ELECTROMAGNETIC SIMULATIONS

Foreword and apology:

> The two software tools described here were written by Mr. Charles Borges de Lima while at The University of Akron, on Leave from The Federal University of Santa Catarina (Florianopolis, Brazil). You will no doubt find in this document as well as in the menus for the software "awkward" usages of English. Please excuse these. I am working on correcting these as soon as possible. Because of that I suggest you visit the site again in the future and download at least this document as it is likely to change.
>
> N. Ida, August 27, 2006.

Software download and installation

1. Download the following two files by clicking on them

   Electromagnetic_Simulations.zip
   The_Smith_Chart.zip

2. Unzip the files if your system does not do so automatically. These files are actually folders and contain a number of files each. After unzipping, you should have two folders as follows:

   Electromagnetic_Simulations
   The_Smith_Chart

3. Place the two files into the work folder of Matlab.

To run any of the simulations:

1. Start Matlab 7.0 or 7.01 (or higher)
2. Select Open
3. Click on one of the two folders (Electromagnetic_Simulations or The_Smith_Chart)
4. Click on one of the two folders (For_MatLab70 or For_MatLab_701) depending on the version of Matlab (7.0 or 7.01 (or higher)) you have installed.
5. Select the m-file (Electromagnetic_Simulations.m or The_Smith_Chart.m)
6. Select "run"
7. The menu for the appropriate simulation will appear on screen
8. Select the simulation you wish to work with and follow screen instructions. Refer to this guide as necessary.

Introduction

The software described in this document was developed to help students understand the physics of electromagnetic. It was written to accompany the book "Engineering Electromagnetics", 2nd Edition, by Nathan Ida [1]. Some of its components are totally general and can be used in conjunction with any other textbook but some are intricately linked with the contents of the reference mentioned. There are many opportunities for electromagnetic

fields plotting, including design of structures for analysis of static electromagnetic fields and waves. The electromagnetic problems solved using this tool were solved using analytical equations and numerical methods (FDM, FEM, FDTD) but these methods are transparent to the user and their understanding is not necessary. Only the input and the output are visible to the user. The tool was written in Matlab and requires the user to run it under Matlab (see section on download and installation).

The various programs are intended to demonstrate concepts in electromagnetics. They are not intended for general purpose design or for computation of electromagnetic fields beyond that of demonstrations as no provisions have been made for accuracy and robustness. Nevertheless, they can give reasonable answers under the constraints described in this document. To simplify the use of the various programs, specific constraints on geometry (e.g. rectangular shapes, uniform distributions, etc.) have been used and are fully described in the document. Some of the programs use numerical techniques beyond the scope of the present textbook [1]. The numerical techniques are described briefly but additional information may be found in references [2] and [3] and in chapter 6 of reference [1]. The understanding of the numerical techniques is not necessary in order to use the programs.

The software has two separate parts as follows:
   a. Electromagnetic_Simulations
   b. The_Smith_Chart

The **Electromagnetic_Simulations** part is further divided into the following components

   1 – Vector Addition
   2 – Contour Lines - Gradient
   3 – Electrostatic fields
           Point Charges
           Potential
   4 – Electrical Current
   5 – Magnetostatic fields
   6 – Electromagnetic waves

The_Smith_Chart part deals with chapter 15 of [1] and will be described separately below. It is different than part (a) in that it does not attempt to provide a general purpose Smith Chart tool but rather to allow solution and display of the examples in chapter 15. Nevertheless, the individual examples can be modified by user input and in this fashion used to solve other problems of the same general type. With care and perhaps by combining the use of more than one example, one can use this tool for almost any problem in Chapter 15 as well as general problems in transmission lines and impedance matching, well beyond the scope of chapter 15.

a. Electromagnetic_Simulations
   Many of thes components in this software are self explanatory, and for each there is a ready-made example that can be run to clarify its use. In some cases the variables of the example can be changed by the user. In these instances, the variables are entered with <space> between them and <Enter> after the variables have been entered (if more than one

line of values is called for, type <Enter> at the end of each line). A <;> can be used instead of <Enter>.

1 – VECTOR ADDITION

This function is provided to help understand the meaning of vectors, with the possibility of plotting vectors in 2 and 3 dimensions and rotating them in space. It shows vectors and their projections on axes (in 2D) or on planes (in 3D). The system of coordinates is rectangular, with axes denoted as x, y and z.

