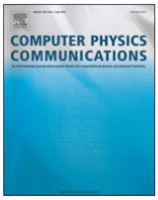
Computer Physics Communications 184 (2013) 2588–2595

Contents lists available at ScienceDirect

Computer Physics Communications

journal homepage: www.elsevier.com/locate/cpc



ERMES: A nodal-based finite element code for electromagnetic simulations in frequency domain[☆]

Ruben Otin [*]

*International Center for Numerical Methods in Engineering (CIMNE), Parque Mediterráneo de la Tecnología (PMT), c/ Esteve Terradas n° 5 - Edificio C3,
08860 Castelldefels, Barcelona, Spain*

ARTICLE INFO

Article history:
Received 29 January 2013
Accepted 11 June 2013
Available online 25 June 2013

Keywords:
Electrostatics and electromagnetics (10)
Finite element analysis (02.70.Dh)
Computational electromagnetics
Time-harmonic fields
Nodal elements
Weighted regularized Maxwell equations

ABSTRACT

In this work we present a new finite element code in frequency domain called ERMES. The novelty of this computational tool rests on the formulation behind it. ERMES is the C++ implementation of a simplified version of the weighted regularized Maxwell equation method. This finite element formulation has the advantage of producing well-conditioned matrices and the capacity of solving problems in the low (quasi-static) and high frequency regimens. As a consequence of this versatility, ERMES has been applied successfully to microwave engineering, antenna design, electromagnetic compatibility and eddy currents problems. This paper describes the main features of ERMES and explains how to use this numerical tool for computing electromagnetic fields in frequency domain.

Program summary

Program title: ERMES

Catalogue identifier: AEPV_v1_0

Program summary URL: http://cpc.cs.qub.ac.uk/summaries/AEPV_v1_0.html

Program obtainable from: CPC Program Library, Queen's University, Belfast, N. Ireland

Licensing provisions: Standard CPC licence, http://cpc.cs.qub.ac.uk/licence/licence.html

No. of lines in distributed program, including test data, etc.: 223528

No. of bytes in distributed program, including test data, etc.: 35954099

Distribution format: tar.gz

Programming language: C++.

Computer: Any computer with Microsoft Windows (32-bits or 64-bits) installed.

Operating system: Microsoft Windows 32-bits or 64-bits.

RAM: Problem dependent. See [1] for examples of computational performance.

Classification: 10.

External routines:
GiD [5] is used for geometrical modeling, data input, meshing and visualization of results.

Nature of problem:
Time-harmonic Maxwell equations.

Solution method:
Finite element formulation based on the weighted regularized Maxwell equation method [2, 1, 3, 4].

Additional comments:
!!!!! The distribution file for this program is approximately 35 Mbytes and therefore is not delivered directly when the download or Email is requested. Instead a html file giving details of how the program can be obtained is sent. !!!!!

Running time:
Problem dependent (see [1]).

[☆] This paper and its associated computer program are available via the Computer Physics Communication homepage on ScienceDirect (http://www.sciencedirect.com/science/journal/00104655).

[*] Tel.: +34 93 413 41 79; fax: +34 93 413 72 42.
E-mail addresses: rotin@cimne.upc.edu, ruben.otin.bcn@gmail.com.

http://dx.doi.org/10.1016/j.cpc.2013.06.010

References:

[1] R. Otin, L. E. Garcia-Castillo, I. Martinez-Fernandez, D. Garcia-Donoro, Computational performance of a weighted regularized Maxwell equation finite element formulation, Progress In Electromagnetics Research 136 (2013) 61–77.

[2] R. Otin, Regularized Maxwell equations and nodal finite elements for electro-magnetic field computations, Electromagnetics 30 (2010) 190–204.

[3] M. Costabel, M. Dauge, Weighted regularization of Maxwell equations in polyhedral domains, Numerische Mathematik 93 (2) (2002) 239–277.

[4] C. Hazard, M. Lenoir, On the solution of the time-harmonic scattering problems for Maxwell's equations, SIAM Journal on Mathematical Analysis 27 (1996) 1597–1630.

[5] GiD, The personal pre and post processor, International Center for Numerical Methods in Engineering (CIMNE), Barcelona, Spain. [Online] Available: http://www.gidhome.com.

1. Introduction

ERMES (*E*lectric *R*egularized *M*axwell *E*quations with *S*ingularities) is a finite element (FEM) code in frequency domain which implements in C++ a simplified version of the weighted regularized Maxwell equation method [1]. This finite element formulation produces well-conditioned matrices which can be solved efficiently with low-memory consuming iterative methods [2,3]. Also, thanks to the null kernel of its differential operator [4,5], it operates indistinctly in the low (quasi-static) and the high frequency regimens. Therefore, ERMES is a versatile tool which can be used in a wide variety of situations. For instance, it has been applied to microwave engineering [1], specific absorption rate computations [3,6,7], electromagnetic compatibility [8] and electromagnetic forming [9,10].

But, despite of the advantages of the formulation behind ERMES, there are a few drawbacks that make difficult its implementation in a computational electromagnetic software. The main drawback is the special treatment that must be given to the field singularities and discontinuities [1,5]. This treatment makes the regularized formulation more difficult to model and implement than the best known FEM formulation based on edge elements and the double curl Maxwell equations [11,12,2].

Then, although ERMES offers more easily solvable matrices than the edge-based formulations [2], the cost is a more difficult implementation and modeling. The aim of this paper is to show how ERMES minimizes the drawbacks of the regularized formulation, thanks to its user-friendly graphical interface and the object oriented design of its source code. Also, we present ERMES in detail and explain how to work with this numerical tool.

2. Finite element formulation

The FEM formulation implemented inside ERMES is explained in detail in [1–3]. ERMES solves numerically the weak form of the regularized Maxwell equations [4,5,13]. That is, it provides a FEM approximation with tetrahedral nodal (Lagrangian) elements of the electric field $\mathbf{E} \in \mathbf{H}_0(\mathbf{curl}, \mathrm{div}; \Omega)$ which satisfies that $\forall \mathbf{F} \in \mathbf{H}_0(\mathbf{curl}, \mathrm{div}; \Omega)$ holds:

$$\int_\Omega \frac{1}{\mu}(\nabla \times \mathbf{E}) \cdot \left(\nabla \times \bar{\mathbf{F}}\right) + \int_\Omega \frac{1}{\bar{\varepsilon}\varepsilon\mu}(\nabla \cdot (\varepsilon\mathbf{E})) \cdot \left(\nabla \cdot \left(\bar{\varepsilon}\,\bar{\mathbf{F}}\right)\right) - \omega^2 \int_\Omega \varepsilon\,\mathbf{E} \cdot \bar{\mathbf{F}} - \mathbf{R}.\mathbf{B}.\mathbf{C}.|_{\partial\Omega} = i\omega \int_\Omega \mathbf{J} \cdot \bar{\mathbf{F}}, \tag{1}$$

where

$$\mathbf{H}_0(\mathbf{curl}, \mathrm{div}; \Omega) := \{\mathbf{F} \in \mathbf{L}^2(\Omega) \,|\, \nabla \times \mathbf{F} \in \mathbf{L}^2(\Omega), \nabla \cdot (\varepsilon\mathbf{F}) \in L^2(\Omega), \hat{\mathbf{n}} \times \mathbf{F} = 0 \text{ in PEC}, \hat{\mathbf{n}} \cdot \mathbf{F} = 0 \text{ in PMC}\},$$

