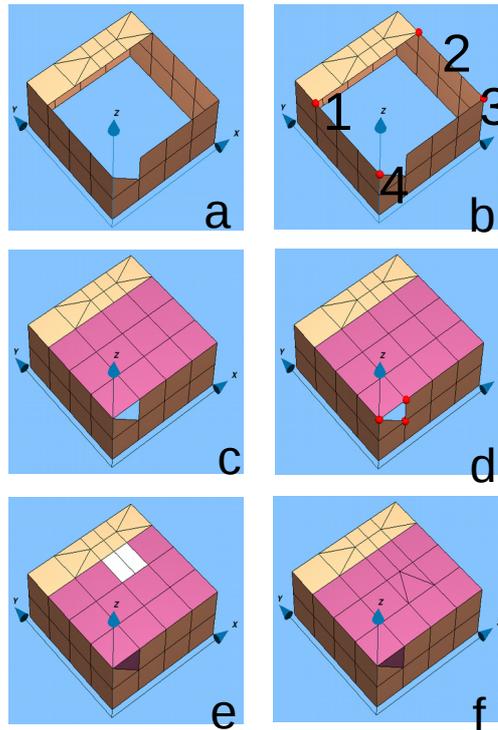


Fig 4.7: Using the plate tool of the **Cad tools 1** section

- Click on  → a window according to Fig 4.8 opens:

Enter the output file (*surf.top* in our case). Enter the number of 'Meshes' between the nodes.

Another option would be to enter 'Frequency in MHz' and the 'No. of Basis functions'.

- Clicking on **OK** provides the mesh as depicted in Fig. 4.7 c). *Surf.top* has been created with a certain color. Change color by **Color** → **Single structure** → select *surf.top* → select the color.
- As can be easily seen in Fig 4.7 c) there is still a small triangular region which needs to be closed, hence select the three nodes as indicated in Fig. 4.7 d).

Under 'Meshes' according to Fig. 4.8 we have to enter four times a "1", the output file remains the same. Click on  → We see the mesh as illustrated

in Fig. 4.7e).



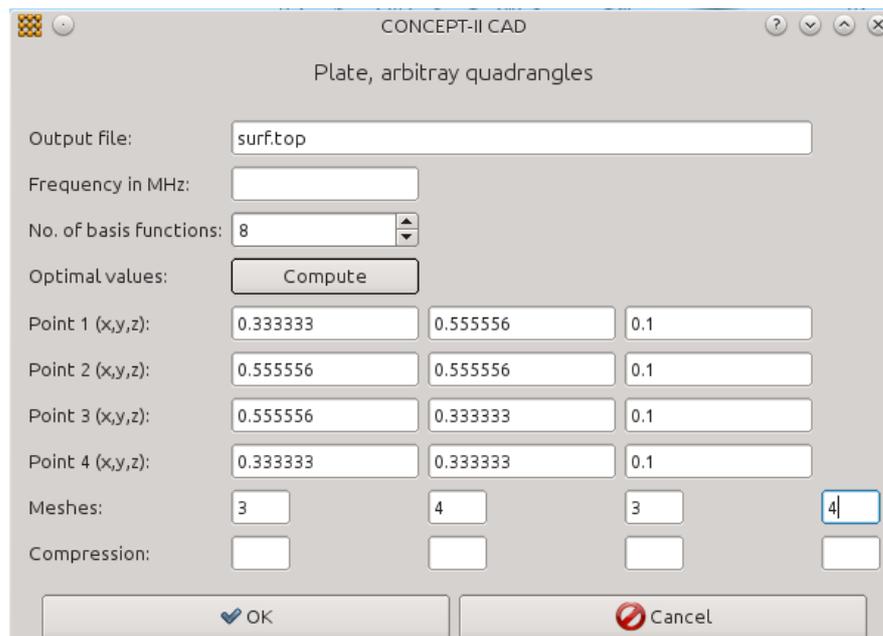


Fig 4.8: Discretizing plates under CAD tools 1 with the plate tool

- The patch grid is not closed, see sub-figures c) and d). To repair this we have to refine the grid locally as has been indicated by the preview pattern in Fig. 4.7 e). Clicking on  provides the final result.

To create a new plate the following possibilities are available:

1. The nodes of an existing surface patch grid and/or the nodes of wires can be selected. There is no need that such wires belong to the structure under investigation, they may simply serve for marking the outline of a structure part or similar. Only those wires that have been imported into the **Simulation** section are part of the structure to be analyzed by the MoM.
2. Additional points in space can be created: **Cad tools 1** → **Points** → **Edit a new point file** → a window (similar) to the one shown in Fig. 4.9 opens.

Here the xyz coordinates can be entered. Such points are shown as small spheres in the display area. Points can also be generated from the structure itself by right-clicking on nodes.



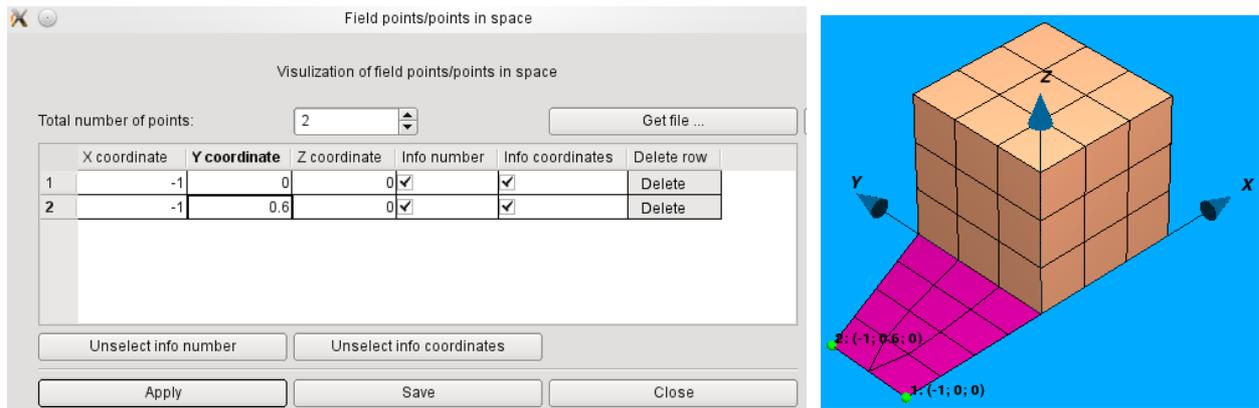


Fig 4.9: Left: A window for entering arbitrary points in space with two example entries. Right: The corresponding points are used to discretize the additional bottom plate.

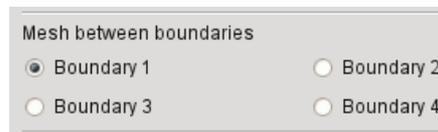
Hint: A separate plate tool with an enhanced number of features is available under **Cad tools 2** , .

#### 4.4 Using the boundary tool

Surfaces with curved boundaries can be discretized with the boundary tool:

Fig 4.10: The boundary tool

( see **Cad tools 1** )



The required steps for discretizing a surface with two non-uniform boundaries shall be explained by Fig. 4.11. The missing surface part within the disk section has to be filled (Fig 4.12 a)

- Set the radio button "*Boundary 1*" (see Fig. 4.10) to active and select the nodes of the first boundary by right mouse clicks in a sequence. We get the row of blue marked nodes in Fig. 4.11b).



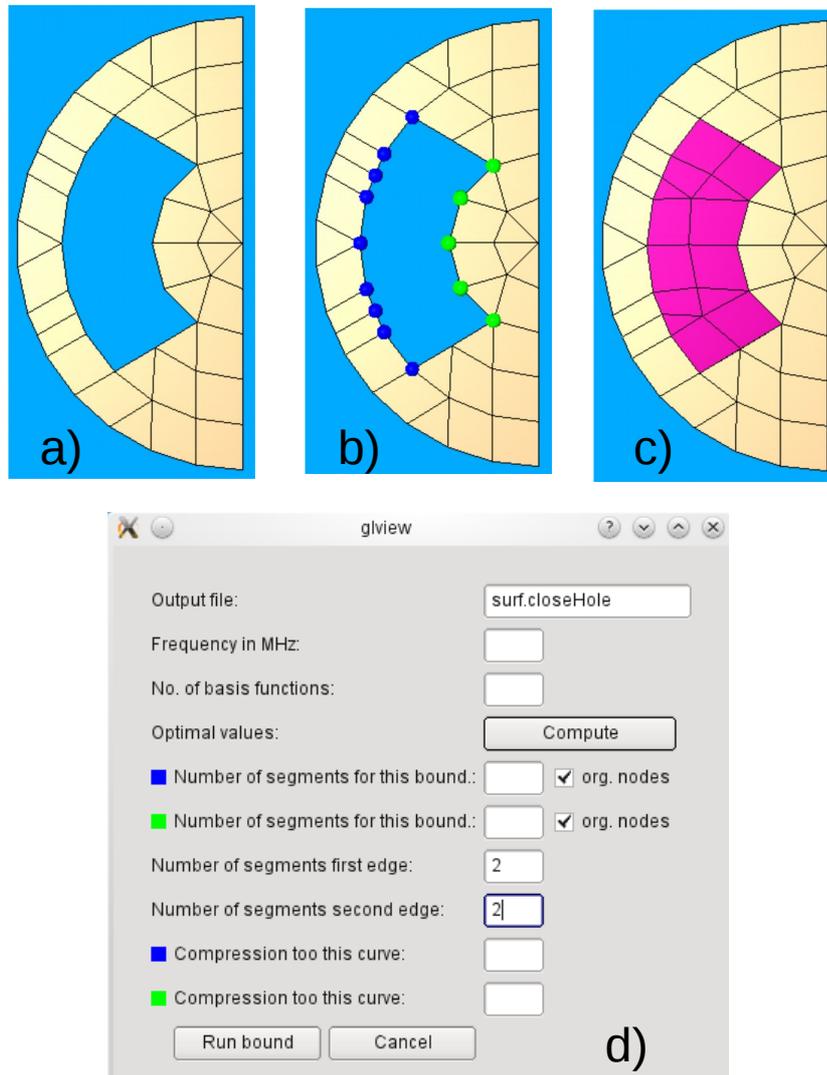


Fig 4.11: Working with the boundary tool

- Set the radio button “*Boundary 2*” to active and select the nodes of the opposite boundary - in our example we have the green marked nodes. It does not matter which boundary is selected first.
- Click on  or type <e> → the window according to Fig. 4.11 d) opens. The selected boundary nodes shall become grid nodes, therefore we enable the two check boxes ‘*org. nodes*’ and enter a ‘2’ for the ‘*Number of segments first edge*’ and ‘*Number of segments first edge*’ for the remaining boundary sections of the area to be filled
- Clicking on **Run bound** closes the aperture by a surface patch grid as illustrated in Fig. 4.11 c)

Hint: Using the boundary tool the coordinates of selected nodes are immediately



written in an ASCII file. Boundary 1 is active → the file *bound.1* is generated, Boundary 2 is active → the file *bound.2* is generated. This may be helpful for many purposes, for example for a copy and paste action of complicated coordinates. The mentioned files can easily be loaded by **Cad tools 1 → Points → Load point file(s)**

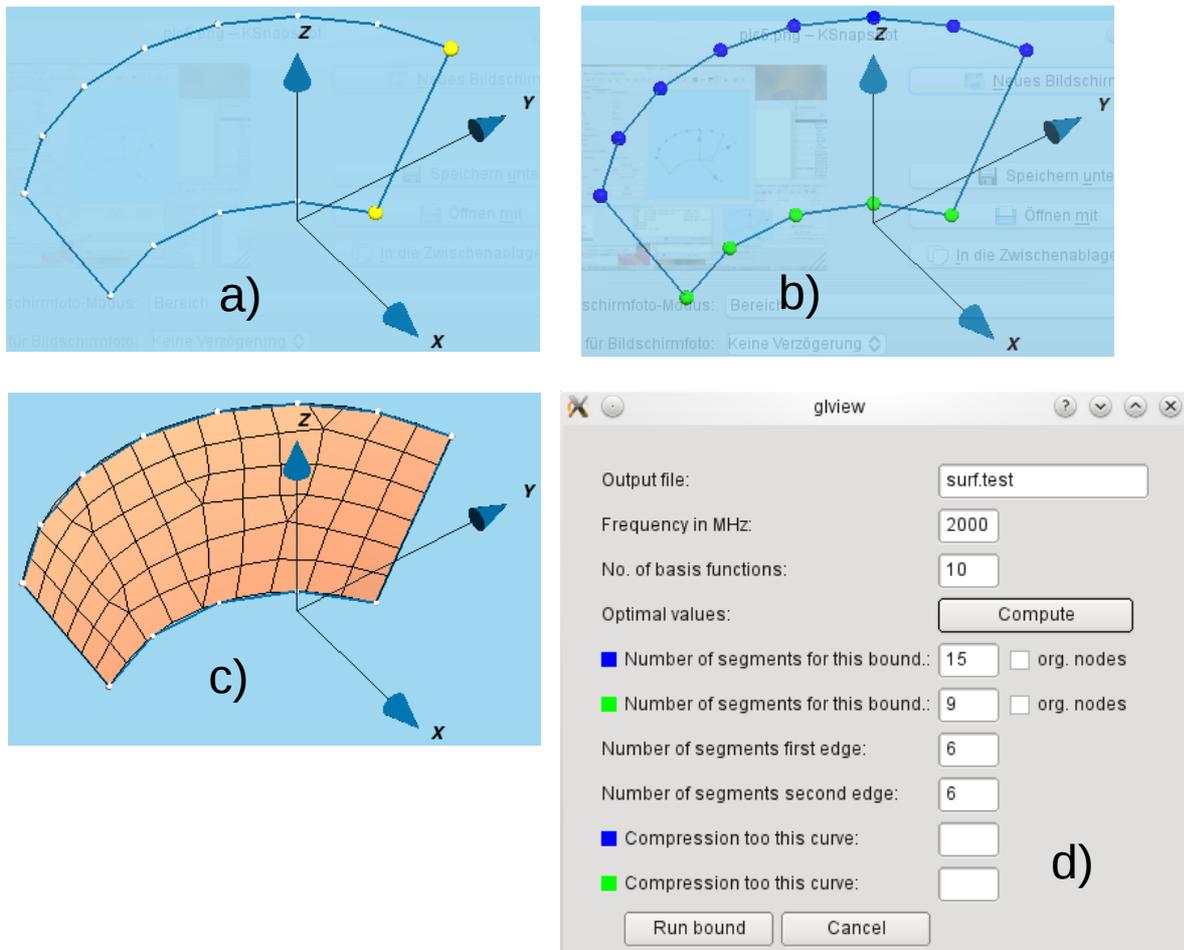


Fig 4.12: Working with the spline mode of the boundary tool

Another option for using the boundary tool shall be demonstrated by means of Fig. 4.12. The task is to discretize the area which is bounded as illustrated in Fig 4.12a) by a straight wire sections (part of a conical surface). The distance between the yellow dots is 8 cm and the assumed frequency is 2 GHz. Therefore the intervals between the white dots are too large to be directly taken as patch nodes. To generate a grid according to the '10-patches-per-wavelength' rule of thumb the following is suggested:

- Mark the upper and lower curved boundaries (see Fig. 4.12 b)) as has been described in the previous example



- Click on  → the window according to Fig. 4.12 d) opens. Now enter the 'Frequency in MHz' and the 'No. of basis functions' (per wavelength) and click on **Compute** → the number of segments along all 4 boundaries are evaluated automatically.
- Click on **OK** → the mesh according to Fig. 4.12 c) is generated. Notice that the original nodes are not used any more; i.e., the curved boundaries are approximated by spline functions and the necessary nodes have been equidistantly inserted.
- Be careful when generating grids in the same way on connected surface parts. It is strongly recommended in such a case to check the grid by means of the check tool  to find out if it is closed at the common boundary. If small gaps appear with nodes very close to each other, the grid can easily be closed by means of the combine tool  (see **Cad tools 2**). Set 'Distance between nodes' appropriately.

## 4.5 Turn a wire contour model into a surface patch model

In practical cases the contour of a physical structure may be available. An example is depicted in Fig. 4.13 a). We have the outline of a ship generated from wire data. Because of symmetry only half the model needs to be considered setting up the desired surface patch model (see `$CONCEPT/examples/ship/ship_symmetry/ship.wire`).

The following steps are suggested to generate a mesh suitable for a maximum frequency (here 30 MHz, see below), part of the final model can be seen in Fig. 4.13 b).

1. Open tab **Cad tools** and load the wire-contour model (double click on **Wires, aux.lines** → **Load wire files(s)...**). In the general case it is recommended to clean and complete the wire structure, redundant wires should be deleted. This can be accomplished by activating "Wires" and "Delete", tab **Cad tools 1**. Mark the wires to be erased by a right click, then press .

Wire can be divided into two parts: use the wire-split option (Set "Wires" and "Split" to active).

Use **Cad tools 1** → **Points** → **Edit a new point file** to create points in space, i.e., enter the coordinates. Spatial points appear as tiny green dots by default.

Additional wires can be created by setting "Wires" and "Add" to active. Select



the start point (or node) and end point (or node) and click .