A vector is defined by 2 points:

$$\text{in 2D:} \quad P_1(x_1, y_1) \quad \text{---------> } P_2(x_2, y_2);$$
$$\text{in 3D:} \quad P_1(x_1, y_1, z_1) \text{ ------> } P_2(x_2, y_2, z_2).$$

The arrow indicating the vector is directed from P_1 to P_2. The easiest way to enter the vectors' coordinates is to write x_1 y_1 x_2 y_2 <Enter> (keyboard) in the field provided for this purpose. The same can be done in 3D but now the sequence is x_1 y_1 z_1 x_2 y_2 z_2 <enter>.

After entering one or more vectors, one must press ENTER. The vectors will be plotted in blue. Each vector is held in memory; after that, one can use ADD to add to the previously entered vectors (new vectors will be plotted in red). The first vector is used as reference, so that all subsequent vectors are placed in the correct position with respect to this first vector. Pressing on the CLEAR field erases the memory. The vector component on the z axis in 3D will not have an arrow at its end, but rather a $<+>$.

The projection of the vectors can be displayed by selecting the field *Projections*. In 2D the projection will be on de axes. In 3D, the projection will be shown on the planes *xy, xz, yz*. The vector projections are plotted in cyan.

Note: This part of the software should be used in conjunction with Chapter 1 [1].

2 – CONTOUR LINES - GRADIENT

This function is intended to show how scalar functions (in three dimensions) look and how contour lines and gradients relate to the functions. Gradients are indicated as vectors (arrows). There are 4 different examples of functions starting from the simpleest to the more complex. To run an example the field *Run Example* must be selected.

Any function can be entered in the field *Function z = f(x,y)*. The correct way to type the function is the same as that used in MATLAB$^{©}$, a dot must be used before the symbols of multiplications, division, and exponential. The values used to calculate z (x and y) are assigned in the field *xy Values* (in the following way: *start value: increment: end value*. For example: -1:0.1:1 indicates the values starting with -1 at increments of 0.1 and ending with 1. One must pay attention not to allow the space increment to be zero at some point if the function divides by x or y (division by zero).

Note: This part of the software should be used in conjunction with Chapter 2 [1].

3 – ELECTROSTATICS

POINT CHARGES

In this part of the software it is possible to see the interaction between point charges in free space (including forces between them) and near perfect conductors. Positive charges are plotted as red points, negative charges as blue points. The potential and field lines are shown as well. The individual forces between the charges are plotted in red and the total forces are plotted in black. The method of images is used to see the behavior of the electric field in the presence of perfect conductors.

The following simulations can be carried out:

Free Space

2D

    1Q – One point charge, one can change the sign of the charge to see the fields.

    2Q – You can enter the values of two point charges (the difference between de charge value is used). You can choose to see the arrows (vectors) in the plot.

    NQ – One can enter any number of charges and their respective *xy* coordinates, in the following order: *Q x y* <Enter>. You can choose to see the arrows (vectors) in the plot.

    Capacitor – This is not an ideal capacitor, but it is a simulation of a capacitor using point charges over two parallel plates. One can choose the number of charges per plate (indicating charge density), the length of the plates and the distance between them. Positive charges are plotted as +, negative charges as points.

3D – the same, as in 2D.

Note: This part of the software should be used in conjunction with Chapter 3 and 4 [1].

Method of images

    1Q – one point charge over a perfect ground conductor. One can change de value of the charge and its height, and choose to see the image.

    NQ – any number of charges over a perfect ground conductor, with their respective *xy* coordinates, in the follow order   *Q x y* <Enter>

    Tilted planes – one can choose between different angles of perfect conductors and then entering the position of the charge and the charge value. The *x* and *y* coordinate of the charge must be inside of the plane section defined by the conducting planes, otherwise the software will change their value or indicate an error.

    Metal Box – this function uses the method of images to calculate the electric field produced by a point charge inside a perfectly conducting metal box. One can change the value of the charge, position, length and height of the box. In this simulation only the first 4 image charges and the original charge are used. An exact solution would require an infinite number of image charges. Nevertheless, the result obtained, while approximate, is quite good. The metal box is assumed to be infinite in the dimension perpendicular to the display plane.

Note: This part of the software should be used in conjunction with Chapter 5 [1].