$L^2(\Omega)$ is the space of square integrable functions in the domain Ω, $\mathbf{L}^2(\Omega)$ is the space of vectorial functions with all its components belonging to $L^2(\Omega)$, PEC represents a perfect electric conductor boundary, PMC represents a perfect magnetic conductor boundary, $\hat{\mathbf{n}}$ is the boundary unit normal, the bar over a magnitude denotes its complex conjugate, μ is the complex magnetic permeability, ε is the complex electric permittivity, $i = \sqrt{-1}$ is the imaginary unit, ω is the angular frequency, $\mathbf{J}$ is an imposed current density and $\mathbf{R}.\mathbf{B}.\mathbf{C}.|_{\partial\Omega}$ is the term, properly adapted to the regularization, that takes into account the boundary conditions. Its general expression is

$$\mathbf{R}.\mathbf{B}.\mathbf{C}.|_{\partial\Omega} = \int_{\partial\Omega} \frac{1}{\mu}(\nabla \times \mathbf{E}) \cdot \left(\hat{\mathbf{n}} \times \bar{\mathbf{F}}\right) + \int_{\partial\Omega} \frac{1}{\mu\varepsilon\bar{\varepsilon}}(\nabla \cdot (\varepsilon\mathbf{E})) \cdot \left(\hat{\mathbf{n}} \cdot \left(\bar{\varepsilon}\bar{\mathbf{F}}\right)\right). \tag{2}$$

The time-harmonic variation of fields and sources is defined as $\mathbf{F}(t) = \mathbf{F}e^{-i\omega t}$, where $\mathbf{F}$ is a complex function that only depends on the spatial coordinates.

ERMES uses a simplified version of the weighted regularized Maxwell equation method [1,5] to overcome the known problem exhibited by the regularized formulation in the presence of field singularities. That is, instead of using a singularity dependent weight over all the domain (as in [5]), ERMES simply cancels the divergence term of (1) in the elements near a singularity [1].

In the surfaces between different materials, where the field is discontinuous, ERMES follows the strategy explained in [1]. This strategy consists in placing distinct nodes at the same position (one on each side of the discontinuity surface) and relating them by means of a matrix that contains information about the materials involved.

3. Code description

ERMES source code was developed from the first version of the C++ open source library Kratos [14,15]. ERMES is the customization of Kratos for solving electromagnetic problems in frequency domain. Therefore, the main structure and characteristics of the ERMES C++ code can be extracted from [16,17,14,15].

The current version of ERMES (version 7.0) is multi-processor (OpenMP) and it runs on Microsoft Windows 32-bits and 64-bits. The C++ source code has been compiled with Microsoft Visual C++ 2005. On computers that do not have installed Visual C++ 2005, it is necessary the installation of the Microsoft Visual C++ 2005 Redistributable Package to run ERMES with more than one processor.

The recommended system requirements depend on the size of the problems we intend to solve. As a reference, in a desktop computer with a CPU Intel Core 2 Quad Q9300 at 2.5 GHz, 8 GB

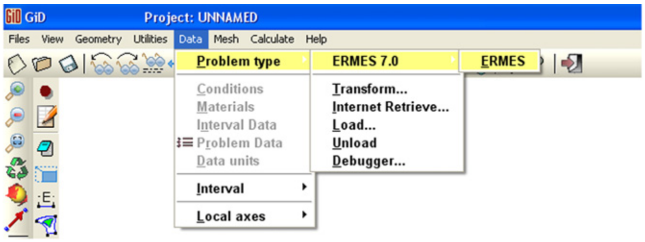


Fig. 1. ERMES starts after clicking on the *ERMES* button located at the upper menu of GiD: *Data* → *Problem type* → *ERMES* 7.0 → *ERMES*. Alternatively, ERMES can be started with the command *Load* (*Data* → *Problem type* → *Load*).

of RAM memory and the operative system Microsoft Windows XP Professional x64 Edition v2003, it can be solved problems with a FEM mesh of 1e6 isoparametric 2nd order tetrahedral nodal elements. In the references [2,3,6,8,1] are shown more examples of computational performance.

ERMES has a user-friendly interface created with Tcl/Tk and integrated in the commercial software GiD [18]. GiD is employed for geometrical modeling, data input, meshing and visualization of results. The graphical user interface (GUI) has been tested successfully for GiD versions 10 and 11.

4. Installation

ERMES is a problem type of GiD. Therefore, to install it, we only have to *copy & paste* the folder *ERMES 7.0* in the folder *problemtypes* (usually in the path: *C:/Program Files/GiD/GiD 10/problemtypes*) before starting GiD. To open ERMES we must go to the upper menu of GiD: *Data* → *Problem type* → *ERMES* 7.0 → *ERMES* and click on *ERMES* (see Fig. 1). Alternatively, we can start ERMES with the command *Load* (*Data* → *Problem type* → *Load*). Then, after a splash window that shows up for a few seconds, the ERMES menu bar appears at the left hand side of GiD (see Fig. 2).

5. Pre-process

Before running a simulation with ERMES, we need to create a geometry, define materials, apply boundary conditions and set the problem parameters. Once the problem is set up properly, we only have to mesh the geometry (clicking on the *Generate mesh* button) and, finally, execute ERMES (clicking on the *Calculate* button).

The computer-aided design (CAD) geometry can be generated inside GiD or imported from another CAD tool. We must remind that ERMES needs two different surfaces, located at the same position, to model the field discontinuities at the interface between different materials (see Section 2). We have to click on the upper menu of GiD *Geometry* → *Create* → *Contact* → *Volume* and select a surface to generate two overlapping surfaces connected by a contact volume. Then, we must assign each surface to its correspondent volume and mesh the geometry as usual. GiD will write a file with information relative to the nodes resting on the discontinuity surface (at which element they belong and what nodes are facing each other) before starting the computations. ERMES reads this file and applies the strategy explained in [1] to overcome the problem of modeling field discontinuities with nodal elements.