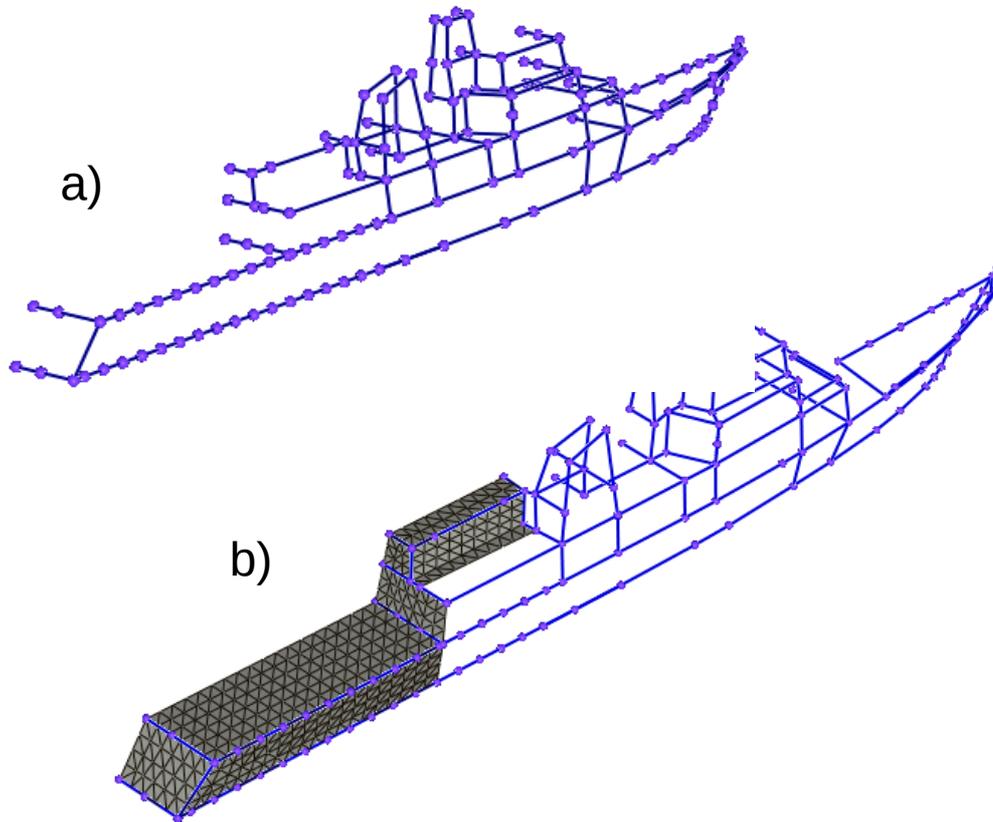
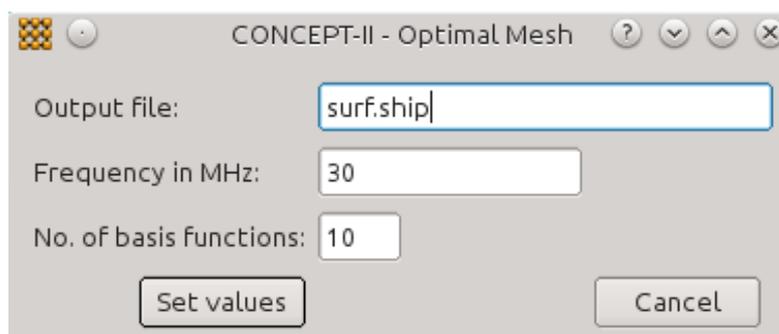


Fig 4.13: Using a wire grid model for generating a surface patch model. The original mesh of the depicted surface parts had both quadrangles and triangles. A mesh of only triangles was generated by clicking on  (Tool bar). See `$CONCEPT/examples/ship/ship_symmetry, file ship-sym.surf`).

2. The discretization of sub-plates can be accelerated by:

a) **Cad options** → **Set meshing parameters...** → the following window opens:



Notice the entries. The overall length of the model under consideration is 72 m.

b) **Cad options** → set **Use meshing parameters** to active

Having selected the corner points of a new sub-plate to be introduced (see Sec. 4.3), the corresponding mesh will immediately be generated once the user has entered <e> (key “e”) or clicked on  ('Plate, arbitrary quadrangles' is active)

For the rapid positioning of the mouse pointer turn the mouse wheel for quickly zooming in or out. Press the <CTRL> key + click the left mouse button and drag to shift the depicted structure in the display area. Once all sub-plates have been introduced it is recommended to check the grid.

Click on  (**Cad tools 2**). A window opens. Set check mark 'Find free

nodes' to active. Click on **Apply** → the contour between free nodes is shown in wire format (file: *wire.freenode*). Note that line sections are only allowed to appear along open boundaries. When they appear at places where the grid should be closed, there is an error – the corresponding nodes are separated and the *CONCEPT-II* back end will not be able to compute currents crossing that line. Once the mesh has been generated and errors have been repaired, use the following tool  (**Cad tools 2**) to

complete the structure for the full ship.

Introduce an HF antenna as has been described in Sec. 4.2 and generate a local mesh refinement at the base of the antenna if necessary.

Load the structure into the **Simulation** section ...

## 4.6 The point option for meshing

Points in space or points which are positioned as markers on existing grids can be effectively used for meshing purposes. In Fig. 4.14 a transmission line (wire loop) on a discretized square plate is shown.

Having lines over a discretized ground it is important that the typical edge lengths of patches underneath the horizontal conductor should correspond to the height of the conductor (rule of thumb).

Although this condition is approximately fulfilled in the surface grid according to Fig. 4.14a), it is a bad grid (see Sec 1.7). How to improve this? The following proceeding is possible:

- Mark important points, vertices etc. as indicated in Fig. 4.14 b): **Cad tools 1** → **Points** → **Edit a new point file**. Enter the name of the point file select nodes by right-clicks. **Apply** or **OK** saves the file.



Hint: Clicking into the area of a patch takes the corresponding coordinates.

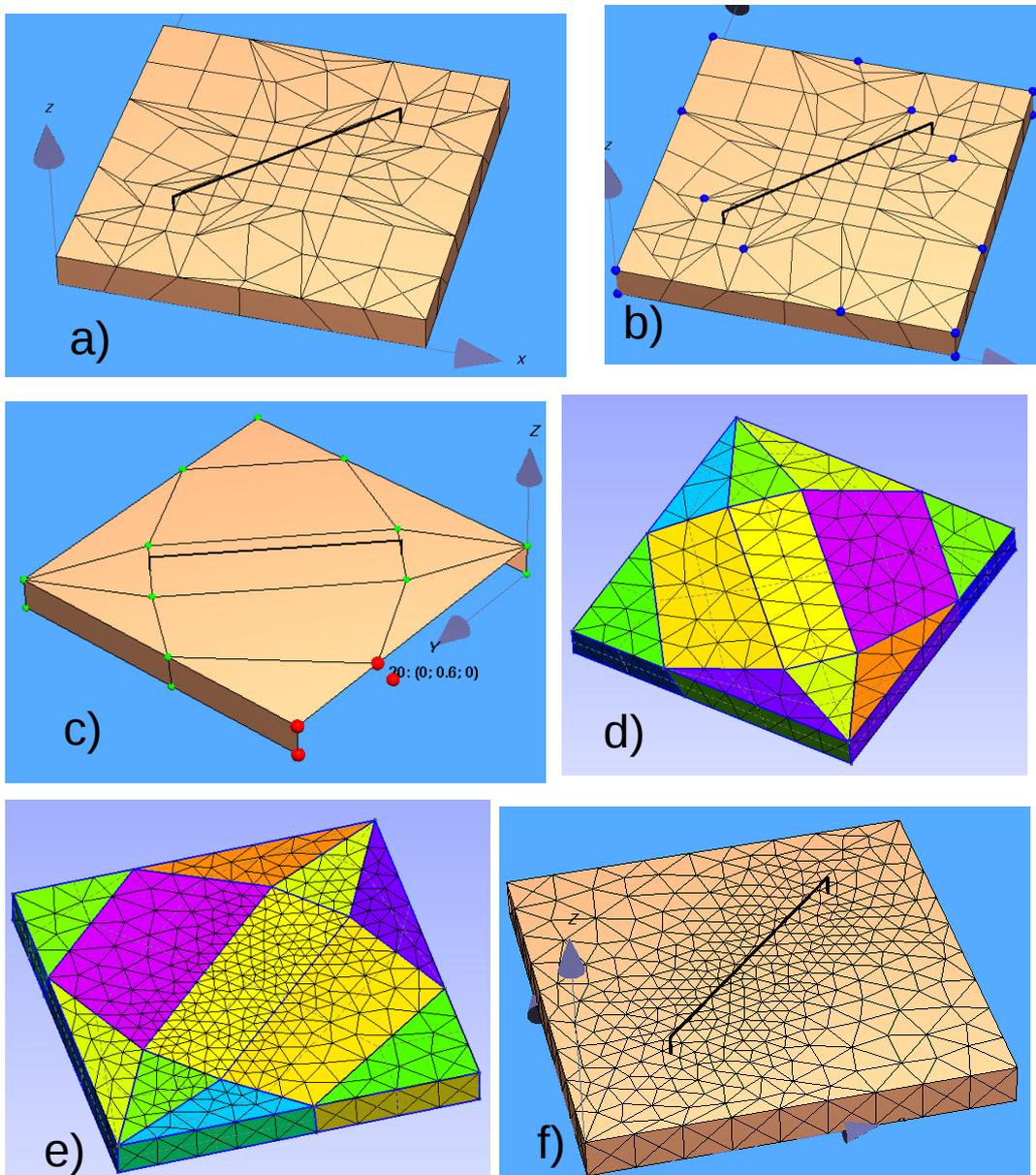


Fig 4.14: Making a good grid out of a bad grid using the *CONCEPT-II* **Import/export** tool in conjunction with *gmsh* (see Sec. 4.8)

- Unload the surface patch file containing the ugly grid; the points and the wire loop remain in the display area. Based on the pattern of points a rough plate grid can easily be created as can be seen in Fig. 4.14 c). Proceed as described in Sec. 4.3. by assuming only a single mesh element on each side of a quadrangular region. Save the resulting surface patch file.

This data shall now be used to create a suitable frequency-dependent grid.



The external tool *gmsh* shall be used to for this purpose (see Sec. 4.8).

*Menu bar* → **Import/export** → the Import/export dialogue window opens → proceed as described under Sec. 4.10. The resulting surface patch grid is displayed in the *gmsh* window as illustrated in Fig. 4.14 d). The assumed frequency was 300 MHz, the square plate is 1m x 1m and the *characteristic edge length* according to *gmsh* has been set to 0.1m. The height of the horizontal wire is 5 cm, hence an additional grid refinement is necessary along the course of the line.

- Start *gmsh* outside the CONCEPT-II GUI. *Gmsh* → '+Mesh' → '+Define' → 'Size at points' → window opens → set 'Element size' to 0.04 → select corner points of center section by left mouse clicks and close this action by typing 'e' → '+Geometry' → 'Reload script' → '+Mesh' → '2D' → the grid according to Fig. 4.14 e) appears.

Before re-meshing the grid it could be advantageous to set: 'Tools' → 'Options' → 'Mesh' → '2D algorithm' → 'Delaunay'

Save the data in .stl or .msh format.

- Turn back into the *CONCEPT-II* GUI and convert the .stl or .msh file into the *CONCEPT-II* surface patch format using the **Import/export** tool.

Note: .msh files can also be directly loaded into the simulation (Tab Simulation → double click Surfaces → Load surface files(s)... .

We get the structure as can be seen in Fig. 4.14 f). Notice the improvement compared with the grid according to Fig. 4.14 a)!

- Practically we might have the situation that the vertical wire is not connected to a surface patch node. This can very easily be corrected:
  - Move the nearest patch node to the wire node (**Cad tools 1** → **Move node** → select the corresponding patch node and then at the wire node)
  - Introduce an appropriate grid refinement at the junction point as is demonstrated in Fig 4.15 a), see also the application examples in Sec. 4.1.
- An example of a computed current distribution is shown in Fig. 4.15b).



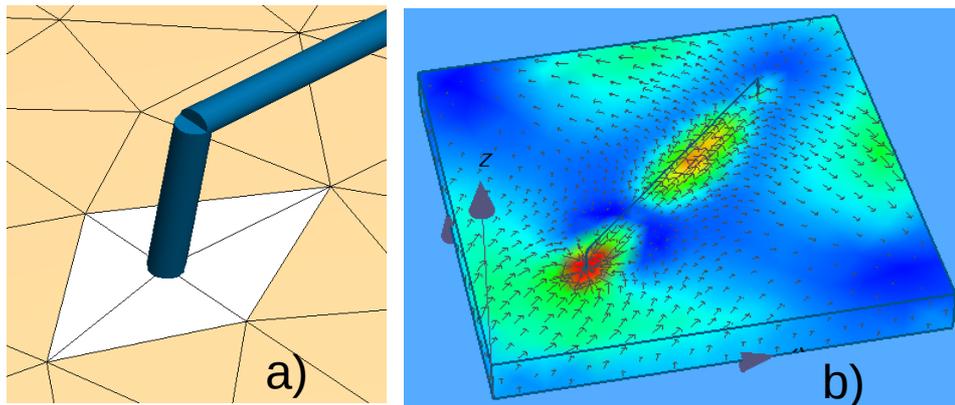


Fig 4.15: Mesh adaption at vertical wires and an example current distribution at a certain phase and frequency.

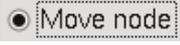
#### 4.7 Joining surface parts separated by unwanted gaps

It might happen that the nodes forming the boundary of different adjacent surfaces which were combined to form a single surface with a unique node and patch numbering do not match (combine tool , **Cad tools 2**).

Example situation: there is a half cylinder and half of a circular disk. These two surface parts shall be combined to form a single structure part as depicted in Fig. 4.16 e). It is immediately clear that there is a grid error at the upper rim having combined the structure parts. In the general case of a complicated grid the situation might not be so obvious but can be detected by means of the grid check tool .

(check box 'Find free nodes' is set to active). Application of the check tool provides the file *wire.freenodes* which is automatically loaded, see Fig. 4.16 b); dots are marking the wire ends.

The following proceeding is suggested to repair the mesh:

- **Cad tools 1** →  → Right click on a node to be shifted, right click onto the position where the selected node shall be shifted to. In some cases

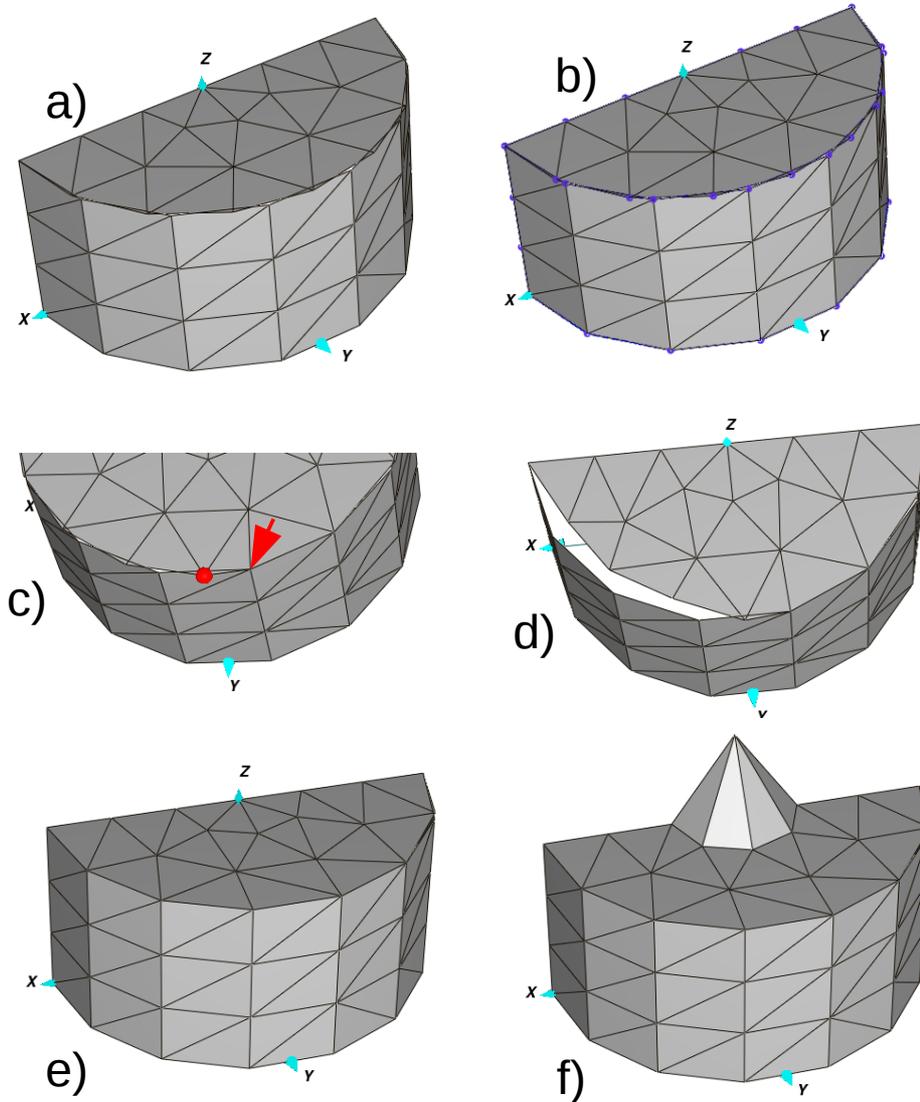


Fig 4.16: Surface patch nodes do not match at the interface line between the top semicircular disk and the half cylinder. a) Grid is not closed along the rim of the top disk. b) Result of the grid check tool. c) Shifting the red dot to the position the arrow is pointing to will erase the top triangle automatically. e) the gap has been closed. d) shifting a node by the **Change node** option.

it is advantageous to make the gap even larger by first shifting the nodes clearly “to one side” as illustrated in Fig. 4.16 d) before closing the gap. In Fig 4.16c the red dot marks the node which shall be shifted to the

position where the red arrow is pointing to. By this action the corresponding triangle will be deleted automatically.



