

**MAXWEL: FEM software module to compute EM fields
within a magnetically confined fusion reactor.**

Master Thesis
presented by

Hernán Domingo Ramos

Thesis Promoters

Prof. Geert Verdoolaege
Ghent University

Prof. Mervi Johanna Mantsinen
Barcelona Supercomputing Center

Thesis Supervisors

Dr. Alejandro Soba Pascual
Dr. Daniel Gallart Escolà
Barcelona Supercomputing Center

August 25, 2024

**MAXWEL: FEM software module to compute EM fields
within a magnetically confined fusion reactor.**

Master Thesis
presented by

Hernán Domingo Ramos

Thesis Promoters

Prof. Geert Verdoolaege
Ghent University

Prof. Mervi Johanna Mantsinen
Barcelona Supercomputing Center

Thesis Supervisors

Dr. Alejandro Soba Pascual

Dr. Daniel Gallart Escolà
Barcelona Supercomputing Center

Erasmus Mundus Program on Nuclear Fusion Science and
Engineering Physics

August 25, 2024

Abstract

The analysis of electromagnetic (EM) wave propagation in an inhomogeneous and magnetically confined plasma is a relevant topic in nuclear fusion, especially for ion and electron heating. In order to understand whether an EM wave is able to heat ions and/or electrons, the accessibility of the wave needs to be assessed. In this thesis, the MAXWEL code has been developed, a finite element code to simulate the EM wave propagation in an anisotropic confined plasma. In particular, MAXWEL computes the EM fields in a bi-dimensional geometry, solving the Helmholtz equation in the frequency domain. The solver has been equipped with linear and quadratic elements and dense meshes to optimally model the studied domains with good computational efficiency. A number of different dielectric media and computational domains, including that of a tokamak, have been modelled and benchmarked to show the reliability and robustness of the code. A first approach to model a plasma by means of the cold plasma permittivity tensor has also been attempted yielding excellent agreement with the well-established EM code ERMES. Several plasma scenarios and relevant cutoffs in tokamaks, such as the R and L cutoffs, have been analysed, providing good agreement with theoretical models. Future work will be carried out in order to include a hot permittivity tensor to account for wave and wave-particle resonances. This code represents another step towards a full reactor integrated modelling effort using the high-performance computing framework ALYA.

Contents

1	Introduction	5
1.1	Nuclear Fusion	6
1.2	Plasma description	7
1.3	Magnetically confined fusion	8
1.3.1	Plasma heating	9
1.4	Numerical Modelling	10
1.5	Alya and Marenostum 5	11
2	Electromagnetic theory	13
2.1	Cold Plasma Model	15
2.2	Wave propagation theory	17
2.2.1	Ordinary Wave	18
2.2.2	Extraordinary Wave	18
3	Finite Elements Method	21
3.1	General framework	21
3.2	Matrix assembly	23
3.3	Element categories	25
3.3.1	Linear Triangular element	26
3.3.2	Quadratic Triangular Elements	28
3.3.3	Bilinear Quadrilateral Elements	29
3.3.4	Quadratic Quadrilateral Element	30
3.4	Weak form in electromagnetism	30
3.5	Solving the linear system	32
3.5.1	Conjugate Gradient Method	32
3.5.2	Bi-conjugate Gradient Method	33
4	Code Development and Validation	35
4.1	Code description and functioning	35
4.1.1	Definition and input values	35
4.1.2	Handling the mesh	36
4.1.3	Building the linear system	37
4.1.4	Post-processing the solution	38
4.2	Results and Validation	38
4.2.1	Validations against literature problems	39
4.2.2	ERMES	40
4.2.3	Tokamak geometry	42

5	Conclusions and Future Work	45
A	Other cases and figures generated	53
B	Different geometries	57
C	In-plane fields	59

Chapter 1

Introduction

Energy production is one of the main pillars for the thrive of humankind. Throughout history, humans have developed ingenious methods to produce, gather, and store energy; which altogether resulted in a high impact on our day-to-day lives. In the current scenario of **climate crisis**, we are being pushed to seek new energy sources. Some of these green alternatives are already well known by the general public, such as the wide spectrum of renewable energies — solar, wind, hydraulic... — or nuclear energy. Despite being a great alternative to fossil fuels, these solutions still present some disadvantages. To mention a few, renewable energy is highly dependent on weather, presents serious geographic limitations and its power output and the power consumption from the grid are generally unmatched (see fig. 1.1). On the other hand, long-term radioactive waste production and the catastrophic potential accidents for nuclear fission make some collectives reluctant to this technology.

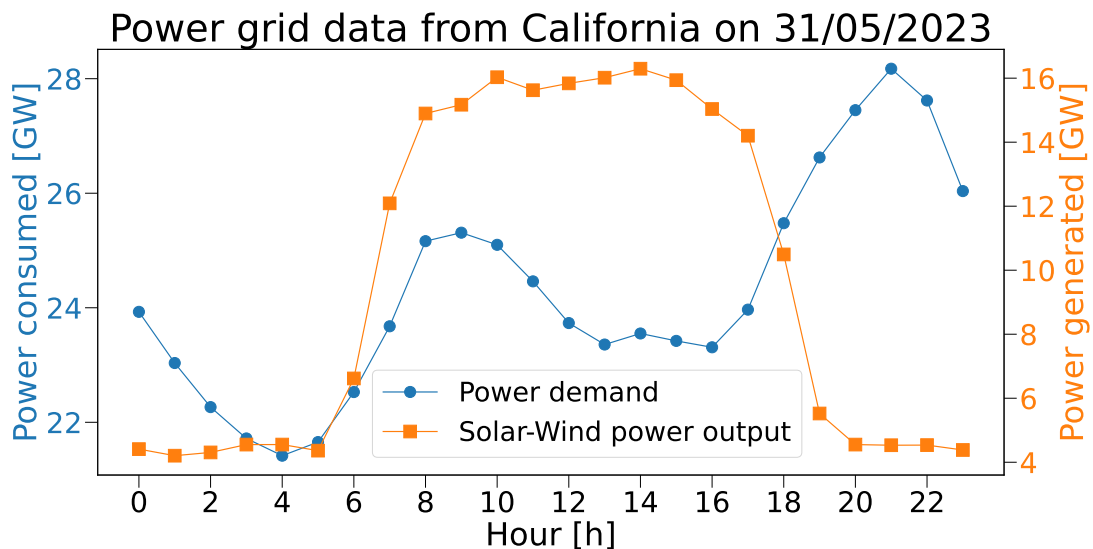


Figure 1.1: Duck curve of power consumption in California and the power generation from solar and wind. The different scales in the y axis help visualise the maximum generation at minimum demand. Data obtained from CAISO [1].

For some of these reasons **nuclear fusion energy** has been proposed as a strong candidate for a clean and robust energy source. In fact, 140 kg would produce as much energy as 800 kg of uranium and is also equivalent to 10^6 t of crude oil [2]. The difference between conventional

nuclear fission and fusion is quite drastic. Both share that the energy production mechanism is brought up by nuclear reactions in its fuel, but while nuclear fission is based on splitting the atom of a heavy element, such as uranium, in lighter species; nuclear fusion aims to combine two lightweight nuclei to produce high amounts of energy. For those curious, a more in-depth insight into nuclear fusion is provided in the following section.

1.1 Nuclear Fusion

Thermonuclear reactions are based on the combination of two nuclei of light-weight atoms; which after fusion, the product mass is smaller than the direct sum of its precursors. According to Einstein's famous formula $E = mc^2$, the energy gain of the reaction, is defined as

$$\Delta E = (m_f - m_i)c^2 \quad (1.1)$$

The energy in eq. (1.1), coming from the mass defect between products and results, is equal to the binding energy of the nucleons. Figure 1.2 depicts useful information regarding not only the physics of fission and fusion energy but also stellar evolution physics. To the left of Fe^{56} , we have light-weight elements. As a rule of thumb, the lighter the element, the bigger the energy produced by fusing. These reactions happen naturally within stellar cores, generating the necessary energy to prevent their gravitational collapse. Massive stars undergo fusion, producing heavier elements until they reach iron. At that point, they cannot produce energy anymore and the outer shells collapse into the nucleus. On the same figure, at the right side of Fe^{56} , we have heavier elements. By fissioning elements such as uranium, lighter elements are produced releasing energy. This is the working principle of fission reactors, which operates with the right-most elements of fig. 1.2.

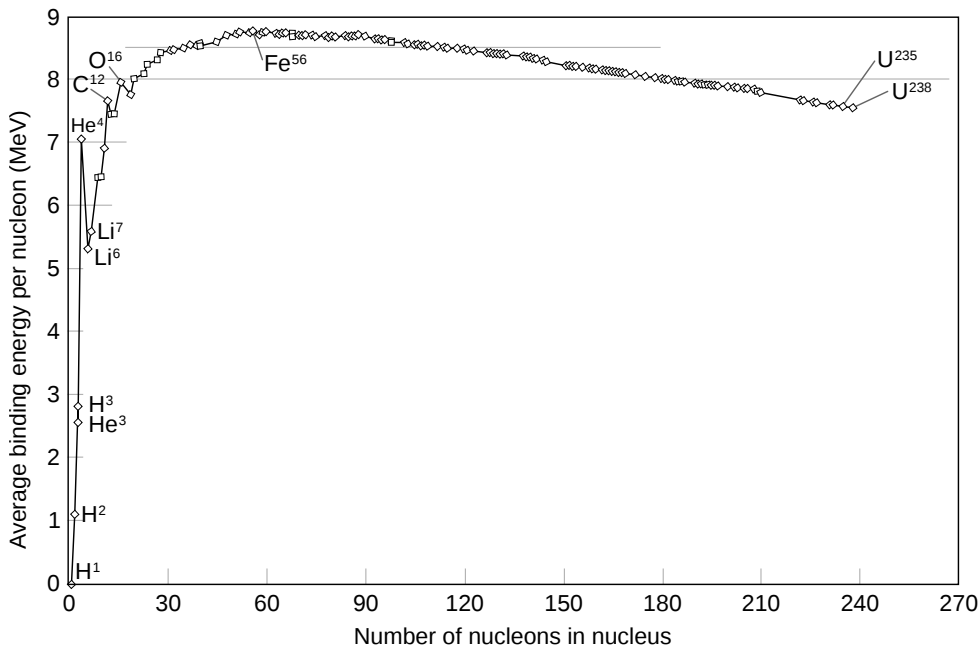
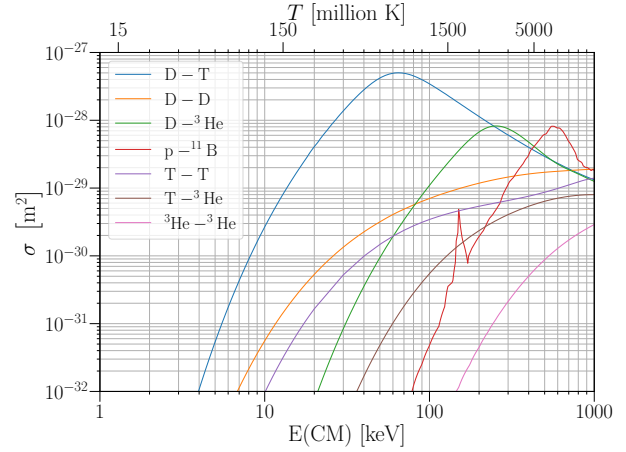
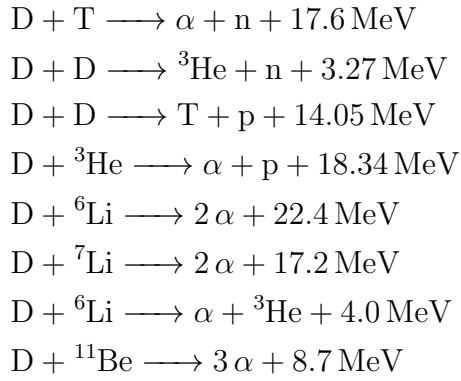
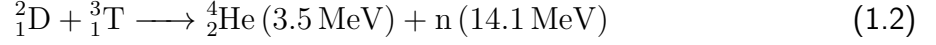


Figure 1.2: Average binding energy of elements. Fe^{56} acts as a limit to exothermic reaction.

There are plenty of nuclear exothermic reactions ($\Delta E > 0$, fig. 1.3a), but their reaction rate, which is proportional to the cross-section $\sigma(v)$ (fig. 1.3b), depends substantially on the temperature of the reactants. To this date, the preferred reaction is fusing deuterium (D)

and tritium (T) eq. (1.2), exclusively due to having the cross-section maximum at the lowest temperature. Some alternatives present certain advantages, such as aneutronic reactions or the use of non-radioactive products such as tritium, but we are still limited by the current technology for that to be achievable. Having aneutronic fusion would avoid the activation of the reactor wall — and hence the use of high-activation materials such as tungsten — and avoiding radioactive fuel such as tritium (half-life of 12.32 years) would suppose bigger reservoirs and no need for breeding process inside the reactor.



(a) Examples of nuclear fusion reactions [3]. (b) Cross-section for some of the reactions. Data extracted from [4, 5].

Figure 1.3: Some reactions present different advantages, but the easiest to achieve is D + T.

1.2 Plasma description

As seen in fig. 1.3b, the range of temperatures to achieve a high cross-section for D-T fusion is of the order of $1.5 \times 10^8 \text{ K}$ (equivalent to $\sim 10 \text{ keV}$). At such temperatures, the hydrogen isotopes become fully ionised, releasing their electrons separately from their nuclei. This state of free-charged gas is known as plasma. Plasma is the most abundant state of matter in the universe, not only the Sun and lightning are examples of plasma, but also the interstellar medium, neon lights, and auroras. A proposed definition for plasma extracted from [6] reads as follow:

A plasma is a quasi-neutral gas of charged and neutral particles which exhibits collective behaviour.

However, there are a set of requirements that need to be fulfilled to present this collective behaviour, and hence, for an ionised gas to be rigorously considered as plasma. These requirements involve a well-known set of natural parameters (λ_{De} , ω_{pe} and Λ_D) that have to fulfil a series of conditions. The Debye length (λ_{De}) could be expressed in words, as the length scale at which quasi-neutrality of the plasma does not necessarily hold, or the length at which a DC perturbation is screened. On the other hand, the plasma frequency (ω_{pe}), could also be understood as the oscillation frequency experienced by electrons under a harmonic perturbation in the charge distribution. Similarly, it can be defined also for the ionic species ω_{pi} . Finally,

the plasma parameter Λ_D simply accounts for the number of particles in a sphere of dimension λ_{De} . The mathematical definitions and constraints are the following [2]:

$$\lambda_{De} = \sqrt{\frac{\varepsilon_0 T_e}{n e^2}} \ll L \quad (1.3a)$$

$$\omega_{pe} = \sqrt{\frac{n e^2}{\varepsilon_0 m_e}} \gg \omega_{Te} \quad (1.3b)$$

$$\Lambda_D = \frac{4}{3} \pi n_e \lambda_{De}^3 \gg 1 \quad (1.3c)$$

Equation (1.3a) is telling that the characteristic length of the system (L) has to be much larger than λ_{De} , in order to shield any static perturbation. Equation (1.3b) represents that the rate of thermal collisions between electrons (ω_{Te}) is much smaller than the electron plasma frequency, allowing screening any harmonic perturbation. The third condition, eq. (1.3c), would tell that for an accurate statistical fluid description of the plasma, we would need a great number of particles in a volume smaller than a Debye's length sphere. If these conditions are fulfilled, then the ionised gas is dominated by long-range collective effects instead of collisionality effects.

In this thesis, we consider one particular type of plasma, which is called *cold plasma*. This approximation assumes that there are virtually no thermal collisions between particles. In some of the simulations done with MAXWEL, we have used this model and its derivations, to observe how an electromagnetic wave would propagate under this type of medium. Plasmas found in fusion reactors, that is *hot plasmas*, are nowhere close to this approximation, but still, cold plasma gives good insight into wave propagation and accessibility in a wide range of scenarios. The cold plasma model will be later detailed in section 2.1.

1.3 Magnetically confined fusion

As we have seen in fig. 1.3, we need to heat the reactants (deuterium and tritium) to extremely high temperatures, of the order of million Kelvin. Furthermore, we need to be able to confine the plasma in a limited region, to prevent the hot material from touching the reactor walls and to improve the heating efficiency. Without a combination of high enough temperature, density, and long enough confinement, nuclear fusion becomes unreachable. Such a condition is known as the Lawson criterion, and expresses the minimum threshold to achieve ignition, in terms of temperature T , density n , and confinement time τ_E .

$$n \tau_E T \geq 3.5 \times 10^{21} \text{ keV s m}^{-3} \quad (1.4)$$

Ignition is the scenario in which fusion becomes self-sustained and there is no need to inject energy to heat the plasma. This concept might be impossible to replicate on Earth, or at least it is being extremely difficult. Plasma presents magnetohydrodynamic instabilities that prevent this ignition condition from happening. Lots of research is being done to minimize these instabilities from different approaches. However, it is not necessary to reach ignition to achieve an energy gain factor bigger than one, which would allow us to extract energy from a fusion reaction. Many methods and technologies are being developed to produce such reactions, but they can be differentiated into two main branches. One would be inertial confinement, where they use laser pulses to heat up and compress the fuel pellets to trigger fusion reactions. On the other hand, magnetic confinement consists of using magnetic fields, usually in toroidal-like devices, to confine the plasma fuel within.

