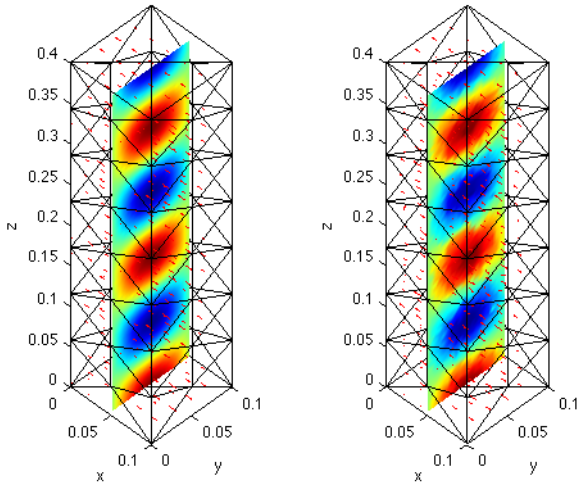
Centre de Mathématiques Appliquées
École Polytechnique
Université Paris-Saclay
UMR 7641 CNRS

Hierarchical Graph-Segmentation of Geometries for a Linear-Complexity Solver for Electromagnetic Heating

March 2013, Palaiseau, France.

Pedro Ramaciotti (pedro.ramaciotti@polytechnique.edu)
Alisa Sedunova (alisa.sedunova@polytechnique.edu)



Abstract

This report describes the results achieved in the investigation and the implementation of a Finite Flement Method for the modeling and simulation of the Maxwell's equations in cavity domains. A general physical modeling is given and a variational formulation is developed. An appropriate Finite Element Method based on vectorial basis functions is presented and discussed together with the computational implementation details regarding performance and efficiency. Finally, a validation consisting in the simulation of propagation inside a rectangular waveguide is presented together with examples of further applications such as microwave heating.

1 Introduction

The goal of this project was investigate a given topic and to develop a computational implementation using several methodologies discussed in the course. The topic chosen was the finite element approximation of the Maxwell's equations in bounded domains or cavities, with focus on microwave heating applications. Even though the original topic was proposed in consideration of scalar Helmholtz equation, the focus of this project was put in the investigation of vector finite elements for the Curl-Curl Helmholtz equation as it was regarded of interest and utility.

The structure of this report described as follows. Firstly, the physical pertinent electromagnetic modeling is presented and discussed and a variational formulation is presented. Well-posedness properties are proven and discussed. A finite element approximation method based on the Nédélec finite element is presented together with properties relevant to the implementation case of the project. A validation is provided and discussed for the case of a rectangular waveguide, for which an analytical solution is developed. Finally, an application is shown regarding the microwave heating phenomenon.

2 Variational Formulations for the Maxwell Equations in Cavities

2.1 The Maxwell Equations

The macroscopic electromagnetic phenomena, and in particular the ones involved in the present case, are described by the Maxwell equations. These equations govern the behavior of the electric field E, the magnetic field H, the electric induction D, the magnetic induction B and the electrical current density J. In differential form, they are written as

$$-\frac{d}{dt}D + \operatorname{curl} H = J, \tag{1}$$

$$\frac{d}{dt}B + \operatorname{curl} E = 0, \tag{2}$$

$$\operatorname{div} D = \rho, \tag{3}$$

$$\operatorname{div} B = 0. \tag{4}$$

The electric field and the electric induction are related by the electrical permittivity $\epsilon = \epsilon_0\epsilon_r$, where ϵ_0 is the electrical permittivity of the vacuum ($8.85418 \cdot 10^{-12} F/m$) and ϵ_r is a function of space modulating the permittivity. Similarly, the magnetic field and the magnetic induction are related by the magnetic permeability $\mu = \mu_0\mu_r$, where μ_0 is the permeability of the vacuum ($4\pi 10^{-7} H/m$) and μ_r is a space modulation function. These relations are expressed as

$$D = \epsilon E, \tag{5}$$

$$B = \mu H. \tag{6}$$

The electric current density is proportional to the electrical conductivity σ. Additionally, a given current density J_d can be imposed inside the domain to produce electrical excitation.

$$J = \sigma E + J_d \tag{7}$$

In such case the current density J_d is given and assumed independent of the other electromagnetic fields.

For the present case of study we are interested in a cavity domain Ω, where the boundary $\Gamma = \partial\Omega$ is considered to be a perfect electrical conductor, and thus the fields satisfy

$$\hat{n} \wedge E = 0 \quad \text{and} \quad \hat{n} \cdot H = 0 \quad \text{over } \Gamma. \tag{8}$$

Additionally, we are interested in the case where the support of the given excitation current $\mathrm{S}_{J_d} := \mathrm{supp}(J_d)$ is compactly contained in Ω and also disjoint from an electromagnetic load, whose support S_L is also compactly contained in Ω and it is the only region where ϵ, μ and σ are different from those of the vacuum. Figure 1 shows the cavity domain, the excitation region and the electromagnetic load.

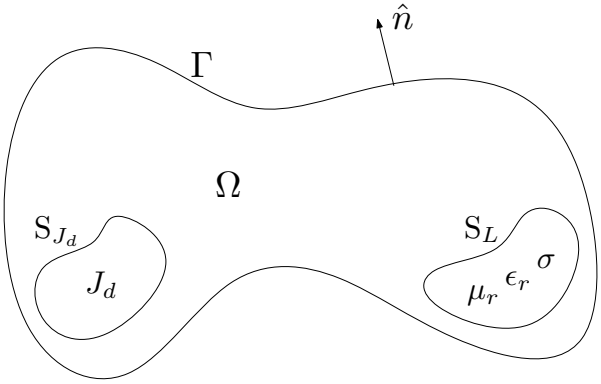


Figure 1: Cavity domain Ω with disjoint excitation region S_{J_d} and an electromagnetic load S_L.

2.2 The Time-Harmonic Electromagnetic Wave Equations

Assuming a time-harmonic behavior represented as a complex phasor using the convention $E(t,x) = Re\left\{E(x)e^{i\omega t}\right\}$, the Maxwell's equations can be rewritten as

$$\begin{cases} -i\omega\epsilon E + \operatorname{curl} H = \sigma E + J_d \\ i\omega\mu H + \operatorname{curl} E = 0 \end{cases} \tag{9}$$

Furthermore, defining $\varepsilon = \epsilon - i\sigma/\omega$ the equations can be written compactly as

$$\begin{cases} -i\omega\varepsilon E + \operatorname{curl} H = J_d \\ i\omega\mu H + \operatorname{curl} E = 0 \end{cases} \tag{10}$$

It is common for time-harmonic fields to group electrical losses due to *dielectric losses* and *conducting losses*, indistinguishable for a single frequency, in a single term, thus defining a complex electrical permittivity $\varepsilon = \varepsilon' + i\varepsilon''$, where ε'' accounts for conducting losses and dielectric losses for a given fixed frequency.

Taking the curl of the Ampere-Maxwell equation (1) and the Faraday-Maxwell equation (2) wave equations can be derived for the electric and the magnetic fields

$$\operatorname{curl}\left(\frac{1}{\omega\mu}\operatorname{curl} E\right) - \omega\varepsilon E = -iJ_d \tag{11}$$

$$\operatorname{curl}\left(\frac{1}{\omega\varepsilon}\operatorname{curl} H\right) - \omega\mu H = \operatorname{curl}\left(\frac{1}{\omega\varepsilon}J_d\right) \tag{12}$$

2.3 A Variational Formulation for Closed Cavities

To determine the electromagnetic behavior inside the cavity Ω under the prescribed circumstances the wave equation must be solved. Since the electric field E and the magnetic field H are related by the Maxwell's equations it is only necessary to solve one of the wave equations. Let's considered the electric field wave equation (11). The strong or classical problem (P_s) to be solved can be stated as

$$(P_s)\begin{cases} \textit{Given } J_d, \in \mathrm{C}(\Omega), \mu \in \mathrm{C}^1(\Omega), \text{ with } \mu > 0, \varepsilon' > 0, \textit{ find } E \in \mathrm{C}^2(\Omega) \text{ such that} \\ \qquad \operatorname{curl}\left(\frac{1}{\omega\mu}\operatorname{curl} E\right) - \omega\varepsilon E = -iJ_d \quad \textit{in } \Omega, \\ \qquad \hat{n} \wedge E = 0 \qquad\qquad\qquad \textit{on } \Gamma. \end{cases}$$

Let us define the following function subspaces:

$$\mathrm{C}^2_\Gamma = \left\{ v \in \mathrm{C}^2\left(\overline{\Omega}\right);\ \hat{n} \wedge v_{|\Gamma} = 0 \right\} \quad \text{and} \quad \mathrm{C}^\infty_\Gamma = \left\{ v \in \mathrm{C}^\infty\left(\overline{\Omega}\right);\ \hat{n} \wedge v_{|\Gamma} = 0 \right\}.$$

Multiplying the electric field wave equation (11) by the conjugate of $E^t \in \mathrm{C}^\infty_\Gamma$ and integrating over the cavity Ω we obtain

$$\int_\Omega \operatorname{curl}\left(\frac{1}{\omega\mu}\operatorname{curl} E\right) \cdot \overline{E^t} dx - \omega \int_\Omega \varepsilon E \cdot \overline{E^t} dx = -i \int_\Omega J_d \cdot \overline{E^t} dx. \tag{13}$$

Using Green's integration formula yields

$$\int_\Omega \frac{1}{\omega\mu}\operatorname{curl} E \cdot \overline{\operatorname{curl} E^t} dx - \omega \int_\Omega \varepsilon E \cdot \overline{E^t} dx = -i \int_\Omega J_d \cdot \overline{E^t} dx + \int_\Gamma \frac{1}{\omega\mu}(\operatorname{curl} E \wedge \hat{n}) \cdot \overline{E^t} ds, \tag{14}$$

which can be rewritten, using the triple product permutation identity, as

$$\int_\Omega \frac{1}{\omega\mu}\operatorname{curl} E \cdot \overline{\operatorname{curl} E^t} dx - \int_\Omega \omega\varepsilon E \cdot \overline{E^t} dx = -i \int_\Omega J_d \cdot \overline{E^t} dx + \int_\Gamma \frac{1}{\omega\mu} \hat{n} \wedge \overline{E^t} \cdot \operatorname{curl} E ds. \tag{15}$$

Applying the boundary conditions to suppress the boundary integral it is finally obtained that an electric field satisfying the classical, or strong, problem (P_s) also satisfies

$$\int_\Omega \frac{1}{\omega\mu}\operatorname{curl} E \cdot \overline{\operatorname{curl} E^t} dx - \int_\Omega \omega\varepsilon E \cdot \overline{E^t} dx = -i \int_\Omega J_d \cdot \overline{E^t} dx. \tag{16}$$

Let us consider the space

$$\mathrm{H}(\operatorname{curl}, \Omega) = \left\{ A \in \left(\mathrm{L}^2(\Omega)\right)^3 : \operatorname{curl} A \in \left(\mathrm{L}^2(\Omega)\right)^3 \right\} \tag{17}$$

with the following associated norm

$$\|A\|_{\mathrm{H(curl},\Omega)} = \left(\|A\|^2_{(\mathrm{L}^2(\Omega))^3} + \|\operatorname{curl} A\|^2_{(\mathrm{L}^2(\Omega))^3} \right)^{1/2}, \tag{18}$$

and the following subspace:

$$\mathrm{H}_0(\mathrm{curl}) = \{A \in \mathrm{H}(\mathrm{curl},\Omega) : \hat{n} \wedge A_{|\Gamma} = 0\}. \tag{19}$$

Using the notation

$$(A, B) = \int_\Omega A \cdot \overline{B} dx, \tag{20}$$

and the fact that $\mathrm{C}_{\infty,\Gamma}$ is dense in $\mathrm{H}_0(\mathrm{curl},\Omega)$, the weak problem (P_w) is stated as

$$(P_w) \begin{cases} \text{Given } J_d \in \left(\mathrm{L}^2(\Omega)\right)^3, \text{ find } E \in \mathrm{H}_0(\mathrm{curl},\Omega) \text{ such that } \forall E^t \in \mathrm{H}_0(\mathrm{curl},\Omega) \\ \left(\frac{1}{\omega\mu} \operatorname{curl} E, \operatorname{curl} E^t\right) - (\omega\varepsilon E, E^t) = -i(J_d, E^t) \end{cases} \tag{21}$$

2.4 Some Well-posedness Results

Lemma 1 (**Existence and Unicity**). *Let $\Omega \in \mathbb{R}^3$ be a bounded, regular domain with connected boundary Γ and let $J \in \mathrm{L}^2(\Omega)$ such that $\operatorname{div} J = 0$ and $\operatorname{supp}(J)$ is compactly included in Ω. Then*

1. *There exists a unique solution $A \in \mathrm{H}_0(\mathrm{curl},\Omega)$ such that*

$$\operatorname{curl}(\mu^{-1} \operatorname{curl} A) = J \quad \text{and} \quad \operatorname{div}(\varepsilon A) = 0 \quad \text{in} \quad \Omega, \tag{22}$$

2. *The application $J \mapsto A$ is compact in $\mathrm{L}^2(\Omega)$.*

Demonstration Let us give a weak form to (22). Let Ψ^0 be the space of functions ψ in $\mathrm{L}^2(\Omega)$ for which $\operatorname{grad}\psi = 0$ in Ω. If Γ is connected then these functions belongs to $\mathrm{H}^1_0(\Omega)$, otherwise it's a slightly bigger space, as ψ can be a non-zero constant over some parts of Γ.