Hint: If unsure if a single node exists or two (or more) nodes hiding each other, shift the doubtful node by a small distance into one of the adjacent patches. If no gap opens there is only a single node. Click on  to get the initial situation back again.

- The coordinates of nodes can be changed to positions which are away from any node or point. Select a node (example: Fig. 4.16 e),. Click on **Change node** → a window opens, enter the new node coordinates → **OK** → the node coordinates are immediately changed, see Fig. 4.16f).

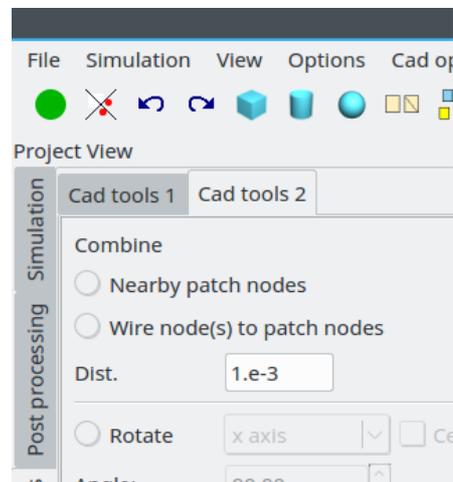


Fig 4.17: A convenient possibility to achieve a closed grid in case of small gaps

Imagine the case of several surfaces forming a dielectric body. Let us assume that the grid is not closed as required. The nodes along the boundaries of adjacent surfaces are not being positioned on top of each other as has been detected by the grid check tool; there is a situation similar to the one as depicted in Fig. 1.5 with the gap hardly being visible by inspection. Under such circumstances we can use the option according to Fig. 4.17, 'Combine → Nearby patch nodes' is active. All adjacent nodes will be combined to a single node once the distance between them is smaller than 1 mm in the example. All loaded surface patch files (see Cad tools 1) will be investigated.

## 4.8 Importing and exporting

The STL format (and the *gms*-specific .msh format) can be converted into the *CONCEPT-II* surface patch format (file names starting with *surf.* or ending with *.surf* will be automatically recognized in the GUI, for details see Sec. 8). *CONCEPT-II* is



able to read .msh files directly. Wires and patches have to be stored in different .msh files.

The STL format is supported by many CAD software packages and refers to triangulated surfaces of three dimensional objects. The format includes ASCII and binary representations, both are supported.

In case of other CAD formats:

*gmsh* can import data of formats such as IGES and STEP and can convert them into the .msh or the STL format.

### 4.8.1 Basic example for data conversion and application of *gmsh*

CONCEPT-II main window, **Misc** → **Import/export** → A window as illustrated in Fig. 4.18 opens.

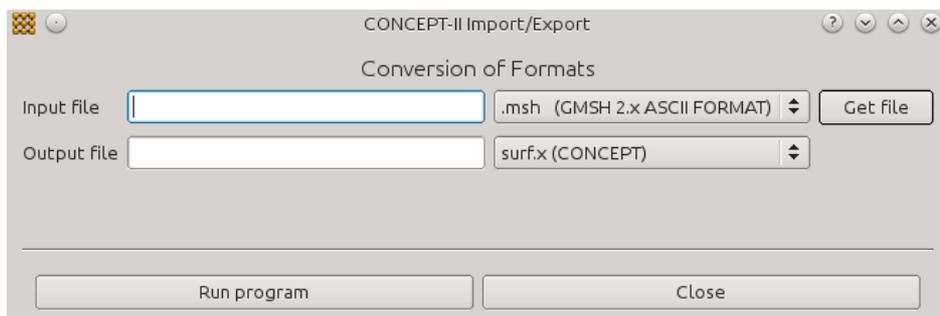


Fig 4.18: Format conversion in *CONCEPT-II*

The program *gmsh* is available on the internet (<http://geuz.org/gmsh/>). It is distributed under the conditions of the General Public Licence (GPL) [C.Geuz-2009] and not part of *CONCEPT-II*.

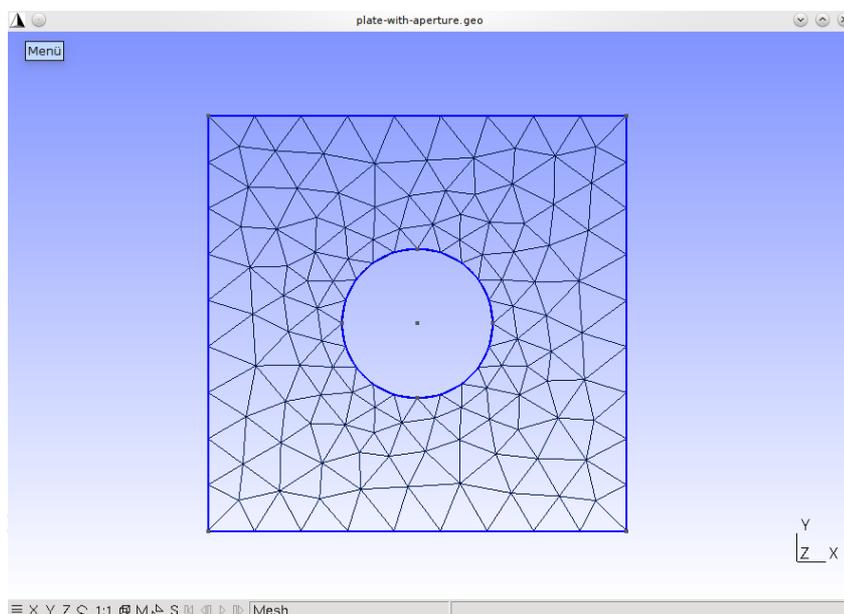


Fig 4.19: A triangulated plate with a circular aperture (*gmsh* display area)

Suppose that a triangulated plate with circular aperture shall be modeled as illustrated in Fig. 4.19 (part of the *gmsh* display area is depicted).

For the application of *gmsh* please study the respective documentation, examples are on the *gmsh* web page. In our case the file to be set up and to be loaded looks like this (suggested name *plate-with-aperture.geo*):

```
// Surface patch model of a plate with hole
// Characteristic length of outer plate boundary
clplate=1;
//Position of hole...
div=2;
// width, height of the plate
b = 10;
h = 10;
// Vertices of the plate
Point(1) = {0.0, 0.0, 0, clplate};
Point(2) = {b, 0.0, 0, clplate};
Point(3) = {b, h, 0, clplate};
Point(4) = {0.0, h, 0, clplate};
// Aperture
// Radius of aperture
rap =0.5;
// characteristic length
clap=0.5;
// Center point of aperture
Point(9) = {b/div, h/div, 0, clap};
// 4 points around the center of the circular aperture
Point(10) = {b/div, h/div+rap, 0, clap}; Point(11) = {b/div,
h/div-rap, 0, clap};
Point(12) = {b/div+rap, h/div, 0, clap}; Point(13) = {b/div-rap,
h/div, 0, clap};
Line(1) = {4, 3}; Line(2) = {3, 2};
Line(3) = {2, 1}; Line(4) = {1, 4};
Circle(5) = {12, 9, 10}; Circle(6) = {10, 9, 13};
Circle(7) = {13, 9, 11}; Circle(8) = {11, 9, 12};
Line Loop(9) = {4, 1, 2, 3};
Line Loop(10) = {6, 7, 8, 5};
Plane Surface(11) = {9, 10};
```



~

Copy the above script into the file *plate-with-aperture.geo* and open it:  
 > *gmsh plate-with-aperture.geo* <Return>

*Gmsh* can export its *.msh* format into the STL data format: *plate-with-aperture.stl* will be generated.

The next figure illustrates the entries to be made to convert the STL format into the CONCEPT-II surface patch format (**Import/export** tool).

|             |  |  |
|-------------|--|--|
| Input file  | <input type="text" value="plate-with-aperture.stl"/> | <input type="text" value=".stl (ASCII FORMAT)"/> |
| Output file | <input type="text" value="surf.plate"/>              | <input type="text" value="surf.x (CONCEPT)"/>    |

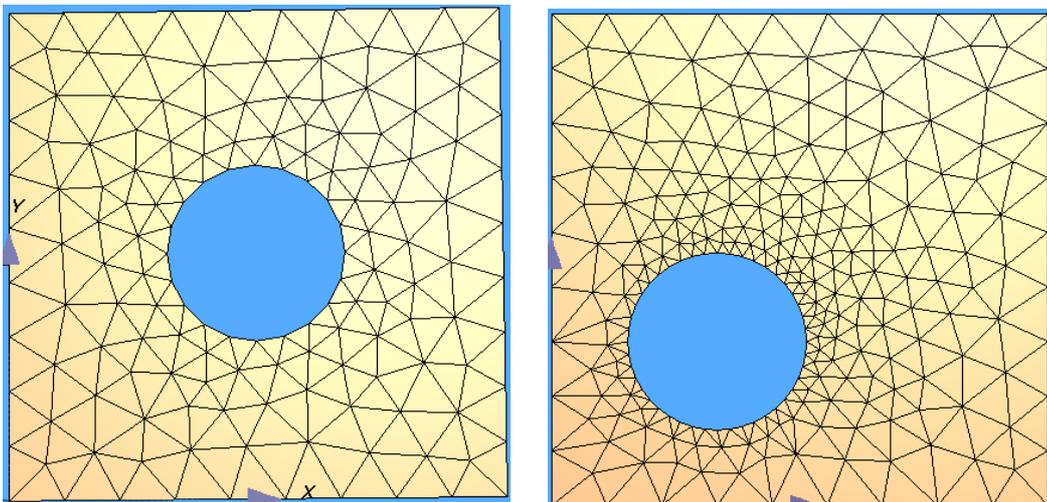


Fig 4.20: STL ASCII files have been converted into the *CONCEPT-II* surface patch format (surf. or .surf)

Fig. 4.20 depicts the result of the conversion. The left surface patch grid is based on the contents of the *plate-with-aperture.geo* file as printed out above.

Note that it is very convenient to modify the plate size (parameters: *b,h*) or the radius of the aperture including the mesh density at the rim of the aperture (parameters: *rap, clap*). For the mesh on the right side the position of the aperture (*//Position of hole...* ) and the mesh density at the aperture boundary (*clap*) were changed. In order to get a modified grid edit the file *plate-with-aperture.geo* by an editor of choice and click on '*Reload script*' in *gmsh* (+*'Modules'* → +*'Geometry'*).



## 4.9 Converting the wire format into the .geo format of gmsh and vice versa

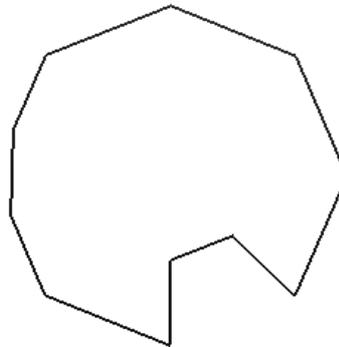


Fig 4.21: Outline of a surface described by wires (*CONCEPT-II* wire format).

The procedure shall be demonstrated by means of an example: let us suppose that the outline of a special plate structure looks as illustrated in Fig. 4.21 .The depicted region shall be triangulated.

Once the outline of an arbitrary structure is described by wires (*CONCEPT-II* wire format) the corresponding file can be converted into the .geo format of *gmsh*.

Let the wire file name of the example boundary according to Fig. 4.23 be *wire.outline* . Then the entries as illustrated in Fig.4.22 are required to generate the .geo file.

|                  |   |   |
|------------------|---|---|
| Input file       | <input type="text" value="wire.outline"/> | <input type="text" value="wire.x (CONCEPT)"/>             |
| Output file      | <input type="text" value="outline.geo"/>  | <input type="text" value=".geo (GMSH)"/>                  |
| Frequency in MHz | <input type="text" value="3000"/>         | Number of basis functions <input type="text" value="10"/> |

Fig 4.22: Entries for a the conversion of an outline file given in the *CONCEPT-II* wire format to the .geo format of *gmsh*

The diameter of the contour as shown in Fig. 4.21 is 0.2 m, hence setting the frequency to 3 GHz will provide a rather fine mesh already. For “Number of basis functions” 10 is a good first choice. Running the conversion tool provides *outline.geo*, which is an ASCII file, which can easily be modified by an editor, for example by setting the frequency to another value.

At the prompt:

> *gmsh outline.geo* <CR> → the *gmsh* window opens, showing the outline.

Continue with: '+Geometry' → '+Elementary entities' → '+Add' → 'Plane surface'  
→ select the boundary by left mouse clicks → type 'e' → it is indicated that a surface



is ready to be discretized (dashed crossed lines) → '+Mesh' → '2D'

The mesh as shown in Fig. 4.23 is generated. The grid can be exported in STL format and can then be converted back into the *CONCEPT-II* surface patch format as has been described in the last section. Note that *CONCEPT-II* can directly read gmesh files that have been exported as .msh files. Starting with October 18 both "Version 2.0 and Version 4.0 ASCII formats are supported).

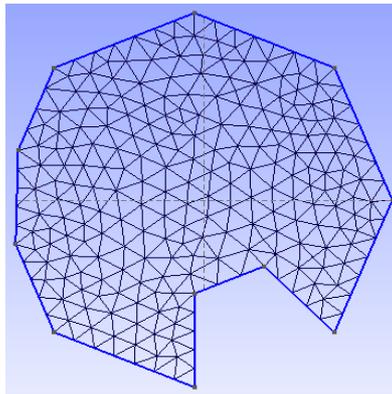


Fig 4.23: The triangulated plate according to Fig.

Note that files containing arbitrary wire structures can be converted into the .geo format. In general wires are used as auxiliary lines for describing the outline of the structures.

Wire structures can also be generated by means of *gmsh*. Choose 1D elements (for the discretization of lines). Export or save as a .msh file, then load the .msh file into the simulation. Default radius is 1 and the default number of basis functions is 3. 'Physical groups' as might have been defined in *gmsh* . are converted in separate files. The file name are generated as follows:

xxx-mshnr.wire with xxx as the gmesh name without suffix and nr as a sequential number.



## 4.10 Using the CONCEPT-II patch format for mesh generation in gmsh

Surface patch files of *CONCEPT-II* format can be converted into the *.geo* format of *gmsh*. Each patch is converted into the surface description of *gmsh*.

Again, let us consider an example. The structure according to Fig. 4.24 can be found under `$CONCEPT/examples/ship/ship_symmetry`, file `ship-geo.surf`. The

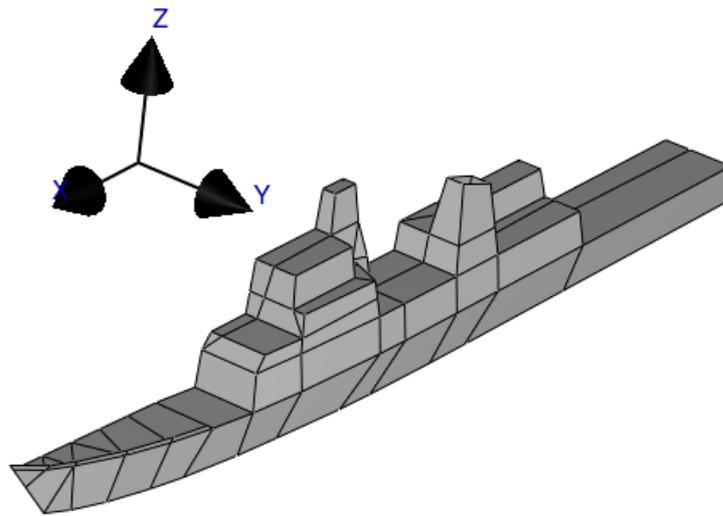


Fig 4.24: A very rough surface patch model of a ship (here only the symmetric part for  $y>0$ ). How to set up such a structure has been described in Secs. 4.3 and 4.5

length of the ship is around 72 m. All partial regions have to be converted into the corresponding *gmsh* format.

Let us suppose that we want to have a patch grid which is suitable for a 100 MHz application. A triangulated model is required for this frequency assuming 8 basis functions per wavelength, output file: `ship.geo`, see Fig. 4.25. Click on **OK**.