Since the early stages of magnetic fusion confinement, there have been many proposed reactors and machines that have been tested and studied. However, the ones that lead to the most promising results are tokamaks and stellarators (fig. 1.4). Both present a toroidal shape, whereas a tokamak resembles a standard torus while stellarators present a twisted shape and consequently, they are not axisymmetric. However, new stellarators are being tested that rely on 2D-printed coils on the surface of the torus. This allows the machine itself to be axisymmetric, despite the plasma shape configuration remaining the same. This technology is under investigation by the company Renaissance Fusion [7]. Tokamaks induce a toroidal plasma current by means of the central solenoid, while stellarators use their helical geometry to stabilise the plasma.

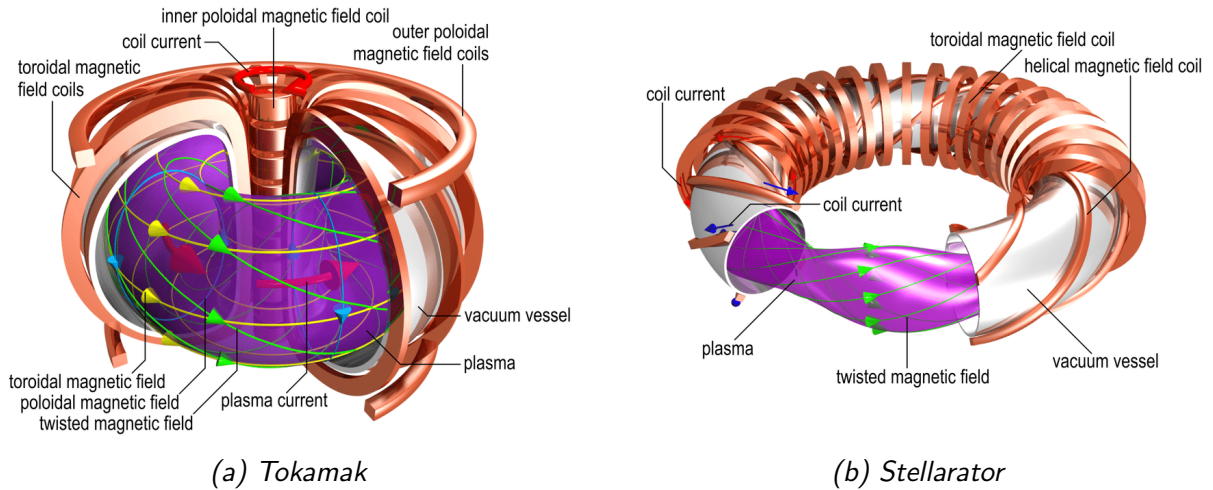


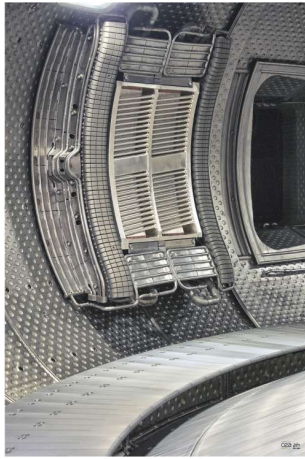
Figure 1.4: Schematic concepts of tokamaks and stellarators. Images extracted from [8].

1.3.1 Plasma heating

Indistinctly of which reactor is used, plasma heating is one of the fundamental topics to be addressed to achieve an energy gain factor $Q > 1$. Several approaches are being implemented; however, we will mention only the following three due to their major relevance. Each one of these three presents different regions of efficiencies, advantages, and difficulties. The first method is known as ohmic heating, in which the kinetic energy of electrons is transferred by elastic collisions to the different species within the plasma. Ohmic heating is logically more dominant in tokamaks than stellarators since the former operates with large toroidal currents. However, ohmic heating becomes less effective at higher temperatures. To overcome this issue, there are alternative methods such as microwave heating and Neutral Beam Injection (NBI). NBI consists of injecting neutral particles which are not affected by the magnetic fields within the reactor. This allows those particles to reach deeper into the plasma, allowing them to transfer kinetic energy and momentum.

Finally, Ion and Electron Cyclotron Resonance Heating (ICRH, ECRH) are both two complementary methods to heat the plasma using microwave radiation. For this method, some antennas present on the wall of the reactor — as can be seen in fig. 1.5 — inject electromagnetic pulses with such frequency that resonates with the cyclotron frequencies of electrons or ions, transferring energy from the wave to their kinetic energy perpendicular to the magnetic field. Consequently, these electromagnetic waves will travel across the plasma interacting with its species.

Simulating these waves would allow to better understand and test, really cost-efficiently, different configurations of ICRH and ECRH in working plasma conditions. MAXWEL module tries to fill this gap in the computational plasma physics domain, by using the *Finite Element Method* (FEM) to solve numerically a propagating wave equation. The main idea is to have a code that is agnostic to geometry, tokamak or stellarator, to plasma position, core or *scrape-off layer* (SOL), and to range of frequencies, ICRH or ECRH, as opposed to most of the codes available in the community, as [9, 10, 11] to cite a few. Furthermore, with the added value that the module will be integrated into a suite of codes for integrated modelling within the ALYA framework. For now, the software is limited to a poloidal cross-section of a tokamak, considering an invariant geometry across the toroidal direction. Further details of the process will be given in chapters 2 and 3.



(a) ICRH of WEST tokamak.



(b) ICRH of stellarator Wendelstein 7-X.

Figure 1.5: Example of ICRH antennas. Images extracted from [12, 13] respectively.

Plasma heating, in particular ICRH and NBI, was one of the key aspects to achieving the present fusion energy world record reached at the Joint European Torus (JET) during the second and third deuterium-tritium campaigns (DTE2 and DTE3) in 2021 and 2023. Many recent studies based on plasma heating were carried out to prepare for these experiments which served as a guideline towards this success, e.g. [14, 15, 16, 17, 18, 19, 20].

1.4 Numerical Modelling

Numerical modeling is a broad multidisciplinary field, in which computational resources are used as a tool to simulate mathematical or theoretical models. It combines many aspects of computer science, mathematics, engineering, and physics. Numerical methods have the advantage of being independent of the problem or geometry of the domain, compared to analytical derivations, whose solutions are usually limited to a set of functions that can change depending on the domain and geometry. Consequently, numerical methods can be applied to numerous types of problems, of varied physics and applied to different geometries. This allows us to access a framework of virtual simulations in order to reproduce and experiment with the physical paradigm of our interest.

In this work, we will use this powerful tool to simulate the wave propagation in a 2D cross-section of a tokamak, with different considerations and configurations. This task serves its purpose as a first step towards a complete simulation of microwave heating, in which electro-

magnetic pulses are sent from the antennas to heat the plasma, as explained in section 1.3.1. So we will focus on solving partial differential equations derived from Maxwell's laws to simulate and compute the electromagnetic fields within the torus. To do this, we will use the *Finite Element Method* (FEM). In the initial stages of the project, we used Dr. Özgün and Dr. Kuzuoğlu's textbook [21] as a reference and inspiration, so the reader may find some similarities in the notation and theoretical framework. For a more comprehensive description of the numerical methods used and simulated scenarios, we recommend [22].

1.5 Alya and Marenostum 5

This work was carried out as a student member of the Fusion group at the Barcelona Supercomputing Center (BSC). BSC is a Spanish Severo Ochoa Center of Excellence, the founding member of the Partnership of Advance Computing in Europe (PRACE), and also the Spanish National Supercomputing Center (*Centro Nacional de Supercomputación* - CNS). The Fusion group is hosted in the Computer Applications for Science and Engineering (CASE) department, which is staffed by a highly multidisciplinary team with around 100 researchers of different backgrounds (physicists, engineers, mathematicians, computer scientists, technicians, etc.) working together. The Fusion Group at CASE is focused on the research in nuclear fusion which aims to develop fusion as a safe, clean, and virtually limitless energy source for future generations. The CASE department has been involved in the field of nuclear fusion research since 2008 when BSC-CNS joined as one of the fourteen partners in the EU-funded project EUFORIA (EU fusion for ITER Applications) to provide the infrastructure that links high-performance computing (HPC) to the fusion modelling community for ITER sized plasmas. Several members of the Fusion Group have led and coordinated many international experiments and modelling tasks such as in JET or AUG.

The BSC CASE department, since its birth in 2004, has developed a software program called Alya [23, 24]. Alya is more than a Finite Element Method (FEM) code; it is a development framework on which different physics are implemented according to the interest of the developer. Alya uses a single main numerical tool, finite elements, although over the years numerous alternative techniques have been included to optimise its answer depending on the physics to be solved. Alya provides the parallelism tools in an integrated and transparent way to the specific developer. Needless to say, Alya was designed from the very first line of code, to get the best possible performance on a supercomputer. That is why efficiencies close to 100% are obtained even when running problems on 100,000 processors [24].

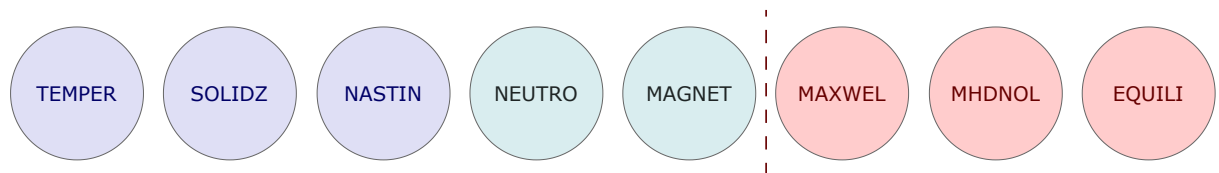


Figure 1.6: In blue, modules originally developed by CASE. In green, modules developed exclusively by the Fusion Group. In red, modules being currently under development.

Alya's kernel has a mesh partitioning system, the IO, a huge number of linear systems and eigenvalue solvers, and an extensive finite element library, its main numerical tool. The specific modules are mounted on this kernel to solve the different problems of interest. Currently, there are modules to solve thermal problems, incompressible fluids, turbulence, mechanical problems, combustion, neutronics, and magnetism, among others. The long-term goal within

the Fusion group is to develop a group of modules that will be added entirely to Alya to account for different problems that occur within confined plasma, and its interaction with the walls and components of the reactor that contains it. In this sense, some modules already created in Alya need to be adapted to work with this problem. There are currently modules to simulate thermal hydraulics and thermal mechanics (TERMAL, SOLIDZ, NASTIN), a module to describe neutron transport (NEUTRO) [25, 26] and another to describe magnetic behaviour in superconductors (MAGNET) [27]. In the same way that those modules were built, we have the intention to create new modules: one for plasma equilibrium (EQUILI) [28], one to analyse the electromagnetic response of the plasma and the structure (MAXWEL), and another to describe linear and nonlinear magnetohydrodynamics to study problems as breeding blanket and plasma disruptions (MHDNOL) — see fig. 1.6 —. In this way, Alya would possess the required plasma physics, based on the same language, the same IO, the same numerical technology, and the natural communication proposed by the KERNEL. As an added advantage, this software will have Alya's high computational efficiency and continuous support for the kernel operation on different architectures.

Chapter 2

Electromagnetic theory

In this chapter we will present the electromagnetic formulation and derivation of the differential equations of interest, to be solved by the FEM method. To do so, we will part from the well-known Maxwell equations.

$$\nabla \times \mathcal{E}(\mathbf{r}, t) = -\partial_t \mathcal{B}(\mathbf{r}, t) \quad (2.1)$$

$$\nabla \times \mathcal{H}(\mathbf{r}, t) = \mathcal{J}(\mathbf{r}, t) + \partial_t \mathcal{D}(\mathbf{r}, t) \quad (2.2)$$

$$\nabla \cdot \mathcal{D}(\mathbf{r}, t) = \rho(\mathbf{r}, t) \quad (2.3)$$

$$\nabla \cdot \mathcal{B}(\mathbf{r}, t) = 0 \quad (2.4)$$

Where $\mathcal{D}$ [C/m²] is the electric displacement field, $\mathcal{E}$ [V/m] is the electric field, $\mathcal{H}$ [A/m] is the magnetic field intensity, $\mathcal{B}$ [T] is the magnetic flux density, ρ [C/m³] is the electric charge density and $\mathcal{J}$ [A/m²] is the current density. Considering a linear relation between $\mathcal{D}$ and $\mathcal{E}$ and between $\mathcal{B}$ and $\mathcal{H}$, we can express these relations in the frequency domain such as:

$$\mathbf{D}(\mathbf{r}, \omega) = \boldsymbol{\varepsilon}(\mathbf{r}, \omega) \cdot \mathbf{E}(\mathbf{r}, \omega) \quad (2.5)$$

$$\mathbf{B}(\mathbf{r}, \omega) = \boldsymbol{\mu}(\mathbf{r}, \omega) \cdot \mathbf{H}(\mathbf{r}, \omega) \quad (2.6)$$

Where $\boldsymbol{\varepsilon}$ [F/m] and $\boldsymbol{\mu}$ [H/m] are the complex permittivity and permeability tensors. For this work, we have considered monochromatic and time-harmonic waves, so we can extract frequency dependence in fields from eqs. (2.1) to (2.6), by taking out the time phase $e^{i\omega t}$. Consequently, now Maxwell's equations look like

$$\nabla \times \mathbf{E}(\mathbf{r}) = -i\omega \boldsymbol{\mu} \mathbf{H}(\mathbf{r}) \quad (2.7)$$

$$\nabla \times \mathbf{H}(\mathbf{r}) = \mathbf{J}(\mathbf{r}) + i\omega \boldsymbol{\varepsilon} \mathbf{E}(\mathbf{r}) \quad (2.8)$$

$$\nabla \cdot \mathbf{E}(\mathbf{r}) = \boldsymbol{\varepsilon}^{-1} \rho(\mathbf{r}) \quad (2.9)$$

$$\nabla \cdot \mathbf{H}(\mathbf{r}) = 0 \quad (2.10)$$

Now we have two sets of coupled equations, from which we can extract a constitutive relation for $\mathbf{H}$ and $\mathbf{E}$. Applying the curl to eq. (2.7), and substituting the right-hand term by eq. (2.8), we obtain the following expression:

$$\nabla \times (\boldsymbol{\mu}^{-1} \cdot \nabla \times \mathbf{E}) - \omega^2 \boldsymbol{\varepsilon} \cdot \mathbf{E} = -i\omega \mathbf{J} \quad (2.11)$$

and doing the same process but in the opposite order, we reach the following equation

$$\nabla \times (\varepsilon^{-1} \cdot \nabla \times \mathbf{H}) - \omega^2 \boldsymbol{\mu} \cdot \mathbf{H} = \nabla \times (\varepsilon^{-1} \cdot \mathbf{J}) \quad (2.12)$$

It has to be considered that $\mathbf{J}$ has two different components; $\mathbf{J}_s$, originated by externally input currents or sources, and $\mathbf{J}_{\text{ind}}$, from the induced current due to non-zero conductivity. According to Ohm's law $\mathbf{J}_{\text{ind}} = \boldsymbol{\sigma} \cdot \mathbf{E}$. Now Ampere's law eq. (2.8) is presented as

$$\begin{aligned} \nabla \times \mathbf{H} &= \mathbf{J}_s + \boldsymbol{\sigma} \cdot \mathbf{E} + i\omega\varepsilon_0\boldsymbol{\varepsilon}_r \cdot \mathbf{E} \\ &= \mathbf{J}_s + i\omega\varepsilon_0\boldsymbol{\varepsilon}_{rc} \cdot \mathbf{E} \end{aligned} \quad (2.13)$$

with $\boldsymbol{\varepsilon}_{rc} = \boldsymbol{\varepsilon}_r + i\frac{\boldsymbol{\sigma}}{\omega\varepsilon_0}$. It has to be noted that $\boldsymbol{\varepsilon}_r$ is strictly a real tensor and $\boldsymbol{\varepsilon}_{rc} \in \mathbb{C}$. As an abuse of notation, when a permittivity tensor presents complex values, we might note it as $\boldsymbol{\varepsilon}_r$ although it inherently refers to $\boldsymbol{\varepsilon}_{rc}$.