The weak formulation yields: find $A \in \mathrm{H}^1_0(\mathrm{curl},\Omega)$, such that $\forall A' \in \mathrm{H}^1_0(\mathrm{curl},\Omega)$

$$(\mu^{-1} \operatorname{curl} A, \operatorname{curl} A') = (J, A')$$

and $\varepsilon A \in V$, where

$$V = \{v \in \mathrm{L}^2(\Omega); \quad \forall \psi \in \Psi^0 \, ((v, \operatorname{grad}\psi) = 0)\}.$$

By definition of Ψ^0 the kernel of curl in $\mathrm{H}^1_0(\mathrm{curl},\Omega)$ is exactly Ψ^0, which assures unicity. Let us then take $A_0 = (\chi * J)_{|\Gamma}$, with χ being a compact application $J \mapsto A_0$ and

$$\chi : \; x \mapsto \frac{\mu}{4\pi|x|}.$$

Let us now choose $\psi \in \Psi^0$ such that $\forall \psi' \in \Psi^0$

$$(\varepsilon A_0 + \operatorname{grad}\psi, \operatorname{grad}\psi') = 0 \quad .$$

Then, $A = A_0 + \operatorname{grad}\psi$ is the searched solution, and since an application $A_0 \mapsto \operatorname{grad}\psi$ is continue in $\mathrm{L}^2(\Omega)$, the application $J \mapsto \operatorname{grad}\psi$ is compact as a composition of continue and compact applications. It follows that $J \mapsto A$ is compact in $\mathrm{L}^2(\Omega)$. ∎

It is worth remarking that Lemma 1 implies the compacity of the operator $G : J_d \mapsto \varepsilon A$, defined the subspace V.

Theorem 1 (**Well-posedness**). *For any non-singular value of ω the problem (P_w) admits a unique solution and the application $J_d \mapsto E$ is continue from $\mathrm{L}^2(\Omega)$ to $\mathrm{H}(\operatorname{curl}, \Omega)$.*

Demonstration If ω is not singular then there is unicity. We will search then for a solution of the form

$$E = -i\omega A - \operatorname{grad}\psi, \quad \text{with } A_0 \in \mathrm{H}_0(\operatorname{curl}, \Omega), \varepsilon A \in V, \text{ and } \psi \in \Psi^0.$$

Fixing $E' = \operatorname{grad}\psi'$ in (P_w) with $\psi' \in \Psi^0$ we obtain

$$\int_\Omega i\omega\varepsilon(A + \operatorname{grad}\psi)\operatorname{grad}\psi' = \int_\Omega J_d \operatorname{grad}\psi' \quad \forall \psi' \in \Psi^0.$$

Since εA is orthogonal to $\operatorname{grad}\psi'$, then

$$\int_\Omega i\omega\varepsilon \operatorname{grad}\psi \operatorname{grad}\psi' = \int_\Omega J_d \operatorname{grad}\psi' \quad \forall \psi' \in \Psi^0$$

is a well-posed problem in Ψ^0 from where we deduce the continuity of $J_d \mapsto \operatorname{grad}\psi$ in $\mathrm{L}^2(\Omega)$.

Additionally A must satisfy (P_w), from where we obtain, $\forall A' \in \mathrm{H}_0(\operatorname{curl}, \Omega)$, that

$$((i\omega\mu)^{-1} \operatorname{curl} A, \operatorname{curl} A') = (J_d - i\omega\varepsilon \operatorname{grad}\psi, A') + \omega^2(\varepsilon A, A').$$

We identify this equation the Fredholm's equation of the second kind:

$$(1 - \omega^2 G)(\varepsilon A) = G(J_d - i\omega\varepsilon \operatorname{grad}\psi),$$

from where, using Fredholm's alternative when ω is a non-singular value, we get the desired result. ∎

The following lemma assures conditions over which there are no singular values of ω, which is to say, the problem (P_w) is well posed for any pulsation ω (for any frequency f).

Lemma 2 (**Absence of Singular Values**). *If $S_{\varepsilon''} := \operatorname{supp}(\varepsilon'')$ has positive measure, then there are no singular values of ω.*

Demonstration[Sketched] The Proposition 1 leads to the definition of a compact operator $J \mapsto \operatorname{grad}\psi$ that works over all the subspaces of V. This yields

$$(i\omega\varepsilon E, E') + ((i\omega\mu)^{-1}\operatorname{curl} E, \operatorname{curl} E') = 0 \quad \forall E' \in \mathrm{H}_0(curl, \Omega).$$

Choosing $E' = E$, we obtain that

$$\omega \int_\Omega \varepsilon''|E|^2 + i\omega \int_\Omega \varepsilon'|E|^2 + \int_\Omega (i\omega\mu)^{-1}|\operatorname{curl} E|^2 = 0.$$

Taking the real part, $E = 0$ on $S_{\varepsilon''}$, thus $\operatorname{curl} E = 0$. There exists then a non-zero field E, which has support in $\Omega \setminus S_{\varepsilon''}$ that verifies $-\Delta E = k^2 E$ and the boundary conditions

$$\hat{n} \wedge E = 0 \quad \text{and} \quad \hat{n} \wedge \operatorname{curl} E = 0 \quad \text{on} \quad \partial S_{\varepsilon''}.$$

Being these too many conditions over $\partial S_{\varepsilon''}$, which would imply that E and its derivatives are equal to zero on $\partial S_{\varepsilon''}$, this gives $E = 0$ over Ω thanks to the analyticity of E. ∎

2.5 A Variational Formulation for Boundary Values

If the solution to the Maxwell equations is known to be E_d and H_d and we take a subdomain Ω_c of Ω cutting the extremes, the boundaries can be separated in $\Gamma_c = \partial\Omega_c = \overline{\Gamma_0} \cup \overline{\Gamma_1} \cup \overline{\Gamma_2}$ as shown in the Figure 2.

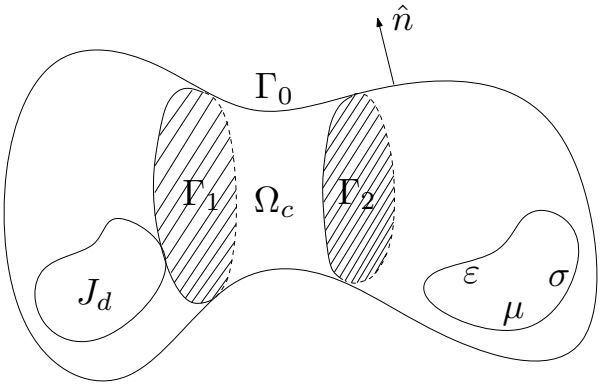


Figure 2: Cut domain Ω_c obtained from Ω used to pose boundary value problems.

The problem of finding the electric field E in Ω_c will be a new strong boundary value problem.

$$(P_s^c)\begin{cases} \textit{Given } J_d, \mu, \in C(\Omega), \textit{ find } E \in C^2(\Omega), \\ \operatorname{curl}\left(\frac{1}{\mu}\operatorname{curl} E\right) - \omega^2 \varepsilon E = -i\omega J_d \quad \textit{in } \Omega \\ \hat{n} \wedge E = 0 \qquad\qquad \textit{on } \Gamma_0 \\ \hat{n} \wedge E = \hat{n} \wedge E_d \qquad\qquad \textit{on } \Gamma_1 \cup \Gamma_2 \end{cases} \tag{23}$$

To derive a variational formulation we look for $E \in \mathrm{C}^2_{\Gamma_c}$ using a test function $E^t \in \mathrm{C}^\infty_{\Gamma_c}$, where

$$\mathrm{C}_{2,\Gamma} = \left\{ v \in C^2\left(\overline{\Omega}\right);\ \hat{n} \wedge v_{|\Gamma_0} = 0 \right\}, \tag{24}$$

$$\mathrm{C}_{\infty,\Gamma} = \left\{ v \in C^\infty\left(\overline{\Omega}\right);\ \hat{n} \wedge v_{|\Gamma_0} = 0 \right\}. \tag{25}$$

From equation (14) it follows that the solution to (P_s^c) satisfies

$$\int_\Omega \frac{1}{\omega\mu} \operatorname{curl} E \cdot \overline{\operatorname{curl} E^t} dx - \int_\Omega \omega\varepsilon E \cdot \overline{E^t} dx = -i \int_\Omega J_d \cdot \overline{E^t} dx + \int_{\Gamma_c} \frac{1}{\omega\mu} (\operatorname{curl} E \wedge \hat{n}) \cdot \overline{E^t} ds, \tag{26}$$

which, applying the boundary conditions on Γ_0 and the Faraday-Maxwell equation (2) can be written as

$$\int_\Omega \frac{1}{\omega\mu} \operatorname{curl} E \cdot \overline{\operatorname{curl} E^t} dx - \int_\Omega \omega\varepsilon E \cdot \overline{E^t} dx = -i \int_\Omega J_d \cdot \overline{E^t} dx + i \int_{\Gamma_1 \cup \Gamma_2} \frac{1}{\mu} (\hat{n} \wedge B_d) \cdot \overline{E^t} ds \tag{27}$$

Using the constitutive law relating the magnetic field and the magnetic induction (6) the equation is rewritten as

$$\int_\Omega \frac{1}{\omega\mu} \operatorname{curl} E \cdot \overline{\operatorname{curl} E^t} dx - \int_\Omega \omega\varepsilon E \cdot \overline{E^t} dx = -i \int_\Omega J_d \cdot \overline{E^t} dx + i \int_{\Gamma_1 \cup \Gamma_2} (\hat{n} \wedge H_d) \cdot \overline{E^t} ds. \tag{28}$$

If we define this time the space

$$\mathrm{H}_0(\operatorname{curl}, \Omega_c) = \left\{ A \in H(\operatorname{curl}, \Omega_c) : \hat{n} \wedge A_{|\Gamma_0} = 0 \right\}, \tag{29}$$

the weak formulation for the boundary value problem on the subdomain Ω_c is written as

$$(P_w^c)\begin{cases} \textit{Given } J_d \in \left(\mathrm{L}^2(\Omega_c)\right)^3, \textit{ find } E \in \mathrm{H}_0(curl, \Omega_c) \textit{such that } \forall E^t \in \mathrm{H}_0(curl, \Omega_c), \\ \left(\frac{1}{\omega\mu} \operatorname{curl} E, \operatorname{curl} E^t\right) - (\omega\varepsilon E, E^t) = -i(J_d, E^t) + i \int_{\Gamma_1 \cup \Gamma_2} (\hat{n} \wedge H_d) \cdot \overline{E^t} ds. \end{cases} \tag{30}$$

3 A Variational Approximation for the Maxwell's Equations in Cavities

This section describes the finite elements and the finite spaces to be used for a numerical approximation of the solution to the weak problem and they resume from [2] the necessary elements to give a basic theoretical background to the method used.

3.1 The First Order Nedelec Finite Element

Let us consider Ω to be polyhedral and let us define a mesh.

Definition 1 (Mesh)**.** *Let $\Omega \subset \mathbb{R}^3$ be an open, bounded, polyhedral domain. A tetrahedral mesh of Ω is a set of N_T subsets $T_k \subset \overline{\Omega}$, $k \in \{1, .., N_T\}$, such that*

1. $\overline{\Omega} = \bigcup_{k=1}^{N_T} T_k$.
2. *Each subset T_k of the mesh is a tetrahedron.*
3. *The intersection of two different tetrahedra T_k and T_l is either a common face, a common edge, a common vertex, or void.*

N_V, N_E, N_F will designate respectively the number of vertexes, the number of edges and the number of faces of the mesh.

Definition 2 (Basis Functions)**.** *Let us consider a single edge e_i, $i \in \{1, .., N_E\}$, of a mesh and a tetrahedron T_k to which it belongs. Let i_1 and i_2 be the extreme vertexes of the edge e_i. Inside the tetrahedron T_k for which e_i is an edge, the basis function N_i associated to this edge is*

$$N_i = l_i \left(\lambda_{i_1}^k \nabla \lambda_{i_2}^k - \lambda_{i_2}^k \nabla \lambda_{i_1}^k \right), \tag{31}$$

where λ_j^k is the barycentric coordinate function associated to the vertex j of the tetrahedron T_k and l_i is the length of the edge e_i. The basis functions over each edge are defined by convention such that $i_1 < i_2$.