The materials, boundary conditions and problem parameters can be defined and assigned from the windows associated to the ERMES menu bar (see Figs. 2–4). This vertical menu bar has the following buttons (from up to down): Help–Material properties–Rectangular waveguide port properties–Coaxial waveguide port properties–Generic Robin condition coefficients–Current source properties–Dirichlet boundary conditions–Robin boundary conditions–Current sources–Field integrals–Solving parameters–Results–Generate mesh–Calculate. Each button opens its correspondent window or executes a command. In the following subsections we detail the parameters required at each window and how this parameters are incorporated into ERMES.

After assigning materials, boundary conditions and sources, we can proceed to mesh the geometry. We recommend to set the GiD meshing parameters as it is shown in Fig. 5. Also, it is advisable to mesh with special care around possible sources of field singularities (reentrant corners and edges of PECs, corners and edges of dielectrics and on the intersection of several dielectrics). We can assign smaller element sizes to specific parts of the geometry from the GiD upper menu: *Mesh* → *Unstructured* → *Assign sizes on....*

5.1. Material properties

From the *Materials* window we can assign materials to the volumes of the geometry and introduce the values of its electromagnetic properties. The material properties required by ERMES are the complex electric permittivity and the complex magnetic permeability:

$$\varepsilon = \epsilon_r'\epsilon_0 + i\left(\epsilon_r''\epsilon_0 + \frac{\sigma}{\omega}\right) \tag{3}$$

$$\mu = \mu_r'\mu_0 + i\,\mu_r''\mu_0 \tag{4}$$

where ϵ_r' is the real part of the relative electric permittivity, ϵ_r'' is the imaginary part of the relative electric permittivity, ϵ_0 is the vacuum electric permittivity ($\approx 8.8541878176e{-}12$ F/m), σ is the electric conductivity (S/m), ω is the angular frequency of the problem (Hz), μ_r' is the real part of the relative magnetic permeability, μ_r'' is the imaginary part of the relative magnetic permeability and μ_0 is the vacuum magnetic permeability ($\approx 1.2566370614e{-}6$ H/m). In the window *Materials* we introduce the values of σ, ϵ_r', ϵ_r'', μ_r' and μ_r'' for each medium.

5.2. Rectangular waveguide port

We define the properties of a rectangular waveguide port from the window *RWPortTE10*. The parameters required are the identification number (*Port type* = 0 for an input port, *Port type* > 0 for an output port) and the Cartesian coordinates of the lower left corner (00 *X*, *Y*, *Z*), upper left corner (*High X*, *Y*, *Z*) and lower right corner (*Width X*, *Y*, *Z*). The defined port is assigned to a rectangular surface from the window *Robin conditions* → *RW Port TE*10 *conditions*.

We assume that only the fundamental mode TE$_{10}$ is propagating in the rectangular waveguide ports. That is, we apply the boundary condition [11]

$$\hat{\mathbf{n}} \times \nabla \times \mathbf{E} = \gamma\,(\hat{\mathbf{n}} \times \hat{\mathbf{n}} \times \mathbf{E}) + \mathbf{U} \tag{5}$$

where γ is the propagation constant of the mode TE$_{10}$, which is $\gamma = \pm i\sqrt{k_0^2 - k_c^2}$ when $k_0 > k_c$ and $\gamma = \mp\sqrt{k_c^2 - k_0^2}$ when $k_0 < k_c$, being $k_0 = \omega\sqrt{\epsilon_0\mu_0}$ and $k_c = \pi/a$, with a being the width of the rectangular waveguide. The sign of γ depends on the direction of propagation. If the port selected is an output port (*Port type* > 0) then

$$\mathbf{U} = 0. \tag{6}$$

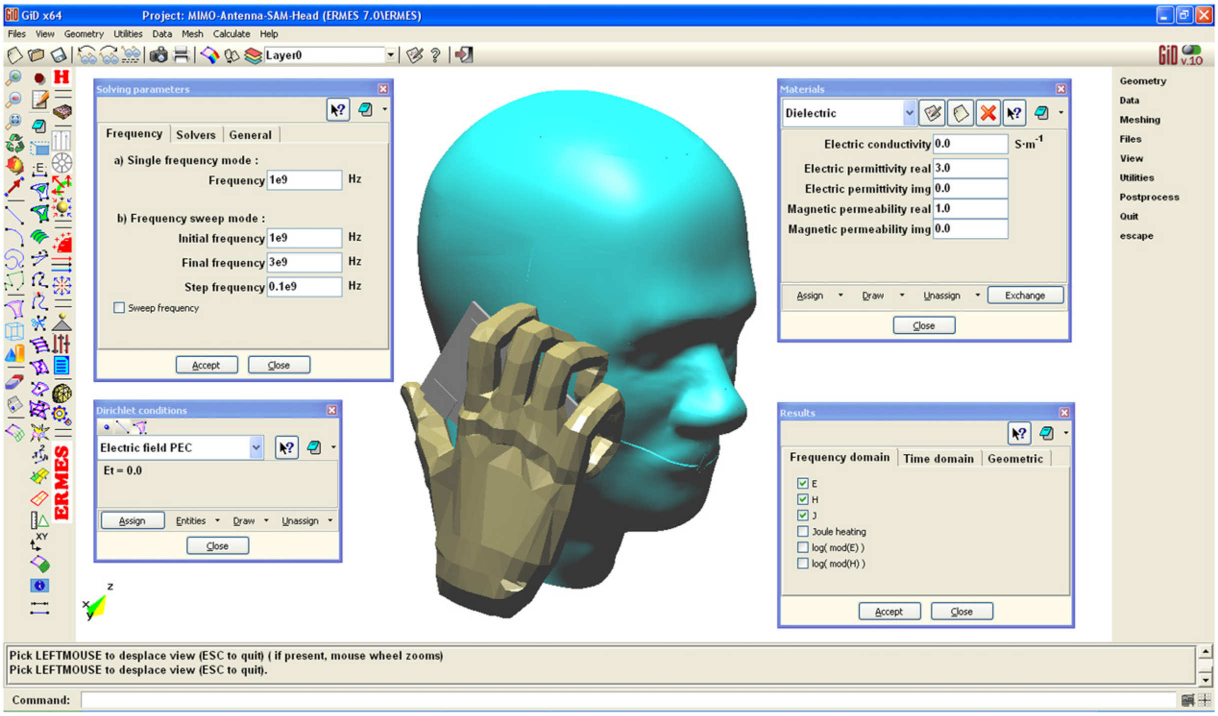
If the port selected is an input port (*Port type* = 0) then

$$\mathbf{U} = -2\,\gamma\,(\hat{\mathbf{n}} \times \hat{\mathbf{n}} \times \mathbf{E}_0), \tag{7}$$

where the field $\mathbf{E}_0$ is the imposed TE$_{10}$ mode

$$\mathbf{E}_{10} = -\sqrt{\frac{2i\omega\mu}{ab\gamma}}\,\sin(k_c x)\,e^{\gamma z}\,\hat{\mathbf{y}}, \tag{8}$$

being γ the propagation constant of TE$_{10}$, a the width of the rectangular waveguide and b is its height. In (8) we have considered