ELECTRIC POTENTIAL

In this section of the software one can see the behavior of electrostatic fields when electric potentials are applied on conducting surfaces and different types of dielectric materials are placed between the conductors. The simulation relies on the finite difference method. Although a fairly complex method, its application in simple cases is fairly easy and is briefly explained next. The method requires geometrical modeling of the objects and space in which a solution is needed. See also Chapter 6 in [1]

Note: the following is given here as a very short introduction to the finite difference method (FDM) and the finite element method (FEM). A full understanding of the methods is not needed in order to take full advantage of the software. However the sequence of entering values must be followed exactly as described here.

FDM

The Finite Differential Method is fully described in Chapter 6 but its essentials are described here in a few sentences so that the user has some feel as to what the software does. The finite difference method uses square cells to solve for static electric fields within defined boundaries. The problem discussed here is a metallic box (actually an infinite channel with metallic walls) in free space, in which each wall is separated from the others by a very small gap. The potential on the walls can be assigned V_1, V_2, V_3, *and* V_4. The dimension of the cell is defined by *Dxy*. *I* and *J* are the number of cells in the *x* and *y* directions respectively. *# of iterat* is the number of iterations used by the FDM algorithm. The algorithm will converge to the answer after a number of iterations [1,3].

FEM

Finite Element Method [1-3] (also described fully in Chapter 6) uses triangular cells to solve the electrostatic field equations. To simplify the geometric modeling required, the finite elements are restricted to be right angle triangles. One can define the dimension of the triangular cells through *Dx* and *Dy*. The number of cells in the *x* and *y* directions are *I* and *J* respectively. See Fig. 1 for an example ($I = 4$, $J = 3$, $Dx = 1$, $Dy = 1$).
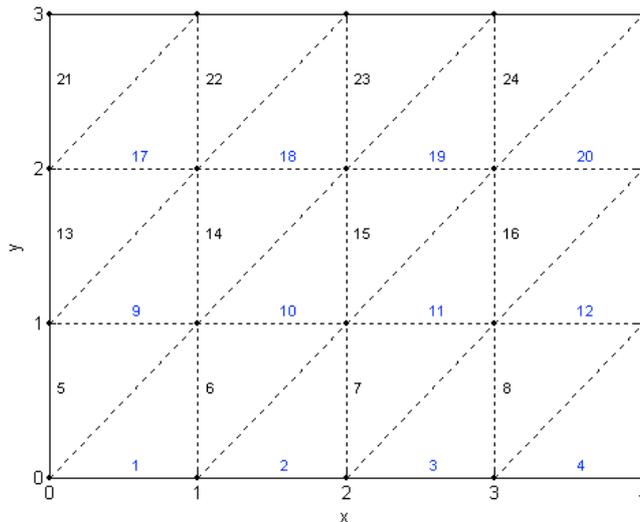
Figure 1. FEM mesh. The numbers inside the mesh are the element numbers (triangles).

To solve the problem it is necessary to assign boundary conditions. These are known potentials over the boundaries.

The _boundaries_ are assigned in the following order:

x_1 y_1 x_2 y_2 V_b <Enter> - The two points define a line on which the value is to be assigned. If only one point needs to be assigned use the following form: x_1 y_1 x_1 y_1

V_b -> potential over the boundary condition $[(x_1, y_1)$--$(x_2, y_2)]$. The line can be tilted, but must have the same number of cells in the x and y directions (projection).

The properties of the dielectric are assigned through _properties_ in the following order:

x_1 y_1 x_2 y_2 x_3 y_3 x_4 y_4 _Per_   <Enter>. The four points define a box. For a triangular geometry use $x_4 = x_3$ and $y_4 = y_3$. it MUST always be isosceles (same number of CELLS on x and y direction) with a right angle (e. g. |_) . The points (x, y) MUST be numbered in anti-clockwise direction starting from the lowest x and lowest y values (x_{min}, y_{min}), see Fig. 2.

Per -> relative electric permittivity inside the xy area. The permittivities are assigned in descending order (first element to last one). This means that the second line of permittivities will overwrite the first one and so on. Elements not assigned will have _Per_ = 1. At least one geometrical property must be assigned.

FEM Cell   Geometry of the materials – coordinate of the points

```
  / |                    3      3     3 --2   3 --2    4---3
 / | Dy                  / |    | \    \ |    | /      |   |
 / |                     / |    | \    \ |    | /      |   |
 ---                   1 --2   1 --2    1      1     1---2
 Dx                      A      B       C      D        E
```

Figure 2. Possible geometries and assignment of properties for the FEM mesh.