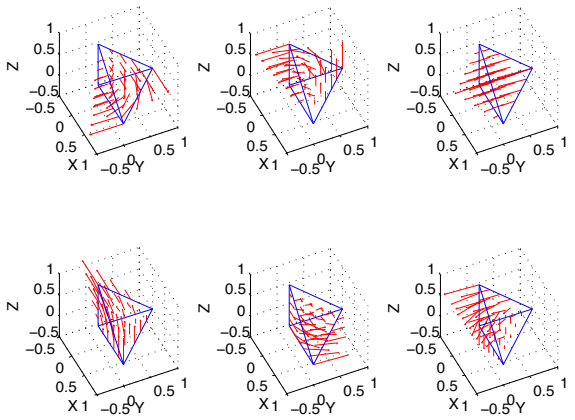


Figure 3: The six basis functions as described in Definition 2 for each local edge of the reference tetrahedron.

Definition 3 (Local Finite Element Spaces)**.** *Inside each tetrahedron* T_k*,* $k \in \{1, .., N_T\}$*, of a mesh the local function space spanned by the basis functions associated to each one the 6 edges is called* R_k

Lemma 3 (Zero Tangential Coponents of Subspaces of R_k)**.** *The subspace of* R_k *of the functions spanned only by the basis function associated to edges not contained in a given face of the tetrahedron* k *are perpendicular to that face when evaluated on it.*

Demonstration Without loss of generality we can consider a local coordinate system (x', y', z') such that a face of the tetrahedron is contained in the (x', y') plane, such that only one local node, say node 1 is not on this plane. In this local coordinate system, as shown in Figure 4, the points have coordinates

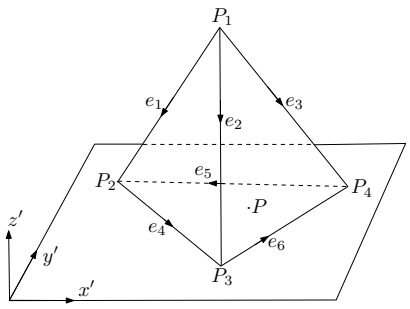


Figure 4: Local plane (x', y') associated to an arbitrary face of a a tetrahedron and a point P on that face.

$$P_1 = \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix}, P_2 = \begin{pmatrix} x_2 \\ y_2 \\ 0 \end{pmatrix}, P_3 = \begin{pmatrix} x_3 \\ y_3 \\ 0 \end{pmatrix} \text{ and } P_4 = \begin{pmatrix} x_4 \\ y_4 \\ 0 \end{pmatrix}.$$

Using the barycentric coordinates associated to each point, a function of the subspace of R_k spanned be edges e_1, e_2 and e_3 , evaluated on a point P on the surface contained in the (x',y') plane can be written as

$$\alpha_1 N_1 + \alpha_2 N_2 + \alpha_3 N_3$$

which expanding into the basis component terms yields

$$\alpha_1 l_1(\lambda_1 \nabla\lambda_2 - \lambda_2 \nabla\lambda_1) + \alpha_2 l_2(\lambda_1 \nabla\lambda_3 - \lambda_3 \nabla\lambda_1) + \alpha_3 l_3(\lambda_1 \nabla\lambda_4 - \lambda_4 \nabla\lambda_1)$$

Remarking the fact that the barycentric coordinate associated to the first vertex can be written as $\lambda_1 = z'/z_1$ we deduce that the previous expression, evaluated at the point P on the (x',y') plane (and thus having $z' = 0$) can be written as

$$-\frac{1}{z_1}\left(\alpha_1 l_1 \lambda_2 + \alpha_2 l_2 \lambda_3 + \alpha_3 l_3 \lambda_4\right) \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix},$$

which is certainly perpendicular to the (x', y') plane. ∎

Definition 4 (Degrees of Freedom)**.** *Let* R_k *be the function space spanned by the basis functions associated to each edge of the tetrahedron* T_k*. We associate to each edge* e_i *the degree of freedom* $\sigma_i \in (R_k)'$ *for* $A \in R_k$ *as*

$$\sigma_i(A) = \int_{e_i} A \cdot \hat{t}\, dl, \tag{32}$$

where $\hat{t}$ *is the unitary vector tangent to* e_i*.*

3.2 A Finite-Dimensional Conformal Subspace of $\mathrm{H}(curl, \Omega)$

It can be checked (see [2]) that with this definition of local spaces and degrees of freedom the resulting finite element (called the First Order Nédélec/Whitney Finite Element) is unisolvent and $\mathrm{H}(curl, \Omega)$-conforming.

Given a tetrahedral mesh of a polyhedral domain Ω (characterized by its mean edge length h) and using Definition 2, a finite subspace of $\mathrm{H}(curl, \Omega)$ can be constructed. According to the definition, we consider for each edge e_i in the mesh and its associated basis function N_i to construct the conforming finite-dimensional subspace defined as

$$\mathrm{H}_h(curl, \Omega) = \left\{ A \in \mathrm{H}(curl, \Omega); \forall T_k (A_{\restriction_{T_k}} \in \mathrm{R_k}) \right\} \tag{33}$$

Using Lemma 3 it can be showed that the tangential components of a field belonging to $\mathrm{H}_h(curl, \Omega)$ on a point on the boundary Γ only depend on the basis functions spanned by the edges of the triangular face to which this point belongs. Thus, the space $\mathrm{H}_{0,h}(curl, \Omega) \subset \mathrm{H}_0(\mathrm{curl}, \Omega)$ can be easily represented as a subspace of $\mathrm{H}_h(curl, \Omega)$ considering only the basis functions spanned by edges not contained in the boundary Γ as it will be detailed in the next subsection.

3.3 A Discrete Variational Formulation

Being $\mathrm{H}_{0,h}(curl,\Omega)$ a finite subspace of $\mathrm{H}_0(curl,\Omega)$ we consider to represent the electric field in this space to produce a Galerkin approximation.

Let $\mathcal{E}$ bet the set of edges of the mesh of Ω, considered to be polyhedral and bounded. We recall that $N_E = \#\mathcal{E}$. Let $\mathcal{E}_\Gamma$ be the set of edges belonging to the boundary Γ of the domain Ω, $\mathcal{E}_I = \mathcal{E}\backslash\mathcal{E}_\Gamma$ and $N_{E_I} = \#\mathcal{E}_I$. After Definition 2 and Lemma 3 a finite approximation $E_h \in \mathrm{H}_{0,h}(\mathrm{curl},\Omega)$ of $E \in \mathrm{H}_0(\mathrm{curl},\Omega)$ can be written using only the internal edges $\mathcal{E}_I$ as

$$E_h = \sum_{j=1}^{N_{E_I}} \alpha_j \, N_j, \tag{34}$$

where N_j is the basis function associated to edge $e_j \in \mathcal{E}_I$ and $\alpha_j \in \mathbb{C}$ is its associated coefficient. Using this approximation and testing for every basis function associated to $\mathcal{E}_I$ we can write the approximated variational formulation (P_w) as

$$(P_d) \begin{cases} \textit{Find } \alpha = \left\{\alpha_1,..,\alpha_{N_{E_I}}\right\} \in \mathbb{C}^{N_{E_I}} \textit{ such that for } N_i \textit{ associated to } e_i \in \mathcal{E}_I, \forall i \in \{1,..,N_{E_I}\}, \\ \displaystyle\sum_{j=1}^{N_{E_I}} \alpha_j \int\limits_{S_i \cap S_j} \left(\frac{1}{\omega\mu} \operatorname{curl} N_i \cdot \operatorname{curl} N_j - \omega\varepsilon N_i \cdot N_j \right) dx = -i \int\limits_{S_i} J_d \cdot N_i \, dx \end{cases} \tag{35}$$

where $S_i = \mathrm{supp}(N_i)$ and $S_j = \mathrm{supp}(N_j)$. The sets $S_i \cap S_j$ of non zero measure are either a single tetrahedron or a set of tetrahedra sharing a single edge. This fact means that the integrals can be easily divided into integrals over each tetrahedron of the mesh as it will be detailed in the next section.

4 Computational Implementation

4.1 Mesh Reading

In order to implement the previous methodology in a computer we make use of a mesh generator. In the case of this project we have used Gmsh[3], a 3D finite element grid generator with a build-in CAD engine and post-processor, which provides a controlled size mesh consisting of tetrahedra for a 3D domain and triangles for the 2D surfaces present in the model. Additionally, it allows for definition of physical objects in two and three dimensions with different labels to set boundary and interface conditions. Figure 5 provides an example of an object designed and meshed using Gmsh.

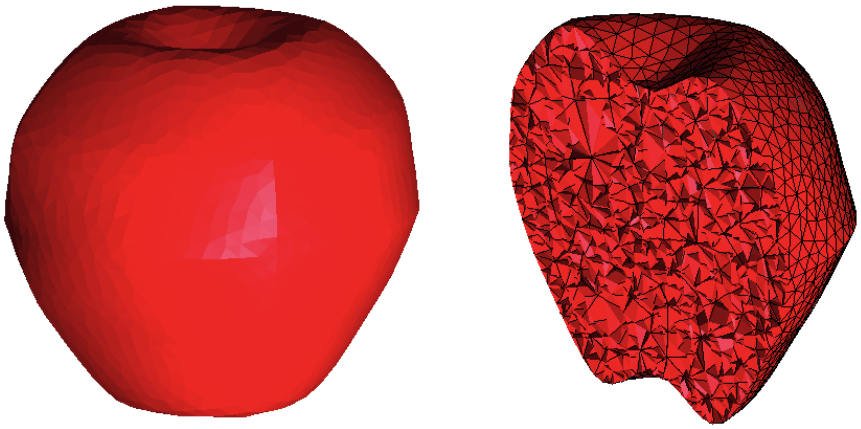


Figure 5: Example of an apple meshed with Gmsh using 4255 points, 3512 boundary triangles, 21233 tetrahedra and 27243 edges (mean edge = 2.15mm, max edge = 2.3mm)

To store the relevant information contained in the file generated by Gmsh different classes have been created for the physical objects, the nodes, the triangles, the tetrahedra, the edges and the degrees of freedom of the model to use in a simulation.

The *Object* class stores the information for each physical object of the model, indicating if it is a surface, a volume, and what are their physical properties. Other elements of a mesh can then be linked to physical objects to determine their properties and the degrees of freedom associated to them. Listing 1 shows the implementation of the class in C++.

Listing 1: *Object* Class

```
class Object {
public:
  string Name;
  int Dim;
  int Label;
  complex<double> mu;
  complex<double> eps;
  Object() : Name(""), Dim(0), Label(0), mu(), eps() {}
```

```
  Object(string InName, int InDim, int InLabel) : Name(InName), Dim(InDim),
      Label(InLabel), mu(complex<double>(0.0,0.0)), eps(complex<double
      >(0.0,0.0)) {}
  ~Object() {}
};
```

The *Node* class stores the information for each node of the model, providing its coordinates (for which it inherits properties from the *R3* class of FreeFem++), and the list of objects to which it belongs. Listing 2 shows the implementation of the class in C++.

Listing 2: *Node* Class

```
class Node : public R3 {
public:
  int Index;
  int NumberOfObjects;
  int ObjectList[MAX_OBJECTS_PER_NODE];
  int NumberOfTetrahedra;
  int TetrahedraList[MAX_TETRA_PER_NODE];
  Node() : R3(), Index(0), NumberOfObjects(0), ObjectList(),
      NumberOfTetrahedra(0), TetrahedraList() {} // Constructeur
  ~Node() { }           // Destructeur
};
```

The *Triangle* class stores the information for each triangle of the model, providing its vertices, a list of tetrahedra to which it belongs and other relevant information such as the unit normal, its measure (area) a list mapping its local edges to the global list of edges and their relative orientation. Listing 3 shows the implementation of the class in C++.

Listing 3: *Triangle* Class

```
class Triangle {
public:
  int N1,N2,N3;
  Node * Nodes[3];
  int ObjectLabel, SurfaceLabel;
  R3 Normal;
  double Mes;
  int LocalEdgeList[3],LocalEdgeOrientation[3];
  Triangle() : N1(-1), N2(-1), N3(-1), Nodes(), ObjectLabel(0), SurfaceLabel
      (0), Normal(), Mes(0.0), LocalEdgeList(), LocalEdgeOrientation() {}
};
```

The *Tetrahedron* class stores the information for each tetrahedron of the model providing a list of its vertices and triangles, its measure (volume), the object to which it belongs, and a list mapping its local edges to the global list of edges and their relative orientation. It also stores the coefficients of the P1 functions ($\lambda_i(x,y,z) = a_i + b_i x + c_i y + d_i z$) associated each one of its four vertices in arrays called A, B, C and D. Listing 4 shows the implementation of the class in C++.