Solving parameters

Frequency | Solvers | General

a) Single frequency mode :

Frequency 1e9 Hz

b) Frequency sweep mode :

Initial frequency 1e9 Hz

Final frequency 3e9 Hz

Step frequency 0.1e9 Hz

☐ Sweep frequency

Dirichlet conditions

Electric field PEC

Et = 0.0

Materials

Dielectric

Electric conductivity 0.0 $S{\cdot}m^{-1}$

Electric permittivity real 3.0

Electric permittivity img 0.0

Magnetic permeability real 1.0

Magnetic permeability img 0.0

Results

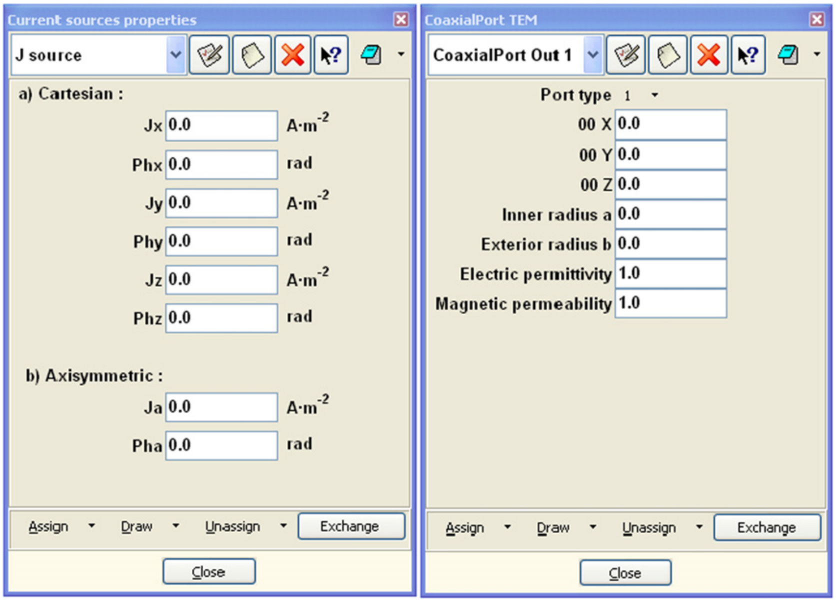
Frequency domain | Time domain | Geometric

☑ E
☑ H
☑ J
☐ Joule heating
☐ log(mod(E))
☐ log(mod(H))

Pick LEFTMOUSE to desplace view (ESC to quit) (if present, mouse wheel zooms)
Pick LEFTMOUSE to desplace view (ESC to quit).

Command:

Fig. 2. ERMES pre-process interface integrated in GiD. At the left hand side is located the ERMES menu bar. It is also shown a CAD geometry and some of the windows that can be activated using the ERMES menu bar.



Current sources properties

J source

a) Cartesian :

Jx 0.0 $A{\cdot}m^{-2}$

Phx 0.0 rad

Jy 0.0 $A{\cdot}m^{-2}$

Phy 0.0 rad

Jz 0.0 $A{\cdot}m^{-2}$

Phz 0.0 rad

b) Axisymmetric :

Ja 0.0 $A{\cdot}m^{-2}$

Pha 0.0 rad

CoaxialPort TEM

CoaxialPort Out 1

Port type 1

00 X 0.0

00 Y 0.0

00 Z 0.0

Inner radius a 0.0

Exterior radius b 0.0

Electric permittivity 1.0

Magnetic permeability 1.0

Fig. 3. Left: current sources properties window. Right: coaxial waveguide port properties window.

that the x-axis is along the width of the rectangular waveguide, the y-axis is along its height and the z-axis is perpendicular to the xy-plane.

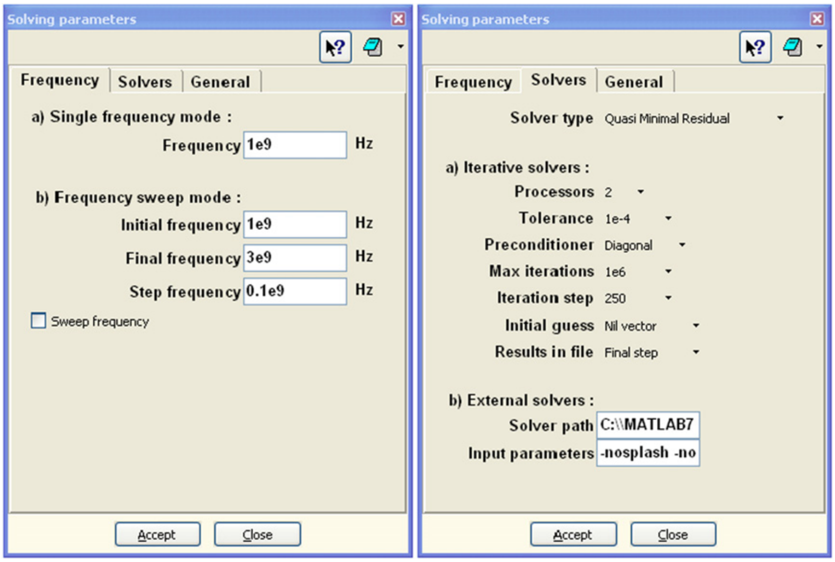
Because of the peculiarities of the FEM formulation implemented inside ERMES, the full boundary condition that must be applied to the waveguide port is [4,13]

$$\begin{aligned}&\hat{\mathbf{n}} \times \nabla \times \mathbf{E} = \gamma \left(\hat{\mathbf{n}} \times \hat{\mathbf{n}} \times \mathbf{E}\right) + \mathbf{U},\\ &\hat{\mathbf{n}} \cdot \mathbf{E} = 0\end{aligned} \tag{9}$$

which is the regularized version of the boundary condition (5). Therefore, after assigning (5) with *Robin conditions* $\rightarrow$ *RW Port TE*10 *conditions* we must apply the second condition of (9) with *Dirichlet conditions* $\rightarrow$ *Electric field TEPort*.