Examples considering Fig. 1 and Fig. 2:

Boundaries: 0 0 2 0 V_1 --
              1 1 3 3 V_2 /
              0 3 3 0 V_3 \

Properties - A : 0 0 0 2 2 2 2 2 Per_A
              B: 1 1 3 1 1 3 1 3 Per_B
              C: 4 0 4 3 1 3 1 3 Per_C
              D: 2 1 4 3 2 3 2 3 Per_D
              E: 1 1 3 1 3 2 1 2 Per_E

There are 5 examples that can be run. Of these, 3 examples deal with capacitors in which the permittivities can be changed. Also, there are some options for plotting. To run an example the field *Run Example* must be selected. After running an example the values used will appear in the properties and boundary conditions fields and can be viewed on the screen. At this point it is still possible to change values in the boundary and properties fields, but in this case the *Run Example* must not be selected again. The boundaries will be shown in blue and the properties contour in black.

Note: The FEM algorithm is not protected against entering wrong coordinates. One must be sure to enter the correct position of the properties and boundaries according to the mesh. The mesh will change according to the Boundary and Properties values. Should you enter incorrect values the results obtained may not be predictable.

Note: This part of the software should be used in conjunction with Chapters 4, 5 and 6 [1].

4 – ELECTRICAL CURRENT

In this section one can see the magnetic field lines around parallel infinite conductors. In 2D it is also possible to calculate and display the force between the conductors. The red point indicates a current flowing out of the screen, the blue point indicates a current flowing into the screen. In 3D the field plot is a rough plot of the magnetic field lines. The values of the currents can be changed.

Note: This part of the software should be used in conjunction with Chapters 8 and 9 [1].

5 – MAGNETOSTATICS

In this section the Finite Element Method is used in a fashion similar to that used above for the electrical potential. Therefore the mesh and the parameters entered are similar. The material properties and the boundary conditions are entered in the same way as for the electrical potential.

One can use two kinds of FEM calculations for static magnetic fields; one uses the scalar potential (voltage) and the other the vector potential [2].

One can define the dimension of the triangle cells by *Dx* and *Dy*. The number of cells in the *x* and *y* directions are defined by *I* and *J* respectively.

The *boundaries* and boundary conditions are assigned in the following order:

$x_1 \, y_1 \, x_2 \, y_2 \, P_b$ <Enter>

P_b -> potential over the boundary condition $[(x_1, y_1)\text{--}(x_2, y_2)]$. For vector potential the boundaries surrounding the mesh have $P_b=0$.

The *properties* are assigned in the following order:

$x_1 \, y_1 \, x_2 \, y_2 \, x_3 \, y_3 \, x_4 \, y_4 \, Mu$ <Enter>

Mu -> relative magnetic permeability inside the *xy* area. The geometry of the material is the same as for static fields. Elements not assigned will have *Mu* = 1.

The *source* is assigned in the following order:

$x_1 \, y_1 \, x_2 \, y_2 \, J_s$ <Enter> - the same as for boundaries condition.

J_s -> current density on the point or line $[(x_1, y_1)\text{--}(x_2, y_2)]$.

There are 4 examples for vector potential and one for scalar potential that can be run. For these examples a number in the name of the example, e. g. *[2M]* means that 2 material properties can be changed in the permeability field. The order is from top to bottom, that is, from material 1 to material 5. To run an example the field *Run Example* must be selected. After that the values used will appear in the properties and boundary conditions fields and can be seen on the screen. It is still possible to change values over the boundary and properties fields, but in this case the *Run Example* must not be pressed again.

Note: This part of the software should be used in conjunction with Chapters 8, 9 [1].

6 – WAVES

In this section one can see the interaction between electromagnetic waves and matter including potential and current in transmission lines. Electromagnetic propagation in two dimensions is shown. This is calculated using the Finite Difference Time Domain (FDTD) algorithm. Although the FDTD is used, the user has no interaction with the model: the model is internal and any change in parameters are handled by the program as inputs. Thus the details of the FDTD method are not important.

Reflection and Transmission

In this section, the reflection and transmission of a plane wave in different media is explored. The first option, *Metallic Walls*, is an example of a Gaussian pulse hitting metallic walls; the reflection of the electric and magnetic fields can be seen as they occur. The number of iterations can be changed to suit. In the next part, one can see a sinusoidal planar wave (electric field) crossing different types of media. There are 3 possibilities (1, 2 and 3 media); the electric properties of each medium can be assigned at will. *Length [P]* is the length of the medium in periods of excitation. *Time [P]* is the number of excitation periods to simulate.