Listing 4: *Tetrahedron* Class

```
class Tetrahedron {
public:
  Node * Nodes[4];
  int Index;
  int ObjectLabel, VolumeLabel;
  double Mes;
  double A[4], B[4], C[4], D[4];
  int LocalEdgeList[6],LocalEdgeOrientation[6];
  int NumberOfLocalTriangles;
  int LocalTriangleList[4];
  Tetrahedron() :Nodes(), Index(0), ObjectLabel(0), VolumeLabel(0), Mes(0.0)
      , A(), B(), C(), D(), LocalEdgeList(), LocalEdgeOrientation(),
      NumberOfLocalTriangles(0), LocalTriangleList() {}
};
```

The *Edge* class stores the information for each edge of the model providing information of its extreme vertices, the objects to which it belongs, its length, its unitary tangent and a label indicating the degree of freedom to which it is associated if it is the case, and an indicator that tells wether it is or not part of the boundary of the model. A relevant convention for this object is that the first node of the edge always has a smaller index than the second one, meaning that the global orientation of edges is from lower index of nodes to higher index of nodes whenever they are connected by an edge. Listing 5 shows the implementation of the class in C++.

Listing 5: *Edge* Class

```
class Edge{
public:
  int N1,N2;
  int Index;
  int NumberOfObjects;
  int ObjectList[MAX_OBJECTS_PER_EDGE];
  double Length;
  R3 Tangent;
  int DOF;
  bool EdgeOnBoundary;
  Edge() : N1(-1), N2(-1), Index(-1), NumberOfObjects(0), ObjectList(),
      Length(0.0), Tangent(), DOF(-1), EdgeOnBoundary(false) {}
};
```

The *DOF* (Degree of Freedom) class stores information for each degree of freedom in the model, indicating the edge to which it is associated. Listing 6 shows the implementation of the class in C++.

Listing 6: *DOF* Class

```
class DOF{
public:
  int Edge;
  DOF() : Edge(-1) {}
};
```

Finally, the *Mesh* class unites the whole information about the model containing arrays of objects, nodes, triangles, tetrahedra, edges and degrees of freedom either read directly from a mesh file generated using Gmsh or by post-processing of this data. The *Mesh* class also stores other relevant information such as the volume of the domain the minimum, maximum and mean edge length, the size of the domain in the three dimensions of space and the conventions for the local numbering and orientation of nodes and edges on triangles and tetrahedra (see Figure 6). Listing 7 shows the implementation of the class in C++.

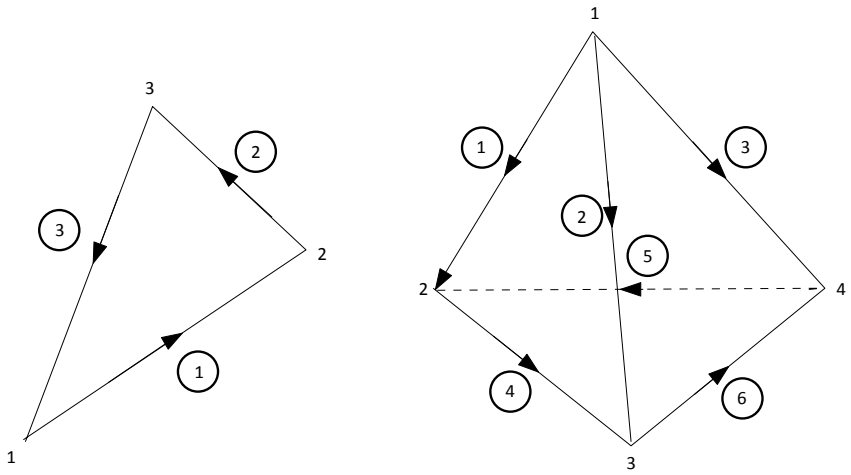


Figure 6: Diagrams of the internal edge numbering and orientation on triangles and tetrahedra using the local nodes.

Listing 7: *Mesh* Class

```
class Mesh {
public:
  // Donnees globales du maillage
  double Mes;      // Mesure du volume du maillage
  double h_min;    // Taille de la plus petite arete du maillage
  double h_max;    // Taille de la plus grande arete du maillage
  double h_mean;     // Taille moyenne des arete du maillage
  double x_min, x_max;// Limites du domaine maille en X
  double y_min, y_max;// Limites du domaine maille en Y
  double z_min, z_max;// Limites du domaine maille en Z
  double Lx, Ly, Lz;  // Taille du domaine maille
  string MeshVersion; // Version du maillage
  // Convention de numerotation des aretes
  int IndTetra[6][2]; // Numerotation des points des aretes de chaque
      tetraedre
  int IndTrian[3][2]; // Numerotation des points des aretes de chaque
      triangle
  // Donnees des groupes physiques du maillage
  int Nobjects;    // Nombres de groupes physiques du maillage
  Object * ObjectList;// Tableu avec les objets physiques
```

```
  // Donnees des sommets du maillage
  int Nnodes;      // Nombre de points du maillage
  Node * NodeList;  // Tableu avec les noeuds du maillage
  // Donnees des elements (Tableau temporale)
  int Nelems;      // Nombre d'elements du fichier de maillage
  int ** ElemData;  // Tableau temporel de stockage des donnees des elements
  // Donnees des Triangles
  int Ntrian;      // Nombre de Triangles
  Triangle * TriangleList;// Tableau avec les triangles du maillage
  // Donnees des Tetraedres
  int Ntetra;      // Nombre de Tetraedres
  Tetrahedron * TetraList;//Tableau avec les tetraedres du maillage
  // Donnees des Aretes
  int Nedges;      // Nombre d'aretes
  int NedgesOnTriangles; // Nombre d'aretes sur triangles
  Edge * EdgeList;  //Tableau avec les aretes du maillage
  // Donnees des degrees de liberte
  int NDOF;      // Nombre de degrees de liberte
  DOF * DOFList;    // Tableau avec les degrees de liberte
  // Fonctions membres et operateurs
  Mesh(const char * file);  // Constructeur
  //~Mesh() { delete [] ObjectList; delete []NodeList; []TrianList; []
      TetraList; []EdgeList; []DOFList;}    // Destructeur
private:
  Mesh(const Mesh &);
  const Mesh & operator = (const Mesh &);
};
```

The most used geometrical element of the *Mesh* class is its list of edges to which the basis functions and degrees of freedom are associated. In order to build the edge list from the triangle and tetrahedra list contained in the mesh file delivered by Gmsh without incurring in computational cost of $\mathcal{O}(N_E^2)$ we used an algorithm of search over the edges having the same first number of node. This algorithm allows for the construction of edges in a time of the same order of the number of edges, in opposition to a quadratic time over this number ($\mathcal{O}(N_E^2)$) obtained when searching exhaustively over the list of edges whenever a new candidate is tested. Figure 7 shows the required time to build the *Mesh* class by file reading and processing showing and empirical asymptotic law close to $\mathcal{O}(N_T^{1.5})$.

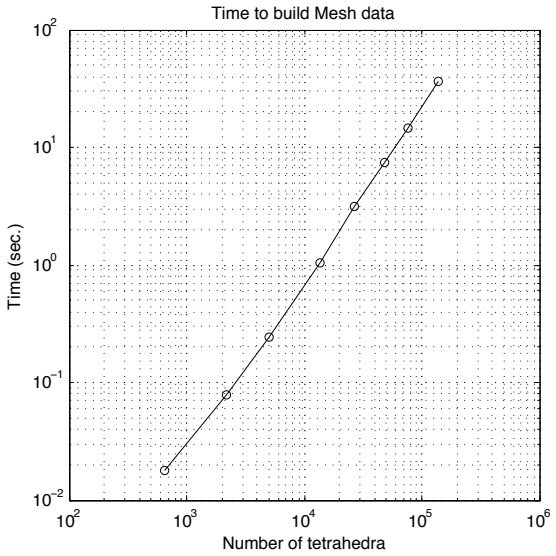


Figure 7: Computational time required to build the *Mesh* class from a Gmsh file and using post-processing as described and obtained for mesh models with increasing number of tetrahedra.

4.2 Construction of the Octree

Many parts and routines of the simulation code require to determine the tetrahedron to which a given point belongs. This procedure is used heavily in the generation of results over a set of points in space (e.g. the values of the electrical field over points in a plane cutting the domain). Since the exhaustive search over the list of tetrahedra for a given point will result in a computational time in the order of the number of points to examine times the number of tetrahedra it is necessary to improve the search algorithm to achieve practical execution times. For this purpose and Octree structure will be constructed for the associated mesh. The Octree structure divides the space consecutively in subdomains so that a given point can be fastly associated with a subdomain. In this way, the search for the tetrahedron to which the point belong is performed only on the list of tetrahedra present in such subdomain.

The *Octree* class contains the structure of the Octree, and it consists of a tree with nodular elements called cells, implemented in the *Cell* class. Each cell belongs to a level of subdivisions of the domain, and on each level the subdivisions form the whole domain. Each cell keeps information of the lower level cell to which it belongs and of the higher level cells that belong to it, allowing for fast determination of the highest level cell to which a given point belongs. Each cell also keeps a list of the tetrahedra present in it. Listings 8 and 9 exhibit the C++ classes used to implement the cell and the Octree structures.

Listing 8: *Cell* Class

```
class Cell {
```

```
public:
  // Membres
  int Octant;
  int Level;
  int Ntetra;
  Tetrahedron ** TetraList;
  double x_max,x_min,y_max,y_min,z_max,z_min;
  int ParentIndex;
  int ChildrenIndex[8];
  // Constructeur
  Cell () : Octant(0), Level(0), Ntetra(0), TetraList(), x_max(0.0), x_min
      (0.0), y_max(0.0), y_min(0.0), z_max(0.0), z_min(0.0), ParentIndex(0),
       ChildrenIndex() {}
};
```

Listing 9: *Octree* Class

```
class Octree {
public:
  // Membres
  int Ncells;
  int Nlevels;
  Cell * CellList;
  double x_max,x_min,y_max,y_min,z_max,z_min;
  double Lx,Ly,Lz;
  // Constructeur
  Octree(const Mesh & Th, bool verbose_octreebuild);
};
```

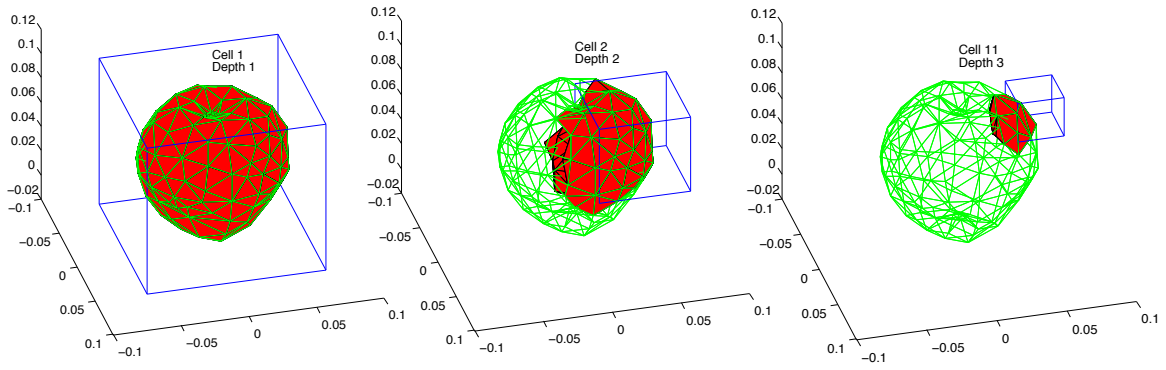


Figure 8: Examples of arbitrary cells with different depths of the Octree structure constructed for the a mesh of the apple model. The tetrahedra belonging to each cell are highlighted in red.

The construction of the Octree is done recursively, first creating a mother cell that contains the whole domain and thus all the tetrahedra. This mother cell is divided in eight octants, each of which will be a new higher level mother cell to subsequent cells of even higher level that

will be children. Figure 9 shows the required computation time required to build the *Octree* class from the mesh structure, which has an asymptotic performance of $\mathcal{O}(N_T)$.

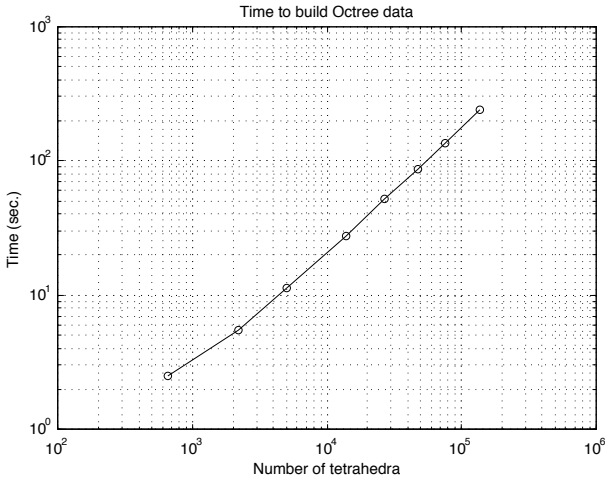


Figure 9: Computational time required to build the *Octree* class from the mesh structure for an increasing number of tetrahedra in the model.