5.3. Coaxial waveguide port

We define the properties of a coaxial waveguide port from the window *CoaxialPortTEM* (see Fig. 3). The parameters required are the identification number (*Port type* $=$ 0 for an input port, *Port type* $>$ 0 for an output port), the Cartesian coordinates of the center (00 X, Y, Z), the coaxial inner radius (*Inner radius a*), the coaxial exterior radius (*Exterior radius b*) and the values of ϵ_r' (*Electric permittivity*) and μ_r' (*Magnetic permeability*) for the



a) Single frequency mode :
Frequency 1e9 Hz

b) Frequency sweep mode :
Initial frequency 1e9 Hz
Final frequency 3e9 Hz
Step frequency 0.1e9 Hz
□ Sweep frequency

Solver type Quasi Minimal Residual

a) Iterative solvers :
Processors 2
Tolerance 1e-4
Preconditioner Diagonal
Max iterations 1e6
Iteration step 250
Initial guess Nil vector
Results in file Final step

b) External solvers :
Solver path C:\\MATLAB7
Input parameters -nosplash -no

Fig. 4. Solving parameters window. Left: problem frequency settings tab. Right: solver settings tab.

medium inside the coaxial. The defined port is assigned to a surface from the window *Robin conditions* → *Coaxial TEM conditions*.

We assume that only the fundamental mode TEM is present in the coaxial waveguide ports. That is, we apply the boundary conditions (5)–(7) but now with $\gamma = i\omega\sqrt{\epsilon_r'\epsilon_0\mu_r'\mu_0}$ and $\mathbf{E}_0$ being the imposed fundamental TEM mode

$$\mathbf{E}_{\mathrm{TEM}} = \sqrt{\frac{\eta}{2\pi\ln(b/a)}}\left(\frac{e^{\gamma z}}{r}\right)\hat{\mathbf{r}}, \tag{10}$$

where $\eta = \sqrt{\mu_r'\mu_0/\epsilon_r'\epsilon_0}$, a is the coaxial inner radius, b is the coaxial exterior radius, z is the propagation direction, r is the radial coordinate and $\hat{\mathbf{r}}$ is the unitary vector of r.

As in the previous case, the full boundary condition that must be applied in the waveguide port is (9). Therefore, after assigning (5) with *Robin conditions* → *Coaxial TEM conditions* we must apply the second condition of (9) with *Dirichlet conditions* → *Electric field TEPort*.

5.4. Generic Robin boundary condition

We apply the generic Robin boundary condition

$$\hat{\mathbf{n}} \times \nabla \times \mathbf{E} = \gamma\,(\hat{\mathbf{n}} \times \hat{\mathbf{n}} \times \mathbf{E}) \tag{11}$$

from the window *Robin conditions* → *Generic Robin condition*. The real and imaginary parts of the coefficient γ are defined in the window *Generic Robin coefficients*.

5.5. Far field boundary condition

We apply the regularized version of the first order absorbing boundary condition [4,1]

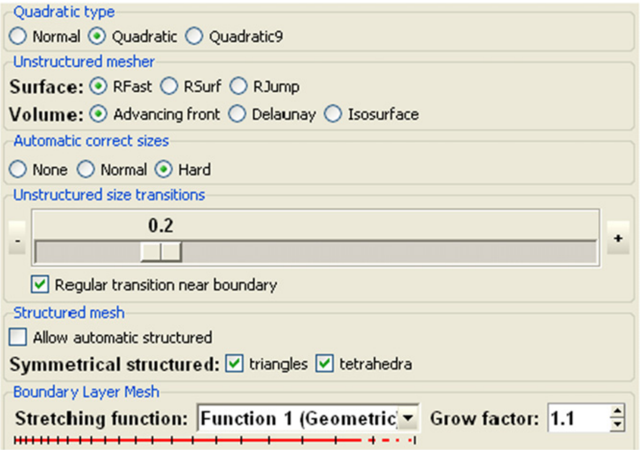
$$\begin{aligned} &\hat{\mathbf{n}} \times \nabla \times \mathbf{E} = i\omega\sqrt{\epsilon_0\mu_0}\,(\hat{\mathbf{n}} \times \hat{\mathbf{n}} \times \mathbf{E})\,, \\ &\nabla \cdot \mathbf{E} = i\omega\sqrt{\epsilon_0\mu_0}\,(\hat{\mathbf{n}} \cdot \mathbf{E}) \end{aligned} \tag{12}$$

from the window *Robin conditions* → *Far field condition*.

5.6. Volumetric current density

We define the properties of a volumetric current density **J** from the window *Current sources properties* (see Fig. 3) and assign it to a volume from the window *Current sources*.

We can define **J** by the modulus and phase of its Cartesian components or, if **J** is an axis-symmetric current density around the *Y* axis, by the modulus and phase of its angular component. This last case is very useful when modeling axis-symmetric coils [10], loop antennas [6], etc. When assigning the phases, we must remind that the time-harmonic variation of the sources in ERMES is defined as $\mathbf{J}(t) = \mathbf{J}e^{-i\omega t}$ (see Section 2).



Quadratic type
○ Normal ● Quadratic ○ Quadratic9

Unstructured mesher
Surface: ● RFast ○ RSurf ○ RJump
Volume: ● Advancing front ○ Delaunay ○ Isosurface

Automatic correct sizes
○ None ○ Normal ● Hard

Unstructured size transitions
0.2
☑ Regular transition near boundary

Structured mesh
□ Allow automatic structured
Symmetrical structured: ☑ triangles ☑ tetrahedra

Boundary Layer Mesh
Stretching function: Function 1 (Geometric Grow factor: 1.1

Fig. 5. Detail of the tab *Meshing*, inside the window *Preferences*. We can access to this window from the GiD upper menu bar: *Utilities* → *Preferences* → *Meshing*. The values showed in the picture are the recommended meshing parameters. The parameter *Unstructured size transitions* can be set to any value in the interval [0.1, 0.3]. The parameter *Smoothing* (located at the bottom of the same window) must be set to *HighAngle*.

5.7. Field singularities

As it is mentioned in Section 2 and detailed in [1], it is necessary to cancel the divergence term of (1) around the field singularities to obtain a physically sound solution with the regularized formulation.

From the window *Dirichlet conditions* → *Ungaged layers* we assign a positive integer number to a point or line in the geometry. This number represents the layers of elements that will be incorporated into the FEM matrix without the divergence term (see Fig. 6). We must assign *Ungaged layers* to the points and lines of the domain where the field can be singular (see Fig. 7). That is [19]: reentrant corners and edges of perfect electric conductors, corners and edges of dielectrics and intersection of several dielectrics.

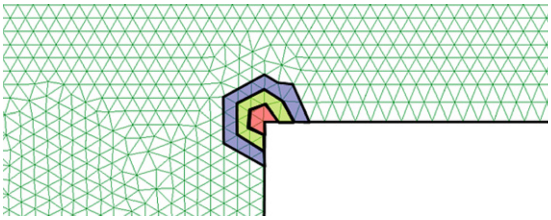


Fig. 6. The colored zone represents three layers (*Ungaged layers*) of elements around a reentrant corner.