Consequently eqs. (2.11) and (2.12) now look like

$$\nabla \times (\boldsymbol{\mu}_r^{-1} \cdot \nabla \times \mathbf{E}) - k_0^2 \boldsymbol{\varepsilon}_{rc} \cdot \mathbf{E} = -ik_0\eta_0 \mathbf{J}_s \quad (2.14)$$

$$\nabla \times (\boldsymbol{\varepsilon}_{rc}^{-1} \cdot \nabla \times \mathbf{H}) - k_0^2 \boldsymbol{\mu}_r \cdot \mathbf{H} = \nabla \times (\boldsymbol{\varepsilon}_{rc}^{-1} \cdot \mathbf{J}_s) \quad (2.15)$$

with $k_0 = \omega\sqrt{\varepsilon_0\mu_0}$ and $\eta_0 = \sqrt{\frac{\mu_0}{\varepsilon_0}}$. Both eqs. (2.14) and (2.15) represent a 3D vector wave equation. However, we have considered an invariant geometry across a given direction, in this case, the z axis; and, if the current density $\mathbf{J}$ does not depend on z , it would be enough condition to reduce eqs. (2.14) and (2.15) to a 2D case. But doing a further simplification, we have considered a source-free medium ($\mathbf{J}_s = 0$) and electric and magnetic fields have been decomposed into two fundamental polarisations, *transverse magnetic* (TM) and *transverse electric* (TE), giving place to the following *Helmholtz equations*.

$$\nabla \cdot (\boldsymbol{\Lambda}_\mu \cdot \nabla E_z) + k_0^2 \varepsilon_{zz} E_z = 0 \quad (2.16)$$

$$\nabla \cdot (\boldsymbol{\Lambda}_\varepsilon \cdot \nabla H_z) + k_0^2 \mu_{zz} H_z = 0 \quad (2.17)$$

Where eq. (2.16) describes a TM wave ($H_x, H_y \neq 0 \Rightarrow E_z \neq 0$) and eq. (2.17) corresponds to a TE wave ($E_x, E_y \neq 0 \Rightarrow H_z \neq 0$). In these equations, parameters are defined as

$$\boldsymbol{\Lambda}_\mu = \frac{(\boldsymbol{\mu}_r)_{\text{sub}}^T}{|(\boldsymbol{\mu}_r)_{\text{sub}}^T|} \quad \text{and} \quad \boldsymbol{\Lambda}_\varepsilon = \frac{(\boldsymbol{\varepsilon}_{rc})_{\text{sub}}^T}{|(\boldsymbol{\varepsilon}_{rc})_{\text{sub}}^T|} \quad (2.18)$$

where

$$\boldsymbol{\varepsilon}_{rc} = \begin{pmatrix} \varepsilon_{xx} & \varepsilon_{xy} & 0 \\ \varepsilon_{xy} & \varepsilon_{yy} & 0 \\ 0 & 0 & \varepsilon_{zz} \end{pmatrix} = \begin{pmatrix} (\varepsilon_{rc})_{\text{sub}} & 0 \\ & 0 \\ 0 & 0 & \varepsilon_{zz} \end{pmatrix} \quad (2.19a)$$

$$\boldsymbol{\mu}_r = \begin{pmatrix} \mu_{xx} & \mu_{xy} & 0 \\ \mu_{xy} & \mu_{yy} & 0 \\ 0 & 0 & \mu_{zz} \end{pmatrix} = \begin{pmatrix} (\mu_r)_{\text{sub}} & 0 \\ & 0 \\ 0 & 0 & \mu_{zz} \end{pmatrix} \quad (2.19b)$$

To be able to compute either H_z or E_z fields, they will be decomposed using the *scattered field formulation*. For example, $H_z = H_z^{\text{inc}} + H_z^{\text{scat}}$, where "inc" refers to the incident wave and "scat" to the scattered field. Consequently, eqs. (2.16) and (2.17) can be re-written as:

$$-\nabla \cdot (\boldsymbol{\Lambda}_\mu \cdot \nabla E_z^{\text{scat}}) - k_0^2 \varepsilon_{zz} E_z^{\text{scat}} = \nabla \cdot (\boldsymbol{\Lambda}_\mu \cdot \nabla E_z^{\text{inc}}) + k_0^2 \varepsilon_{zz} E_z^{\text{inc}} \quad (2.20)$$

$$-\nabla \cdot (\mathbf{\Lambda}_\varepsilon \cdot \nabla H_z^{\text{scat}}) - k_0^2 \mu_{zz} H_z^{\text{scat}} = \nabla \cdot (\mathbf{\Lambda}_\varepsilon \cdot \nabla H_z^{\text{inc}}) + k_0^2 \mu_{zz} H_z^{\text{inc}} \quad (2.21)$$

Equations (2.20) and (2.21) are a particular case of a more generic expression

$$-\nabla \cdot (\mathbf{p} \cdot \nabla u) + q u = f \quad (2.22)$$

where, for a TM polarisation,

$$f(x, y) = \nabla \cdot (\mathbf{\Lambda}_\mu \cdot \nabla E_z^{\text{inc}}) + k_0^2 \varepsilon_{zz} E_z^{\text{inc}} \quad (2.23a)$$

$$\mathbf{p} = \mathbf{\Lambda}_\mu \quad (2.23b)$$

$$q = -k_0^2 \varepsilon_{zz} \quad (2.23c)$$

$$u^{\text{inc}} = E_z^{\text{inc}} \quad (2.23d)$$

$$u = E_z^{\text{scat}} \quad (2.23e)$$

and for a TE polarisation

$$f(x, y) = \nabla \cdot (\mathbf{\Lambda}_\varepsilon \cdot \nabla H_z^{\text{inc}}) + k_0^2 \mu_{zz} H_z^{\text{inc}} \quad (2.24a)$$

$$\mathbf{p} = \mathbf{\Lambda}_\varepsilon \quad (2.24b)$$

$$q = -k_0^2 \mu_{zz} \quad (2.24c)$$

$$u^{\text{inc}} = H_z^{\text{inc}} \quad (2.24d)$$

$$u = H_z^{\text{scat}} \quad (2.24e)$$

This new notation will be of great help when computing the equation's weak form in section 3.4 by simplifying their expressions

2.1 Cold Plasma Model

In this work, we have not considered the thermal distribution of particles nor the effects of the heating methods such as ICRH. Instead, we have treated our plasma with a cold plasma approximation, in which we considered the thermal velocity of particles much smaller than the phase velocity of an injected wave. This approximation is necessary due to the early stage of MAXWEL software, in which we solve a time-harmonic equation, and hence we need a time-independent permittivity tensor. However, the study of EM wave accessibility is of great importance. The code assesses whether an EM wave can propagate in an arbitrary linear medium, such as plasma. Notice that the cold plasma model offers a fairly accurate idea of the capability of the wave to reach the plasma region of interest albeit its lack of wave-particle interaction.

To compute the permittivity tensor of a cold plasma we need to start from the definition of ε_{rc} , that has been introduced previously, but considering that $\varepsilon_r = \mathbb{I}$. This is forced essentially to simplify calculations. It is commonly considered plasma as a vacuum-like material; for which all information regarding the permeability, and most importantly, the permittivity is stored in the conductivity tensor. A more rigorous derivation of ε_{rc} is shown in the following section.

$$\varepsilon_{rc} = \mathbb{I} - i \frac{\boldsymbol{\sigma}}{\omega \varepsilon_0} \quad (2.25)$$

Consequently, there is only the conductivity $\boldsymbol{\sigma}$ as an unknown, because ω is an input parameter. To compute $\boldsymbol{\sigma}$ we need to derive the linear expressions of the particles' velocity first and then use the total current expression, assuming that there are no external input currents

$$\mathbf{J} = en_0 (\mathbf{v}_i - \mathbf{v}_e) = \boldsymbol{\sigma} \mathbf{E} \quad (2.26)$$

Starting from the initial dynamic equations (2.27), assuming small perturbations and taking the first-order linear terms we can retrieve the velocity functions for both ions and electrons (eq. (2.28)). The cold plasma model allows us to consider the fluid stress tensor $\Phi_j = 0$, valid when the wave phase velocity is much greater than the thermal velocity of species.

$$m_e \left(\frac{\partial \mathbf{v}_e}{\partial t} + \mathbf{v}_e \cdot \nabla \mathbf{v}_e \right) = -e (\mathbf{E} + \mathbf{v}_e \times \mathbf{B}) - \nabla \cdot \Phi_e \quad (2.27a)$$

$$m_i \left(\frac{\partial \mathbf{v}_i}{\partial t} + \mathbf{v}_i \cdot \nabla \mathbf{v}_i \right) = e (\mathbf{E} + \mathbf{v}_i \times \mathbf{B}) - \nabla \cdot \Phi_i \quad (2.27b)$$

$$v_x = \frac{q}{m} \frac{i\omega E_x - \omega_c E_y}{\omega^2 - \omega_c^2} \quad (2.28a)$$

$$v_y = \frac{q}{m} \frac{i\omega E_y + \omega_c E_x}{\omega^2 - \omega_c^2} \quad (2.28b)$$

$$v_z = \frac{iq}{m\omega} E_z \quad (2.28c)$$

Where $\omega_c = \frac{qB_0}{m}$ is the cyclotron frequency. Now, using eqs. (2.25) and (2.26) we can compute the cold plasma permittivity tensor. Expressing it in Stix notation, the tensor is [29, 30, 31]

$$\varepsilon_r = \begin{pmatrix} \varepsilon_{\perp} & -i\varepsilon_{xy} & 0 \\ i\varepsilon_{xy} & \varepsilon_{\perp} & 0 \\ 0 & 0 & \varepsilon_{\parallel} \end{pmatrix} = \begin{pmatrix} S & -iD & 0 \\ iD & S & 0 \\ 0 & 0 & P \end{pmatrix} \quad (2.29)$$

where

$$\varepsilon_{\perp} = S = \frac{R+L}{2} = 1 - \sum_j \frac{\omega_{pj}^2}{\omega^2 - \omega_{cj}^2} \quad (2.30a)$$

$$\varepsilon_{xy} = D = \frac{R-L}{2} = \sum_j \frac{\omega_{cj}\omega_{pj}^2}{\omega(\omega^2 - \omega_{cj}^2)} \quad (2.30b)$$

$$\varepsilon_{\parallel} = P = 1 - \sum_j \frac{\omega_{pj}^2}{\omega^2} \quad (2.30c)$$

$$R = S + D = 1 - \sum_j \frac{\omega_{pj}^2}{\omega(\omega + \omega_{cj})} \quad (2.30d)$$

$$L = S - D = 1 - \sum_j \frac{\omega_{pj}^2}{\omega(\omega - \omega_{cj})} \quad (2.30e)$$

The plasma frequency is defined as $\omega_{pj}^2 = \frac{n_j (Z_j e)^2}{\varepsilon_0 m_j}$

2.2 Wave propagation theory

The cold plasma permittivity tensor (eq. (2.29)) plays a crucial role when solving the dispersion relation. From this expression, one can extract many insights about cutoff regions and resonances or at which velocity the wave will propagate, among other relevant information. To compute the cold plasma dispersion relation (CPDR) it is necessary to obtain the full wave equation first. This time, instead of retrieving a particular case such as the Helmholtz equation as previously, we will aim for a more general and three-dimensional expression. To do so, we will rewrite eqs. (2.1) and (2.2) in the frequency domain. Have in mind that after the Fourier transformation $\nabla \rightarrow -i\mathbf{k}$ and $\partial_t \rightarrow i\omega$

$$i\mathbf{k} \times \mathbf{E} = i\omega \mathbf{B} \quad (2.31)$$

$$i\mathbf{k} \times \mathbf{B} = -\boldsymbol{\mu}(\mathbf{J} + i\omega\boldsymbol{\varepsilon}\mathbf{E}) \quad (2.32)$$

Combining these two equations, we can write

$$\mathbf{k} \times (\mathbf{k} \times \mathbf{E}) = i\omega\boldsymbol{\mu}(\mathbf{J} + i\omega\boldsymbol{\varepsilon}\mathbf{E}) \quad (2.33)$$

which by means of Ohm's law, $\mathbf{J} = \boldsymbol{\sigma} \cdot \mathbf{E}$, we obtain

$$\mathbf{k} \times (\mathbf{k} \times \mathbf{E}) = i\omega\boldsymbol{\mu}\boldsymbol{\sigma}\mathbf{E} - \omega^2\boldsymbol{\varepsilon}\boldsymbol{\mu}\mathbf{E} \quad (2.34)$$

$$\mathbf{k} \times (\mathbf{k} \times \mathbf{E}) = -\frac{\omega^2}{c^2} \left(\mathbb{I} - i\frac{\boldsymbol{\sigma}}{\omega\boldsymbol{\varepsilon}_0} \right) \mathbf{E} \quad (2.35)$$

And consequently,

$$\mathbf{n} \times (\mathbf{n} \times \mathbf{E}) + \boldsymbol{\varepsilon}_{rc} \cdot \mathbf{E} = 0 \quad (2.36)$$

where $\mathbf{n} = \mathbf{k} \frac{c}{\omega}$ is the refraction index and $\boldsymbol{\varepsilon}_{rc} = \mathbb{I} - i\frac{\boldsymbol{\sigma}}{\omega\boldsymbol{\varepsilon}_0}$. The refraction index can be split into two components $\mathbf{n} = \mathbf{n}_\perp + n_\parallel \hat{a}_z$; but for our purpose, $n_\parallel$ will be considered to be 0 since our wave propagation always happens in the xy plane — that is, perpendicular to the z direction —. With all these considerations and using eq. (2.29), eq. (2.36) can be rewritten as [30]

$$\begin{pmatrix} S - n^2 \cos^2 \theta & -iD & n^2 \cos \theta \sin \theta \\ iD & S - n^2 & 0 \\ n^2 \cos \theta \sin \theta & 0 & P - n^2 \sin^2 \theta \end{pmatrix} \cdot \mathbf{E} = 0 \quad (2.37)$$

where θ is the angle between $\mathbf{k}$ and the z axis. For our solutions, $\theta = \frac{\pi}{2}$ always. Solving this set of homogeneous equations is equivalent to finding the CPDR. The non-trivial solutions are obtained by the roots of the matrix determinant. Finding these roots is not necessarily an easy task, and many possible solutions can be found. Each of these solutions or *modes* has a specific name or label, which may vary according to the author, context, or discipline. But for our interest, we will mention just a few of these cases.

When the propagation of the electric field is parallel to the z direction ($\theta = 0$) there are three possible solutions:

- $P = 0$. Langmuir waves.
- $n^2 = R$. Right-handed waves.
- $n^2 = L$. Left-handed waves.

As we mentioned, these modes will not be observable in the solutions of our problem, since the propagation occurs in a 2D cross-section plane perpendicular to z . For $\theta = \frac{\pi}{2}$, there are two propagation modes:

- $n^2 = P$. Ordinary wave or slow wave.
- $n^2 = \frac{RL}{S}$. Extraordinary wave or fast wave.

In eq. (2.30) it is shown all full expressions for parameters P , S , R and L . For those regions in which these parameters make n diverge, or become 0; we will have *resonant conditions* and *cutoff conditions*, respectively.

2.2.1 Ordinary Wave

The polarization of the ordinary or slow mode corresponds to a TM wave since the electric field is parallel to the background magnetic field B_z . The refraction index of an ordinary wave only depends on the plasma frequency, and the magnetic field does not affect the dispersion relation whatsoever. As presented above, the dispersion relation is given by

$$n_O^2 = P = 1 - \sum_j \frac{\omega_{pj}^2}{\omega^2} \quad (2.38)$$

Consequently, there is no resonance condition, other than the trivial $\omega = 0$. However, there is a cutoff condition at which $n_O^2 \leq 0$ for $\omega \leq \omega_p$. When n_O^2 is negative, the wave vector $\mathbf{k}$ will have a complex component which eventually will add a negative exponent coefficient, attenuating the wave.

For the ordinary wave to propagate, it can be derived from the general CPDR that the wave frequency has to be lower than the ion cyclotron frequency, and consequently, it should be launched from a high-field side antenna. At the same time, the wave would not penetrate due to a resonance surface in the low-density edge of the plasma [32]. Therefore, the slow wave has poor penetration and can not be used for ICRH. For this reason, the focus will be given to TE propagation, i.e., extraordinary waves, as ICRF heating mechanisms use this polarization to heat ions in present fusion devices.