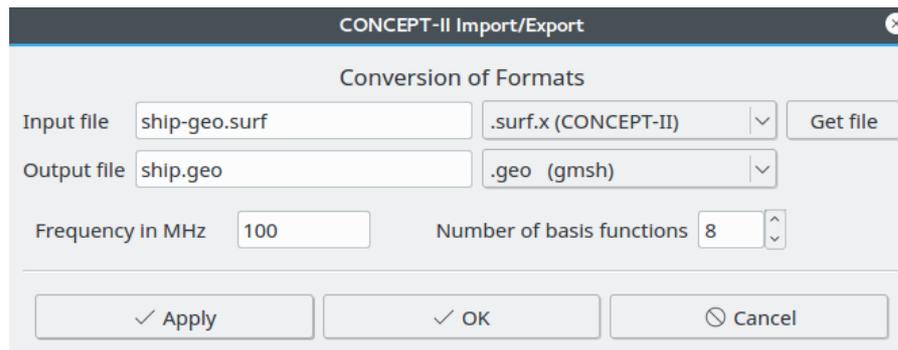


Fig 4.25: *CONCEPT-II*, patch conversion to the .geo format of *gmsh*. (**Misc.** → **Import/export**)

Continue with:

- *gmsh ship.geo* <CR> (or start *gmsh* and open *ship.geo*) →  
 ('+Geometry' → '+Elementary entities' → '+Add' → '+Plane surface' → all surfaces are indicated by crossed dashed lines →)  
 '+Mesh' → '2D'

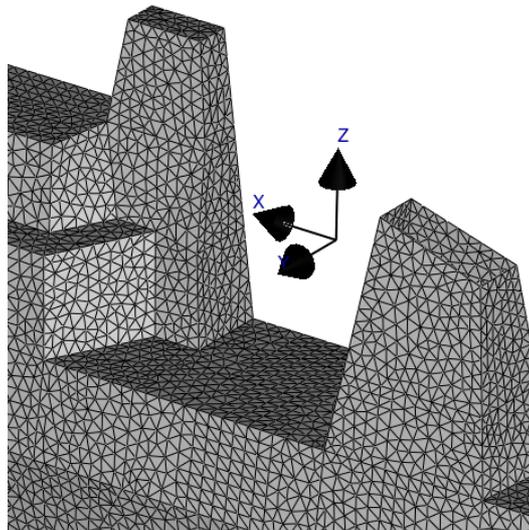


Fig 4.26: Triangular mesh for 100 MHz in the funnel region

- In *gmsh*) 'File' → 'Export' → *surf.msh* → 'OK' → Format 'Version 2 ASCII' or Format 'Version 4 ASCII' or simply 'Save Mesh'
- *Surf.msh* can be loaded directly into the *CONCEPT-II* simulation: **Simulation** → **Surfaces** → **Load surface file(s)** → 'Files of type:' **Gmsh files (\*.msh)** → *ship.msh*.
- A part of the triangulated model of the ship is depicted in Fig. 4.26.



At the the prompt:

`gmsh -2 ship.geo <CR> → ship.msh` is generated.

From the outlined example follows that it is easy the generate a mesh refinement for any higher frequency based on an arbitrary existing surface patch model.

The head section of the `.geo` file under discussion looks like this:

```
//Speed of light
vv00=299.7922517;
//Frequency in MHz
freq=100;
//Number of basis functions per wavelength
nzsw=8;
well=vv00/freq;
len=well/nzsw;
P1=newp;
Point(P1)={18.060499, -4.681880, 0.000000, len};
P2=newp;
Point(P2)={18.060499, 4.681880, 0.000000, len};
P3=newp;
.
.
.
```

As can be seen the characteristic length is computed according to the frequency (freq) and the number of basis functions per wavelength (nzsw = 8).

It has to be emphasized again that an arbitrary CONCEPT-II surface patch file can be discretized into a finer mesh using `gmsh`:

Convert the discretized surface with an arbitrary grid including triangles and quadrangles of the CONCEPT-II surface patch format into a `.geo` file. This can be accomplished by clicking on **Misc.** → **Import/export ...**  
Enter the desired frequency and the number of basis functions. The `.geo` file will be written. Continue at the command line:

```
> gmsh -2 filename.geo → the new grid is in filename.msh which can be
directly read by CONCEPT-II; filename-msh1.surf will appear. In case of
several 'Physical groups' (see input tree of gmsh) there will be
filename-msh1.surf, filename-msh2.surf,...
```

Note that when the entered frequency is too small it is not possible to generate a coarser grid! It is easy now to change the frequency (variable freq) to a higher value by an editor. Note that a new finer grid has no impact on the curvature of a surface, any new grid is based on the original flat



surfaces (i.e. patches).

### 4.11 Visualizing field and current distributions by means of gmsh

Field and current distributions which have been created in the post-processing stage can be displayed by *gmsh* (version 2.6 or newer, experimental stage, Jan. 16).

Start *gmsh*: **File** → **Open...** → vis\_cur.msh → display current distribution. Tool 

must have been finished.

Item **Post-processing** → entries **[1]**, **[2]**

**Tools** → **Options** → **View[0]** → *Intervals type* → **Logarithmic ...** → Color

**View[1]** → *Intervals type* → **Logarithmic ...**

Many options are available for professional looking graphics files. For details refer to the *gmsh* documentation.

(Currently only magnitude, Oct. 18) gmsh outfile.msh <CR> → 2D field computation (  ) must have been finished.

-----  
 [C. Geu-2009] C. Geuzaine and J.-F. Remacle. Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. International Journal for Numerical Methods in Engineering, Volume 79, Issue 11, pages 1309-1331, 2009  
 -----

## 5 Time domain system responses by an IFT

*CONCEPT-II* is based on frequency-domain integral equations (see Sec. 9). For time-domain quantities it is necessary to compute a series of sample frequencies in a frequency loop and to perform a subsequent inverse Fourier transform (IFT). Choosing the basic frequency, the highest frequency and the sampling rate in the right way is essential for a successful simulation. Hence some preliminary considerations are recommended as described in detail in Sec. 5.2.



## 5.1 The necessary steps in short

1. Discretize the structure in the usual way. Power generators are not

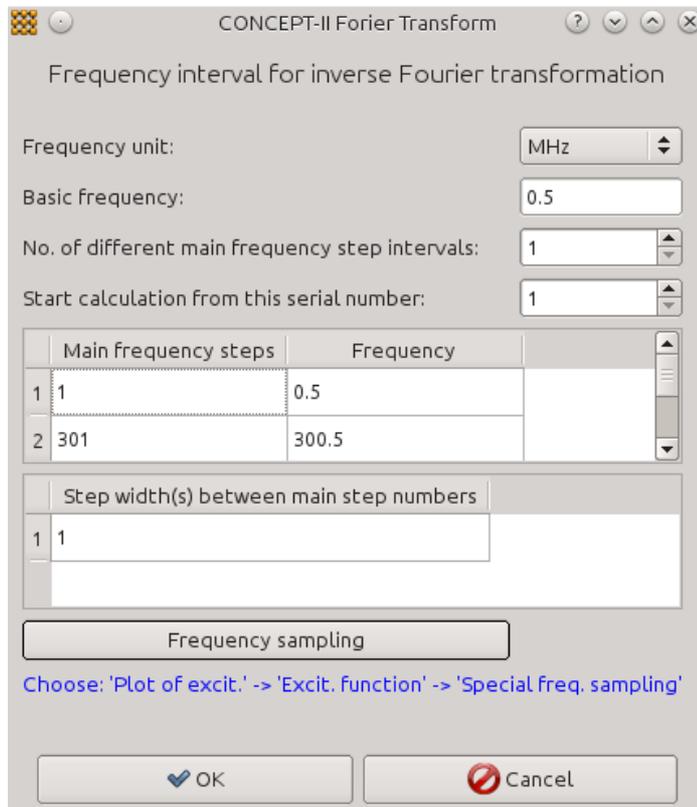


Fig 5.1:  
Frequency  
sampling for  
an IFT

allowed. The key point for a successful application of an IFT is the frequency sampling **Simulation** tab → **Frequencies** → **Frequency interval for inverse Fourier transform** → a window according to Fig 5.1 opens.

2. At the beginning set the following items to 1: *No of different main frequency step intervals, Start calculation from this serial number, Step width(s) between main step numbers*
3. All remaining input data have to be completed in a the standard way for the structure under consideration; then **Simulation** → **Run simulation**. Have in mind that a long computation time might happen as the matrix equation has to be set up and solved according to the chosen number of frequency samples.
4. Click on  (tab **Postprocessing**) on investigate the computed system

responses as a function of frequency. This is always recommended for validation. It is important that all system resonances are well approximated (always investigate the real part and the imaginary part). The displayed curves should look “smooth” and the value for the basic frequency should not



deviate too much from the static value for  $f = 0$ , which cannot be computed numerically by *CONCEPT-II*.

5. Click on  (time-domain tool) and select the type excitation function. Note

that there is a clear relationship between frequency sampling (including the basic and the limiting frequency) and the possible parameters of the excitation function, for example the rise time, the total signal duration etc.. After the IFT has been performed the time-dependent curves appear in the

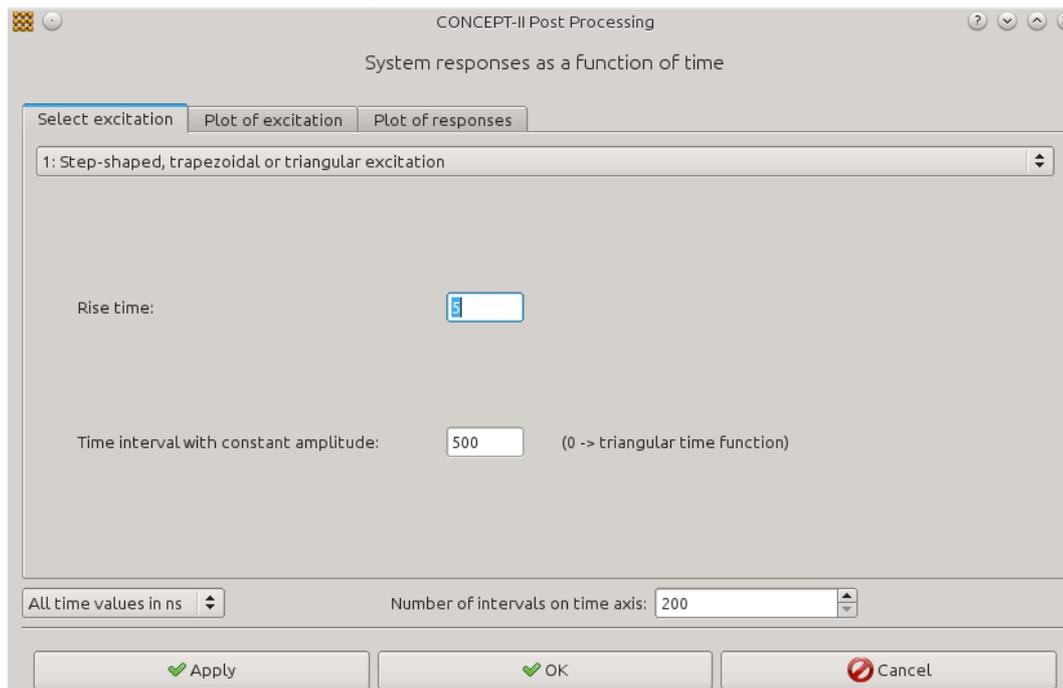


Fig 5.2: The Select excitation tab of the time-domain tool *Display area*.

Currently the following excitation functions are available (top combo box in Fig. 5.2):

- Step-shaped, trapezoidal or triangular functions
- Double-exponential function
- Arbitrary time function. When chosen, a special window opens, where the number of time steps, the time steps and the corresponding function values can be entered. Hence the time function is approximated by piecewise linear sections
- sine-square pulse
- Lightning current function according to Heidler [Hei]



Time values can be chosen in ns or in microseconds (in Fig 5.3 'All time

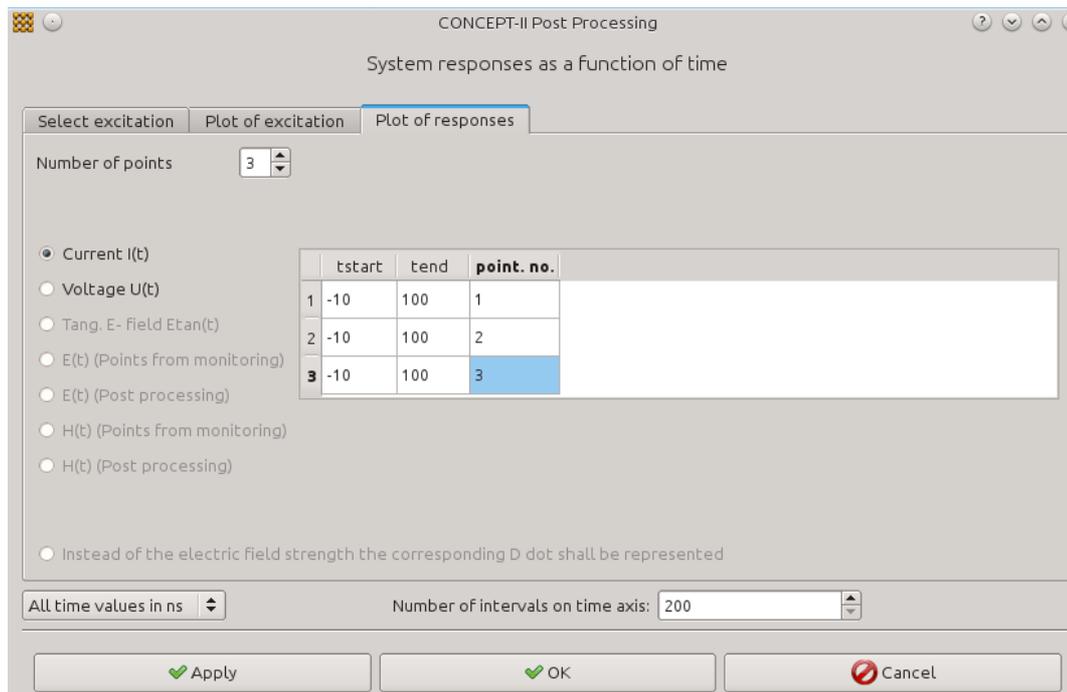


Fig 5.3: The **Plot of system responses** tab of the time domain tool

values in ns' has been selected). Under the tab **Plot of responses** (Fig. 5.3) is indicated which types of system responses are ready for an inverse Fourier transform (IFT). In Fig. 5.3 *Current I(t)* *Voltage u(t)* are indicated and the current at observation points 1 2 3 are to be plotted in the time interval ranging from -10 ns to 100 ns.

Generally the system response which are subjected to an IFT should be specified in the **Simulation Project tree** (Simulation tab) already, at least voltage observation points and current observation points, also field points. Additional field points can be specified later by using  (see **Post processing** tab)

Files: *time-e-n.asc*, *time-h-n.asc*, *time-u-n.asc*, *time-i-n.asc*, n of point (curve).

The example of a line excited by a voltage generator can be found under `$CONCEPT/examples/Transient_line`.

## 5.2 Key points for performing an IFT

The computation of time-domain system responses shall be demonstrated by means of an example which can be found under the directory `$CONCEPT/examples/Transient_line`

### Problem description:

A wire loop above an ideal ground shall be investigated.



- Height of the wire loop: 0.1 m
- Length of the loop (horizontal wire x-directed): 5.0 m
- Radius of all wires: 1.75 mm

A voltage generator with zero internal impedance is located at the base point of the left vertical wire, corresponding to the origin of the rectangular coordinate system. At the opposite end, i.e. at the base of the right vertical section a resistor of 75 Ohm has been placed.

The structure can be seen in Fig. 5.4 .



Fig 5.4: Single-wire transmission line over an ideal ground excited by a voltage jump

Our goal is to compute the time dependence of the current at the beginning, the midpoint, and the end of the horizontal wire.

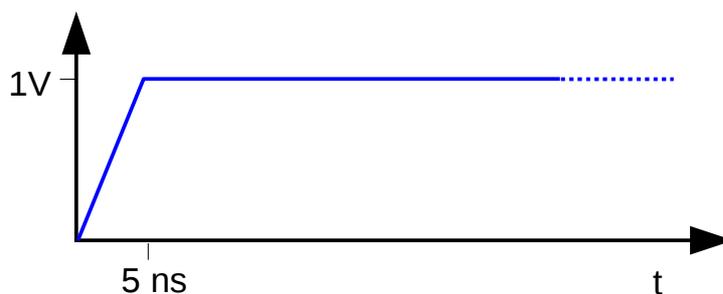


Fig 5.5: The excitation function

due to the excitation function provided by the generator (Fig. 5.5), which is a voltage step from 0 V to 1 V at a rise time of 5 ns.

### General remarks

The user should note that “frequency sampling” means automatically that the corresponding time-domain responses will be repeated periodically although the generator in the physical structure provides a non-periodic excitation. In other words: a periodic time-domain function cannot be avoided due to the principles and rules of the inverse Fourier transform.

Furthermore it is important to note that a frequency-domain MoM code cannot consider



extremely low frequencies;  $f = 0.0$  Hz is not possible.