Using the Octree structure an evaluation of the solution field based on tetrahedral information over a given number of points is performed in short times. Figure 10 shows the computational time required to build such information over a plane cutting the domain with 6400 sampling points for meshes with an increasing number of tetrahedra.

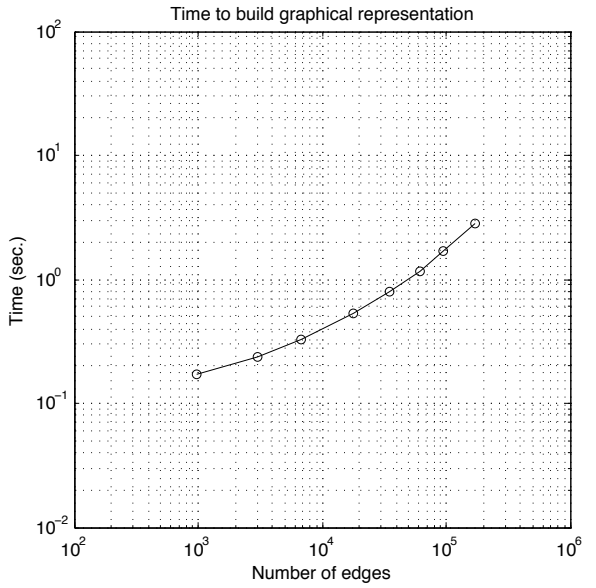


Figure 10: Computational time required to build such information over a plane cutting the domain with 6400 sampling points for meshes with an increasing number of tetrahedra.

4.3 Elemental Matrices

The assembly of the system matrix needs the computation of the terms

$$\int_{S_i \cap S_j} curl N_i \cdot curl N_j dx \quad \text{and} \quad \int_{S_i \cap S_j} N_i \cdot N_j dx. \tag{36}$$

The support $S_i \cap S_j$ can be either the set of all tetrahedra sharing edge e_i if $i = j$, a tetrahedra common to e_i and e_j or void. In either case the integral over $S_i \cap S_j$ can be decomposed in contributions from integrals over the tetrahedra of the mesh. For this reason the core of the matrix assembling routine is the determination, for each tetrahedron, of the integration of the interaction of its local edges i (of extreme nodes i_1 and i_2) and j (of extreme nodes j_1 and j_2), with the convention given by Figure 6, which will be represented by local 6x6 matrices E^k and F^k for each tetrahedron T_k:

$$\mathrm{E}^k_{ij} = \int_{T_k} \mathrm{curl}\, N_i \cdot \mathrm{curl}\, N_j dx \tag{37}$$

$$\mathrm{F}^k_{ij} = \int_{T_k} N_i \cdot N_j dx \tag{38}$$

From (31) in Definition 2

$$\begin{aligned} curl N_i &= l_i \left(\mathrm{curl}(\lambda_{i_1} \nabla \lambda_{i_2}) - \mathrm{curl}(\lambda_{i_2} \nabla \lambda_{i_1})\right) \\ &= l_i \left(\lambda_{i_1} \mathrm{curl}(\nabla \lambda_{i_2}) + \nabla \lambda_{i_1} \times \nabla \lambda_{i_2} - \lambda_{i_2} \mathrm{curl}(\nabla \lambda_{i_1}) - \nabla \lambda_{i_2} \times \nabla \lambda_{i_1}\right) \\ &= 2 l_i \nabla \lambda_{i_1} \times \nabla \lambda_{i_2} \end{aligned}$$

which, expanding the expression of the barycentric coordinates, can be written as

$$curl N_i = 2 l_i \begin{pmatrix} c_{i_1} d_{i_2} - d_{i_1} c_{i_2} \\ d_{i_1} b_{i_2} - b_{i_1} d_{i_2} \\ b_{i_1} c_{i_2} - c_{i_1} b_{i_2} \end{pmatrix}. \tag{39}$$

From the previous expression it is easy to check that

$$E^k_{ij} = 4 l_i l_j V_k \begin{pmatrix} c_{i_1} d_{i_2} - d_{i_1} c_{i_2} \\ d_{i_1} b_{i_2} - b_{i_1} d_{i_2} \\ b_{i_1} c_{i_2} - c_{i_1} b_{i_2} \end{pmatrix} \cdot \begin{pmatrix} c_{j_1} d_{j_2} - d_{j_1} c_{j_2} \\ d_{j_1} b_{j_2} - b_{j_1} d_{j_2} \\ b_{j_1} c_{j_2} - c_{j_1} b_{j_2} \end{pmatrix}. \tag{40}$$

We consider now terms associated to F^k_{ij}:

$$N_i \cdot N_j = l_i l_j \left(\lambda_{i_1} \lambda_{j_1} f_{i_2 j_2} - \lambda_{i_1} \lambda_{j_2} f_{i_2 j_1} - \lambda_{i_2} \lambda_{j_1} f_{i_1 j_2} + \lambda_{i_2} \lambda_{j_2} f_{i_1 j_1}\right), \tag{41}$$

where $f_{mn} = b_m b_n + c_m c_n + d_m d_n$. It follows from this that

$$F_{ij}^k = l_i l_j \left(f_{i_2 j_2} \int_{T_k} \lambda_{i_1} \lambda_{j_1} dx - f_{i_2 j_1} \int_{T_k} \lambda_{i_2} \lambda_{j_1} dx - f_{i_1 j_2} \int_{T_k} \lambda_{i21} \lambda_{j_1} dx + f_{i_1 j_2} \int_{T_k} \lambda_{i_2} \lambda_{j_2} dx \right) \quad (42)$$

All the terms involved in E_{ij}^k and F_{ij}^k can be easily computed knowing the barycentric coordinates of each vertex of the tetrahedron T_k.

4.4 Assembly of the System's Matrix

The fastest way to assemble matrix of the electromagnetic model is with a loop over each tetrahedron computing the contributions to each element of the matrix according to whether the local edges of the tetrahedron are degrees of freedom of the model.

for t from 1 to N_T **do**
  Compute E^k and F^k.
  Recover local values of μ and ε.
  for local edge i from 1 to 6 **do**
    for local edge j from 1 to 6 **do**
      Recover the global edge I corresponding local edge i.
      Recover the global edge J corresponding local edge j.
      if global edges I and J are degrees of freedom **then**
        $Sign_i$ = orientation local edge i.
        Recover d.o.f. m of edge I.
        $Sign_j$ = orientation local edge j.
        Recover d.o.f. n of edge J.
        $Z_{m,n} = Z_{m,n} + Sign_i * Sign_j * ((\omega\mu)^{-1} E_{i,j}^k - (\omega\varepsilon) F_{i,j}^k)$.
      end if
    end for
  end for
end for

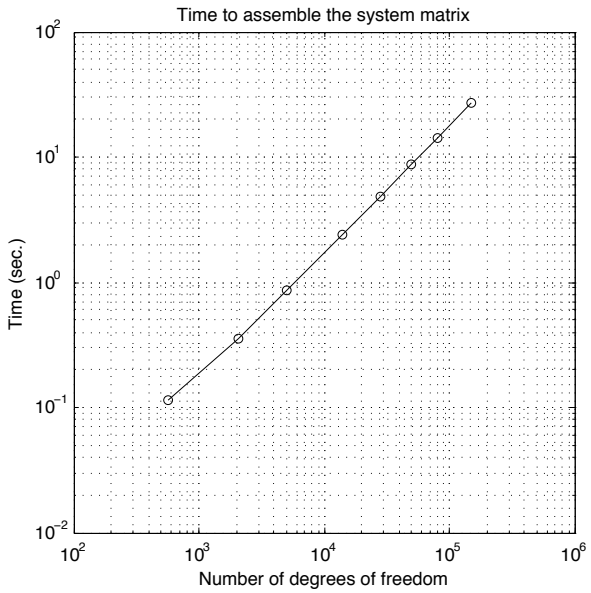


Figure 11: Computational time required to assemble the matrix of the electromagnetic finite element model for different numbers of degrees of freedom.

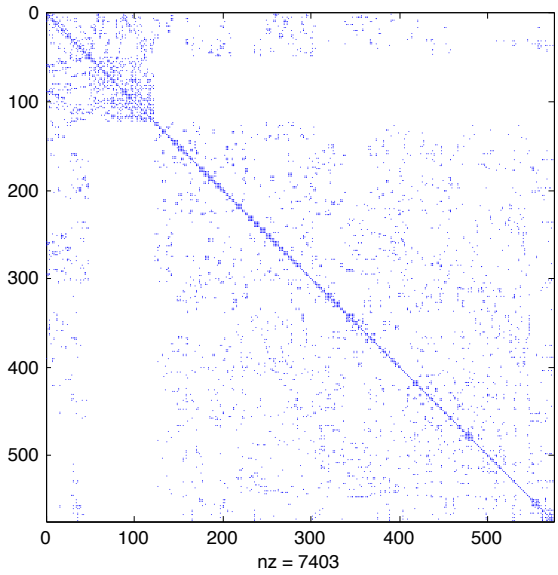


Figure 12: Structure of the system matrix assembled for a model with 573 degrees of freedom and 7403 non-zero elements.

4.5 Resolution of the Linear System

Once the system matrix is assembled and the excitation vector has been computed for the specific case to be simulated the linear system is solved using UMFPACK, an unsymmetric

multi-frontal LU solver for sparse matrices. UMFPACK uses multi-frontal renumbering parallel sparse factorization of the system matrix using BLAS algorithms. For the present case, the BLAS library and the UMFPACK library have been built for a 2.4GHz Intel Core 2 Duo 64 bits architecture. Figure 13 shows the required computational time to solve the linear system for different numbers of degrees of freedom arising from different models with different meshes.

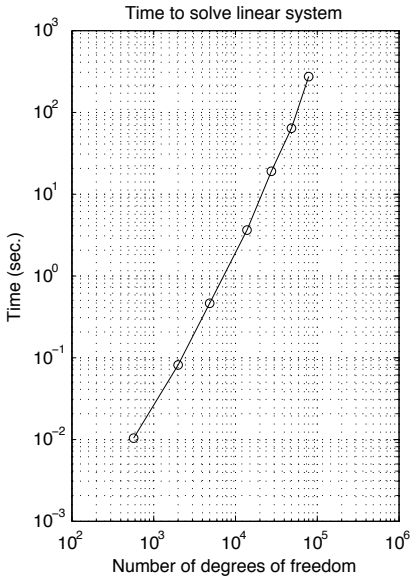


Figure 13: Computational time required to solve the linear system using the **UMFPACK** library for systems with different number of degrees of freedom.

A secondary, less efficient, method has also ben tried for the sake of comparison. Figure 14 and Figure 15 show the required computational time and the required number of iterations to solve the system using the Conjugate Gradient Method for different numbers of degrees of freedom.

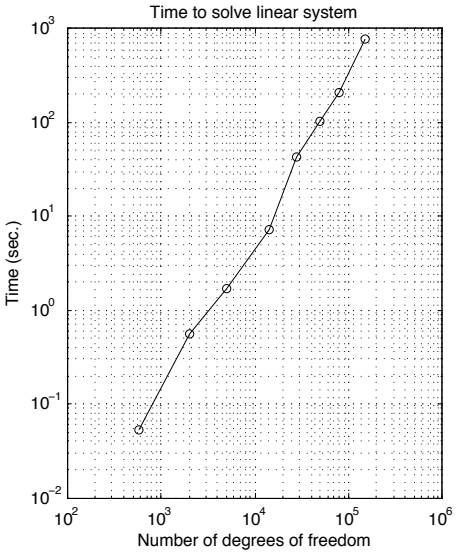


Figure 14: Computational time required to solve the linear system using the **Conjugate Gradient Method** for systems with different number of degrees of freedom.

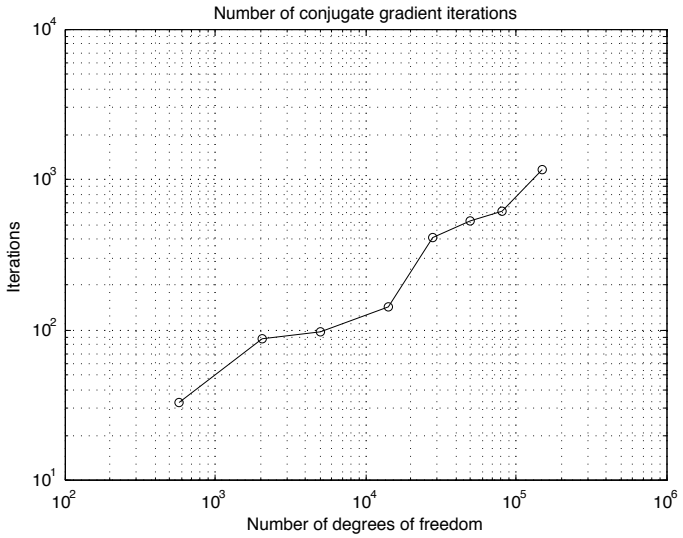


Figure 15: Number of iterations required to solve the linear system using the **Conjugate Gradient Method** for systems with different number of degrees of freedom.