2.2.2 Extraordinary Wave

The solution of the extraordinary wave dispersion relation for two species (electrons and ions) is

$$n_X^2 = \frac{RL}{S} = \frac{[(\omega + \omega_{ci})(\omega - \omega_{ce}) - \omega_p^2][(\omega - \omega_{ci})(\omega + \omega_{ce}) - \omega_p^2]}{(\omega^2 - \omega_{ci}^2)(\omega^2 - \omega_{ce}^2) + \omega_p^2(\omega_{ce}\omega_{ci} - \omega^2)} \quad (2.39)$$

where $\omega_p^2 = \omega_{pe}^2 + \omega_{pi}^2$ and ω_{cj} is the cyclotron frequency. The resonance conditions ($n_X \rightarrow \infty$) would be those for which the denominator becomes 0. This condition happens when

$$\omega^2 = \frac{\omega_e^2 + \omega_i^2}{2} \pm \sqrt{\left(\frac{\omega_e^2 - \omega_i^2}{2}\right)^2 + \omega_{pe}^2 \omega_{pi}^2} \quad (2.40)$$

where $\omega_j = \omega_{pj}^2 + \omega_{cj}^2$. By neglecting the terms $\frac{m_e}{m_i}$ due to the big difference in mass, we can simplify eq. (2.40) as

$$\omega_{UH}^2 = \omega_{pe}^2 + \omega_{ce}^2 \quad (2.41)$$

$$\omega_{LH}^2 = \omega_{ce}\omega_{ci} \left(\frac{\omega_{pe}^2 + \omega_{ce}\omega_{ci}}{\omega_{pe}^2 + \omega_{ce}^2} \right) \quad (2.42)$$

These are the two roots of eq. (2.40), which are named as *upper hybrid resonance* ω_{UH} and *lower hybrid resonance* ω_{LH} . Similarly, the cutoffs ($n_X = 0$) are found at the zeros of the numerator in eq. (2.39).

$$\omega_X = \sqrt{\left(\frac{\omega_{ce} + \omega_{ci}}{2} \right)^2 + \omega_p^2} \pm \frac{\omega_{ce} - \omega_{ci}}{2} \quad (2.43)$$

Cutoff frequencies act as a limit for which the wave can penetrate the plasma. We can distinguish two different conditions, $L = 0$ or L-cutoff and $R = 0$ or R-cutoff. Since plasma heating using ICRH is performed with extraordinary waves, it is of great relevance to correctly identify such conditions. An important aspect of ICRH efficiency is to allow the waves to reach the inner regions of the plasma, but this not always can be easily achieved. For example, the injected wave may cross and tunnel through as an evanescent wave, bounce back, or even switch from a propagating mode to another when finding these cutoff regions [33]. This behaviour can be complex to describe and depends on many parameters, such as the wave harmonics or the plasma composition, to mention a few. Having a full-wave simulation software, able to capture the physics of this scenario, would help to better understand and analyse different configurations. However, notice that the cold plasma model fails at describing properly wave resonances and wave-particle interactions, to account for these effects one requires a hot permittivity tensor, which is out of the scope of this thesis. This is the main reason most antenna codes use the cold plasma model as analysis is carried out at the scrape-off layer (SOL) as thermal effects can be neglected, whereas wave solvers for the plasma core region use a hot plasma model, especially if resonances occur.

Chapter 3

Finite Elements Method

In the present chapter, we will present the theoretical framework used for the numerical simulations performed in this work. The Finite Elements Method is a well-established numerical tool, that allows solving linear and non-linear partial differential equations (PDE) in a physical domain, limited by a set of boundary conditions. These kinds of problems are known as *boundary value problem* (BVP). The algorithm is not limited to a particular domain shape or size nor a set of PDEs, allowing it to be highly adaptable and applicable to different fields of physics. The Finite Elements Method was first used in the 1940's decade by Alexander Hrennikoff and Richard Courant, and it was initially applied to the calculation of mechanical stress and elastic deformations in continuous solids [34, 35].

3.1 General framework

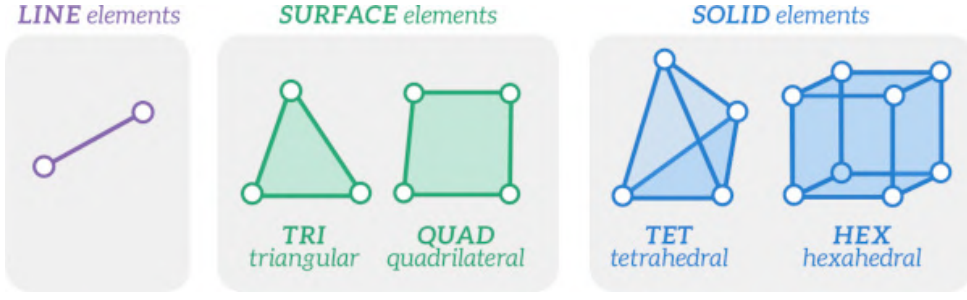
The main principle of FEM is the division of a computational domain into a set of *finite elements* (fig. 3.1a), building up a mesh structure where our differential equation will be solved. Every element has a series of nodes, or coordinate points, depending on the element's shape and the order of approximation. For example, the most common elements in use are triangles and quadrilaterals for 2D meshes or tetrahedra and hexahedra for 3D meshes. Within an element, the unknown function to be found is represented as a linear combination of a series of basis functions. These basis functions are usually called *shape functions* and can be linear, quadratic, or even higher polynomials. Within each element, a local system of algebraic equations is built, and using the connectivity between elements, a global set of linear equations is constructed. The number of nodes for a given element depends only on the polynomial order of the shape functions; for example, an element can be linear or quadratic with a triangular or quadrilateral shape (fig. 3.1b). Higher order elements do also exist, such as cubic elements or a specific case of the quadrilateral quadratic element, called the Lagrangian element.

One of the most simple methods to be used when the field equation and boundary conditions are known, as in a BVP problem, is the called Galerkin method [37, 38]. In particular, and when the problem allows it, by using the same weighted function used to approximate the elements we can reduce the problem to a well-conditioned linear system.

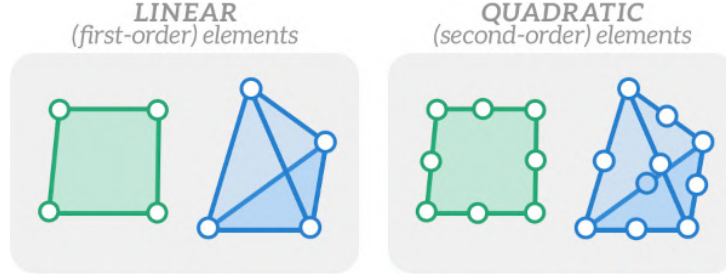
A typical BVP can be represented as

$$L u = f \tag{3.1}$$

where L is a partial differential operator, f is the non-homogeneous term and u is the unknown



(a) Most common shapes of finite elements; in 1D, 2D and 3D.



(b) Linear elements have nodes on their vertices, while quadratic elements also have nodes on their edges.

Figure 3.1: Overview of types of elements and their shapes. Figures extracted from [36].

function to be determined. Also, the boundary enclosing the domain of the differential equation is called Γ . Some examples of relevant and well-known PDE that govern BVP are

- Transport equation $L = \partial_t + c\partial_x$
- Wave equation $L = \partial_t^2 - c^2\partial_x^2$
- Heat equation $L = \partial_t - k\partial_x^2$
- Laplace equation $L = \nabla^2$

Since our domain is decomposed in several elements, we can express eq. (3.1) locally, restricted to the domain of a given element

$$L u^e = f^e \quad (3.2)$$

where, u^e and f^e are the unknown and independent functions of the e -th element. u^e is approximated as a series of weighted functions N_j^e , also called shape functions (eq. (3.3)). These functions have special properties that will be explained later, but it is important to mention that eq. (3.2) will only be fulfilled by the exact solution. Once we approximate the unknown function to a linear combination of shape functions, eq. (3.2) will be consequently fulfilled only approximately. For this reason, we can define the residual $R^e = L u^e - f^e$, which ultimately is aimed to be minimised.

$$u^e = \sum_{j=1}^n u_j^e N_j^e \quad (3.3)$$

So the goal would be to find these u_j^e that minimise the residual, that means $R^e \approx 0$. Then, instead of finding a solution for $R^e = 0$, we look for solutions that *on average* are equal to zero, by evaluating its inner product with respect to a series of weight functions w_i^e , where i

always go from 1 to n (being n the number of nodes per element) [37]. This ensures that the number of unknowns and equations is the same.

$$\langle R^e, w_i^e \rangle = \int_{\Omega^e} R^e w_i^e d\Omega^e = 0 \quad (3.4)$$

This method is called *weighted residuals*. w_i^e can be of different nature, and corresponding to their shapes they can be named as:

- Collocation method: $w_i^e = \delta(x - x_i)$
- Subsectional collocation method: w_i^e is nonzero in a subdomain within its element.
- Garlekin method: $w_i^e = N_i^e$
- Petrov-Garlekin method: $w_i^e \neq N_i^e$
- Least squares method: $w_i^e = L N_i^e$

In this work, we have used the Garlekin method because it is the simplest and most adequate for the conditioning of the equations we need to solve. With this approach, eq. (3.4) can be transformed to

$$\int_{\Omega^e} w_i^e L \left(\sum_{j=1}^n u_j^e N_j^e \right) d\Omega^e = \int_{\Omega^e} w_i^e f^e d\Omega^e \quad (3.5)$$

$$\sum_{j=1}^n u_j^e \left(\int_{\Omega^e} N_i^e L N_j^e d\Omega^e \right) = \int_{\Omega^e} N_i^e f^e d\Omega^e \quad (3.6)$$

Generally, the differential operator L can be of second order, and this will be the case for many of the electromagnetic problems that we will be focused on solving. Thankfully, by integrating by parts — or its 2D and 3D generalization —, we can reduce these second-order derivatives to first-order derivatives. This new way of expressing eq. (3.6) is called *weak form*. In any case, one can already visualise in eq. (3.6) the expression of a linear system of equations such as

$$\sum_{j=1}^n u_j^e (a^e)_j^i = b_i^e \quad (3.7a)$$

$$(a^e)_j^i = \int_{\Omega^e} N_i^e L N_j^e d\Omega^e \quad (3.7b)$$

$$b_i^e = \int_{\Omega^e} N_i^e f^e d\Omega^e \quad (3.7c)$$

or expressed in a matrix equation formulation

$$\mathbf{A}^e \mathbf{u}^e = \mathbf{b}^e \quad (3.8)$$

3.2 Matrix assembly

But the equations shown so far, such as eq. (3.8), are specific to a given element from the full mesh or domain. To compute the full unknown u , we will need to assemble all of the local matrix equations, resulting in a linear system with as many unknowns as the number of nodes

composes our mesh. Consequently, we will obtain a linear system of size $N \times N$ represented as:

$$A_{i_g j_g} \longleftarrow A_{i_g j_g} + A_{ij}^e \quad (3.9a)$$

$$b_{i_g} \longleftarrow b_{i_g} + b_i^e \quad (3.9b)$$

$$\mathbf{A} \mathbf{u} = \mathbf{b} \quad (3.9c)$$

Where eqs. (3.9a) and (3.9b) shows the assembly process (i_g and j_g are the global indices, and i and j are the local indices) and eq. (3.9c) is the global system of equations. Local indexes will always be numbered from 1 to n , being n the number of nodes per element, while the global indexes can be numbered from 1 to N , being N the total number of nodes in the whole mesh. Clearly, a node can have an arbitrary global index, while its local index will be limited to a small set of possibilities. The relation between local and global indexes is stored in the *connectivity matrix* or connectivity table.

M_e	n_1	n_2	n_3
1	11611	11676	11577
2	12042	11975	11926
3	2562	2520	2591
$\vdots$	$\vdots$	$\vdots$	$\vdots$
1500	7584	7410	7488
$\vdots$	$\vdots$	$\vdots$	$\vdots$
M	n_1	n_2	n_3

Table 3.1: Example of connectivity matrix for a linear triangular element (only three nodes).

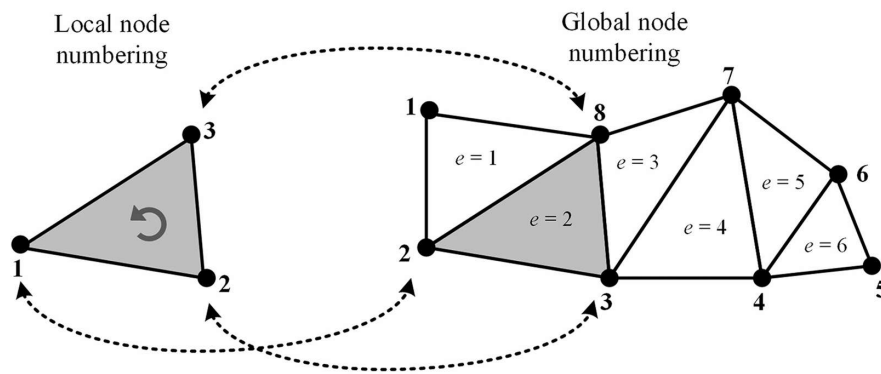


Figure 3.2: Visual representation between local and global numbering. Extracted from [21].

In table 3.1, it is represented a real case example of a connectivity matrix, where M_e is the element numbering; n_1 , n_2 and n_3 columns are the first, second, and third local indices and the numbers within the table are the corresponding nodes in the global numbering.

Since a given node will be only in contact with a reduced number of other nodes, the assembly process shown in eq. (3.9a) will result in a *sparse matrix*. Sparse matrices present

the characteristic of having non-zero values in their diagonal and null values in most of their non-diagonal coefficients. This allows for saving up a lot of memory and computational time, as all zero-valued elements do not need to be stored if the right method is used. Some of the most common formats to store sparse matrices are *Compressed Sparse Row* (CSR) and *Compressed Sparse Column* (CSC). MAXWEL code uses CSR format, so we will focus on explaining only this method, but CSC is analogous to the same principle.

Imagine an arbitrary shape matrix $\mathbf{A}$. There will be at least 3 arrays needed to store its values; one that stores the numerical values of $\mathbf{A}$, a second to store the column at which the values are located, and the third, which encodes the rows at which non-zero values are found. Optionally, the array that stores the numerical values of $\mathbf{A}$, can be split in two; one for diagonal terms and another for non-diagonal terms. As a little example, let us say that $\mathbf{A}$ is the non-zero values array of a 4 by 4 matrix $\mathbf{A}$

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 5 & 8 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 6 & 0 & 0 \end{pmatrix} \quad (3.10a)$$

$$\mathbf{A} = \begin{bmatrix} 5 & 8 & 3 & 6 \end{bmatrix} \quad (3.10b)$$

$$\mathbf{JA} = \begin{bmatrix} 0 & 1 & 2 & 1 \end{bmatrix} \quad (3.10c)$$

$$\mathbf{IA} = \begin{bmatrix} 0 & 0 & 2 & 3 & 4 \end{bmatrix} \quad (3.10d)$$

$\mathbf{JA}$ simply indicates column indices of elements found in $\mathbf{A}$. And $\mathbf{IA}$ is the cumulative number of non-zero values going row by row. So to extract a row of data from $\mathbf{A}$, lets say $\text{row} = 1$, we simply:

$$\begin{aligned} \text{row_start} &= \mathbf{IA}[\text{row}] \\ \text{row_end} &= \mathbf{IA}[\text{row} + 1] \end{aligned}$$

that is $\text{row_start} = 0$ and $\text{row_end} = 2$, so the extracted values are $\mathbf{A}[0 : 2] = \begin{bmatrix} 5 & 8 \end{bmatrix}$ with columns $\mathbf{JA}[0 : 2] = \begin{bmatrix} 0 & 1 \end{bmatrix}$.

3.3 Element categories

As shown in fig. 3.1, there are different types of elements, according to their shape, dimensions, and approximation order. In this section, we will present in more depth the properties of these elements and the particularities of their shape functions N_j^e . We will limit the explanation to bi-dimensional elements, since 1D elements are of little interest to our particular problem, and 3D elements are yet not implemented in order to solve a 3D wave equation.

Elements in an arbitrary domain can have arbitrary coordinates in each of their nodes and vertices. However, always exist a linear transformation from the physical domain (x, y) to a *master domain* (ξ, η) , such that in the master domain $(\xi, \eta) \in [0, 1]$. Also, we may omit the e notation since is evident that shape functions are not globally defined but element-wise.