It follows that the total time duration of transient quantities to be computed is limited. The lowest frequency that *CONCEPT-II* can handle is in the range of 1 kHz for structures with typical dimensions of 1 ... 10 m. The total duration of transients should not exceed 0.25 ... 0.5 ms. Transient quantities, exciting functions and system responses must always start at zero amplitude and end with zero amplitude.

The computation of time-domain system responses caused by pulses with very short rise times can only be accomplished if sufficiently high frequency components are included. At these high frequencies the dimension of the system under consideration may comprise already many wavelengths leading to large equation systems and long computation times. Remember that the solve time for the equation system is proportional to  $N^3$  with  $N$  as the number of unknowns and the LU decomposition is applied.

If currents tend to oscillate for a long time because energy can hardly be radiated or dissipated to heat in resistors it may get difficult to achieve satisfying results. Such systems, for example a single-wire transmission line close to the ground, terminated by a short-circuit or an open end, have sharp resonances. The "true" peaks of these resonances have to be captured resulting in a very short step width between the frequency samples. For each frequency step the system matrix has to be set up and the equation system has to be solved with the consequence of large computation times even in case of a moderate number of unknowns.

In *CONCEPT-II*, the frequency sampling is carried out according this scheme:

$f_0, 3 \cdot f_0, 5 \cdot f_0, 7 \cdot f_0, \dots, f_g$  with  $f_0$  as the basic frequency and  $f_g$  as the highest frequency.

In order to model a generator function as shown in Fig. 5.5 the transmission line has to be excited by a pulse as can be seen in Fig. 5.6.

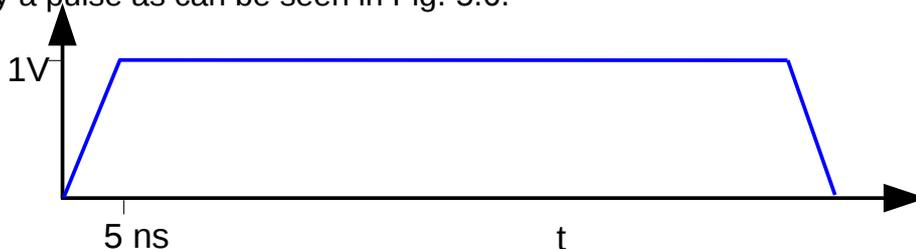


Fig 5.6: An excitation function must start with zero amplitude and end with zero amplitude

**Important: Sampling in the frequency domain causes a periodic continuation of the signal in the time domain.** Thus, due to the sampling according to mentioned scheme the excitation function which is **numerically** applied to the transmission line looks like the pink curve in Fig. 5.7. In this picture the basic frequency is  $f_0 = 0.5\text{MHz}$  corresponding to a cycle duration of  $2\ \mu\text{s}$ , starting at  $T_0=0$  and ending at  $T_p=2\ \mu\text{s}$ .



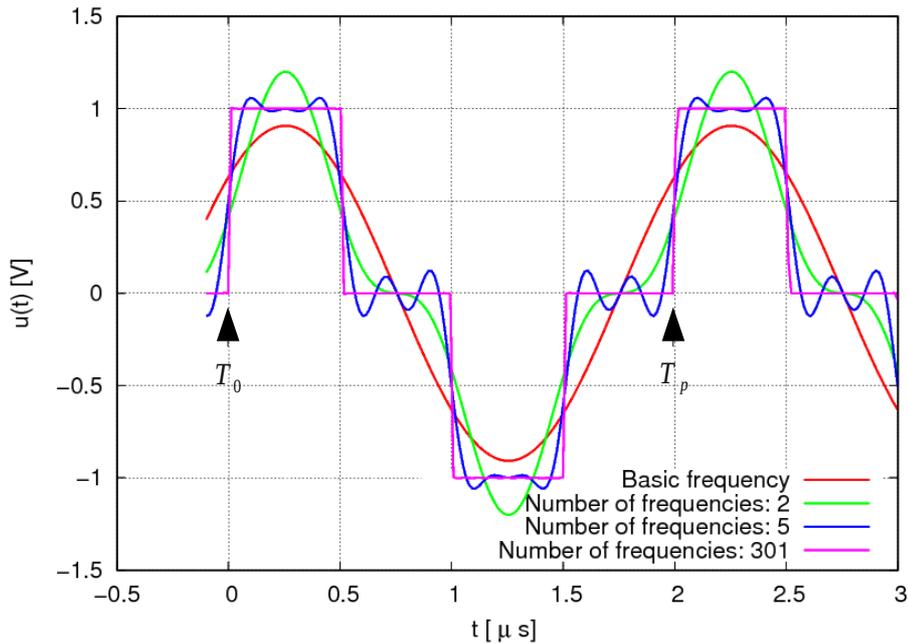


Fig 5.7: Forming of a time-domain signal  $u(t)$  by the superposition of sine functions. Pink curve: time dependence of generator voltage due to frequency sampling. The first frequency sample is  $f_0=0.5$  MHz

**We see that our transmission line is actually excited four times**, namely at  $t_1=0 \mu s$ ,  $t_2=0.5\mu s$ ,  $t_3=1\mu s$  and at  $t_4=1.5 \mu s$ . Rise and fall times of the pink curve are 5 ns. Due to the frequency sampling according to  $f_0, 3 \cdot f_0, 5 \cdot f_0, 7 \cdot f_0, \dots, f_g$  the generator voltage is a time-periodic bipolar signal.

At  $t=2 \mu s$  the voltage of the generator starts again. The transient voltages and currents which are caused by the last voltage jump from -1 V to 0 V at 1.5  $\mu s$  should be damped to negligible amplitudes, otherwise aliasing will occur.

In order to get the **step response** of the system, the stationary state ("direct current") must be reached at  $t_1=0.5 \mu s$ , because at that instant of time the generator voltage begins to drop again.

From the overall length of the loop (5 m + 0.1 m + 0.1 m) we find the propagation time as follows:

$$t_{\text{travel}} = 5.2\text{m}/3.0\text{e}8 \text{ s} = 5.2\text{m}/c = 17.33 \text{ ns} .$$

The reflection factor is given by

$$r = \frac{R_2 - \Gamma}{R_2 + \Gamma} = \frac{(75 \Omega - 284.32 \Omega)}{(75 \Omega + 284.32 \Omega)} = -0.58 \quad \text{with } R_2 \text{ as the terminating resistor and } \Gamma \text{ as the}$$

characteristic impedance of the horizontal wire which can be calculated by the the well-known formula

$$\Gamma = 60.0 \cdot \ln(2h/r_w) ,$$

$h$ : height of the horizontal wire,  
 $r_w$ : radius of the horizontal wire.

From the magnitude of the reflection factor one can expect that current and voltage waves will travel back and forth for several times until the steady current of

$I = 1 \text{ V} / 75 \text{ Ohm} = 13.333 \text{ mA}$  is reached.

We guess that 20 travel times are necessary to get the final current value. At  $t_1, t_2, t_3$  quite similar transients will be excited resulting in

$T_p = 4 * 20 * 17.33 \text{ ns} = 1.36 \text{ } \mu\text{s}$  ( $T_p$ : periodic time of the numerical exciting time function, we have chosen  $T_p$  as  $2 \text{ } \mu\text{s}$ , pink curve, basic frequency  $f_0$ , see Fig. 5.7)

Note: In the general case it may be difficult to predict  $T_p$  as one does not know the system behavior in advance. Low-loss system which easily tend to oscillate (example short-circuited line, open-ended line) require a very narrow frequency sampling. In such a case we will have sharp resonances in the frequency domain and long system responses in the time domain. As a rule of thumb one can say that transmission-line type structures (waveguide-type structures) require a very fine frequency sampling if they are open-ended and/or short-circuited. In such cases energy can hardly be dissipated because radiation is negligible.

It is important to recognize that  $T_p$  includes the duration of the excitation function **and** the duration of the system responses, both have to reach their final values at  $T_p$ . In the case of lightning induced electromagnetic quantities on vehicles (ships, aircrafts) the excitation function (impressed current) may have a dominant influence, i.e., traveling waves which might have been induced on the system are already damped to zero, while the exciting impressed current continues to flow.

$T_p = 2 \cdot \text{total duration of excitation function}$  (Lightning impact on vehicles)

On the other hand in the case of a sharp “needle-like” pulse excitation a structure will essentially be excited two times leading to  $T_p = 2 \cdot \text{No. of travel times} \cdot \text{travel time}$

In general  $T_p$  depends on the following items:

- Typical geometrical extension of the configuration. Large structure cause long travel times
- Duration of the excitation function (may especially get dominant for lightning simulations)
- Type of the structure, antenna-type, transmission-line type. Lines can hardly radiate energy, the forming of resonance peaks are mainly caused by losses in the terminating resistors.

Antennas radiate electromagnetic energy, therefore they not as critical as low-loss



lines with respect to the frequency sampling. We usually get first resonances which are relatively broad although ideal conducting metal is assumed and no resistors exist

- Losses due to resistors or finite conductivity
- Other kinds of loads, for example inductors and/or capacitors, time constants  $L/R$ ,  $R \cdot C$  due to lumped loads.

In the example under discussion we have  $T_p = 1.36$  microsecond which leads to the basic frequency of

$$f_0 = \frac{1}{T_p} = 0.74 \text{ MHz}$$

We have choose  $f_0 = 0.5$  MHz, allowing even more travel times along the loop. So frequency sampling starts at 0.5 MHz with a step width of 1 MHz.

Known from experience, good results in the time domain can be achieved by means of

$$T_{\text{rise}} \cdot f_g = 0.8 \cdots 1.5 \quad (2)$$

with  $T_{\text{rise}}$  as the rise time in microseconds and  $f_g$  as the highest frequency in MHz.

From the last equation one finds

$$f_g = 1.5 / 0.005 = 300 \text{ MHz} \quad (\text{chosen rise time is } 5 \text{ ns})$$

Hence the number of frequency steps is around  $300 \text{ MHz} / 1 \text{ MHz} = 300$ . The equation system has to be set up and solved 300 times by the *CONCEPT-II* back end.

-----  
 Note: If high-quality system responses shall be achieved it is recommended to set the discretization of all wires according to the requirements of the highest frequency:

$$N_{\text{wire}} = \frac{L_{\text{wire}}}{\lambda_{\text{min}}} \cdot 8$$

with

$N_w$  : number of unknowns (segments)

$\lambda_{\text{min}}$  : wavelength at the highest frequency

8: minimum number of unknowns (basis functions) per wavelength.

The drawback of such a proceeding is a relatively high computation time if there are dominating wire parts in the structure under consideration.



If the propagation of current waves along long wires shall be displayed (  , **Results Section**), the number of basis functions along that wire is not allowed to change.

---

Fig. 5.8 shows the frequency dependence of the computed wire current at the left vertical

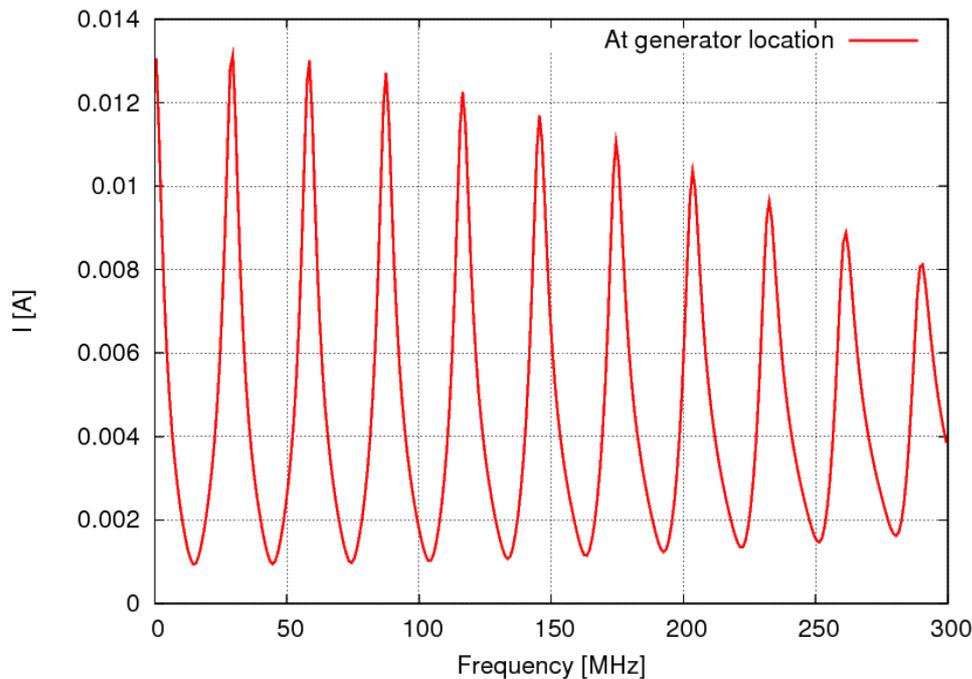


Fig 5.8: Magnitude of the wire current at the left vertical wire as a function of frequency

wire of the loop (  , **Results section**). Two important features have to be emphasized, which can be observed:

1. At  $f_0$  (first frequency step) we have already a very good approximation of the static current which is  $1V/75 \text{ Ohm} = 13.3 \text{ mA}$ . Therefore the time domain signal is supposed to have the right information once all propagation waves are damped to zero.

General rule:  $f_0$  should be such small that all physical quantities are already very close to their static values.

2. All resonance peaks are arranged “in a chain”. The envelope of all resonances appears as a smooth function which should always be the case. This indicates that the resonance peak values have been computed properly.

General rule: Carefully observe if the true peak values of resonances are contained in the system responses.

This requirement might be difficult to meet in the case of highly resonant structures.



It is always recommended to check the frequency domain system responses before performing an inverse Fourier transform. Time domain signals can be computed by means of time-domain tool: **Post processing** tab → 

Mathematically we have to carry out an inverse Fourier transform of  $Y(\omega)$

$$Y(\omega) = H(\omega) \cdot X(\omega)$$

with  $H(\omega)$  as the transfer function computed by *CONCEPT-II*, based on an excitation with constant amplitude, and with  $X(\omega)$  as the spectrum of the excitation function. By means of the **Select excitation** tab the user can specify various time-domain excitation functions (even arbitrary excitation functions) in order to perform the outlined operation.

In our example we select **Step-shaped trapezoidal or triangular function** with a *Rise time* of 5 ns and a *Time interval with constant amplitude* of 500 ns. The time domain system response(s) shall be sampled by 200 time steps (*Number of intervals on time axis*). Especially complicated signals of long durations might require a much larger number of time steps.

Next the input positions under the **plot of responses** tab have to be completed (Fig. 5.9). We observe that data for the Current  $I(t)$  and Voltage  $U(t)$  and  $E(t)$  is available and activate the radio button for current plotting. The  $E(t)$  has the addition (*Post processing*), meaning the observation points have been specified and computed under the Post processing tab with the fields tool  and were not set in the front end (**Simulation** tab). Here the

currents at the beginning, middle and end of wire 2 (horizontal) shall be plotted, each beginning at  $T_{start} = -10\text{ns}$  and at  $T_{end} = 250\text{ns}$  with a resolution of 200 time steps.



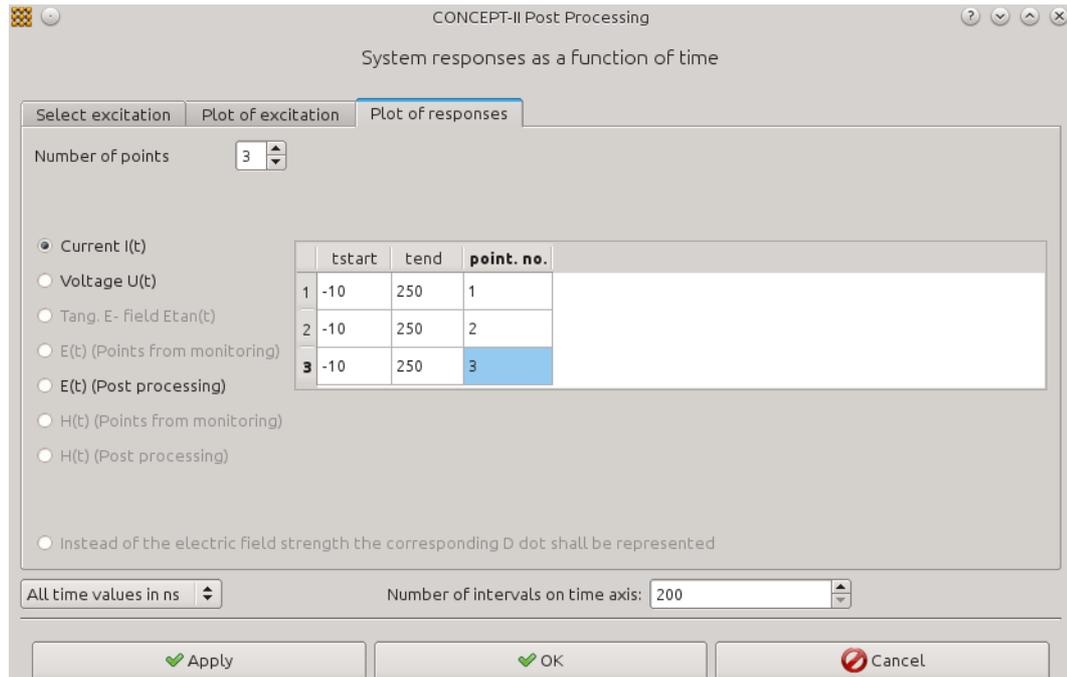


Fig 5.9: The 'Plot of responses' card of the inverse Fourier transform tool

Clicking on **Apply** provides the transient currents, see Fig. 5.10.