5 Validation

The purpose of this section is to asses the capability of the developed algorithm to approach the exact solution of the Maxwell cavity problem. In order to determine and measure this capability a numerically approximated solution is compared to the exact solution for a case for which the latter is known. Using the previously exhibited physical modeling of the phenomenon, variational formulations and numerical algorithms, it will be shown that the approximated solution converges to the exact solution in the variational space where the problem is defined.

5.1 Analytical Solution for an Infinite Waveguide

The chosen case of consideration, for which an exact solution is known, is that of an infinite waveguide of rectangular section. Let us consider such a waveguide of rectangular section of dimension a along de X axis, b along the Y axis, and extending infinitely along the Z axis.

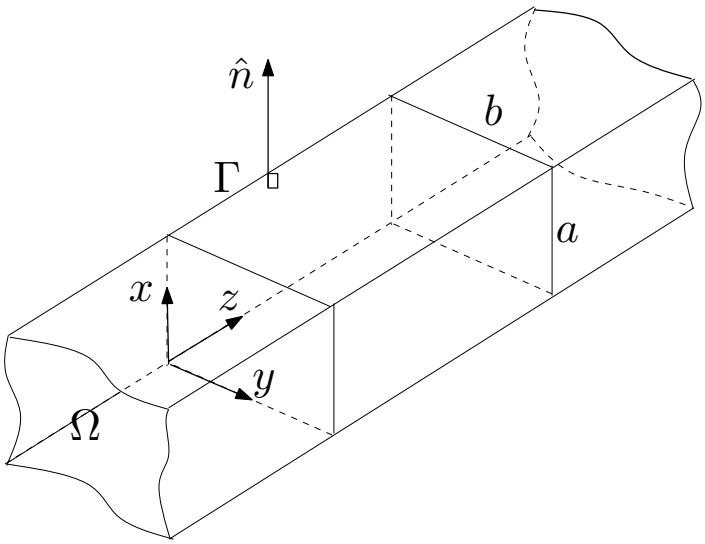


Figure 16: Diagram of the domain of the problem used for validation, exhibiting the waveguide Ω of boundary Γ.

In the case where the interior of the waveguide is filled with air the electric pemittivity and magnetic permeability are those of the vacuum. Additionally, no electric currents or charges will be found on the inside. If the walls are considered to be perfectly conducting surfaces, it follows from the Maxwell equations (1), (2), (3) and (4), and from the boundary conditions (8) that the electric field inside a waveguide Ω of boundary Γ is given by

$$\begin{cases} \Delta E = \frac{1}{c_0^2}\frac{\partial^2}{\partial t^2}E \quad in\ \Omega, \\ \\ \hat{n} \wedge E = 0 \quad on\ \Gamma, \end{cases} \tag{43}$$

where $c_0 = 1/\sqrt{\epsilon_0 \mu_0}$ is the speed of light in the vacuum. It can be proved that each

component of the solution electric field can be written as a factor of functions of a single argument, allowing for the utilization of the method of separation of variables [6]. In this manner the X component E_x, for example, can be written as

$$E_x(x,y,z,t) = E_{x_1}(x)E_{x_2}(y)E_{x_3}(z)E_{x_4}(t) \tag{44}$$

Replacing in the equation, separating the term of independent variables and setting constants $-k_{11}^2$, $-k_{21}^2$, $-k_{31}^2$ and $-k^2$, a set of independent ordinary differential equations is obtained:

$$\begin{aligned} \tfrac{\partial^2}{\partial x^2}E_{x_1} &= -k_{11}^2 E_{x_1}, \quad \tfrac{\partial^2}{\partial y^2}E_{x_2} = -k_{21}^2 E_{x_2}, \\ \tfrac{\partial^2}{\partial z^2}E_{x_3} &= -k_{31}^2 E_{x_3}, \quad \tfrac{\partial^2}{\partial t^2}E_{x_4} = -k^2 E_{x_4}. \end{aligned} \tag{45}$$

As usual, the pulsation is defined as $\omega = 2\pi f = c_0 k$. It is also remarked that for the partial differential equation to hold the constants comply with $k_{11}^2 + k_{21}^2 + k_{31}^2 = k^2$. The solutions of the ordinary differential equations in the bounded directions X and Y are

$$\begin{aligned} E_{x_1} &= A_{11}\cos(k_{11}x) + B_{11}\sin(k_{11}x), \\ E_{x_2} &= A_{21}\cos(k_{21}y) + B_{21}\sin(k_{21}y). \end{aligned} \tag{46}$$

To express wave propagation in the positive Z sense we take $E_{x_3}(z)E_{x_4}(t) = Re\left\{E_{0_x}e^{-i(k_{31}z-\omega t)}\right\}$, thus yielding the unconstrained solution

$$E_x(x,y,z,t) = E_{0_x}(A_{11}\cos(k_{11}x)+B_{11}\sin(k_{11}x))(A_{21}\cos(k_{21}y)+B_{21}\sin(k_{21}y))\,e^{-i(k_{31}z-\omega t)}. \tag{47}$$

Similarly, we obtain

$$E_y(x,y,z,t) = E_{0_x}(A_{12}\cos(k_{12}x)+B_{12}\sin(k_{12}x))(A_{22}\cos(k_{22}y)+B_{21}\sin(k_{22}y))\,e^{-i(k_{32}z-\omega t)}, \tag{48}$$

$$E_z(x,y,z,t) = E_{0_x}(A_{13}\cos(k_{13}x)+B_{13}\sin(k_{13}x))(A_{23}\cos(k_{23}y)+B_{23}\sin(k_{23}y))\,e^{-i(k_{33}z-\omega t)}. \tag{49}$$

Imposing boundary conditions on $x = y = 0$ yields $A_{21} = A_{12} = A_{13} = A_{23} = 0$. Furthermore, imposing $\operatorname{div} E = 0$ yields $B_{11} = B_{22} = 0$ and

$$\begin{aligned} k_{11} = k_{12} = k_{13} &:= k_1, \\ k_{21} = k_{22} = k_{23} &:= k_2, \\ k_{31} = k_{32} = k_{33} &:= k_3. \end{aligned} \tag{50}$$

Imposing the boundary conditions at $x = a$ and $y = b$ gives $k_1 = m\pi/a$ and $k_2 = m\pi/b$ for $m, n \in \mathbb{Z}$.

Gathering the remaining multiplicative constants per component, the solution electric field is written as

$$E = \begin{pmatrix} E_{0_x} \cos(k_1 x) \sin(k_2 y) e^{-i(k_3 z - \omega t)} \\ E_{0_y} \sin(k_1 x) \cos(k_2 y) e^{-i(k_3 z - \omega t)} \\ E_{0_z} \sin(k_1 x) \sin(k_2 y) e^{-i(k_3 z - \omega t)} \end{pmatrix}. \tag{51}$$

The Faraday-Maxwell equation (2) gives the magnetic field

$$B = \frac{i}{\omega} \begin{pmatrix} (k_2 E_{0_z} + i k_3 E_{0_y}) \sin(k_1 x) \cos(k_2 y) e^{-i(k_3 z - \omega t)} \\ -(k_1 E_{0_z} + i k_3 E_{0_x}) \cos(k_1 x) \sin(k_2 y) e^{-i(k_3 z - \omega t)} \\ (k_1 E_{0_y} - i k_2 E_{0_x}) \cos(k_1 x) \cos(k_2 y) e^{-i(k_3 z - \omega t)} \end{pmatrix}. \tag{52}$$

In order to compute a numerical approximation particular values should be chosen. For this validation case the TE_{01} mode will be taken into account, resulting (using $m = 0$ and $n = 1$ and $E_{0_x} = 1$) in

$$E = \begin{pmatrix} \sin(\pi y/b) e^{-i(k_3 z - \omega t)} \\ 0 \\ 0 \end{pmatrix}, \tag{53}$$

and

$$B = \frac{1}{\omega} \begin{pmatrix} 0 \\ k_3 \sin(\pi y/b) e^{-i(k_3 z - \omega t)} \\ -i(\pi/b) \cos(\pi y/b) e^{-i(k_3 z - \omega t)} \end{pmatrix}. \tag{54}$$

For a given frequency f the pulsation is computed as $\omega = 2\pi f$, the wavenumber as $k = \omega/c_0$ and k_3 as $k_3 = -\sqrt{k^2 - (\pi/b)^2}$. It is noteworthy that the waveguide has a cutoff frequency where $k = \pi/b$, that is $f = c_0/2b$, below which propagation is attenuated exponentially along Z positive.

As means of validation of the proposed finite element method we present the concrete case of a waveguide section with parameters detailed in Table 1. Two frequencies will be used, one above and one below the cut-off frequency of the waveguide. Figure 17 and Figure 18 show respectively the exact solution over two planes cutting the waveguide.

Table 1: Physical dimensions used in the interpolation of the exact solution to the waveguide problem.

Physical Parameter	Value
X-Width a	0.1m
Y-Width b	0.1m
Section Length	0.4m
Cut-Off Frequency	1.5GHz
Frequencies Used	1GHz, 2GHz
Mode	TE_{01}
Permittivity ε	ϵ_0
Permeability μ	μ_0

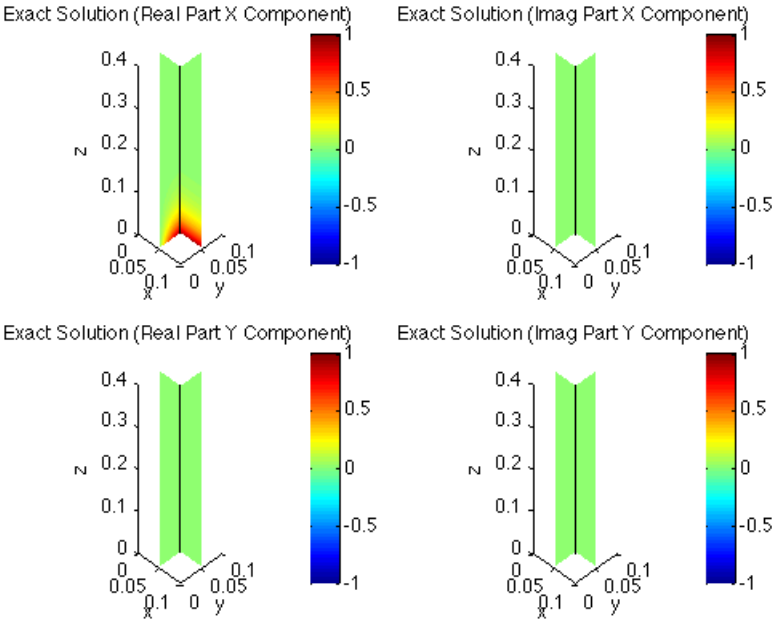


Figure 17: **Exact electrical field strength** in V/m on the waveguide section with **1GHz** frequency using the parameters detailed in Table 1.

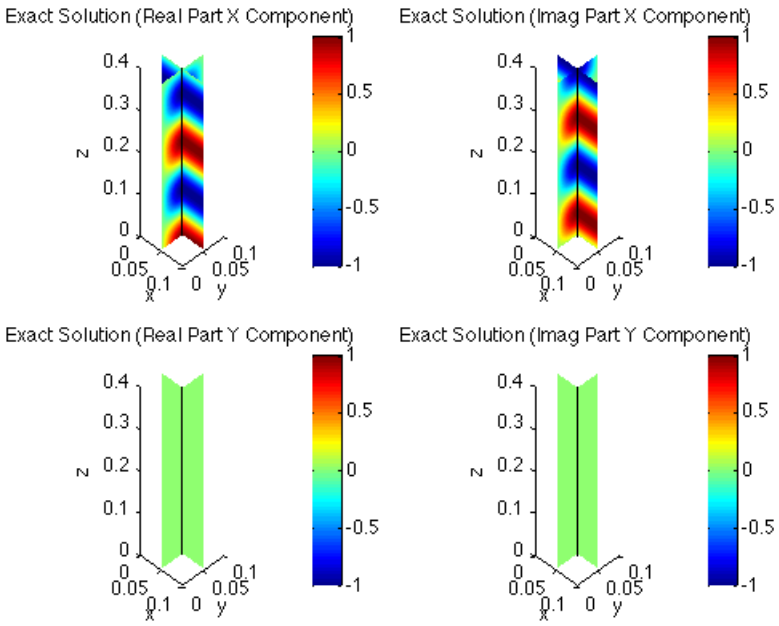


Figure 18: **Exact electrical field strength** in V/m on the waveguide section with **2GHz** frequency using the parameters detailed in Table 1.