The number of *Ungaged layers* depends on the size and order of the element used [1]. The optimal combination is second order elements with three *Ungaged layers* [1,3,8]. Higher order elements with one *Ungaged layers* is also possible, but computationally more expensive.

It is advisable to reduce the size of the mesh elements around the singular lines to capture the strong variations of the fields around these lines (from the upper menu of GiD: *Mesh* → *Unstructured* → *Assign sizes on lines*). GiD writes a list with the elements in the *Ungaged layers* zones after meshing the geometry and before starting the computations. ERMES reads this file and removes the divergence term from the elements in the GiD list while building the FEM matrix.

5.8. PEC/PMC boundary conditions

From the window *Dirichlet conditions* → *Electric field PEC* we assign the perfect electric conductor boundary condition

$$\hat{\mathbf{n}} \times \mathbf{E} = 0. \tag{13}$$

From *Dirichlet conditions* → *Electric field PMC* we assign the perfect magnetic conductor boundary condition

$$\hat{\mathbf{n}} \cdot \mathbf{E} = 0. \tag{14}$$

In *Solving parameters* → *General* → *Normal type* we can select if the unit normal $\hat{\mathbf{n}}$ at each node of a PEC/PMC surface is calculated as the average area weighted unit vector (*Area weighted*) or as the geometric average (*Geometric average*).

5.9. Periodic boundary conditions

From the window *Dirichlet conditions* → *Electric field PBC* we assign the periodic boundary condition

$$\mathbf{E}_{S_1}(\mathbf{r}) = \mathbf{E}_{S_2}(\mathbf{r}), \tag{15}$$

where S_1 and S_2 are the boundary surfaces of the unit cell of a periodic geometry (see Fig. 8). We can select two couples of surfaces S_1–S_2: *Front–Back* and *Right–Left*. *Front–Back* surfaces must placed in the *XY*-plane of GiD. The *Left* surface must be placed in the 00–*YZ*-plane. For cylindrical symmetry (as in Fig. 8) the central axis must be placed along the *z*-axis.

This boundary condition do not require equal meshes at the surfaces S_1–S_2 for its application. Each node at S_1 is expressed as a function of the nodes at S_2 before applying (15). In the window *Solving parameters* → *General* → *PBC Tolerance* we set the accuracy of this function. In the current version of ERMES (version 7.0), this boundary condition has been only implemented for isoparametric 2nd order elements and cyclic periodicity in the sides (as in the example of Fig. 8).

5.10. Field integrals

From the window *Field integrals* we can select a surface (*Field surface integral*) or a volume (*Field volume integral*) where the fields **E**, **H** and **J**, its modulus ($|\mathbf{E}|$, $|\mathbf{H}|$, $|\mathbf{J}|$) and the square of its modulus ($|\mathbf{E}|^2$, $|\mathbf{H}|^2$, $|\mathbf{J}|^2$) will be integrated. In a volume, it also integrated the quantity $\bar{\mathbf{J}} \cdot \mathbf{E}$.

The integrals are performed element by element through the FEM mesh. Then, we must be cautious if we select a surface located inside the problem domain. If the surface selected is meshed with triangles belonging to two different tetrahedra then, the value of the surface integral will be twice its actual value. This will not happen in the surfaces which are defining a discontinuity (see Section 5) if we select only one of the available surfaces.

From the window *Field integrals* we can also select the surface of a rectangular waveguide port (*Projection RWTE10*) or the surface of a coaxial waveguide port (*Projection CoaxialTEM*) to compute the scattering parameters S_{ii} and S_{ji}. These parameters are defined in ERMES by

$$S_{ii} = \frac{\int_{\Gamma_i} (\mathbf{E} \times \mathbf{H}_0) \cdot \hat{\mathbf{n}} \, \mathrm{d}\Gamma_i}{V_i^{\mathrm{imp}}} - 1, \qquad S_{ji} = \frac{\int_{\Gamma_j} (\mathbf{E} \times \mathbf{H}_0) \cdot \hat{\mathbf{n}} \, \mathrm{d}\Gamma_j}{V_i^{\mathrm{imp}}}, \tag{16}$$

where Γ_i is the input port, Γ_j is the output port, $\mathbf{H}_0$ is the magnetic field of the fundamental mode $\mathbf{E}_0$ (see Sections 5.2 and 5.3)

$$\mathbf{H}_0 = \frac{1}{i\omega\mu} (\nabla \times \mathbf{E}_0), \tag{17}$$

and V_i^{imp} is given by the expression

$$V_i^{\mathrm{imp}} = \int_{\Gamma_i} (\mathbf{E}_0 \times \mathbf{H}_0) \cdot \hat{\mathbf{n}} \, \mathrm{d}\Gamma_i. \tag{18}$$

5.11. Solving parameters

The window *Solving parameters* (see Fig. 4) set the problem parameters. It contains three tabs: *Frequency*, *Solvers* and *General*.

In the *Frequency* tab we set the frequency of the problem. If the checkbox *Sweep frequency* is checked then ERMES will do a frequency sweep starting at *Initial frequency*, ending at *Final frequency* and with a step *Step frequency*.

In the *Solvers* tab we set the solver parameters. We can select between the iterative solvers: *Quasi Minimal Residual* (recommended), *Bi Conjugate Gradient* and *Conjugate Gradient*. Also exists the possibility of using an *External solver*. If we select an in-core solver then we can set the number of CPU processors to solve the linear system in parallel, the residual tolerance ($\|b - Ax\|/\|b\| <$ *Tolerance*), the preconditioner (*Diagonal*) and the initial guess (*Nil vector* or *Read from file*). If we select *External solver* then the linear system generated by ERMES is written in a file and the solver located in *Solver path* is executed with the parameters given in the text-box *Input parameters*.