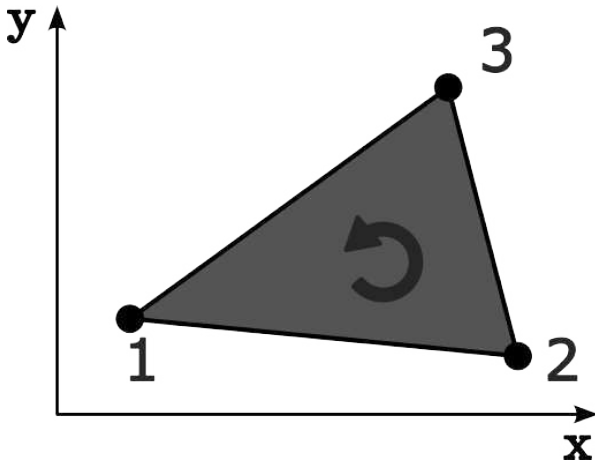
3.3.1 Linear Triangular element

Linear Triangular elements present only 3 nodes, one in each vertex, and they have straight edges. As a general convention for all element types, the nodes are labelled in the anti-clockwise direction. Each node has an associated shape function because one of the main properties of all given elements is that

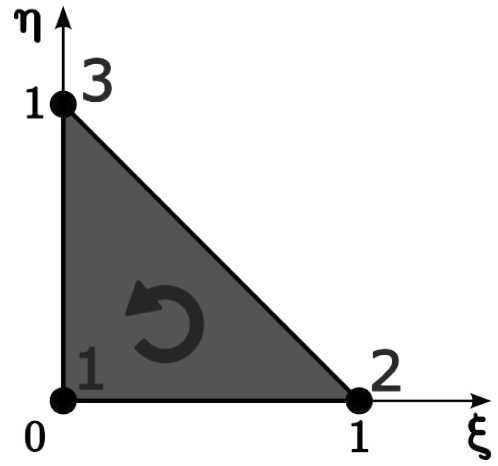
$$N_j(\xi_i, \eta_i) = \delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \quad (3.11)$$

so $N_1 = 1$ at node 1, $N_2 = 1$ at node 2, $N_3 = 1$ at node 3, and so on for higher order elements. Another essential property for shape functions is that the sum of all N_j within an element must add up to 1.

$$\sum_{j=1}^n N_j(\xi, \eta) = 1 \quad (3.12)$$



(a) Linear triangular element in an arbitrary domain.



(b) Linear triangular element in the master domain.

Figure 3.3: There is a linear transformation from a conventional domain (x, y) to the master domain (ξ, η) .

But to know the exact form of this shape function we need to derive them. Since we are using a linear element, we know that the order of our solutions u^e and the order of the shape functions will be one, and consequently they will only depend linearly with x and y , or with ξ and η . To take advantage of knowing the exact coordinates of nodes in the master domain will help computing the shape functions; and later, it will be just needed to do an inverse mapping to the conventional domain.

Considering a solution in the form of

$$u^e(\xi, \eta) = c_1 + c_2\xi + c_3\eta \quad (3.13)$$

which are evaluated at the nodes in the master domain:

$$\begin{cases} u^e(\xi_1, \eta_1) = u_1^e = c_1 + c_2 \cdot 0 + c_3 \cdot 0 = c_1 \\ u^e(\xi_2, \eta_2) = u_2^e = c_1 + c_2 \cdot 1 + c_3 \cdot 0 = c_1 + c_2 \\ u^e(\xi_3, \eta_3) = u_3^e = c_1 + c_2 \cdot 0 + c_3 \cdot 1 = c_1 + c_3 \end{cases} \quad (3.14)$$

so the coefficient values are

$$c_1 = u_1^e \quad (3.15a)$$

$$c_2 = -u_1^e + u_2^e \quad (3.15b)$$

$$c_3 = -u_1^e + u_3^e \quad (3.15c)$$

and substituting this coefficients in eq. (3.13) and re-arranging the terms, it is obtained

$$u^e(\xi, \eta) = (1 - \xi - \eta) u_1^e + \xi u_2^e + \eta u_3^e \quad (3.16)$$

which comparing with eq. (3.3), we can easily identify

$$N_1 = 1 - \xi - \eta \quad (3.17a)$$

$$N_2 = \xi \quad (3.17b)$$

$$N_3 = \eta \quad (3.17c)$$

The mapped coordinates also follow the same relationship due to the linear transformation

$$x(\xi, \eta) = \sum_j x_j^e N_j(\xi, \eta) = (1 - \xi - \eta) x_1^e + \xi x_2^e + \eta x_3^e \quad (3.18a)$$

$$y(\xi, \eta) = \sum_j y_j^e N_j(\xi, \eta) = (1 - \xi - \eta) y_1^e + \xi y_2^e + \eta y_3^e \quad (3.18b)$$

As it has been shown in eq. (3.6) and other equations, differential operators will be applied over N_j in the weak form of our differential equation. That means that we will need to compute the derivatives over N_j . These derivatives might be useful as well at other stages of the derivation, such as post-processing the solution. The derivatives $\partial_\xi N_j$ and $\partial_\eta N_j$ in the master domain are rather trivial, and using the Jacobian matrix (3.19), we can easily compute the derivatives in the conventional domain $\partial_x N_j$ and $\partial_y N_j$.

$$\mathbf{J}_N = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{pmatrix} = \begin{pmatrix} x_2^e - x_1^e & y_2^e - y_1^e \\ x_3^e - x_1^e & y_3^e - y_1^e \end{pmatrix} \quad (3.19)$$

Consequently

$$\begin{pmatrix} \frac{\partial N_j}{\partial \xi} \\ \frac{\partial N_j}{\partial \eta} \end{pmatrix} = \mathbf{J}_N \begin{pmatrix} \frac{\partial N_j}{\partial x} \\ \frac{\partial N_j}{\partial y} \end{pmatrix} \quad (3.20)$$

$$\begin{pmatrix} \frac{\partial N_j}{\partial x} \\ \frac{\partial N_j}{\partial y} \end{pmatrix} = \mathbf{J}_N^{-1} \begin{pmatrix} \frac{\partial N_j}{\partial \xi} \\ \frac{\partial N_j}{\partial \eta} \end{pmatrix} \quad (3.21)$$

So, in the case of linear triangular elements

$$\begin{pmatrix} \frac{\partial N_1}{\partial \xi} \\ \frac{\partial N_1}{\partial \eta} \end{pmatrix} = \frac{1}{|J_N|} \begin{pmatrix} y_2^e - y_3^e \\ x_2^e - x_3^e \end{pmatrix} \quad (3.22a)$$

$$\begin{pmatrix} \frac{\partial N_2}{\partial \xi} \\ \frac{\partial N_2}{\partial \eta} \end{pmatrix} = \frac{1}{|J_N|} \begin{pmatrix} y_3^e - y_1^e \\ x_3^e - x_1^e \end{pmatrix} \quad (3.22b)$$

$$\begin{pmatrix} \frac{\partial N_3}{\partial \xi} \\ \frac{\partial N_3}{\partial \eta} \end{pmatrix} = \frac{1}{|J_N|} \begin{pmatrix} y_1^e - y_2^e \\ x_1^e - x_2^e \end{pmatrix} \quad (3.22c)$$

We can observe that the derivatives of the shape functions are constants and only depend on the nodes' coordinates. That facilitates the integration of the weak form equation, compared to quadrilateral or higher order elements.

3.3.2 Quadratic Triangular Elements

Quadratic Triangular elements present 6 nodes in total; one in each vertex, and one in every edge. The higher-order shape functions allow to present curvilinear and not necessarily straight edges. Despite nodes being labelled in the anti-clockwise direction, as stated in section 3.3.1, there is an established numeration convention of labelling all vertices nodes first and then all the edge nodes. Properties shown in eqs. (3.11) and (3.12) apply to these elements too.

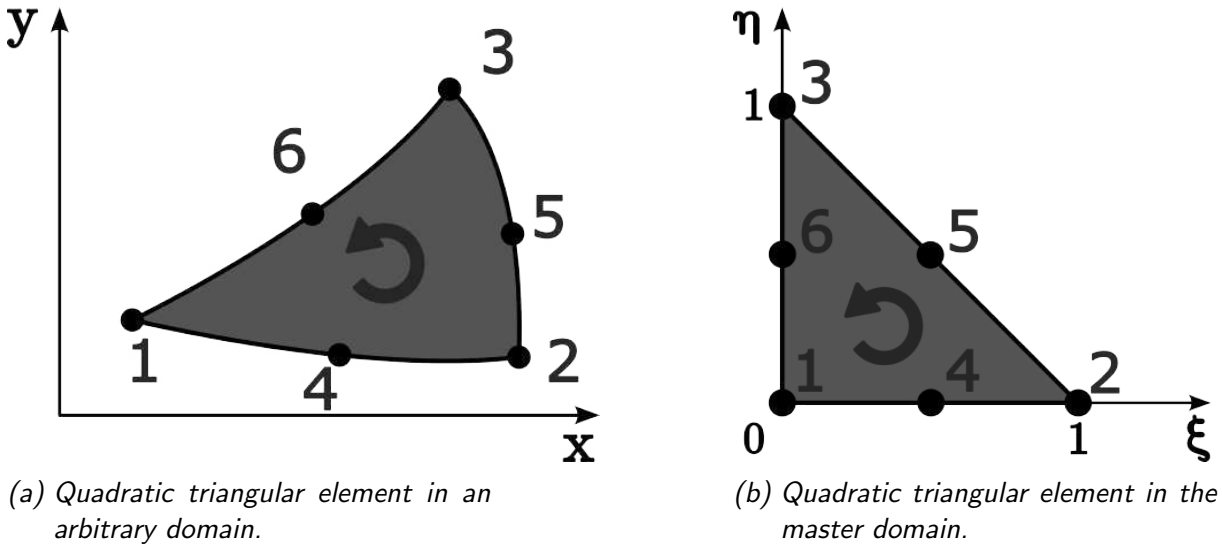


Figure 3.4: Now elements can be curvilinear in (x, y) but they are mapped to the master domain (ξ, η) using the same concept.

Now, we consider a solution of the type

$$u^e(\xi, \eta) = c_1 + c_2\xi + c_3\eta + c_4\xi\eta + c_5\xi^2 + c_6\eta^2 \quad (3.23)$$

and doing the same procedure as in section 3.3.1, it is obtained the corresponding shape functions:

$$N_1 = (1 - \xi - \eta)(1 - 2\xi - 2\eta) \quad (3.24a)$$

$$N_2 = \xi(2\xi - 1) \quad (3.24b)$$

$$N_3 = \eta(2\eta - 1) \quad (3.24c)$$

$$N_4 = 4\xi(1 - \xi - \eta) \quad (3.24d)$$

$$N_5 = 4\xi\eta \quad (3.24e)$$

$$N_6 = 4\eta(1 - \xi - \eta) \quad (3.24f)$$

The derivatives $\partial_x N_j$ and $\partial_y N_j$ are computed directly using eq. (3.21).

3.3.3 Bilinear Quadrilateral Elements

Quadrilaterals are another widely used element shape. These elements present four nodes, one in each vertex. In this case, the solution has the polynomial form of

$$u^e(\xi, \eta) = c_1 + c_2\xi + c_3\eta + c_4\xi\eta \quad (3.25)$$

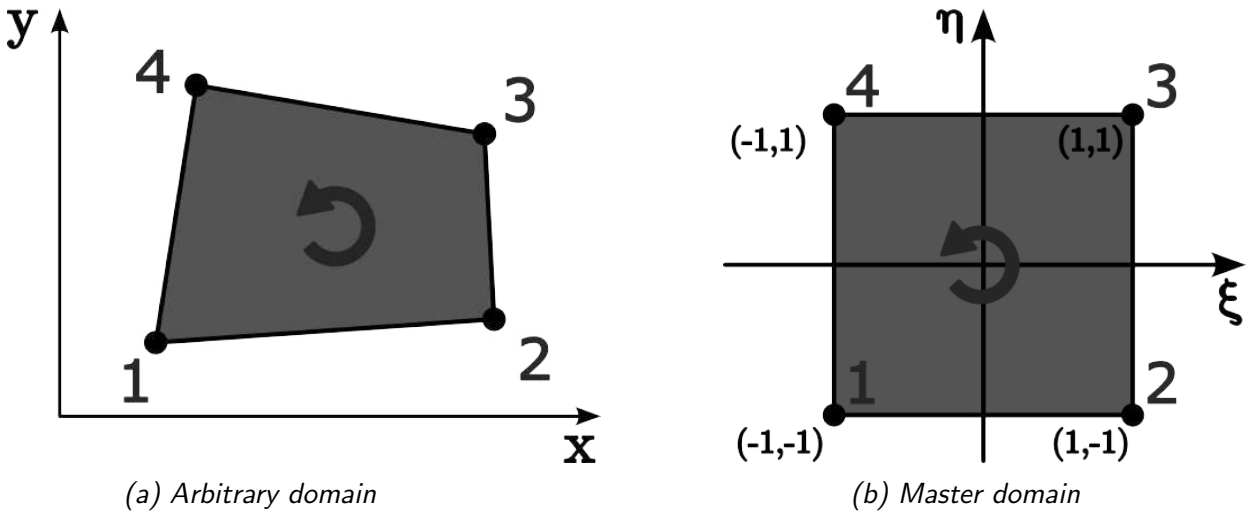


Figure 3.5: Schematic of a bilinear quadrilateral element.

Applying the same logic as in sections 3.3.1 and 3.3.2, now shape functions are

$$N_1 = \frac{1}{4} (1 - \xi) (1 - \eta) \quad (3.26a)$$

$$N_2 = \frac{1}{4} (1 + \xi) (1 - \eta) \quad (3.26b)$$

$$N_3 = \frac{1}{4} (1 + \xi) (1 + \eta) \quad (3.26c)$$

$$N_4 = \frac{1}{4} (1 - \xi) (1 + \eta) \quad (3.26d)$$

3.3.4 Quadratic Quadrilateral Element

There are two different quadratic quadrilateral elements, 8-noded elements (also known as *serendipity* element) and 9-noded elements (also named *Lagrangian* element). Fundamentally, serendipity elements have one node in each vertex, and one node in each edge. Lagrangian elements have an extra node in the centre. Here in this work, we will only present serendipity elements, since lagrangian elements are not implemented in MAXWEL software.

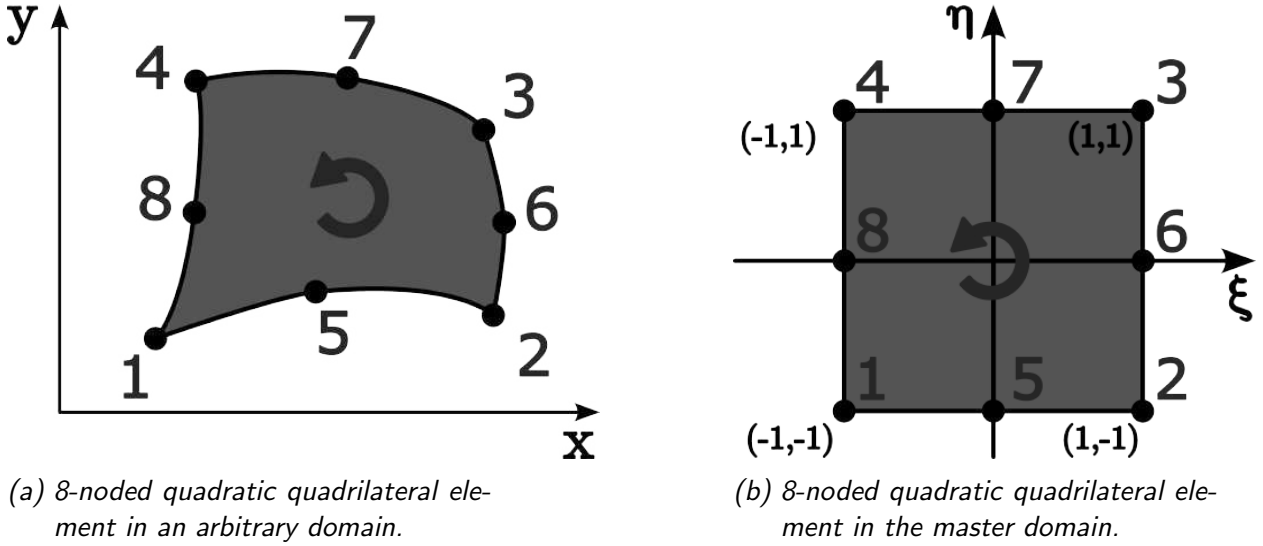


Figure 3.6: Serendipity elements can have curvilinear edges thanks to their higher-order polynomial approximation.

Shape functions for serendipity elements are:

$$N_1 = \frac{1}{4} (1 - \xi) (1 - \eta) (-\xi - \eta - 1) \quad (3.27a)$$

$$N_2 = \frac{1}{4} (1 + \xi) (1 - \eta) (\xi - \eta - 1) \quad (3.27b)$$

$$N_3 = \frac{1}{4} (1 + \xi) (1 + \eta) (\xi + \eta - 1) \quad (3.27c)$$

$$N_4 = \frac{1}{4} (1 - \xi) (1 + \eta) (-\xi + \eta - 1) \quad (3.27d)$$

$$N_5 = \frac{1}{2} (1 - \xi^2) (1 - \eta) \quad (3.27e)$$

$$N_6 = \frac{1}{2} (1 + \xi) (1 - \eta^2) \quad (3.27f)$$

$$N_7 = \frac{1}{2} (1 - \xi^2) (1 + \eta) \quad (3.27g)$$

$$N_8 = \frac{1}{2} (1 - \xi) (1 - \eta^2) \quad (3.27h)$$

3.4 Weak form in electromagnetism

Considering eq. (2.22) and using eq. (3.4) we can obtain an approximate solution for our wave equation. By choosing the appropriate weight functions w_i^e , and reducing the integrand to

first order derivatives (with integration methods such as Gauss' theorem, Stokes' theorem or integration by parts, and other generalisations), eq. (2.22) can be reduced to its weak form. To derive this weak form, we will consider firstly 3D elements, to simplify the procedure. Once it is in its final form, one can assume axial invariance (e.g. along z direction), so all the z derivatives will be 0, and reduce the integration domain to a two-dimensional element.