Freq. [Hz] is the frequency of the excitation in Hertz. In the plot option one can choose what waves to plot, *transmitted* (red), *reflected* (green) and *total* (blue).

Note: This part of the software should be used in conjunction with Chapters 11, 12 and 13 [1].

Transmission Lines

Here one can see the electric potential, current, and input impedance on a transmission line, the potential, input impedance and current at the load are also shown. The phase shown for the load current is the angle assuming zero phase angle for potential at the load. The transmitted fields are plotted in red, the reflected fields in green and the total fields in blue.

The parameters are:
V_{source} = potential of the source [Volts];
Z_{source} = impedance of the source, must be $a + bi$, where i means imaginary number;
Z_{line} = impedance of the transmission line;
Attenuat. = attenuation over the line in Nepers/meter [Np/m];
Phase = phase constant [Rad/m];
Length = length of the transmission line in meters;
Z_{load} = impedance of the load;
Freq. = frequency of the source in Hertz;
Time [P] = number of periods of excitation.

It is also possible to calculate the input impedance Z_{in} at any point on the transmission line, where Z_{in} -> *x[m]* means the distance from the source, so at 0 one has the full length of the transmission line. At the full length the Z_{in} is equal Z_{load}. When this function is used a figure is generated showing the input impedance on the line (magnitude and angle).

Note: This part of the software should be used in conjunction with Chapter 14 [1].

FDTD – TE mode

In this part one can see the interaction of the electromagnetic waves with matter in two dimensions, with the possibility of changing parameters such as excitation and material properties. To compute the fields the Finite Differential Time Domain method is applied in a square mesh ($D = Dx = Dy = 0.0273$ m) in which a TE mode (Transverse Electric Mode) E_x – E_y - H_z. See Fig. 3 and bibliography.

Excitation parameters:
Type = Gaussian Pulse (no frequency assigned, fixed shape) or Sinusoidal (frequency $\leq$ 10^9 Hz);
Field = E_x and/or E_y, or H_z.

Plot parameters:

Field = modulus (magnitude) of electric field ($|E|$), or E_x, or E_y, or H_z;

# of interat. to plot = this value means the number of iteration for a new plot, so one can see the wave propagating during the simulation time. If this field is left empty, the fields values of the last iteration (mesh parameter) will be plotted.

One can choose to see the contour of the metal and/ or properties on the mesh.
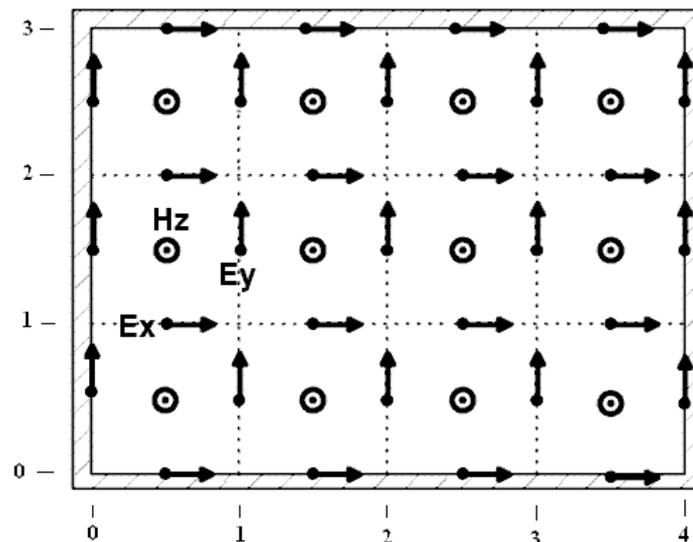
Figure 3. FDTD with square mesh, mode TE. $I = 4$, $J = 3$.

Mesh parameters:

I = number of cells in x direction;

J = number of cells in y direction;

# of iteract. = number of iterations to simulate.

The <u>*excitation*</u> is assigned in the following order:

x_1 y_1 x_2 y_2 <Enter>, excitation points over a line defined by 2 points, must be vertical (E_y or H_z) or horizontal (E_x or H_z) ($0 \le x \le I$, $0 \le y \le J$, entire values), example = [0 0 1 0], vector in the first cell. The excitation can also be tilted as the metal boundaries.