5.2 Finite Element Interpolation

Figure 19 and Figure 20 show the interpolated electric field strength in the waveguide using the physical parameters detailed on Table 1. The parameters of the mesh used for the interpolation are described in Table 2

Table 2: Parameters of the mesh of the waveguide section used in the interpolation and approximation results shown in figures.

Parameter	Value
N. of Nodes	2334
N. Tetrahedra	12831
N. of Edges	15732
N. of D.O.F.	14236
Edge Mean Length	0.9cm

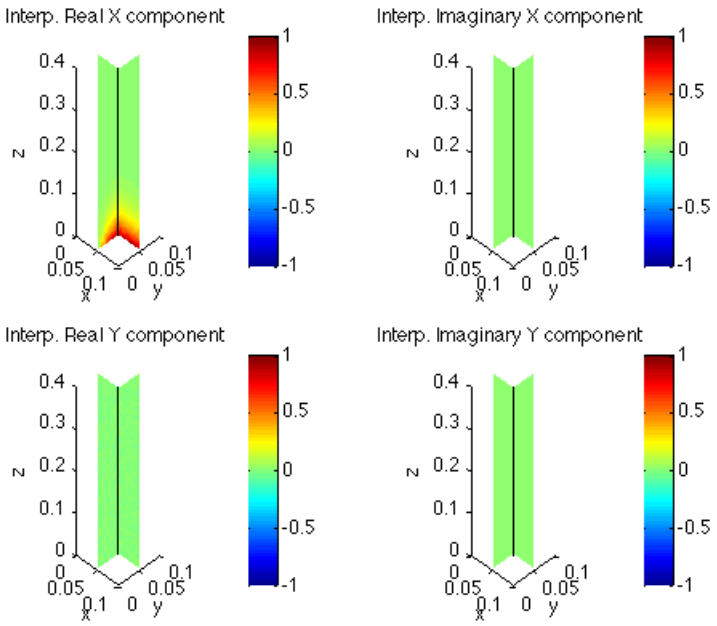


Figure 19: **Interpolated electrical field strength** in V/m on the waveguide section with **1GHz** frequency using the parameters detailed in Table 1 and Table 2.

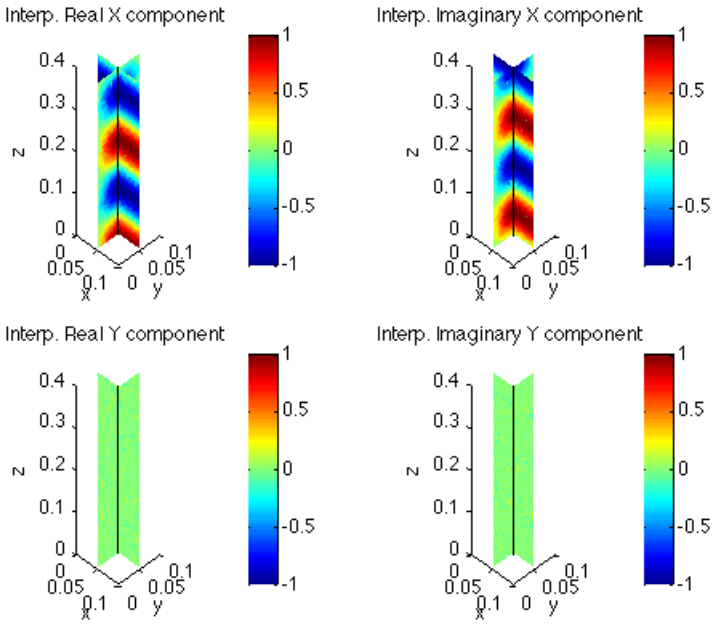


Figure 20: **Interpolated electrical field strength** in V/m on the waveguide section with **2GHz** frequency using the parameters detailed in Table 1 and Table 2.

5.3 Finite Element Approximation

Using the exact electric and magnetic fields of the TE_{01} chosen in for the developed waveguide example in this section the finite element approximation can be obtained using the boundary-valued variational formulation (P_w^c) in (30). Figures 21 and Figure 22 show the approximated electric field for 1GHz and 2GHz frequency on the waveguide using the described Finite Element Method.

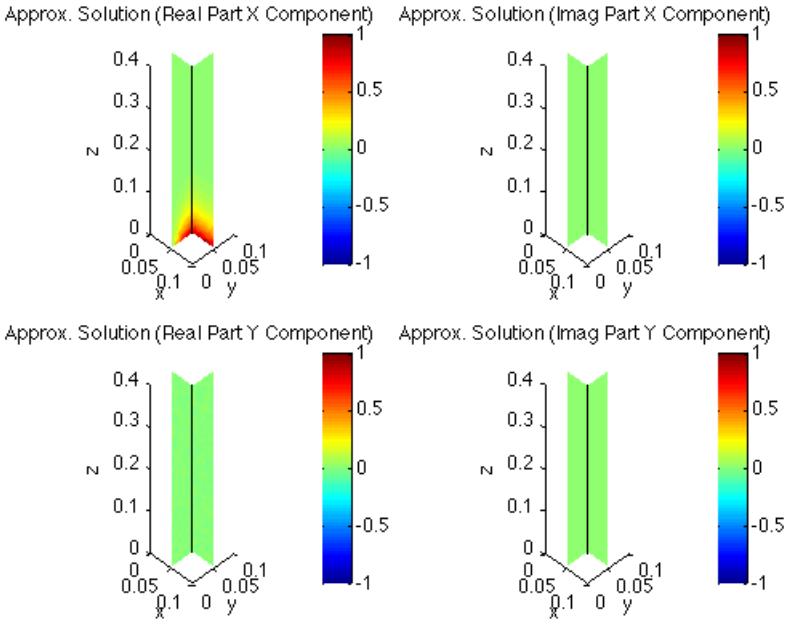


Figure 21: **Approximated electrical field strength** in V/m on the waveguide section with **1GHz** frequency using the parameters detailed in Table 1 and Table 2.

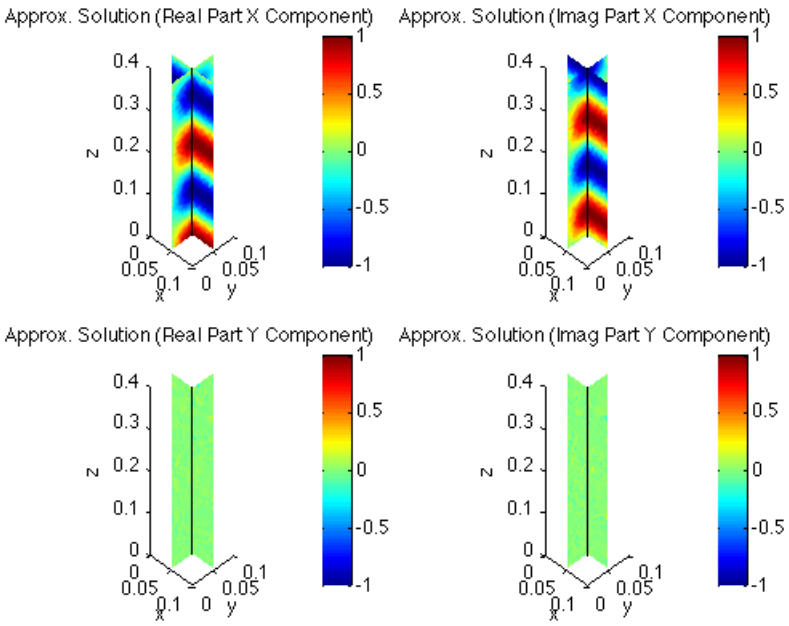


Figure 22: **Approximated electrical field strength** in V/m on the waveguide section with **2GHz** frequency using the parameters detailed in Table 1 and Table 2.

Integrating $E - E_h$ over the waveguide section the L^2 error norm is computed. Figure 23 shows the error convergence for different edge lengths.

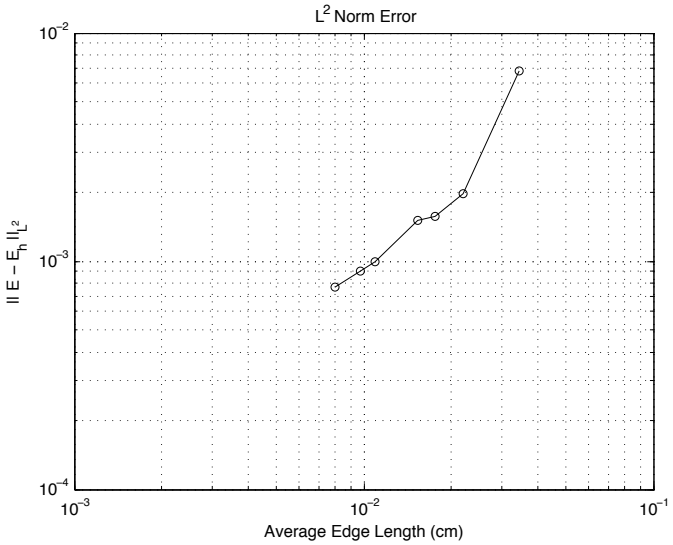


Figure 23: Finite Element Method L^2 approximation error using the parameters detailed in Table 1 and Table 2.

6 Applications

6.1 Microwave Oven

A classical and useful application to the Maxwell's equations in cavities is that of the simulation of microwave heating. Microwave heating is produced by the dissipation of electromagnetic energy in the form of heat in lossy media. The electromagnetic energy loss is composed by conductivity losses (associated with the presence of non-zero electrical conductivity σ) and with dielectric losses. For time-harmonic incident electromagnetic fields both losses are represented indistinguishably by the complex electrical permittivity $\varepsilon = \varepsilon' + i\varepsilon''$. The existence of a non-zero imaginary permittivity component ε'' in a region of the domain causes dissipation of the incident electromagnetic wave E in this region, which is converted to heat with a rate equal to

$$P = \omega|\varepsilon''|E \cdot \overline{E}, \tag{55}$$

in units of Watts [8]. Combining the described method for the solution of the electromagnetic wave in a cavity together with a heat diffusion model the microwave heating can be simulated.

Given a heat diffusivity γ in square meters per second, a heat capacity c in Joules per Celsius degrees, an initial temperature u_0 and a temperature u_{ext} in the exterior of a region to be heated, the evolution of the temperature u in the interior (up to a time T_{max}) can be modeled as the solution to the strong problem (P_s^u):

$$(P_s^u) \begin{cases} \textit{Given } \gamma, c \in \mathrm{C}(\Omega), \gamma, c > 0, P \in \mathrm{C}((0,T_{max}) \times \Omega), \textit{ find } u \in \mathrm{C}^2((0,T_{max}) \times \Omega) \text{ such that} \\ \qquad \frac{\partial}{\partial t}u - \gamma \Delta u = \frac{1}{c}P \quad \textit{in } (0,T_{max}) \times \Omega \\ \qquad u = u_{ext} \qquad \textit{on } (0,T_{max}) \times \Gamma \\ \qquad u = u_0 \qquad \textit{at } t = 0 \end{cases} \tag{56}$$

Since the partial differential equation involves only derivatives, and assuming the coefficient γ does not depend on the temperature, the exterior and initial temperatures can be set to a zero reference for convenience: $u_{ext} = u_0 = 0$. Using a variational formulation in space and a first order backwards finite difference approximation in time (with time step equal to Δt) the problem can be rewritten as

$$(P_w^u) \begin{cases} \textit{Given } \gamma, c \in \mathrm{L}^\infty(\Omega), \gamma, c > 0, P \in \mathrm{L}^2(\Omega), u^n \in \mathrm{H}_0^1(\Omega), \textit{ find } u^{n+1} \in \mathrm{H}_0^1(\Omega) \text{ such that} \\ \qquad \forall v \in \mathrm{H}_0^1(\Omega) \quad \int\limits_\Omega u^{n+1} v dx + \gamma \Delta t \int\limits_\Omega \nabla u^{n+1} \cdot \nabla v dx = \int\limits_\Omega \left(\frac{\Delta t}{c} P + u^n \right) v dx \end{cases} \tag{57}$$

Using a P1 piecewise approximation for the variational formulation spanned by ϕ_i piecewise linear functions on tetrahedra associated to the N_I internal vertices of the object to be heated,

and considering a constant diffusive coefficient $\gamma > 0$, the problem can be written as

$$(P_d^u) \begin{cases} \textit{Given } P \textit{ and } u^n \textit{ piecewise continuous, find } \alpha^{n+1} \in \left\{\alpha_1^{n+1}, .., \alpha_{N_I}^{n+1}\right\} \text{ such that } \forall \phi_i \\ \\ \displaystyle\sum_{j=1}^{NI} \alpha_j^{n+1} \int_\Omega \left(\phi_i \phi_j + \gamma \Delta t \nabla \phi_i \cdot \nabla \phi_j\right) dx = \int_\Omega \left(\frac{\Delta t}{c} P + u^n\right) \phi_i dx \end{cases} \tag{58}$$

This discrete variational formulation can be easily implemented as an iterative matrix problem to perform simulations.