Combining eqs. (2.22) and (3.4) it is obtained the following expression

$$\iiint_{\Omega^e} w_i^e [-\nabla \cdot (\mathbf{p}^e \cdot \nabla u^e) + q^e u^e - f^e] dV = 0 \quad (3.28)$$

and using the identity $\nabla \cdot (\psi \mathbf{A}) = \psi \nabla \cdot \mathbf{A} + \mathbf{A} \cdot \nabla \psi$, where $\mathbf{A}$ and ψ are vector and scalar fields respectively, eq. (3.28) can be turned to:

$$\iiint_{\Omega^e} [(\mathbf{p}^e \cdot \nabla u^e) \cdot \nabla w_i^e - \nabla \cdot (w_i^e \mathbf{p}^e \cdot \nabla u^e) + q^e w_i^e u^e - w_i^e f^e] dV = 0 \quad (3.29)$$

Using Gauss' theorem, we can switch the second term from a volume integration to a surface integration, resulting in

$$\iiint_{\Omega^e} [(\mathbf{p}^e \cdot \nabla u^e) \cdot \nabla w_i^e + q^e w_i^e u^e] dV - \oint_{\Gamma^e} w_i^e (\mathbf{p}^e \cdot \nabla u^e) d\vec{S} - \iiint_{\Omega^e} w_i^e f^e dV = 0 \quad (3.30)$$

Now, considering z as an invariant axis, we can reduce the integration domain to a two-dimensional element:

$$\iint_{\Omega^e} [(\mathbf{p}^e \cdot \nabla u^e) \cdot \nabla w_i^e + q^e w_i^e u^e] dS - \oint_{\Gamma^e} w_i^e (\mathbf{p}^e \cdot \nabla u^e) d\vec{l} - \iint_{\Omega^e} w_i^e f^e dS = 0 \quad (3.31)$$

Equation (3.31) is the weak form expression, valid for all 2D surface elements within a mesh. However, for interior elements, the line integral along their edged cancels out with their adjacent neighbour ($\vec{l}_1 = -\vec{l}_2$). Consequently, the line integral can be omitted for interior elements and only needs to be taken into account for elements having at least one external edge, where boundary conditions must be enforced. Now, keep in mind that we apply the Galerkin method ($w_i^e = N_i$), and by decomposing our solution u^e such as in eq. (3.3):

$$\sum_{j=1}^n u_j^e \left[\iint_{\Omega^e} [(\mathbf{p}^e \cdot \nabla N_j) \cdot \nabla N_i + q^e N_i N_j] dS \right] = \iint_{\Omega^e} N_i f^e dS \quad (3.32)$$

From eq. (3.32) we can directly interpret the terms of the linear system to solve, where u_j^e are the unknowns, the left-hand side terms inside the bracket are the matrix elements and the right-hand side is the independent vector. Now, after substituting q^e , $\mathbf{p}^e$ and f^e from eqs. (2.23) and (2.24), eqs. (3.7b) and (3.7c) present the shape of

$$(a^e)_j^i = \iint_{\Omega} [(\mathbf{\Lambda}_\mu \cdot \nabla N_j) \cdot \nabla N_i - k_0^2 \varepsilon_{zz} N_i N_j] dS \quad (3.33a)$$

$$b_i^e = \iint_{\Omega^e} N_i \nabla \cdot (\mathbf{\Lambda}_\mu \cdot \nabla E_z^{\text{inc}}) + k_0^2 \mu_r^{zz} E_z^{\text{inc}} dS \quad (3.33b)$$

for TM polarisation and

$$(a^e)_j^i = \iint_{\Omega} [(\mathbf{\Lambda}_\varepsilon \cdot \nabla N_j) \cdot \nabla N_i - k_0^2 \mu_r^{zz} N_i N_j] dS \quad (3.34a)$$

$$b_i^e = \iint_{\Omega^e} N_i \nabla \cdot (\mathbf{\Lambda}_\varepsilon \cdot \nabla H_z^{\text{inc}}) + k_0^2 \mu_{zz} H_z^{\text{inc}} dS \quad (3.34b)$$

for TE polarisation.

Using the same logic shown in section 3.2, the following step would be to assemble the element matrices $\mathbf{A}^e \mathbf{u}^e = \mathbf{b}^e$ according to the information stored in the connectivity table, obtaining the full linear system $\mathbf{A} \mathbf{u} = \mathbf{b}$.

3.5 Solving the linear system

As mentioned in section 3.2, the element matrices are assembled to get a full-size matrix with as many rows and columns as nodes are in the mesh. These element matrices are assembled in such a way that the resulting linear system is called a sparse system, or sparse matrix. This is because the unknowns are only affected by the solution of the neighbouring nodes and not by nodes at a faraway distance. Consequently, all coefficients within the diagonal of the matrix will be necessarily non-zero and most of the non-diagonal coefficients will be zero.

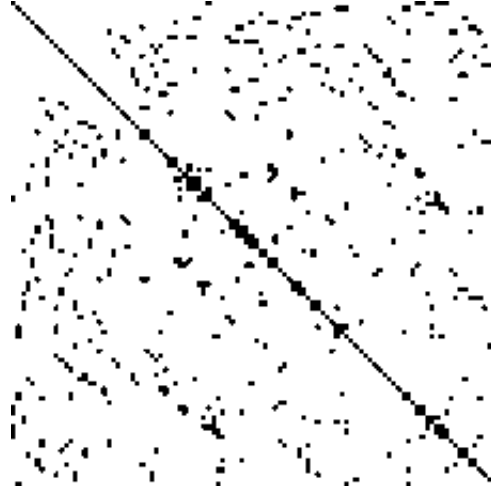


Figure 3.7: Visual example of non-zero coefficients in a sparse matrix.

There are several methods to solve $\mathbf{u} = \mathbf{A}^{-1} \mathbf{b}$; but when matrices are too large, some of them become inefficient and slow. For this reason, the solving method of such systems — where matrices may be 20.000, 60.000, or even 10^6 rows and columns — is quite critical. During the project, we have used two different solvers: *conjugate gradient method* and *bi-conjugate gradient method*. Despite the MAXWEL software being made from scratch by the author of this work, the solver module implemented in the solution is one of the few, if not the only, packages that have been made by a third party. This is mainly because the algorithm is more than 70 years old [39] and has already been described, implemented, and used before with much success.

3.5.1 Conjugate Gradient Method

Conjugate Gradient method (CG) is an iterative method able to solve a linear system such as $\mathbf{A} \mathbf{x} = \mathbf{b}$, assuming that $\mathbf{A}$ is **symmetric** and **positive definite**. The main idea or concept of this method is to define a residual vector such as $\mathbf{r}_0 = \mathbf{b} - \mathbf{A} \mathbf{x}_0$; if the error associated with

$\mathbf{r}$ is sufficiently small, then the solution is given as good. If not, $\mathbf{x}$ and $\mathbf{r}$ are modified such as

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k \quad (3.35)$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k + \alpha_k \mathbf{A} \mathbf{p}_k \quad (3.36)$$

$$\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k \quad (3.37)$$

$$\alpha_k = \frac{\mathbf{r}_k^\top \mathbf{r}_k}{\mathbf{p}_k^\top \mathbf{A} \mathbf{p}_k} \quad (3.38)$$

$$\beta_k = \frac{\mathbf{r}_{k+1}^\top \mathbf{r}_{k+1}}{\mathbf{r}_k^\top \mathbf{r}_k} \quad (3.39)$$

where k is the iteration number. $\mathbf{p}_k$ are the *conjugate gradients* vectors, and on the first iteration $\mathbf{p}_0 = \mathbf{r}_0$. This method is called precisely conjugate gradients because $\mathbf{p}_i^\top \mathbf{A} \mathbf{p}_j = 0$ for $i \neq j$ (that is *A-conjugated*), and because $\mathbf{p}_k$ is the direction that minimises the functional f .

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^\top \mathbf{A} \mathbf{x} - \mathbf{b}^\top \mathbf{x} \quad (3.40)$$

$$\nabla f(\mathbf{x}) = \mathbf{A} \mathbf{x} - \mathbf{b} \quad (3.41)$$

So $\nabla f(\mathbf{x}_k + \alpha_k \mathbf{p}_k) \rightarrow 0$ after enough iterations. We can assure that there is only one possible solution to $\nabla f = 0$ since the Hessian matrix of f is $\mathbf{H}(f) = \mathbf{A}$, which is symmetric positive-definite.

3.5.2 Bi-conjugate Gradient Method

The Bi-conjugate Gradient method [40] (Bi-CG) is a more general case of the CG method, in which $\mathbf{A}$ it is not necessarily symmetric and positive definite. In this method, there are two sets of residual vectors $\mathbf{r}_k, \bar{\mathbf{r}}_k$ and gradient conjugate vectors $\mathbf{p}_k, \bar{\mathbf{p}}_k$. Here, the difference between these vectors are

$$\mathbf{r}_0 = \mathbf{b} - \mathbf{A} \mathbf{x}_0 \quad (3.42)$$

$$\bar{\mathbf{r}}_0 = \mathbf{b} - \mathbf{A}^\top \mathbf{x}_0 \quad (3.43)$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k + \alpha_k \mathbf{A} \mathbf{p}_k \quad (3.44)$$

$$\bar{\mathbf{r}}_{k+1} = \bar{\mathbf{r}}_k + \alpha_k \mathbf{A}^\top \bar{\mathbf{p}}_k \quad (3.45)$$

$$\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k \quad (3.46)$$

$$\bar{\mathbf{p}}_{k+1} = \bar{\mathbf{r}}_{k+1} + \beta_k \bar{\mathbf{p}}_k \quad (3.47)$$

$$\alpha_k = \frac{\bar{\mathbf{r}}_k^\top \mathbf{r}_k}{\bar{\mathbf{p}}_k^\top \mathbf{A} \mathbf{p}_k} \quad (3.48)$$

$$\beta_k = \frac{\bar{\mathbf{r}}_{k+1}^\top \mathbf{r}_{k+1}}{\bar{\mathbf{r}}_k^\top \mathbf{r}_k} \quad (3.49)$$

As it was mentioned in the CG method $\mathbf{p}_0 = \mathbf{r}_0$, and similarly $\bar{\mathbf{p}}_0 = \bar{\mathbf{r}}_0$. Vectors $\mathbf{r}_k, \bar{\mathbf{r}}_k, \mathbf{p}_k$ and $\bar{\mathbf{p}}_k$ present *bi-orthogonality*:

$$\bar{\mathbf{r}}_i \cdot \mathbf{r}_j = \mathbf{r}_i \cdot \bar{\mathbf{r}}_j = 0 \quad j < i \quad (3.50)$$

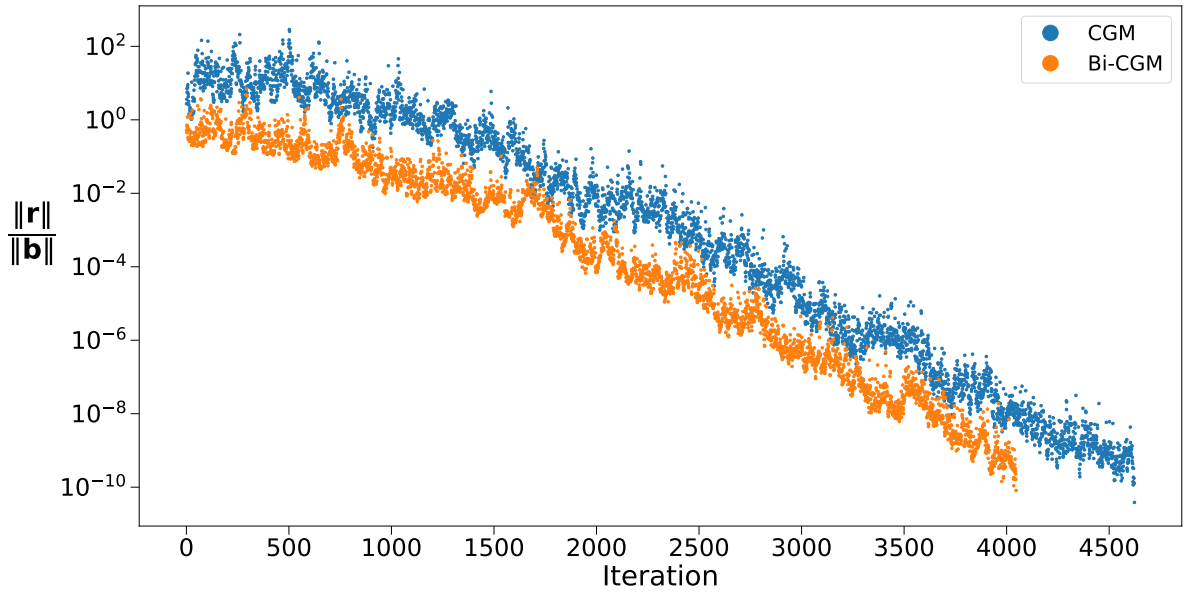
bi-conjugacy:

$$\bar{\mathbf{p}}_i \cdot \mathbf{A} \cdot \mathbf{p}_j = \mathbf{p}_i \cdot \mathbf{A}^\top \cdot \bar{\mathbf{p}}_j = 0 \quad j < i \quad (3.51)$$

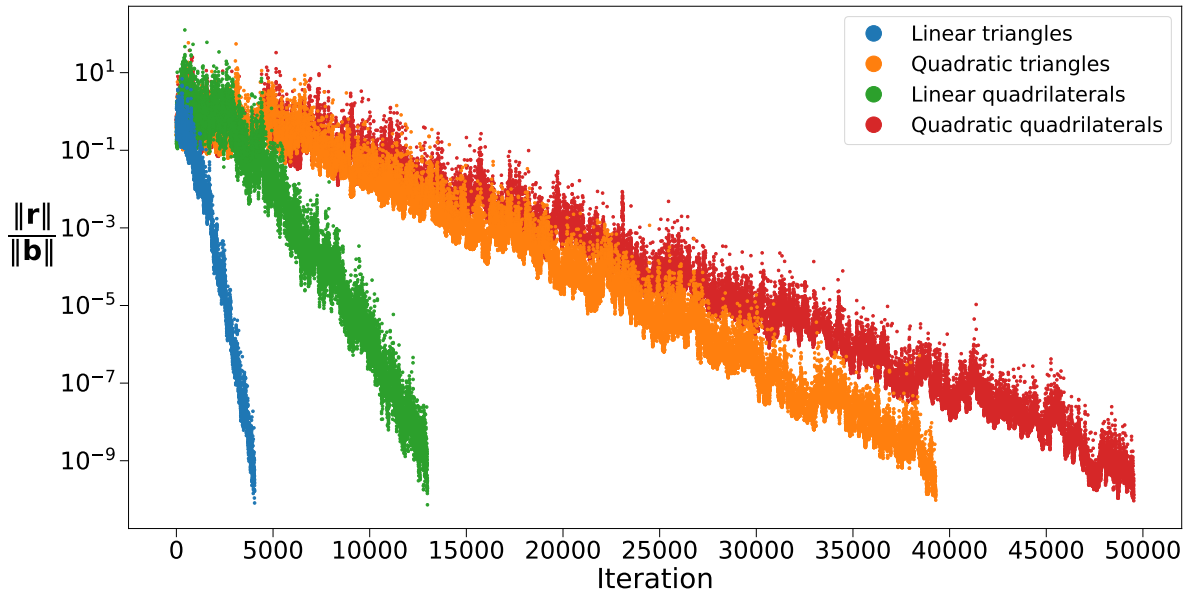
and *mutual orthogonality* properties:

$$\bar{\mathbf{r}}_i \cdot \mathbf{p}_j = \mathbf{r}_i \cdot \bar{\mathbf{p}}_j = 0 \quad j < i \quad (3.52)$$

In fig. 3.8 it can be appreciated, for a particular example, how after several iterations, the error decreases significantly. Figure 3.8a shows that Bi-CGM has a slightly better performance, as it allows to reduce the number of iterations by roughly 500. However, despite higher order elements providing better accuracy in the results, in fig. 3.8b is seen that a bigger number of nodes has a greater computational cost; hence the solution takes longer to converge. Other solving methods could be potentially suitable to our type of problems; but since Alya presents many solvers in its kernel and it is expected that MAXWEL will work on Alya in the near future, we have decided to use these two methods for now.



(a) Convergence speed for CGM and Bi-CGM.



(b) Convergence speed of Bi-CGM for different types of meshing elements.