In the tab *General* we can select the element order (1*st*, 2*nd*, 3*rd*, 4*th*), the length factor (multiplies all the lengths by a given number), the normal type (*Area weighted*, *Geometric average*), the PBC tolerance (accuracy when comparing points in the PBC condition) and the dimension of the problem (3*D*, 3*D-Exy*, 3*D-Ez*, 3*D-Ea*). This last parameter is useful to reduce the computational cost in problems with special symmetries. That is, if *Dimension* is set to 3*D-Exy* then ERMES solves E_x and E_y and makes E_z equal to zero in all the domain. If *Dimension* is 3*D-Ez* then ERMES solves only E_z and makes E_x and E_y equal to zero in all the domain. If *Dimension* is set to 3*D-Ea* then ERMES apply in all the FEM nodes of the domain a change of coordinates from Cartesian (E_x, E_y, E_z) to axis symmetric around the *Y* axis (E_ρ, E_φ, E_y) and makes E_ρ and E_y equal to zero. That is, at

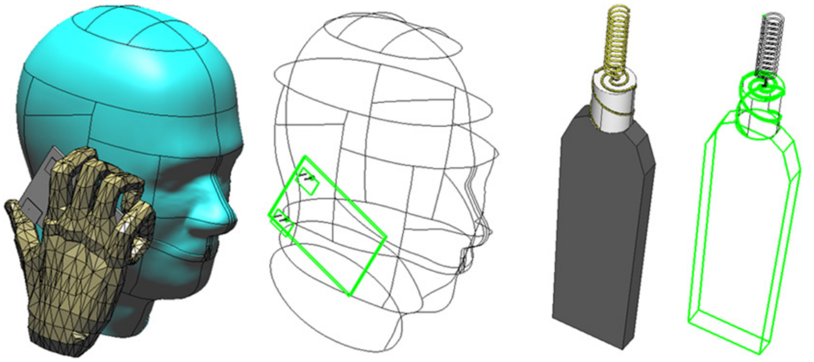


Fig. 7. Lines selected (in green) from the window *Dirichlet conditions* → *Ungaged layers*. The specified number of layers of elements around these lines will be incorporated into the FEM matrix without the divergence term. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

the same time we are building the matrix, we enforce at each node that $E_\rho = E_y = 0$ and

$$\begin{pmatrix} E_\rho \\ E_\varphi \\ E_y \end{pmatrix} = \begin{pmatrix} x/\rho & 0 & z/\rho \\ -z/\rho & 0 & x/\rho \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} E_x \\ E_y \\ E_z \end{pmatrix} \tag{19}$$

where x and z are the Cartesian coordinates of the node and $\rho = \sqrt{x^2 + z^2}$. We preserve the symmetry of the final FEM matrix applying (19) and its transpose as in [13].

We recommend to use 2nd order elements with the quadratic meshing activated (see Fig. 5). If another option is selected a warning will appear in the **.info* file. This message only reminds us that the PBC boundary condition is only implemented for isoparametric 2nd order elements and that we are using an option which is not optimal in ERMES. Nevertheless, this warning can be ignored and the computation will continue without problems.

5.12. Results

The window *Results* selects the results to be displayed in the post-process of GiD. It contains three tabs: *Frequency domain*, *Time domain* and *Geometric*.

In the *Frequency domain* tab we can select the visualization of the complex fields **E**, **H** and **J**. We will visualize the real and imaginary parts of the selected magnitudes as well as its modules. We can also select the visualization of the Joule heating Q defined as

$$\mathrm{Q} = \frac{\sigma + \omega\epsilon_r''\epsilon_0}{2}\,|\mathbf{E}|^2\,. \tag{20}$$

The *Time domain* tab allows the animation of the time-harmonic fields **E**, **H** and **J**. ERMES calculates the time domain results $\mathbf{F}_{td}(t)$ inside the interval $t \in [0, 2\pi/\omega]$ with the formula

$$\mathbf{F}_{td}(t) = \mathrm{Real}\left[(\mathbf{F}_r + i\,\mathbf{F}_i)\,e^{-i\omega t}\right] = \mathbf{F}_r\cos(\omega t) + \mathbf{F}_i\sin(\omega t). \tag{21}$$

From the tab *Geometric* we can select the visualization of the unit normals calculated at the PEC and PMC surfaces (*Boundary normals*) and at the interfaces between different materials (*Contact normals*).

6. Post-process

ERMES gives the results of the simulations in two different modes: single frequency mode and frequency sweep mode. In the following subsections we explain the characteristics of each one.

6.1. Single frequency mode

If the checkbox *Solving parameters* → *Frequency* → *Sweep frequency* is unchecked then ERMES solves the problem for a single frequency. The results selected in the window *Results* are stored in the file **.flavia.res*, where * represents the name of the GiD project. This file is located in the folder **.gid*.

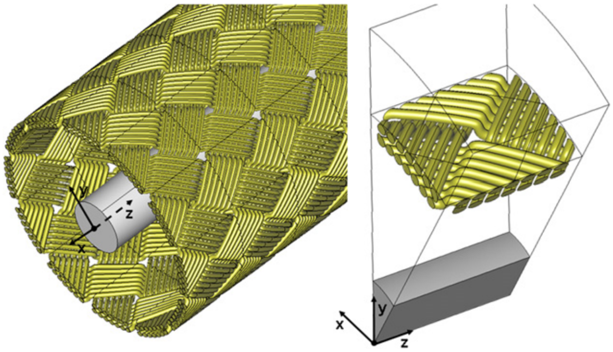


Fig. 8. A periodic geometry can be reduced to a smaller domain (unit cell) thanks to the periodic boundary conditions. This example is a GiD geometry used for computing the transfer impedance of braided wire shields [8]. The calculated electric field is shown in Fig. 9.

To visualize the results we must open the GiD post-processor (see Fig. 9). The frequency domain results are at the time step 0 of the *View Results & Deformations* window that is opened from the upper menu of GiD *Window* → *View results*. The time domain results are in the time steps >0 of the same window.

The information relative to the solver (residual, size of the problem, iterations, time spent), scattering parameters and field integrals can be retrieved from the file **.info*, which is also located in the **.gid* folder.

6.2. Frequency sweep mode

If the checkbox *Solving parameters* → *Frequency* → *Sweep frequency* is checked then ERMES makes a frequency sweep from *Initial frequency* to *Final frequency*. In this mode the results are not present as in Fig. 9. Instead, they are stored in files **.dat* located inside the **.gid* folder. There is a file for every volume and surface selected from the *Field integrals* window.

The name of the files associated to volumes are *Vn.dat*, where n is the ID of the volume. The name of the files associated to surfaces are *Surfn.dat*, where n is the ID of the surface. If the surface is a waveguide port then the name is *S1j.dat*, being $j = 1$ for the input port and $j = 2, 3, \ldots$ for the output ports.

In the files *Vn.dat* and *Surfn.dat* are stored for each frequency the values of the volume and surface integrals of the fields **E**, **H** and **J**, its modulus ($|\mathbf{E}|$, $|\mathbf{H}|$, $|\mathbf{J}|$) and the square of its modulus ($|\mathbf{E}|^2$, $|\mathbf{H}|^2$, $|\mathbf{J}|^2$). In the files *S1j.dat* are stored for each frequency the values of the scattering parameters S_{1j}.