Useful hints: It is always a good idea to start the curve representations at neg. times, i.e. before the generator voltage begins to rise.

Here the time axis starts at -10 ns. Therefore one can see that the current at the generator position begins to rise exactly at 0 ns. Furthermore all depicted currents have zero magnitude before 0 ns. This would not be the case if the terminating resistor had a much smaller value or a much higher value (much higher than the characteristic impedance of the loop) and if the frequency sampling remained unchanged. Under such circumstances aliasing would occur. The user is invited to investigate this.



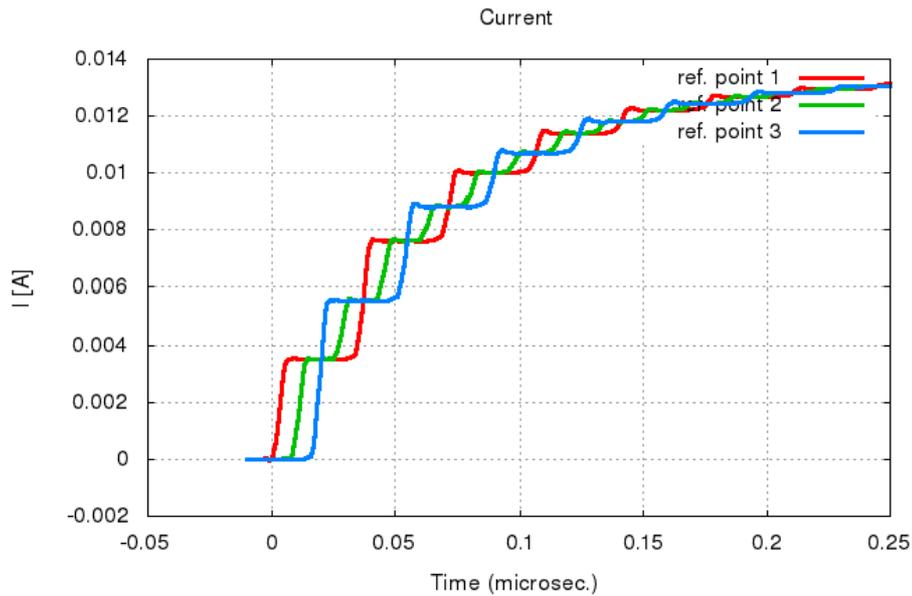


Fig 5.10: Computed wire currents as a function of time. Note the static values and time shifts of the functions according to the position of the observations points along the line.

A much shorter rise time would lead to oscillations with significant amplitudes especially before  $t=0$ . This is due to numerical reasons (bandwidth limitation) and is known as "Gibbs phenomenon". Some small amplitude oscillations can be observed in the current curves by looking more closely before  $t = 0$ .

Validating the results we find:

- All currents start from zero amplitude.
- All curves start according to the corresponding physical travel times.
- The final value of all current is identical. This is  $I = 1V/75 \text{ Ohm} = 13.33 \text{ mA}$ .
- The current at position 1 starts with  $1 \text{ V} / \text{Gamma} = 1V / 284.32 \text{ Ohm} = 3.52 \text{ mA}$



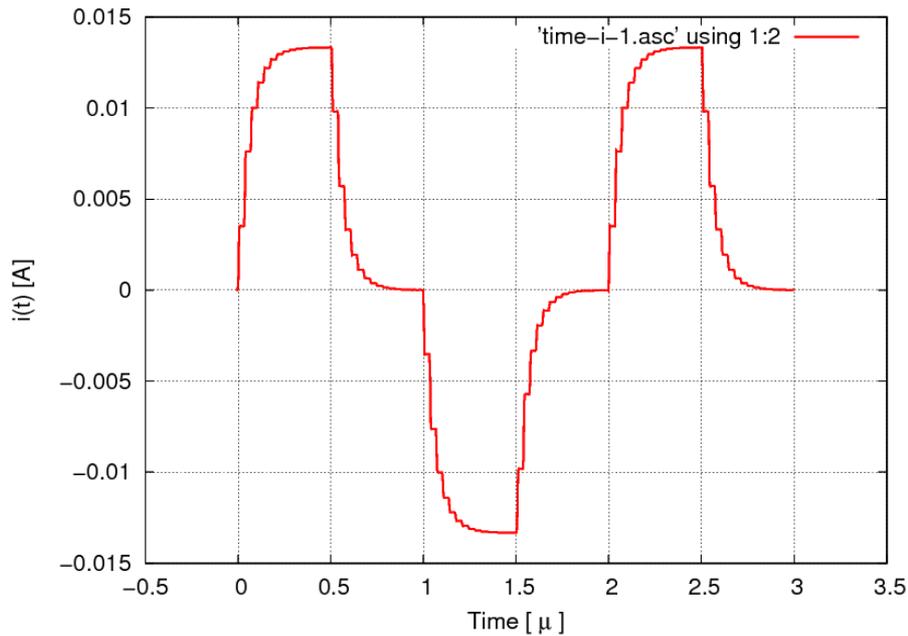


Fig 5.11: Computed wire current (left vertical wire) as a function of time, time interval extended

In Fig. 5.11 the time axis has been extended to  $3\mu\text{s}$ . It can be observed that the wire loop is actually excited four times during  $T_p = 2\mu\text{s}$  according to theory. Each time the transient current decays completely before the next jump of the generator voltage happens.

Note:

- The time  $t = 0$  corresponds to the feed point in the case of a generator excitation or the starting point of an impressed traveling current pulse.
- For a plane wave excitation  $t = 0$  always refers to the origin of the coordinate system.
- Power input is not allowed if transient system responses shall be calculated.

### 5.3 Frequency sampling for an IFT

The frequency sampling for the structure under investigation can be specified in the entry fields of the window shown in Fig 5.1.

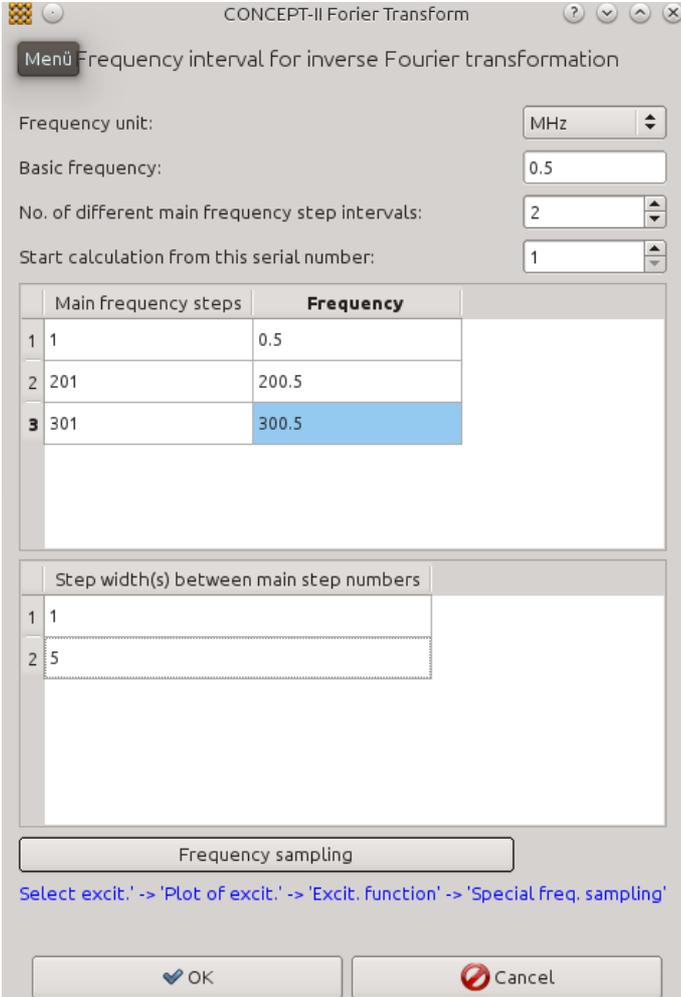
In the case according to Fig 5.1 each frequency step between step 1 and step 301 shall be considered by a full MoM computation, starting at 0.5 MHz (step width 1 MHz, according to the sampling scheme  $f_0, 3 \cdot f_0, 5 \cdot f_0, 7 \cdot f_0, \dots, f_g$ , see last section) and ending at step 301, corresponding to 300.5 MHz. “1” and “301” are called main step numbers or *Main frequency steps*. These numbers have to be entered, the corresponding frequency value will immediately be updated in the neighboring *Frequency* column.



If a *Step widths between main step numbers* were chosen to be 10 instead of 1, then the following sampling scheme between step 1 and step 301 would result: 1,11,21,31,...,301. Thus in such a case only every 10<sup>th</sup> step would be computed based on a full MoM analysis. The missing steps are automatically interpolated later when using  and , respectively.

In case of a suitable behavior of the system responses as a function of frequency we could have an input situation as depicted in Fig. 5.12.

Now the interval between 0.5 MHz and 200.5 MHz is sampled with a step width of



|   | Main frequency steps | Frequency |
|---|----------------------|-----------|
| 1 | 1                    | 0.5       |
| 2 | 201                  | 200.5     |
| 3 | 301                  | 300.5     |

|   | Step width(s) between main step numbers |
|---|---|
| 1 | 1                                       |
| 2 | 5                                       |

Fig 5.12: Frequency sampling for an IFT using different step widths

$2 \cdot 0.5 \text{ MHz} = 1 \text{ MHz}$ . In the remaining region up to 300.5 MHz only every 5<sup>th</sup> step is explicitly taken into account, saving a lot of computation time.

If for example the *Start calculation from this serial number* were set to 2, the

computation would start at step  $206=201+5$ . Precondition would be that the results between the *Main frequency steps* 1 and 201 have already been computed successfully. Such a situation could happen for example for very time-consuming problems where the first run revealed a rather smooth behavior of system responses, so that a rougher sampling seems to be justified for the next run. Another imaginable case: time domain quantities are insufficient due to an excessive appearance of Gibb's phenomenon, hence higher frequencies are to be included. So there is no need to start the whole computation starting at  $f_0$  again.

**Important:** The difference of two successive main frequency steps must be divisible by the corresponding step width without remainder.

Clicking on **Frequency sampling** offers the possibility to investigate if the intended sampling scheme is sufficient for creating the excitation function before starting the back end run. Proceed as follows:

- The "System responses as a function of time" window opens. Click the **Select excitation** tab and choose the desired excitation function. Enter the required values (time constants, rise time etc.)
- Click the **Plot of excitation** tab
- Activate *Excitation function* and enter the start time and the end time for the curve representation into the corresponding input fields.
- Activate *Special frequency sampling* and enter the No. of frequency steps and the step width. If no values are entered the sampling scheme as specified under the *Freq. interval for inverse Fourier transform (Simulation Project tree)* is taken. Note that this can only be successful once the front end run has finished already! Only under this precondition the data is known to the tool.

#### 5.4 Special aspects: Impressed currents and lightning currents

Wire currents with known amplitudes can be taken as a structure excitation. Such currents which are developed in basis functions are related to the voltage vector only (right-hand side of the equation system, Sec. 9). Each basis function with known amplitude has impact on structure parts which are considered by the MoM equation system, in other words such a radiating basis function causes a voltage over each pair of neighboring patches and wire segments respectively - they induce currents. These induced structure currents have no impact back onto the impressed currents. Hence the known currents are 'impressed' ones.

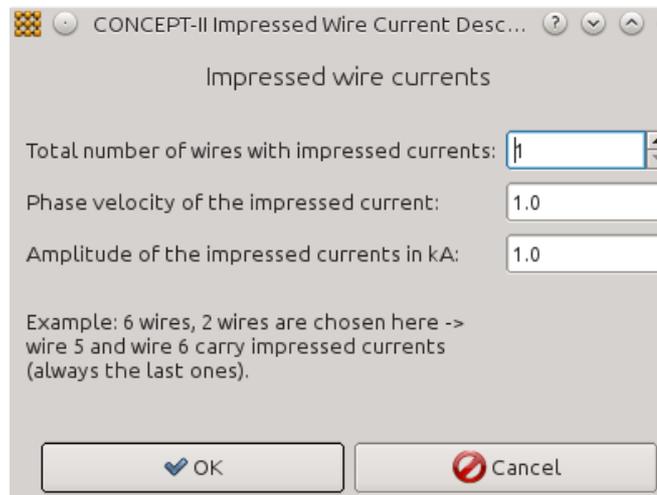
A short section of a wire with a known constant current may be used as a current source. Spatially long wires (also electrically long wires) can be used to model the stroke of a lightning, either as a nearby stroke or as a direct stroke.

The electromagnetic fields of lightning return strokes can be modeled based on the transmission line model originally proposed by Uman [Uman]. The channel is taken into account by wires, which are carrying an impressed, i.e. known current distribution. Wires with impressed current distributions can be specified under the



**Simulation** *Project tree*, item **Excitation** → **Impressed wire current**

The following window opens:



- *Total number of wires with impressed current*

It is important to have in mind that this number always refers to the last wire(s) which have been entered under the **Wires (Simulation Project tree)**.

Only a single wire file is possible in such a case.

- *Phase velocity of the impressed current*

The propagating speed of the time-domain current function.

0 → All basis functions have a constant amplitude all over the wire(s) with an impressed current distribution

1 → Propagation with speed of light (0.333 is frequently taken for lightning simulations)

- *Amplitude of impressed current in kA*

For lightning modeling purposes 0.001 kA is good choice in case of the function according to Heidler [Heidler]. The amplitude of the propagating exciting function can be specified later in the IFT tool  which performs the inverse Fourier transform (**Post processing** tab)

The lightning function according to Heidler: 
$$i = \frac{i_{\max}}{\eta} \frac{\left(\frac{t}{T}\right)^{10}}{1 + \left(\frac{t}{T}\right)^{10}} e^{(-\frac{t}{\tau})}$$

Parameters: Front time T, decay time  $\tau$ , correction coefficient  $\eta$ , see input



window of the IFT tool .

In case of a double-exponential function being used to simulate the event of a lightning the amplitude should be set in the front end run already. This value is related to the magnitude of the sinusoidal channel current distribution in the frequency domain. In the time domain we have this value in front of the exponential terms. This is not the true current peak amplitude of the time-domain function, which also depends on the chosen time constants. The time function that travels along the channel looks like this:

$$i(l, t) = i_{\max} \cdot \left( e^{-\frac{1}{T_1} \cdot (t-l/v)} - e^{-\frac{1}{T_2} \cdot (t-l/v)} \right)$$

with  $l$  as the running coordinate along the channel and  $v$  as the speed of propagation.

- The time domain impulse propagates along the wire, starting at the beginning. If this point is attached to a wire or to the node of the surface patch of the affected structure this marks the point of strike. The type of excitation function can be chosen once the back end has finished ()
- A curved channel in space can be modeled by straight sections where the end of each section is connected to the beginning of the next one. Branches are not allowed and only a single amplitude can be entered. In case of the channel being attached to the structure the continuity equation is fulfilled.
- Modeling of an ideal current source: A short piece of wire with an impressed current is attached to a structure being above a ground with ground connection or in the course of a closed loop.

The presence of a completely closed current loop including the current return path is not necessary when using an impressed current distribution as excitation: the corresponding wire can stop or begin at a certain position in space.

Simulating a lightning return stroke the channel length between the strike point and the cloud should be long enough to sufficiently include the effect of radiating EM fields (good results have been gained for channel length between 500m to 1km). This leads to many functions for a large channel lengths. Each current functions along the channel has an impact on all wire segments and surface patches of the MoM parts of the structure. Thus the computation of the right-hand side of the equation system gets time-consuming.

- Non-linear effect are not regarded in the computations.

According to Fig. 5.13 four different lightning functions are predefined for excitation.



Clicking on **Sampling values** a suggestion for the frequency domain sampling is given under the **Plot of excitation** card. Clicking on **Apply** provides the corresponding curve on the screen. Note that *Step width* corresponds to  $2 \cdot f_0$ .

A good value for the number of time steps (*Number of intervals on the time axis*) for the time-domain curve representation is 200 (default). For complicated signals with superimposed oscillations and a long duration 600...800 intervals (steps) are suggested.

An example for aligning simulation can be found under [\\$CONCEPT/examples/Lightning\\_overhead\\_line](#)

Fig 5.13: Input parameters according the lightning current defined in the international standard IEC 61312-1

Literature:

[Uman] M.A. Uman, D.K. McLain, "Magnetic Field of Lightning return Stroke", Journal of Geophysical Research, vol. 74, Dec. 1969, pp. 6899-6910

[Heidler] F. Heidler, J. Cvetic, "A Class of Analytical Functions to Study the lightning Effects Associated With the Current Front, ETEP Vol. 12 March/April 2002, pp 141-149

## 6 Frequently asked questions

### 1) How can I easily clean the working directory?

**Simulation** (menu bar) → **Clean simulation directory...** → **Delete all files**

**Simulation** (menu bar) → **Clean simulation directory...** → **Delete result and temporary files** → only the simulation control file and data files which are needed



to restart the simulation will not be erased.