The <u>*metal*</u> boundaries are assigned in the following order:

x_1 y_1 x_2 y_2 *0*   <Enter> perfect metallic boundary conditions, a line defined by 2

points, can be vertical, horizontal or tilted. If tilted it MUST always have the same number of CELLS in x and y directions)

For a circle:

px py R Qu1 Qu2   <Enter>; *px, py* -> position [x, y] of the circle center;

R -> radius of the circle; number of cells, must be ≥ 3.

Qu1 and *Qu2* (1 = [0°- 90 °], 2 = [90 ° -180 °], 3 = [180 ° -270 °],

10

$4 = [270\,° - 360\,°]$.

If *Qu1* = *Qu2* just one quarter of the circle is assigned. For a complete circle *Qu1* = 1 and *Qu2* = 4; The properties are assigned in anti-clockwise sequence.

If no metallic boundaries are to be used, this field must be left empty.

The *properties* are assigned in the following order:

$x_1\ y_1\ x_2\ y_2\ x_3\ y_3\ x_4\ y_4$ *Per Cond* <Enter>; four points of a box; for triangle use $x_4 = x_3$ and $y_4 = y_3$; it MUST always be isosceles (same number of CELLS in x and y direction) with a right angle (e. g. |_) . The points (*x, y*) MUST always be entered in anti-clockwise direction starting from the lowest *x* and *x* values, $(x_{min},\ y_{min})$; see Fig. 4. At least one property must be assigned.

For a circle:

px py R Qu1 Qu2 0 0 0 Per Cond <Enter>, *px, py* -> position [*x, y*] of the circle center. *R* -> radius of the circle, number of cells, must be ≥ 3.

Qu1 and *Qu2* (1 = [0°- 90 °], 2 = [90 ° -180 °], 3 = [180 ° -270 °], $4 = [270\,° - 360\,°]$.

If *Qu1* = *Qu2* just one quarter of circle is assigned. For a complete circle *Qu1* = 1 and *Qu2* = 4; The properties are assigned in anti-clockwise sequence.

Per -> relative electric permittivity inside the *xy* area. The permittivities are assigned in descending order (first element to last). This means that the second line of permittivities will overwrite the first line and so on. Elements not assigned will have *Per* = 1.

Cond -> electric conductivity in Siemens/meter inside the *xy* area. The condutivities are assigned in descending order (first element to last). Elements not assigned will have *Cond* = 0.

FDTD Cell   Geometry of the materials – coordinate of the points

```
 __ __            3     3    3 --2  3 --2    4---3
|    | Dy        /|    |\    \ |   | /      |   |          ⊕
|    |          / |    | \    \|   |/       |   |          (circle)
--- ---        1 --2  1 --2   1    1       1---2
   Dx            A     B      C    D         E          F
```
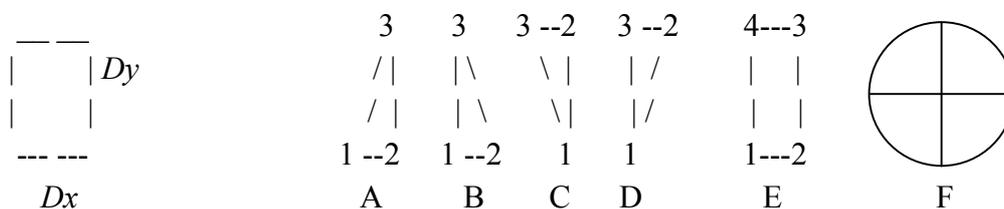
Figure 4. Possible geometries to assign properties into the FDTD mesh.

Examples (see Fig. 3 and Fig. 4):

Excitation: 0 0 2 0   --   Metal: 0 0 3 3 0   /
              0 0 0 3 | 1 3 4 0 0   \
              1 1 2 2   /   1 1 4 1 0   --

$$4\ 0\ 1\ 3 \quad \backslash \qquad\qquad 3\ 0\ 3\ 3\ 0 \quad |$$

Properties - A : 0 0 0 2 2 2 2 2 Per_A $Cond_A$

B: 1 1 3 1 1 3 1 3 Per_B $Cond_B$

C: 4 0 4 3 1 3 1 3 Per_C $Cond_C$

D: 2 1 4 3 2 3 2 3 Per_D $Cond_D$

E: 1 1 3 1 3 2 1 2 Per_E $Cond_E$

Considering a mesh with $I = 100$ and $J = 0$, for circles (for metal will be the same):