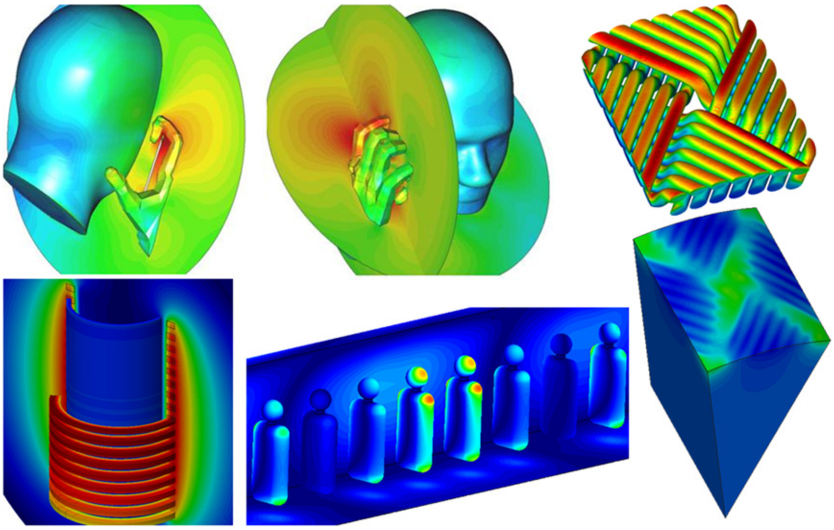


Fig. 9. Examples of visualization of ERMES results (modulus of the electric field) with GiD post-process. ERMES single frequency mode.

7. Conclusions

In this work we have presented a versatile and useful numerical tool that materializes an uncommon finite element formulation. The well-conditioned matrices provided by ERMES allow the simulation of large problems in the high and low frequency band using a desktop computer. Also, the drawbacks of the formulation (difficult modeling and implementation) are minimized, thanks to its user-friendly graphical interface and the object oriented design of its source code.

It is left for future work the improvement of ERMES capabilities. This includes the automatic detection of field singularities, the implementation of better in-core solvers and preconditioners, the hybridization with integral numerical techniques, better radiation and waveguide boundary conditions and the incorporation of new features (e.g. far field radiation patterns).

Acknowledgments

This work was part of the projects CaRDiAN (Cálculo de Radiación electromagnética en presencia de Dieléctricos mediante Análisis Numérico), Spanish Ministry of Science and Innovation, Plan Nacional I+D (2008–2011) Ref.: CIT-370000-2008-10 and SANTTRA (Sistema de Antenas para Transceptores de Radio), Spanish Ministry of Industry, Tourism and Commerce, PROFIT Ref.: FIT-330210-2006-44, framed in the European project SMART (Smart Antennas System for Radio Transceivers), EUREKA Ref.: PIDEA+ 04-161. The author also wants to acknowledge the financial support of the European Community, 7th Framework Programme FP7/2007–2013, project HIRF-SE (High Intensity Radiated Fields Synthetic Environment) Ref.: 205294 and the European Research Council, FP7 Programme Ideas, Starting Grant 258443—COMFUS (Computational Methods for Fusion Technology).

References

[1] R. Otin, Regularized Maxwell equations and nodal finite elements for electromagnetic field computations, Electromagnetics 30 (2010) 190–204.

[2] R. Otin, L.E. Garcia-Castillo, I. Martinez-Fernandez, D. Garcia-Donoro, Computational performance of a weighted regularized Maxwell equation finite element formulation, Progress In Electromagnetics Research 136 (2013) 61–77.

[3] R. Otin, H. Gromat, Specific absorption rate computations with a nodal-based finite element formulation, Progress In Electromagnetics Research 128 (2012) 399–418.

[4] C. Hazard, M. Lenoir, On the solution of the time-harmonic scattering problems for Maxwell's equations, SIAM Journal on Mathematical Analysis 27 (1996) 1597–1630.

[5] M. Costabel, M. Dauge, Weighted regularization of Maxwell equations in polyhedral domains, Numerische Mathematik 93 (2) (2002) 239–277.

[6] R. Otin, Numerical study of the thermal effects induced by a RFID antenna in vials of blood plasma, Progress In Electromagnetics Research Letters 22 (2011) 129–138.

[7] C. Gomez-Calero, N. Jamaly, L. Gonzalez, R. Martinez, Effect of mutual coupling and human body on MIMO performances, in: The 3rd European Conference on Antennas and Propagation, EuCAP 2009, 2009, pp. 1042–1046.

[8] R. Otin, J. Verpoorte, H. Schippers, A finite element model for the computation of the transfer impedance of cable shields, IEEE Transactions on Electromagnetic Compatibility 53 (4) (2011) 950–958.

[9] R. Otin, A numerical model for the search of the optimum frequency in electromagnetic metal forming, International Journal of Solids and Structures 50 (10) (2013) 1605–1612.

[10] R. Otin, R. Mendez, O. Fruitos, A numerical model for the search of the optimum capacitance in electromagnetic metal forming, AIP Conference Proceedings 1383 (2011) 935–942. The 8th International Conference and Workshop on Numerical Simulation of 3D Sheet Metal Forming Processes, NUMISHEET 2011.

[11] J. Jin, The Finite Element Method in Electromagnetics, second ed., John Wiley & Sons, 2002.

[12] M. Salazar-Palma, T.K. Sarkar, L.-E. García-Castillo, T. Roy, A. Djordjevic, Iterative and Self-Adaptive Finite-Elements in Electromagnetic Modeling, Artech House Publishers, 1998.

[13] W.E. Boyse, D.R. Lynch, K.D. Paulsen, G.N. Minerbo, Nodal-based finite-element modeling of Maxwell's equations, IEEE Transactions on Antennas and Propagation 40 (1992) 642–651.

[14] P. Dadvand, A framework for developing finite element codes for multi-disciplinary applications, Ph.D. Thesis, Universitat Politécnica de Catalunya, UPC, Barcelona, Spain, 2007.

[15] J. Mora, R. Otin, P. Dadvand, E. Escolano, M.-A. Pasenau, E. Oñate, Open tools for electromagnetic simulation programs, COMPEL: The International Journal for Computation and Mathematics in Electrical and Electronic Engineering 25 (3) (2006) 551–564.

[16] Kratos, An object-oriented environment for development of multi-physics analysis software, International Center for Numerical Methods in Engineering (CIMNE), Barcelona, Spain. [Online] Available: http://kratos-wiki.cimne.upc.edu.

[17] P. Dadvand, R. Rossi, E. Oñate, An object-oriented environment for developing finite element codes for multi-disciplinary applications, Archives of Computational Methods in Engineering 17 (2010) 253–297.

[18] GiD, The personal pre and post processor, International Center for Numerical Methods in Engineering (CIMNE), Barcelona, Spain. [Online] Available: http://www.gidhome.com.

[19] J.V. Bladel, Singular Electromagnetic Fields and Sources, IEEE Press, 1991.