Figure 3.8: Error evolution for every solver iteration.

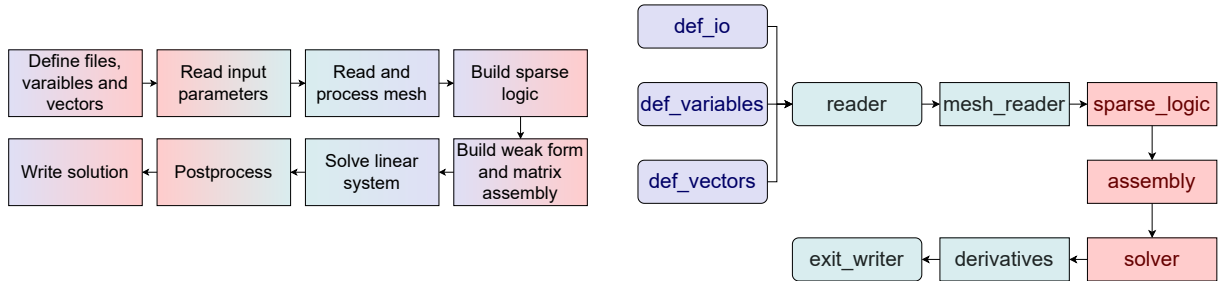
Chapter 4

Code Development and Validation

In this chapter, we will explain how the MAXWEL code works and present some of the results obtained by its simulations. As it has been mentioned before, this code was made from scratch but inspired by the work in [21]. MAXWEL code has been written in Fortran, due to its high speed and to be included in Alya [23, 24], where most of the code is written in Fortran.

4.1 Code description and functioning

MAXWEL is essentially a FEM code, capable of solving the Helmholtz equation in a non-homogeneous, anisotropic media. To do so, the software is split into several modules, each with different tasks and results. Here, we will see the working principle of those most relevant modules in more detail.



(a) Logical flowchart of MAXWEL framework.

(b) Flowchart of relevant MAXWEL modules.

Figure 4.1: Schematic process of MAXWEL software. On the left: Overall logic of the code. On the right: Main modules working on the code.

4.1.1 Definition and input values

A distinguished characteristic of compiled languages such as Fortran, compared to interpreted languages, is that we need to declare a variable before using it. As one can imagine, a code with hundreds or even thousands of lines and several modules and packages, may need many variables to run smoothly and to be understood by the developers. For this reason, the main variables used in MAXWEL are defined in three modules. One for parameters related to file management, writing, and reading; another for scalar parameters, and a third for dimensional parameters, such as vectors and matrices. However, some of these parameters might be desired

to change from simulation to simulation, and hard-coding their values is usually not a good practice. For that purpose, there is a module called `reader`, which takes a file input containing several parameters relevant for experimentation. Some of these parameters are:

- `freq`: Wave frequency in MHz.
- `phii`: Angle of incidence in rad respect to the x axis.
- `pol`: Polarisation mode, 'TE' or 'TM'.
- `elem_type`: Element's order, either linear 'line' or quadratic 'quad'.
- `elem_shape`: Element's shape, either triangular 'tria' or quadrilateral 'squa'.
- `plasma`: Boolean flag whether the permittivity is computed as a cold plasma tensor or given as an input.
- `epsilon` and `mu`: 3×3 matrices with the numerical values for the relative permittivity and permeability, either real or complex values.

Some other parameters regarding the solver (method, tolerance, iterations...) and the mesh origin and structure can also be manually modified through this input file.

4.1.2 Handling the mesh

The domain in which our problem and equation will be solved, and how it is discretised, is a key aspect of FEM. Its structure and organisation must be thoroughly designed to accurately represent the physics of the problem to be solved. For example, the level of refinement (i.e. the element size) has a significant impact on the accuracy of the solution. The elements must be small enough to capture the wave's oscillation but without heavily compromising the execution time. Mainly, we have used circular and tokamak cross-section geometries, to demonstrate MAXWEL feasibility, although other geometries have also been tested.

To produce these meshes two different software were used. Firstly, we modified a meshing function provided in [21], that uses the principle of Delaunay triangulation [41] to discretise a given domain; but at some point, this script was limiting us to a certain number of geometries. Since our goal was to simulate the wave propagation within a tokamak-like shape reactor, we moved toward a more dedicated software, designed to mesh arbitrary domains. GiD [42] is a professional software able to mesh highly complex geometries with a high level of customisation. Ultimately, this tool was used to correctly mesh the geometry of a tokamak cross-section. The tokamak geometry has been extracted from EQUILI[28] software, being currently under development at BSC's Fusion group. EQUILI solves the Grad-Shafranov equation modelling the equilibrium in an axisymmetric plasma using CutFEM, an unfitted mesh Finite Element Method. By solving the Grad-Shafranov equation, one obtains the toroidal magnetic flux surfaces, which is shown in fig. 4.2.

Despite the geometries that have been tested so far are not extremely complex, they need particular settings to work properly in MAXWEL. The most important aspect of meshes is that we need at least two types of materials within the mesh —specified with a numerical label— and every boundary must be labelled too (either boundaries between materials or outer boundaries). These two materials are the inner scattering media and an outermost *locally conformal Perfectly Matched Layer* (LC-PML) region [43]. Generally, we add an extra layer of vacuum in between, to help visualise the wave propagation.

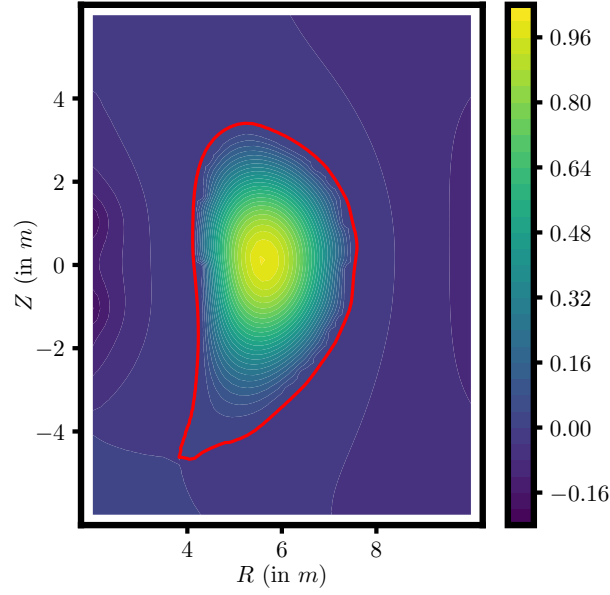


Figure 4.2: Toroidal magnetic flux computed by EQUILI with ITER parameters. The red line corresponds to the separatrix.

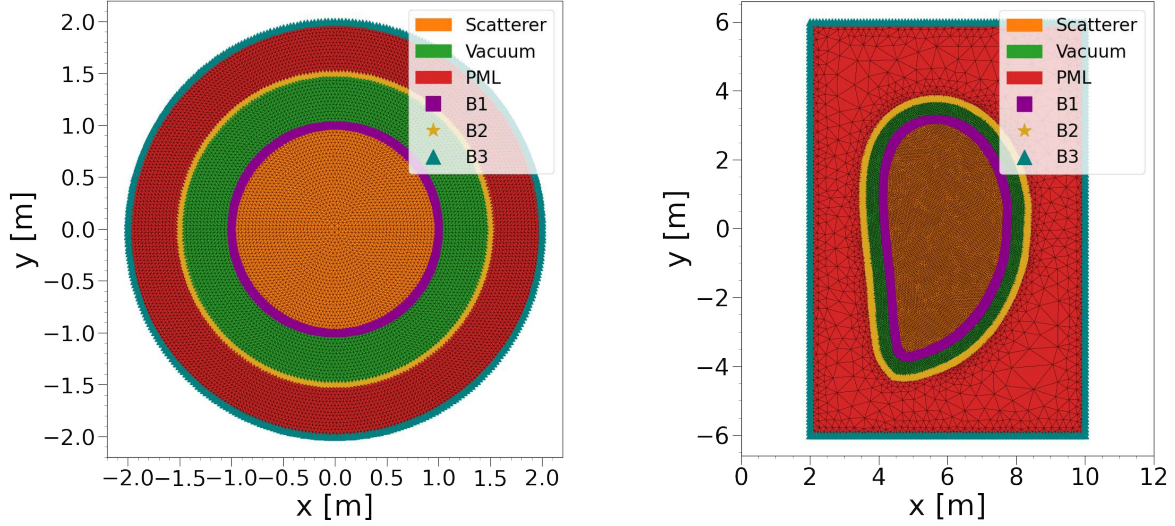
PML domains have been widely used for mesh truncation in numerical simulations of electromagnetic radiation. The key concept is to map a real-valued domain to a complex domain. These complex coordinates introduced in the electromagnetic wave yield to non-Maxwellian fields, that are attenuated after a certain characteristic length. The *locally conformal* variation allows generating PML regions for difficult geometries with abrupt changes or discontinuities in their curvatures.

4.1.3 Building the linear system

After calculating the number of non-zero matrix coefficients and consequently defining the length of the matrix diagonal and non-diagonal arrays, it is time to use eqs. (3.33) and (3.34) to build our linear system of equations. To do so, it is first necessary to define the shape functions N_i and compute their Jacobian matrices and derivatives in the non-master (or conventional) domain. This part is critical for the functioning of the code, and since Fortran has no internal linear algebra libraries, debugging this part was frequently time-consuming. In this section of the code, there are the conditions defined to assign the correct values for permittivity and permeability according to the material type of the mesh element (see fig. 4.3) and the condition regarding the kind of permittivity tensor of the scattering material (either a conventional dielectric or a cold plasma media).

While the element matrices $(\mathbf{a}^e)_j^i$ are being computed, the whole system matrix is being iteratively self-assembled as in eqs. (3.9a) and (3.9b). One remark regarding the independent vector $\mathbf{b}^e$ and $\mathbf{b}$ is that f^e is essentially the same operator that it is applied to the scattered field $\mathbf{u}$, but applied to the incident wave $\mathbf{u}^{\text{inc}}$. Furthermore, being $\mathbf{u}^{\text{inc}}$ a unitary plane wave, when the medium is vacuum (that is $\epsilon_{rc} = \mu_r = 1$), f^e becomes 0. So the only non-zero f^e values are within the scattering nodes. For these two reasons, the independent vector $\mathbf{b}$ is built as

$$\begin{cases} \mathbf{b} = 0 & \text{in vacuum} \\ \mathbf{b} = -\mathbf{A}\mathbf{u}^{\text{inc}} & \text{in scattering media} \end{cases} \quad (4.1)$$



(a) Mesh distribution and parametrisation for a circular geometry.

(b) Mesh for a tokamak cross-section. Geometry extracted from EQUILI.

Figure 4.3: Different meshes that were introduced for simulation in MAXWEL. Beware the difference in sizes between the two.

4.1.4 Post-processing the solution

When using FEM, the output result of the differential equation might not be the final result one desires to obtain. Or maybe there is also interest in extracting extra insight from the solution. Being the latter our case, after we solve the full matrix system $\mathbf{A}\mathbf{u} = \mathbf{b}$ we are also interested in computing the in-plane fields from either H_z or E_z . From eqs. (2.7) and (2.8), we can calculate the in-plane fields as

$$\begin{pmatrix} H_x \\ H_y \end{pmatrix} = \frac{i}{\omega\mu_0} \boldsymbol{\mu}_r^{-1} \begin{pmatrix} \partial_y E_z \\ -\partial_x E_z \end{pmatrix} \quad (4.2)$$

for a TM polarisation, and

$$\begin{pmatrix} E_x \\ E_y \end{pmatrix} = -\frac{i}{\omega\epsilon_0} \boldsymbol{\epsilon}_{rc}^{-1} \begin{pmatrix} \partial_y H_z \\ -\partial_x H_z \end{pmatrix} \quad (4.3)$$

for a TE polarisation.

So in this module, the code computes the derivatives of the solution $\mathbf{u}$ by

$$\partial_j u = \sum_{i=1}^n u_i \partial_j N_i \quad (4.4)$$

where j indicates either the x or y direction and n is the number of nodes per element. In this way, we can extract the complementary in-plane field components, apart from the z component of the electromagnetic fields.

4.2 Results and Validation

The first results from MAXWEL were obtained using a homogeneous dielectric permittivity tensor, either isotropic or anisotropic. For these cases, the linear system resulted as positive

definite and symmetric, since the values outside the diagonal of ϵ_{rc} were zero. These results were benchmarked with Özgün and Kuzuoğlu book [21] since we were already using part of their work as reference. However, while testing the cold plasma tensor implementation, we observed that results were not consistent and varied abruptly depending on non-critical parameters, such as the order of the mesh elements or their shape. This led us to realise that the off-diagonal complex permittivity values in eq. (2.29) induced a non-symmetric linear system. Consequently, in this case, our initial solver did not obtain correct results. After implementing the Bi-CGM solver we benchmarked these results with a different source, the ERMES code [44].

4.2.1 Validations against literature problems

As mentioned before, at the initial stages of the project, we validated our code against several benchmarks provided by [21]. Some of them, listed below, are included in this text as examples and verification of our code:

- TE polarisation with $\epsilon_{rc} = 4$. Figures A.1 and A.2.
- TM polarisation with $\epsilon_{rc} = 4$. Figures A.3 and A.4.
- TE polarisation with $\epsilon_{xx} = 4$, $\epsilon_{yy} = 9$. Figures A.5 and A.6.
- TE polarisation with $\epsilon_{xx} = 9$, $\epsilon_{yy} = 4$. Figures 4.4 and 4.5.

These four cases have an electromagnetic wave of module 1 and a frequency of 300 MHz. One can easily compare the results between the ones obtained by MAXWEL and the ones obtained by [21] in the aforementioned figures. It can be appreciated how these excellent results are identical to the ones used as reference. The rougher discretisation from figs. 4.5, A.2, A.4 and A.6 might be only due to the visualisation tool, since we used a dedicated software called ParaView [45] to plot MAXWEL results. In fig. 4.6 can be appreciated how well the values are adjusted to the ones in literature, as the error between the two is negligible.

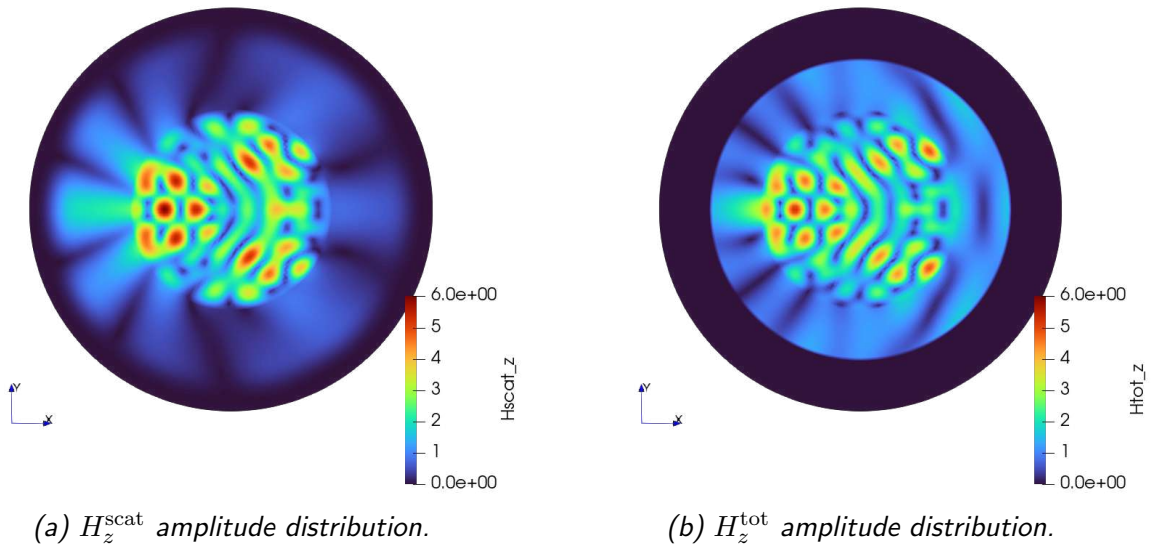


Figure 4.4: TE polarisation with $\epsilon_{xx} = 4$, $\epsilon_{yy} = 9$, simulated with MAXWEL.