## 2) The mesh lines of surfaces appear too small on my screen...

**Options** (menu bar) → **Line width mesh...** → set the width

## 3) I want to create a picture of the display area, how can I do this?

**File** (menu bar) → **Save image as** → a numerous formats are possible

## 4) How to quickly change loads or generators

Put the mouse pointer onto the symbol in the display area and right click → a dialogue window immediately opens with the corresponding entry being highlighted.

## 5) How to load a .msh file, grid from gmsh?

**Simulation** → **Surfaces** → **Load surface files** → Filter: **Gmsh files** (suffix .msh)

## 6) A wire is not directly ending on a patch node – how can I fix this quickly?

**Cad tools** → **Geometry from simulation** → Cad tools 1 → activate '*Move node*' → select nearest patch node by right click → right click on wire node. If necessary refine mesh before dragging a node. If necessary deactivate '*Regard wires*'...

## 7) I have changed the structure and want to run the simulation again. How to quickly save curve data?

The curve is displayed → right click into the display area → a window pops up asking for the new file name. To compare curves start the gnuplot front end  and

load the curve data.

## 8) Where to find the build number?

**Help** → **About...**

## 9) When to apply the 'Single precision' option (Simulation project tree)

Can be used only in conjunction with the LU decomposition solver and should only be applied in case of lack of memory (RAM). The system matrix is stored and solved in single precision mode, no advantage in computation time. Use only for freely radiating structures which are not too small electrically, notice Section 1.11.2.8

## 10) Wires are not OK, they are marked red or pink...

The thin-wire assumptions are hurt. Visualize by

**Options** (menu bar) → **Draw wires as ...** → **Cylinders**

Click on  (tool bar) and notice the ratio of radius/(segment length) or radius/

(edge length of attached patch). Ticks not visible: **Options** (menu bar) → **Settings for wires in cylinder mode...** →

## 11) I always get a warning or an error message with respect to wire radii that are too large or about wires that are running too close to each other (or



something similar). I want the simulation to be completed and I am quite sure that the reported deficiency has almost no impact on the result. What can I do?

Tab **Simulation** → Open **Control options** → Set appropriate menu item to active. Notice the tooltips.

12) I have a plot in the display area, how to change the font?

**Post options** (menu bar) → **2D view** → **select font name** etc → restart the tool

For curves in the display area is also possible:

Right click on the corresponding entry in the *View result* tree → the Gnuplot front end opens → **Get data files(s)**, if data is not yet loaded → **Settings** → “Font size”

13) I see the magnitude of some system responses (for example input impedance as a function of frequency, Post processing tool ) in the display area. I want to improve or enhance the graphics, respectively!

Double click on  **Freq.-domain responses** (“View results section”, tab **Post**

**processing**) → **Show results** → The gnuplot front end opens allowing for numerous improvements and changes: additional ASCII data of curves can be loaded, text size can be changed, real part: “Using “ 1:2, imaginary part: “Using” 1:3, mouse pointer on “Using” input field → Tool tip on file contents. **Options...**, “Output to” .png. .SVG...,

14) I have already computed system responses (for example radiation pattern, field distribution). I want to show the results again. Do I need to compute them again?

This is not necessary: Double click on the corresponding item of the “View results” section (Tab **Post processing**) → Show results → ...

15) I have the error message ‘...Body 1 contains open surfaces...’. How to fix this?

-In case of symmetry – forgotten to specify the symmetry plane(s)?

-Bodies not properly specified?

-Maybe that there is a grid error, carefully inspect the mesh... **Cad tools** → **Cad**

**tools2** → click button  → set ‘Find free nodes, (each loaded ...)’ to active.

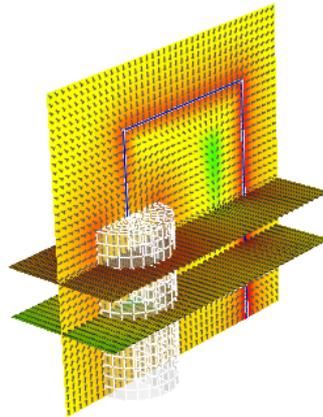
Boundaries of surfaces are indicated in the file *wire.freenode* (wire format, one wire per free edge of a patch). If there are “other wires” shown these are marking grid errors, i.e. the grid is not closed as it should be. Eventually set surfaces to invisible, i.e. disable check marks for surfaces to better display the situation...



Eventually enhance the wire visibility by **Options** → **Settings for wires in line mode...**

Repair the grid by  Move node etc.

## 16) How can I display multiple field distributions at the same time?



Start the field computation tool  several times according to the desired number

of field plots. Change the '*Name of result file*' (default for E: *E-Field-surf\_eh2d.h5* in case of a 2D rectangular region) to a different name, at least starting with the second run. For plotting change the name of the '*Extracted structure file*' (default is *surf.eh2d* in case of a rectangular 2D region) to a different name. Do not unload '*Models already loaded*'. For color or arrow scaling pay attention to specify the corresponding surface file (structure file) under the scale widget on the right side of the display area.

For plotting of existing data: double click on  Surface, point(s) → **Show results ...**

Pay attention to choose different structure file names and that all data is referring to the same number of phases in case of a movie for example. Do not unload previously loaded field distributions.

## 7 Using *CONCEPT-II* for parallel computation

### 7.1 Linux

All machines forming a cluster (distributed memory systems) should have the same operating system, the same size of memory and the same type of CPU.

Note that for configuring a cluster a certain knowledge of the Linux operating system is required! **The cluster will not be configured by the *CONCEPT-II* team.**

Each of the participating computers must be able to connect to any other computer without entering a password.



Format of the machine file which is required on distributed memory systems (MPI).

Node\_name\_1 slots = **Number\_of\_CPU's**

Node\_name\_2 slots = **Number\_of\_CPU's**

Determine the problem size:  $N \cdot N \cdot 16/1024^3$  GByte are needed (double precision), where  $N$  is the number of unknowns. Note that parallel out-of-core solutions are not supported by the *CONCEPT-II* parallel back-end.

Commands at the prompt for distributed PCs, only MPI

```
mpirun -np n -machinefile name_of_machine_file concept.be
```

Command on a single multicore PC, only MPI

```
mpirun -np n concept.be
```

“n” is the number of processors that are available

Use *concept.sbe* for computations in single precision, only for solver LU decomposition

## 7.2 Windows

Currently (Nov. 15) the parallel solvers have only been tested on shared memory machines.

For further details see the installation instructions.

## 8 Format of surface patch files and wire files, special topics

Surface patch files have the following format (ASCII data):

Line 1: Number of nodes  $N_{node}$ , number of surface patches  $N_{element}$

Line 2, coordinates of node 1: x y z

Line 3, coordinates of node 2: x y z

...

Line  $N_{node}+1$  x y z

Line  $N_{node}+2$  node numbers of patch 1

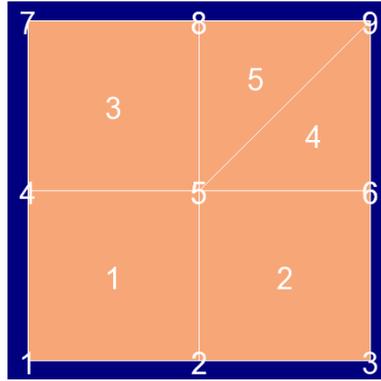
Line  $N_{node}+3$  node numbers of patch 2

...

Line  $N_{node}+1+N_{element}$  node numbers of the last patch

Example: square plate in the x-y plane, edge length: 1 m.





Contents of the surface patch file:

```

9 5 (9 nodes, 5 patches)
0.0 0.0 0.0 ( first node)
0.5 0.0 0.0 (second node)
1.0 0.0 0.0 (third node)
0.0 0.5 0.0
0.5 0.5 0.0
1.0 0.5 0.0
0.0 1.0 0.0
0.5 1.0 0.0
1.0 1.0 0.0 (node number 9)
1 2 5 4 (description of the first patch, see above figure, quadrangle, 4 nodes)
2 3 6 5 (description of the second patch, quadrangle, 4 nodes)
4 5 8 7
5 6 9 0 (triangle, last node number is a "0", patch number 4)
5 9 8 0 (triangle, patch number 5, also triangle)

```

The wire format looks like this (R. Radius, NB: number of basis functions):

```

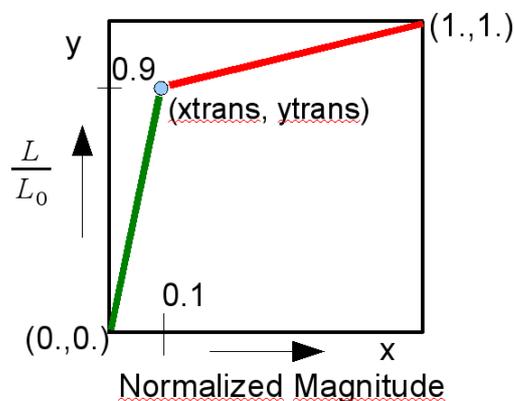
1st line:   number of wires N
2nd line:   Xstart Ystart Zstart Xend Yend Zend R NB → 1st wire
3rd line:   Xstart Ystart Zstart Xend Yend Zend R NB → 2nd wire
4th line:   Xstart Ystart Zstart Xend Yend Zend R NB
...
N+1th line Xstart Ystart Zstart Xend Yend Zend R NB → coordinates Nth wire

```



## 8.1 Arrow scaling, xtrans, ytrans

For the representation of vector diagrams the arrows have to be scaled in an appropriate way. In practice it is sometimes advantageous to display the vector in regions of very small amplitudes in a linear scale. This can be carried out with respect to the surface patch current distribution (  ) or 2D representations of the electric field strength, see  for example. The scaling method is based on a unit square defined in an x-y coordinate system as shown in the next figure.



The standard values of xtrans, ytrans are (0.1, 0.9), see above figure. This means that 90% of the default maximum arrow length  $L_0$  is linearly mapped to all values of the depicted quantity between 0 and 10% of maximum value of this quantity. These arrows are shown in green color. All remaining arrows are allowed to vary between 90% and 100% of  $L_0$  and are shown in red.  $L_0$  depends on the grid size and the average surface patch size, respectively. For a good representation it is recommended not to have more than 10% of all arrows in red.

Setting xtrans to a lower value, for example 0.01, means that regions with very small current or field vectors are magnified, so that their behavior can be observed. As no log. scale is applied one gets a good “feeling” of the physical situation.

Setting both xtrans and ytrans to 1 causes a linear scaling of all arrows – no red arrows will appear.

Setting ytrans to 1 (and leaving xtrans for example at 0.1) causes all arrows larger than 10% of the maximum field or current vector to be drawn in yellow. All yellow arrows have the default maximum arrow length  $L_0$ .



## 8.2 Backup of project directories

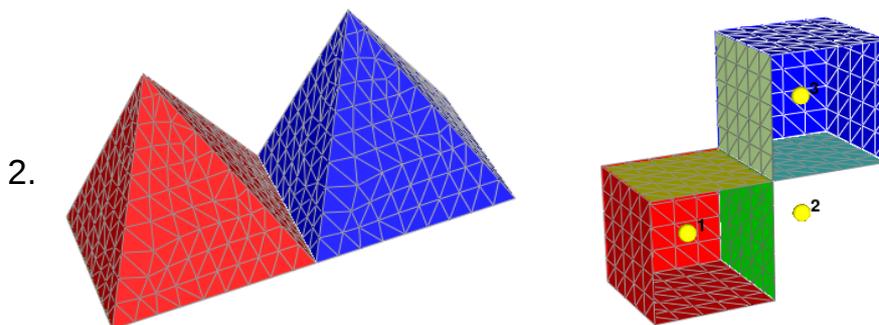
Not all files need to be saved. Several levels can be distinguished:

1. Backup with the lowest storage requirements: Save *project.sim* and all wire and surface files that are named in *project.sim* and *confreq.in*  
Convenient way to clean a working directory:  
**Simulation** → **Clean simulation director...** → all files except for the mentioned ones are erased. Disable the check mark if further files are not to be erased.
2. Add computed currents and fields: *co\_ifl(n).bin*, *co\_ili(n).bin*. In order to be able to use this data, it is necessary to perform the following steps: Load the simulation, then click on the **Simulation** button in the Menu bar → **Run Simulation front end**. The Post processing tools are ready to be used now.
3. Save the whole project directory.

## 8.3 Known problems

Here is a list of currently known bugs and problems (Jan. 2016):

1. Only Windows: If a parallel computation has been killed the files are obviously not free for some time. Hence do not restart the simulation until the files are free again.



Structures as shown in the pictures above (for example 2 pyramids of dielectric material on ideal conducting metal plates or 2 dielectric bodies being attached at a common boundary) cannot be simulated.

3. Proper setting of the clipping plane in the *Display area*.
4. Problem under Windows operating systems when enlarging a part of the structure by dragging the mouse pointer. The corresponding area becomes



non-transparent while selecting (Qt-problem).

## 9 Theoretical background (method of moments, EFIE, MFIE)

### Basics of the method of moments

The method of moments (MoM) has become an important numerical method and is firmly established in the range of applied electromagnetics. It is a general procedure for converting an operator equation into a linear equation system which can be solved on a computer. Its name evolves from the process of taking moments by multiplying by appropriate weighting functions and integrating. The use in electromagnetic problems has become popular especially since the work of R. F. Harrington [Harr1, Harr2]. Recommended for further reading is also [Gibson].

When applying a computer code, the user should have certain knowledge of the mathematical background. For this reason the main principles are shortly repeated here. Following Harrington consider a homogeneous equation of the form

$$L(f) = g \quad (2)$$

where  $L$  is a linear operator (for example the electric field integral equation, EFIE),  $g$  is a known function. This can be for example an impressed plane wave field, illuminating the structure. *CONCEPT-II* only considers cases where  $L$  represents integro-differential equations. The function  $f$  is the response of the excitation and is always represented by pure electric currents, i.e. surface currents and/or wire currents. Both  $f$  and  $g$  are functions of spatial coordinates and frequency; we seek to determine  $u$  with both  $g$  and  $L$  known.

The unknown function  $f$  is represented by a series expansion in the domain of  $L$  as

$$f = \sum_{(n=1)}^{\infty} \alpha_n f_n \quad (3)$$

where the  $\alpha_n$  are constant coefficients which are to be determined. The functions  $f_n$  are called basis functions. Substituting (2) into (3) and assuming linearity gives

$$f = \sum_{(n=1)}^N \alpha_n (L f_n) \simeq g \quad (4)$$

A set of testing, functions  $w_m$  is defined and the inner product, taken to be the integral of the product of functions is formed according to

$$\sum_{(n=1)}^N \alpha_n \langle (L f_n, w_m) \rangle \simeq \langle g, w_m \rangle \quad (5)$$

Eq. (5) represents a set of  $N$  equations and can be written in matrix form as

$$[\tilde{L}] = \begin{bmatrix} \langle L_e(f_1), w_1 \rangle & \langle L_e(f_2), w_1 \rangle & \cdots \\ \langle L_e(f_1), w_2 \rangle & \langle L_e(f_2), w_2 \rangle & \cdots \\ \cdots & \cdots & \cdots \\ \langle L_e(f_1), w_N \rangle & \langle L_e(f_2), w_N \rangle & \cdots \end{bmatrix}, [\tilde{\alpha}] = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_N \end{bmatrix}, [\tilde{g}] = \begin{bmatrix} \langle g_1, w_1 \rangle \\ \langle g_2, w_2 \rangle \\ \vdots \\ \langle g_N, w_N \rangle \end{bmatrix}, \quad (6)$$

$$[\tilde{L}][\tilde{\alpha}] = [\tilde{g}]$$

with  $[\tilde{L}]$  is a  $N \times N$  matrix.

Practically the solution for the  $\alpha_n$  is gained by LU decomposition and according to (3) an approximation of the function  $f$  is known.

Thus, the general procedure starts with the formulation of an integral equation of the physical problem, i.e. we have to find an appropriate integral equation representing the operator  $L$ . By appropriate choices of basis functions and testing functions, the integral equation is turned into a matrix equation to be solved.

### Integral equations for wires and metallic surfaces

For wires the electric field integral equation looks like this:

$$\vec{E}^{inc}(l) \cdot \vec{e}_l = \int_0^L \left[ \frac{j}{\omega \epsilon_0} \frac{dI(l')}{dl'} \frac{\partial G(\vec{r}, \vec{r}')}{\partial l} + j\omega \mu_0 I(l') G(\vec{r}, \vec{r}') \vec{e}_l \cdot \vec{e}_l' \right] dl' \quad (7)$$

$\vec{E}^{inc}(l) \cdot \vec{e}_l$  : Tangential impressed (known) field along a wire with running coordinate  $l$ .