F: 50 50 10 1 4 0 0 0 Per_1 $Cond_1$ - [0 º - 360 º] full circle

30 40 10 1 2 0 0 0 Per_2 $Cond_2$ - [0 º - 180 º]

90 90 20 4 4 0 0 0 Per_3 $Cond_3$ - [270 º - 360 º]

Boundaries surrounding the mesh

Two algorithms can be use, one will use *metallic* boundaries and the other will use absorbing boundary condition (*ABC*) simulation the free space (Perfect Matched layer is used – *PML*, see bibliography).

Examples

Free Space GP - a Gaussian pulse propagates in the free space using PML. E_x E_y and H_z are shown.

3D Cavity – show a Gaussian pulse inside of a metallic cavity (box).

Scattering – shows waves hitting metallic walls and scattering from them.

Diffraction - A planar wave undergoing diffraction at a metallic wall with a gap.

Cylinder – two different materials, one in the shape a circle. Inside it there is a small metal circle.

Many Shapes – many properties and metals with different shapes.

Monopole – a small monopole over a metal ground.

Dipole – a small dipole in free space

Waveguide – a waveguide surrounded by two different media, excited with a planar wave.

In the first 4 examples above the properties cannot be changed. The other examples are self explanatory and their configurations can be changed. To run the examples the *Run Example* button must be pressed.

Note: The software does not protect against wrong coordinates, so one must be sure to use the correct coordinates according to the mesh, and inside the calculation domain. Because de cell dimensions are fixed the coordinate values are integer numbers.

Staircase lines will be used when straight lines are not possible (due to the FDTD mesh).

The plots will be shown after the computation is finished (It takes some time before the plots appear).

When the function PAUSE is used, pressing any key on the keyboard will re-start the plot. Some functions do not stop if PAUSE is pressed just once.

To stop plotting, one must use < Control + c >.

Hint: When the *waves* window is open, do not select *Run Example*, and push the FDTD button (a "ready made" example will be executed).

Note: This part of the software should be used in conjunction with Chapter 13, 17 and 18 [1].

c. The Smith Chart

The second part of the software deals with the Smith Chart and, more specifically, with Chapter 15 of [1]. This is structured somewhat differently than the first part in that the software follows the examples in the book rather than generic examples. However, the software can be used to solve other problems as long as they are structured in the form of one of the examples in the book or can be brought to this form by additional calculations. The purpose is not to provide a general purpose Smith Chart tool but rather to allow better understanding of the examples given and in the process better understand the material.

The Smith Chart menu screen (see section on download and installation) is divided into 7 section. Each section addresses one example although some data is common to more than one example. The examples are 15.1, 15.2, 15.3, 15.4, 15.5, 15.6, and 15.8.

There are a number of options in the simulations
1. Run the examples indicated on the screen. To do so, one selects the example number by clicking in the appropriate circle followed by selection of the *Run* button. Two things happen. First, the appropriate sections are filled with the computed data for the specific example. Second, the Smith chart with all appropriate indications is displayed (and can be printed).
2. Instead of selecting to run the example, another example of the same type but different data may be run by changing the data on the left upper corner of the screen and any specific data in the given example that can be changed (such as distances between stubs, location of load, etc.) and then selecting the Run button (without selecting the example number).
3. Any of the above can be run step by step by selecting the Step by Step button in the right-lower corner of the screen. With this option the process can be followed as it occurs. This can be done by selecting the appropriate button on the right lower corner of the screen.
4. Problems that do not fit any of the given examples, may still be solved but may require additional computation to bring them into the format needed to be solved using one of the examples shown.

As mentioned, this is not a general purpose Smith Chart tool. Such tools exist and may be found on the Internet either as free software or for purchase. A quick search will show many

and varied software tools of all types. The main value of the present tool is the close linkage with the examples in the textbook.

BIBLIOGRAFY

[1] N. Ida, *Engineering Electromagnetics*, Springer Verlag, 2nd ed., 2004.

[2] N. Ida and J.P.A. Bastos, *Electromagnetics and Calculation of Fields*, Springer Verlag, 2nd ed., 1996.

[3] A Taflove, and S.C. Hagness. *Computation Electrodynamics: the Finite-Difference Time-Domain Method*. Artech House, 2nd ed., 2000.