Example values for the parameters of the electromagnetic model can be found abundantly. At microwave oven frequencies, the complex electric permittivity $\varepsilon = \varepsilon' + i\varepsilon''$ of fresh fruits is typically contained in the following ranges: $\varepsilon' \in [70\epsilon_0, 80\epsilon_0]$, $\varepsilon'' \in [-12\epsilon_0, -11\epsilon_0]$ [9]. Examples to the parameters of the heat diffusion model can also be found: $\alpha \in [1.4 \cdot 10^{-7}, 1.5 \cdot 10^{-7}]$ square meters per second, $c \in [3000, 5000]$ Joules per Celsius degree [10]. A common nominal frequency for modern microwave ovens is 2.45GHz.

Table 3: Computational microwave model used in simulations.

Parameter Name	Value
Model Volume	0.03705m^3
Number of Nodes	16139
Number of Tetrahedra	73565
Number of Edges	98356
Degrees of Freedom	72397
Mean Edge Length	1.6cm

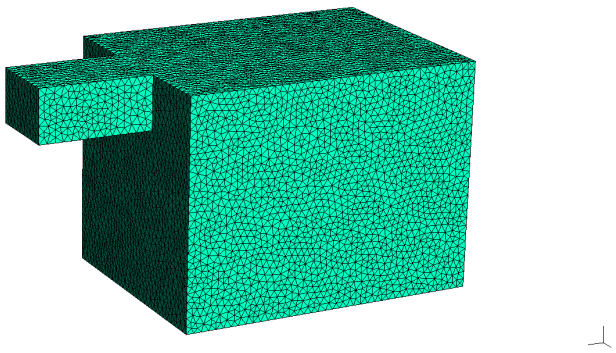


Figure 24: Meshed microwave model used in computational examples of dielectric heating.

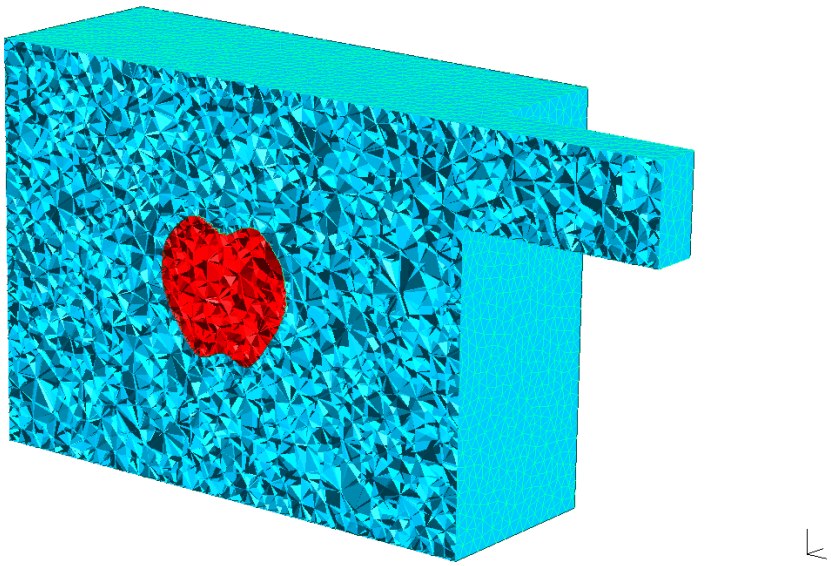


Figure 25: Internal cut of the meshes microwave model used in computational examples of dielectric heating showing an apple inside.

For the following simulations precise physical values will be chosen in the range detailed above for practical microwave heating applications. Table 4 specifies these parameters.

Table 4: Physical parameters of the microwave model.

Parameter Name	Value
Apple's E. Permittivity	$(70 - 11i)\epsilon_0$
Apple's Heat Diffusivity	$1,5 \cdot 10^{-7} m^2/s$
Apple's Heat Capacity	5000 J/C^o
Frequency	$2.45GHz$
Source	Dipole: $2.5mA$

Figure 26 shows the real and imaginary parts for each space component of the electric field inside the microwave oven. Figure 27 shows field intensity inside the oven. For the given color scale, the field strength and field intensity is several orders of magnitude smaller inside the apple, which is explained by the high conductivity and permittivity resulting in high reflective on the interphase. However, the field is not zero in the interior of the apple as it can be deduced by the evolution of the temperature. The wavefront is in fact penetrating the fruit and rapidly decaying in amplitude, heating the outer part by dissipation of the electromagnetic power.

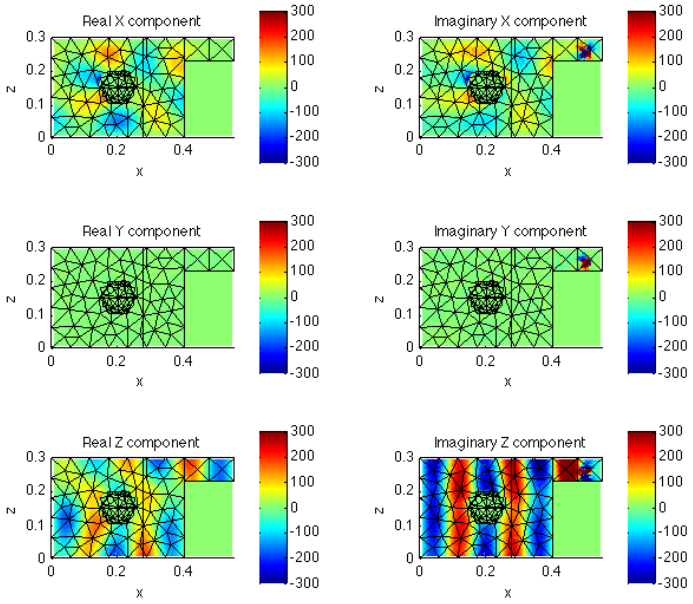


Figure 26: Electric field strength E in V/m inside the microwave oven using parameters of Tables 3 and 4.

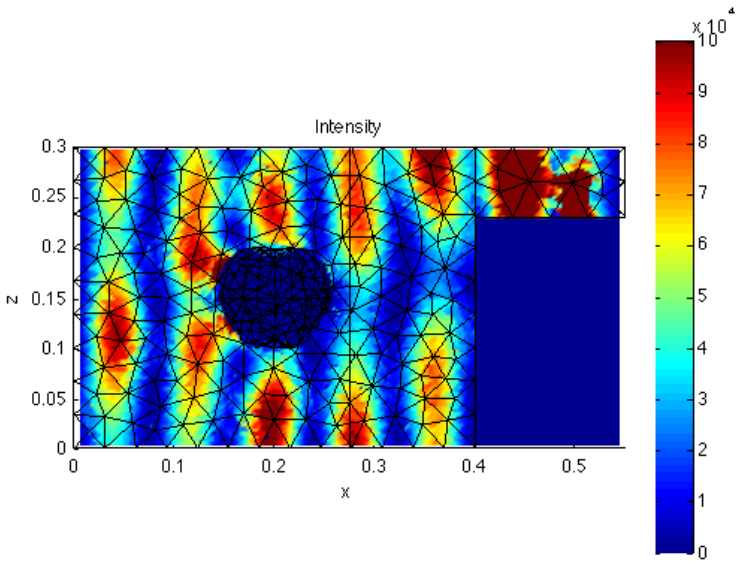
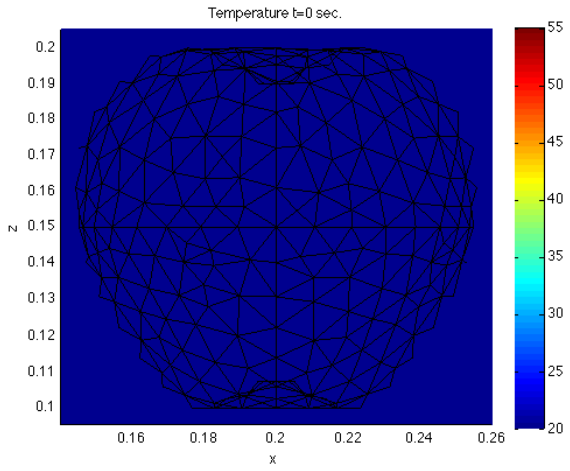
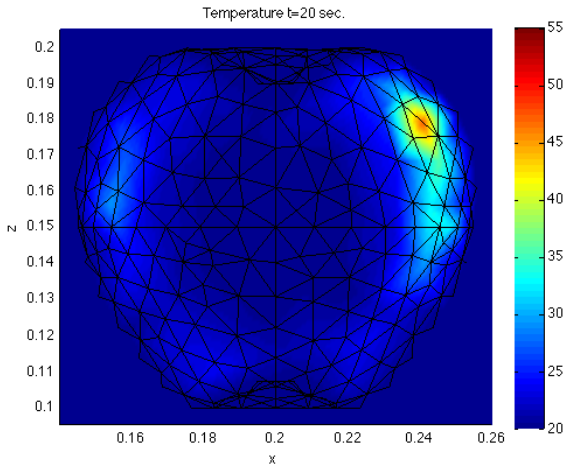


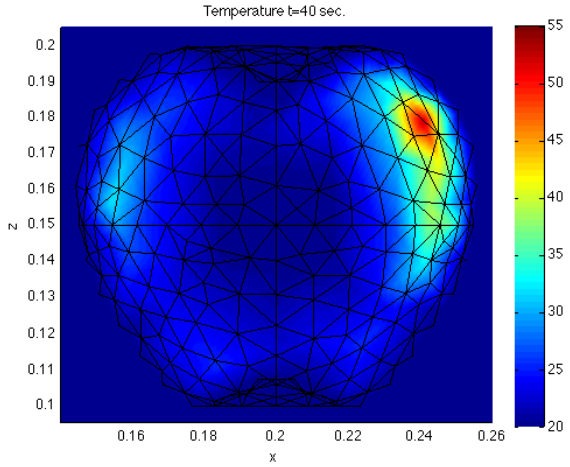
Figure 27: Electric field intensity $E \cdot \overline{E}$ inside the microwave oven using parameters of Tables 3 and 4.



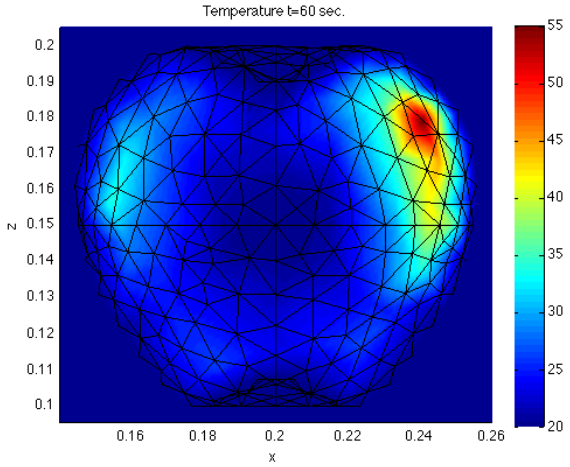
(a)



(b)



(c)



(d)

Figure 27: Evolution of the temperature in the apple for an incoming EM wave front using parameters of Tables 3 and 4. The figure shows from top left to bottom right the temperature in a cut of the apple at t = 0 seconds (a), t = 20 seconds (b), t = 40 seconds (c), and t = 60 seconds (d).

References

[1] A. Bossavit, *electromagnetism, en vue de la modelisation*, SMAI Springer-Verlag, 1993.

[2] J-C. Nedelec, *Mixed Finite Elements in R3*, Numerische Mathematik, 35, p. 315-341 1980.

[3] C. Geuzaine, J.-F Remacle, *Gmsh: a three-dimentional finite element mesh generator with buit-in pre- and post-processing facilities*, International Journal for Numerical Methods in Engineering, 79(11), p. 1309-1331, 2009.

[4] J. Jin, *The Finite Element Method in Electromagnetics*, Second Edition, John Wiley & Sons 2002.

[5] F. Hecht, *Simulation Numerique en C++*, DUNOD Sciences Sup 2003.

[6] M. Schwartz, *Principles of Electrodynamics*, Dover 1972.

[7] C. Bernardi,Y. Maday, F. Rapetti, *Discretisations Variationnelles de problemes aux limites elliptiques*, SMAI Springer-Verlag 2004.

[8] A.C Metaxas,R.J. Meredith, *Industrial Microwave Heating*, IEE, Power Engineering Series 4, 1983.

[9] A. Kaewrawang,S. Swatdiponphallop, A. Siritaratiwat, *Study on Complex Permittivity of Tropical Thai Fruits*, Journal of Sciences 7 (7) p. 1009-1012, 2007.

[10] O.J. Ikegwu,F.C. Ekwu, *Thermal and Physical Properties of Some Tropical Fruits and Their Juices in Nigeria*, Journal of Food Technology 7 (7) p. 38-42, 2009.