$G$  : Green's function of free space

The following electric field integral equation (EFIE) is applicable to general metallic surfaces with ideal conductivity (e.g. plates, shells, closed cavities or cavities with apertures):

$$\vec{n} \times \vec{E}^{inc} = \vec{n} \times \left\{ \frac{1}{4\pi} \iint_S \left[ j\omega \mu \vec{J}_s \frac{e^{-jkR}}{R} + \frac{j \operatorname{div}' \vec{J}_s}{\omega \epsilon_0} \operatorname{grad} \frac{e^{-jkR}}{R} \right] dS' \right\} \quad (8)$$

$$\vec{n} \times \vec{E}^{inc} = \vec{n} \times \vec{E}(\vec{J}_s)$$

$\vec{J}_s$  : Surface current distribution

$\vec{n}$  : Normal vector of the surface

Using line matching with suitable testing functions (matching of the boundary condition  $E_{\tan} = 0$ ) these integral equations can be converted into



$$[Z] \cdot [I] = [U] \quad (9)$$

With  $[Z]$  as the impedance matrix,  $[I]$  as the vector with the unknown complex current amplitudes and  $[U]$  as the voltage vector with all its elements known, compare to (6).

One major limitation of the method is the large matrix size needed to solve a problem accurately if the geometry is large and complicated and/or high frequencies are to be investigated. In these cases, the typical structure dimensions are much larger than the wavelength.

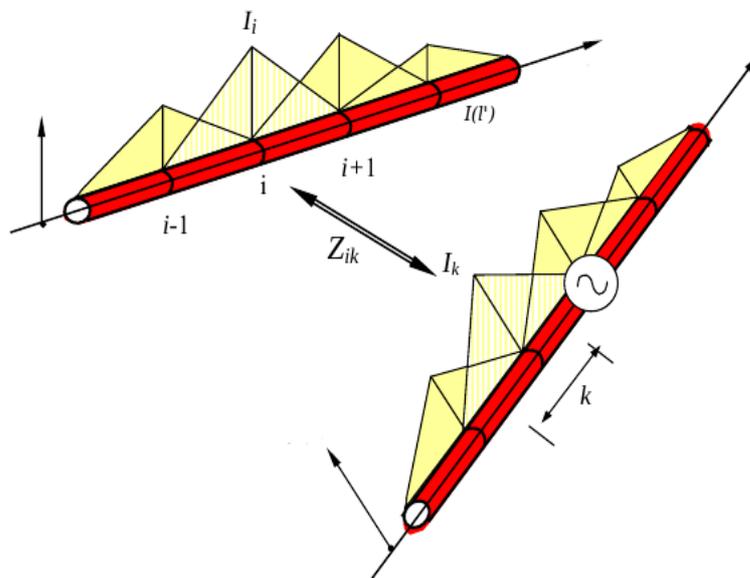
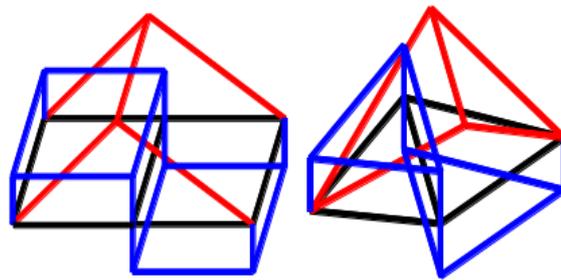


Fig 9.1: Triangular basis functions along two wires arranged close to each other

For wire structures we only have to perform an integration over line sources (current filaments along the axis). In *CONCEPT-II* triangular basis functions are applied as illustrated in Fig 9.1. From the overlapping of neighboring triangles results a piecewise linear approximation of the current. The distribution of the line charges along a wire is approximated by a step function.

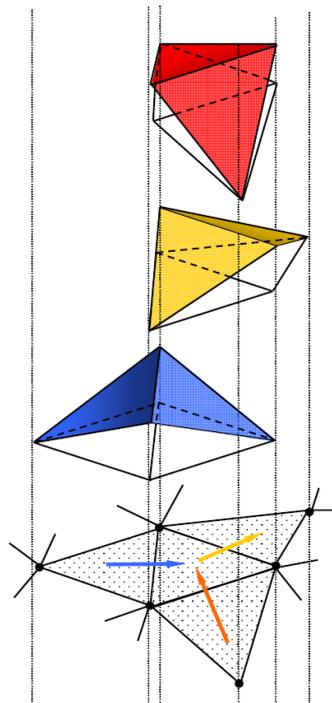


Fig 9.2: Roof-top functions (left) and RWG functions (right)



Setting up the system matrix can be explained as follows:  $Z_{ik}$  is the complex impedance for the impact of the basis function with amplitude  $I_i$  on wire segment  $k$ , causing a voltage  $U_k$  over segment  $k$  (without the corresponding wire being physical present). Then the matching position  $k+1$  is considered and again the effect of all current basis functions is computed in an

Fig 9.3: Superposition of RWG basis functions in a triangle



analogue manner.

Finally we get an equation system with as many lines as currents being defined on the wires (voltages are determined “under the tip of each triangle”).

Due to assumption of infinite conductivity, the line integral of the tangential E field along the wires vanishes except for one segment, where the voltage generator has

been defined. Therefore the voltage vector in (9) contains only zero coefficients except for a single entry, corresponding to the location of the generator.

On metallic surfaces, so-called RWG functions [RWG] defined over adjacent triangles and roof top functions defined over adjacent quadrangles are applied. Triangles can be attached to quadrangles. Fig 9.2 illustrates the surface basis functions: Black lines designate the patch boundaries, the magnitude of the current distribution on the patches is marked in red lines and finally blue lines indicate the surface charge distribution related to the falling and rising current parts. Note that the surface current density vectors are oriented from the left patch to the right patch in each case, they are so-called vector basis functions. Setting up the system matrix  $Z$  works quite similar to the approach already explained for wires, again the boundary condition that the tangential E field vanishes along ideally conductors is applied. Once the equation system is solved the complex amplitudes of the the vector basis functions are known and further quantities like EM fields in arbitrary observation points, along spatial paths or in regions in space can be computed by integrating over all patches (and wires segments, if wires are present).

In Fig 9.3 is illustrates the superposition of the RWG basis functions on a patch; at each edge another patch is connected. Knowing the amplitudes of the blue, yellow and red basis functions the resulting current pattern on the center patch can be computed. In the centroid we will have a certain magnitude and direction of the

current density vector. The tool  works in the described way, displaying the overall surface current distribution in the *Display area*.

## Treatment of dielectric bodies

In the case of a dielectric body we have to consider 2 boundary conditions:

$E_{t1} = E_{t2}$      $H_{t1} = H_{t2}$  We have a set of two integral equations to be solved:

$$\vec{n} \times \vec{H}_1(\vec{J}_1) = \vec{n} \times \vec{H}_2(\vec{J}_2)$$

$\vec{n} \times \vec{E}_1(\vec{J}_1) = \vec{n} \times \vec{E}_2(\vec{J}_2)$  Introducing the corresponding operators we have:

$$\vec{n} \times \left\{ \frac{1}{4\pi} \iint_S \vec{J}_1 \times \text{grad} \frac{e^{-jk_1 R}}{R} dS' + \vec{H}^{inc} \right\} = \vec{n} \times \left\{ \frac{1}{4\pi} \iint_S \vec{J}_2 \times \text{grad} \frac{e^{-jk_2 R}}{R} dS' \right\} \quad \text{and}$$

$$\vec{n} \times \left\{ \frac{1}{4\pi} \iint_S \left[ j\omega\mu_1 \vec{J}_1 \frac{e^{-jk_1 R}}{R} + \frac{j \text{div}' \vec{J}_1}{\omega\epsilon_1} \text{grad} \frac{e^{-jk_1 R}}{R} \right] dS' \right\} =$$

$$\vec{n} \times \left\{ \frac{1}{4\pi} \iint_S \left[ j\omega\mu_2 \vec{J}_2 \frac{e^{-jk_2 R}}{R} + \frac{j \text{div}' \vec{J}_2}{\omega\epsilon_0} \text{grad} \frac{e^{-jk_2 R}}{R} \right] dS' \right\} .$$



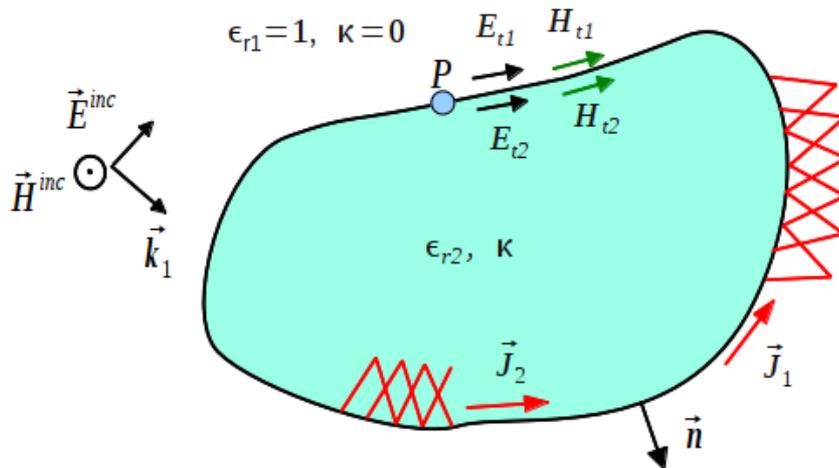


Fig 9.4: A dielectric body illuminated by a plane wave field

As is illustrated in Fig 9.4 we now have two current distributions,  $\vec{J}_1$  causes the scattered field outside the body and  $\vec{J}_2$  causes the internal field. Both currents are equivalent current distributions the purpose of which is to ensure continuous tangential field components at the boundary of the body. According to the above explained MoM approach an equation system has to be set up – compared to the case of metallic body with different boundary conditions and with twice the number of unknowns. A comprehensive study of boundary integral equations can be found in [Harr3]

### Thin layers on dielectric bodies

Fig 9.5 depicts a completely closed cavity. The walls of the cavity have a finite conductivity perhaps in conjunction with a certain relative permeability and are very thin with the thickness  $w$ . It is assumed that the typical dimension of the cavity  $D$  is much larger than  $w$ . In practical cases  $D$  might be in the m range whereas  $w$  is only a fraction of a mm, or even smaller. Such structures can hardly be solved by numerical methods from various reasons. In *CONCEPT-II* a hybrid technique has been implemented to overcome these problems.



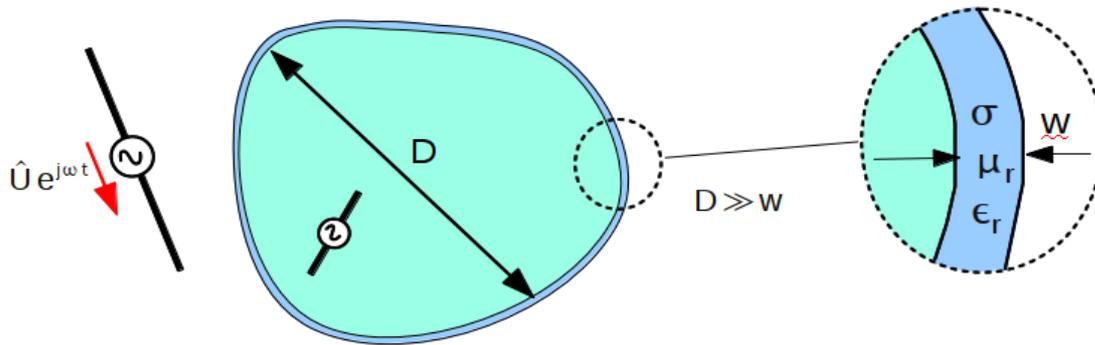


Fig 9.5: A shielded enclosure excited by a monopole antenna

As shown in Fig 9.5 an antenna is illuminating a cavity causing induced currents on the metallic surface. Due to the finite conductivity and small thickness  $w$  of the wall, the external fields are able to penetrate into the interior. Inside the wall the electromagnetic fields are oriented in parallel to the wall boundaries and will propagate almost perpendicular to the walls, i.e., locally we will have a wave propagation that can be described by the following equations.

$$\vec{E} = \vec{A} e^{j\vec{k}\cdot\vec{\zeta}} + \vec{B} e^{-j\vec{k}\cdot\vec{\zeta}}$$

$$\vec{H} = -\frac{\vec{k}}{\omega\mu} \times (\vec{A} e^{j\vec{k}\cdot\vec{\zeta}} - \vec{B} e^{-j\vec{k}\cdot\vec{\zeta}})$$

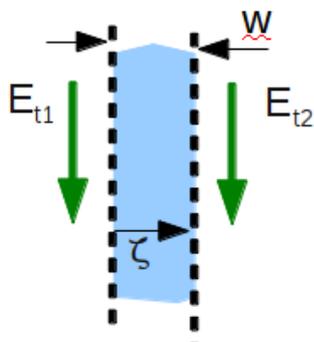


Fig 9.6: Local situation in a cross-section of the cavity walls, compare with the indicated position in Fig 9.5

The external tangential fields (for example the components  $E_{t1}, E_{t2}$  as shown in Fig 9.6 ) are caused by the currents induced on the external and internal sides of the cavity. Requiring that the EM fields are continuous on both sides of the cavity walls the constants  $A$  and  $B$  in the wave equations can be eliminated. In other words: the MoM system of equations is set up in a standard manner for a dielectric body, assuming air ( $\epsilon_r=1$ ) in this special case, see Fig 9.7. Instead of enforcing the boundary conditions directly the given wave equations are “switched in between”. Developing the theory further it turns out that the coefficients of the original equations system have to be weighted and rearranged to be able to compute the current distributions which are responsible for the EM fields inside and

outside the cavity.

Note that the wall thickness and the wall material is contained only in the analytical formulas for the internal wave propagation, hence the fields inside the wall are not resolved explicitly. Note further that the condition  $D \gg w$  has to be fulfilled. In practice the approach finds its limits when  $w$  gets larger than 5...6.times the penetration depth.

Although the cavity according to Fehler: Referenz nicht gefunden has a finite thickness  $w$  it is modeled by single surface patch file, which has to be discretized in the standard way, carrying two electric current distributions. The user can imagine the surface being located at the center of the wall. This approach becomes critical, of course, if  $w$  becomes too large compared with  $D$ .

It is no problem to consider multilayer sheets, however, we have the same restrictions for the electrical and geometrical thickness imposed on the resulting wall thickness as mentioned before for a single layer.

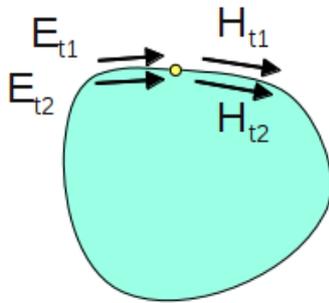


Fig 9.7: The boundary conditions to be fulfilled on a standard dielectric body.

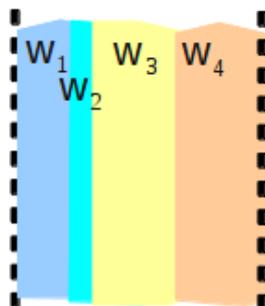


Fig 9.8: A multilayer sheet.

[Harr1] R. F. Harrington, Field Computation by Moment Methods, Macmillan Co., New York, 1968.

[Harr2] R.F. Harrington: Matrix Methods for Field Problems, Proceedings of the



IEEE, vol. 55, no. 2, February, 1967

[Gibson] Walton C. Gibson: The method of Moments in Electromagnetics, Capman & Hall/CRC, 2008

[RWG] S. M. Rao, D. R. Wilton, A. W. Glisson: Electromagnetic Scattering by Surfaces of Arbitrary Shape. IEEE Trans. Antennas and Propagation, vol. AP 30, no. 3, May 1982, pp. 409-418

[Harr3] R.F. Harrington: Boundary Integral Formulations for Homogeneous Bodies, Journal of Electromagnetic Waves and Applications, vol. 3, number 1, 1989 , pp. 1-15(15)

## 10 Licenses

CONCEPT-II uses the following 3rd-Party packages:

| Package                 | License | Link  |
|-------------------------|---------|---|
| OpenBlas 0.2.10 (Linux) | BSD     | <a href="http://www.openblas.net/">http://www.openblas.net/</a>   |
| Qt 5.10.1               | LGPL    | <a href="https://www.qt.io/download-open-source/#section-2">https://www.qt.io/download-open-source/#section-2</a> |

The usage of the software is according to the GPL and the LGPL (see below)

ScaLAPACK 2.02 (Linux) modified BSD  
[http://netlib.org/scalapack/#\\_licensing](http://netlib.org/scalapack/#_licensing)

ARPACK BSD  
<http://www.caam.rice.edu/software/ARPACK/>

MPICH 3.1 (Linux) BSD-like  
<http://www.mpich.org/about/overview/>

HDF5: Linux 1.8.12, Windows 1.8.20 BSD  
<http://www.hdfgroup.org/products/licenses.html>



OpenGL , Trackball  
Graphics, Inc.

(c) Copyright 1993, 1994, Silicon

Special functions  
of special functions" by

See preface of book "Computation

S. Zhang, J. Jin, Wiley & Sons, 1996

Inno Setup Compiler 5.5.5

<http://www.jrsoftware.org/isinfo.php>

For details on the respective licenses/copyrights see the files in the directory  
\$INSTALLATION\_DIRECTORY/licenses. Text files of the GPL and the LGPL are  
included (gpl.txt, lgpl.txt)