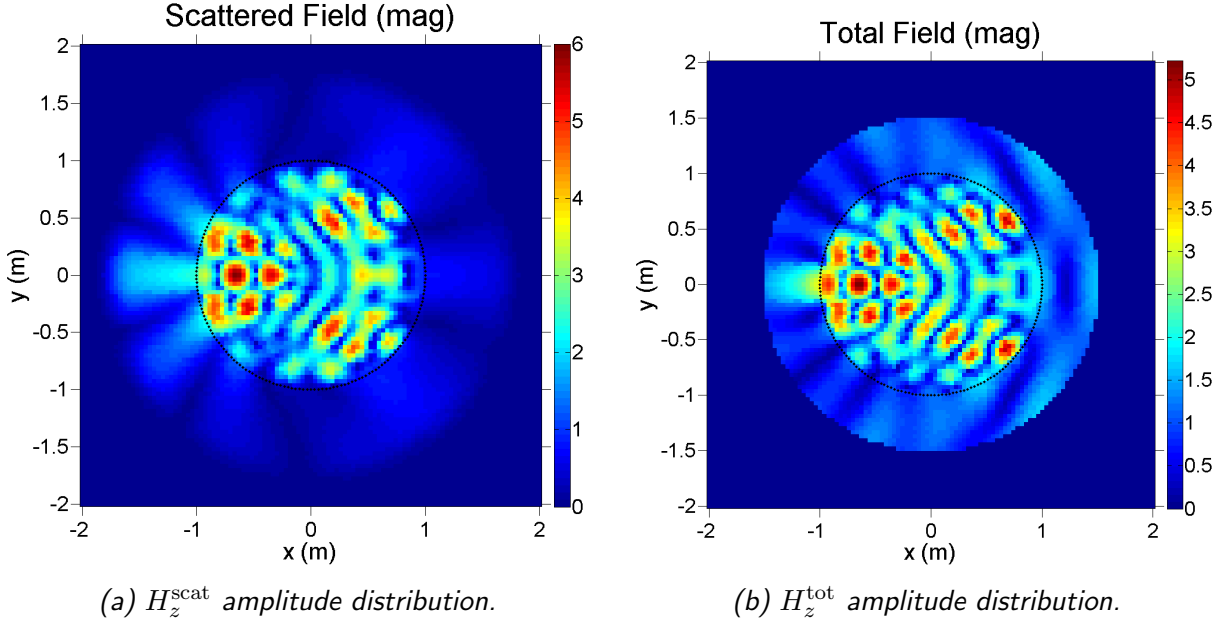


Figure 4.5: TE polarisation with $\varepsilon_{xx} = 9$, $\varepsilon_{yy} = 4$, from [21].

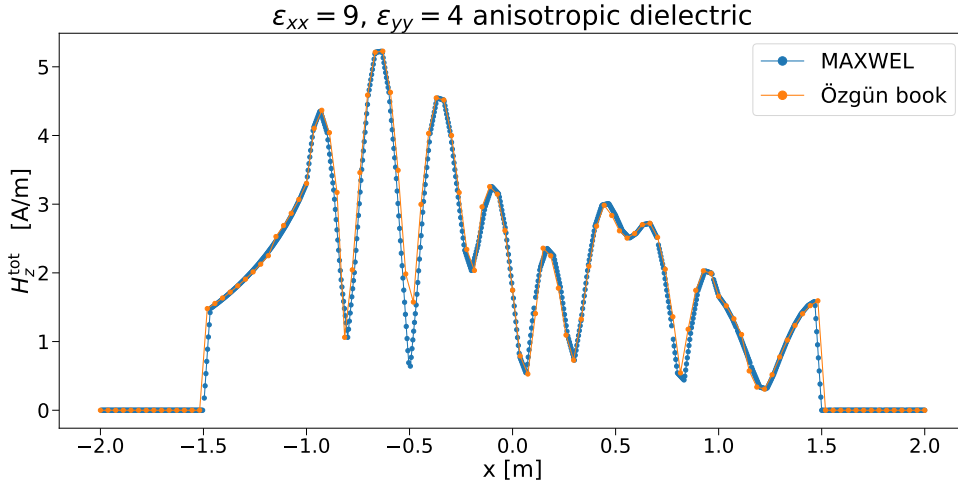


Figure 4.6: Amplitude of H_z^{tot} field across $y = 0$ diameter for an anisotropic dielectric case.

4.2.2 ERMES

Once we simulated complex and non-symmetric permittivity tensors, we used the code ERMES [44] (Electric Regularized Maxwell Equations with Singularities) to perform some benchmarks to validate our code with a robust and established code in the community. In this way, we compared the TE polarised isotropic case (fig. 4.7) and a particular case of a homogeneous cold plasma tensor (fig. 4.8). This was, in fact, really important to asses that MAXWEL was indeed working properly since both cases presented an exceptional agreement between the different codes. The cold plasma parameters are

$$\begin{aligned} R &= 24.521 & L &= -104.14 \\ D &= 64.329 & P &= -44811 & S &= -39.808 \end{aligned} \quad (4.5)$$

As seen in figs. 4.7 and 4.8, both results are qualitatively the same, despite ERMES working quite differently from our code. The simulations in ERMES were done with different meshes

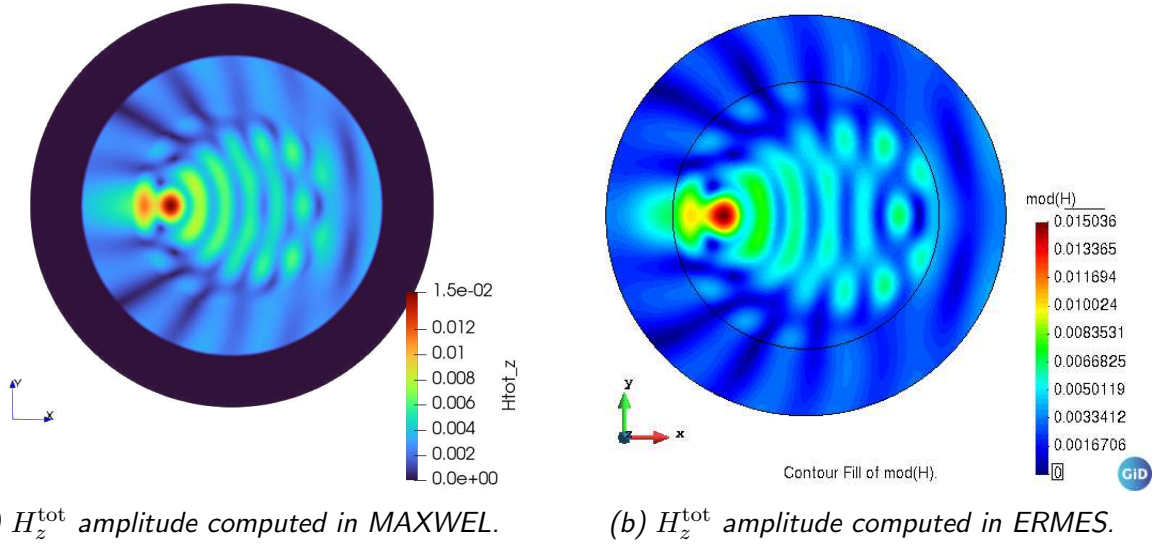


Figure 4.7: Comparing a homogeneous dielectric with $\epsilon_{rc} = 4$. Incident wave has an amplitude of 1 V/m .

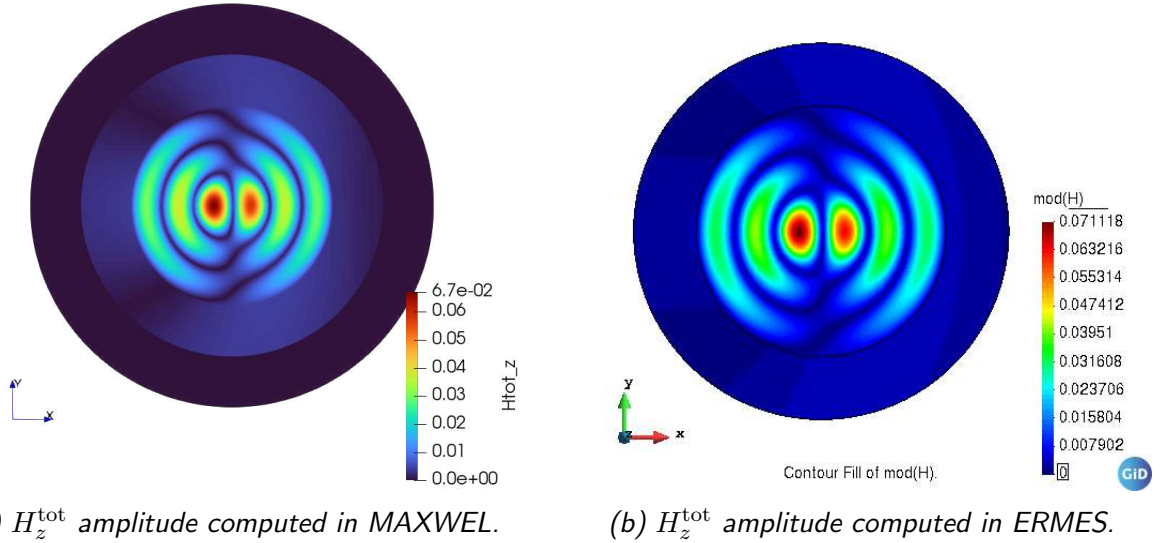


Figure 4.8: Comparing a cold plasma tensor between MAXWEL and ERMES. Incident wave has an amplitude of 1 V/m .

and with *absorbing boundary conditions* instead of PML. It has been tested that for finer meshes, results converge closer to the ones provided by ERMES, and quadratic elements gave much more accurate results than linear ones. It is relevant to mention that ERMES uses *edge* elements, which present different properties than regular nodal elements as those used in this thesis. Also, we had to introduce the vacuum impedance Z_0 as an ad hoc normalization coefficient, whose precision may also affect the numerical result. Despite all of this, in fig. 4.9 it is also appreciated that the behaviour of the wave within the plasma media is in excellent agreement between simulations.

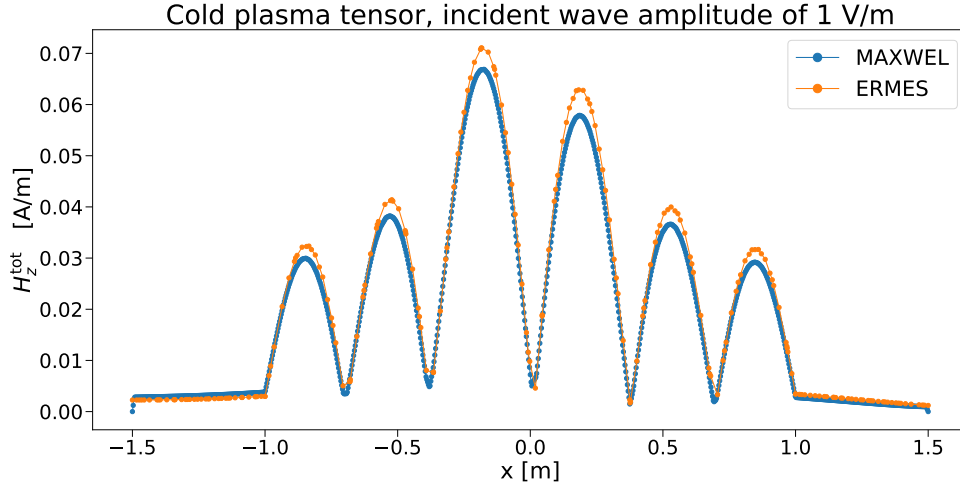


Figure 4.9: Amplitude of H_z^{tot} field across $y = 0$ diameter for a cold plasma medium.

4.2.3 Tokamak geometry

One key step to simulate an EM wave in a tokamak reactor is to have a proper mesh of its geometry. As mentioned in section 4.1.2, the mesh that we used was made thanks to the geometry exported from EQUILI, under current development at BSC by a fellow master student. We have produced results with different combinations of plasma densities, magnetic fields, and wave frequencies, and some of the results are shown in figs. 4.10 and 4.11.

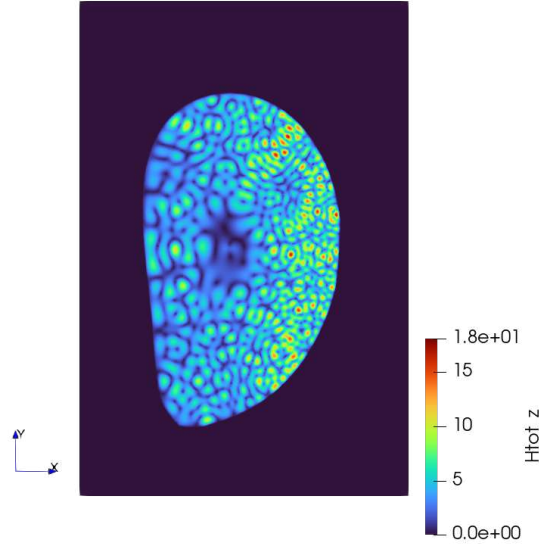


Figure 4.10: H_z^{tot} amplitude of a 60 MHz extraordinary wave dispersion in a D-T plasma with $B_0 = 5$ T and $n_0 = 10^{19} \text{ m}^{-3}$.

The results in fig. 4.10 were obtained with an incident wave of 60 MHz, in a plasma of three species (electrons, deuterons, and tritons) with a toroidal magnetic field of $B_0 = 5$ T and a plasma density of $n_0 = 10^{19} \text{ m}^{-3}$. For this simulation, there is no vacuum gap between the cold plasma scattering media and the PML region. Also, the toroidal magnetic field varies as $\propto \frac{1}{x}$ and the electron density follows the experimental expression

$$n_e(s) = n_0 \left[k + (1 - k) (1 - s^2)^a \right] \quad (4.6)$$

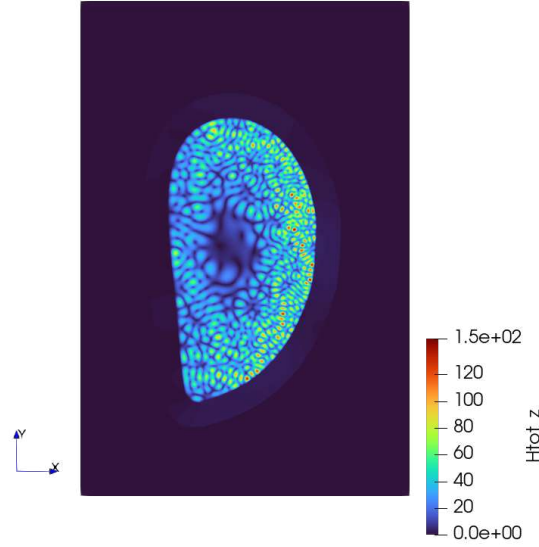


Figure 4.11: H_z^{tot} amplitude of a 53 MHz extraordinary wave dispersion in a D-T plasma with $B_0 = 5.3 \text{ T}$ and $n_0 = 3.5 \times 10^{19} \text{ m}^{-3}$.

where s is the normalised magnetic flux, and k and a are fitting parameters depending on the scenario to be simulated. The normalised magnetic flux is computed by EQUILI but at different coordinates from those of the tokamak mesh. Through interpolation, one can easily retrieve the magnetic flux at the same positions as the mesh's nodes.

In fig. 4.11 we reproduce the 2nd harmonic tritium scenario ($\omega = 2\omega_{cT}$) as described in [46], which is expected to be the main ICRF heating scenario in ITER. In this case, we are using a 50%/50% D-T plasma composition as well, with a magnetic field of $B_0 = 5.3 \text{ T}$ and a plasma density of $n_0 = 3.5 \times 10^{19} \text{ m}^{-3}$. The incident wave has a frequency of 53 MHz. Densities of $8 \times 10^{19} \text{ m}^{-3}$, as mentioned in [46], were not possible to simulate due to the necessity of finer meshes, and the lack of computational power found in the PC running the simulations. However, these results already show good accessibility of ICRF heating in such a potential ITER-like scenario. This simulation was run with quadratic triangular elements and with a vacuum layer between the plasma and the PML region. That is why fig. 4.10 plasma regions look significantly bigger than the one in fig. 4.11.

Additionally, one of the key capabilities of a full wave code is its ability to capture relevant physics regarding wave accessibility. As mentioned in section 2.2, cutoff regions are of special interest as they can prevent the wave from propagating. This is of utmost importance in tokamaks where the L-cutoff and the R-cutoff can lead the wave to an evanescent region and avoid ICRF heating. For example in JET, in front of the antenna, where the plasma density and magnetic field are low, the R-cutoff occurs and it was usually avoided by puffing gas [47]. The L-cutoff is presently used in the three-ion scheme [48]. When the refractive index (eqs. (2.38) and (2.39)) has an imaginary component — that is $n^2 \lesssim 0$ —, the wave should experience an exponential attenuation, not being able to penetrate the cutoff region. This condition naturally depends on the incident wave frequency, plasma frequency and cyclotron frequency. This happens when $P = 0$ for an ordinary wave and when $R = 0$ or $L = 0$ for an extraordinary wave. As shown in figs. 4.12 and 4.13 our code is able to reproduce these important physics for the ordinary wave; and most importantly, for both cutoffs of the extraordinary wave. All three cases correspond to a homogeneous plasma (i.e. no varying

density and magnetic field), with a vacuum gap between itself and the wall. In figs. 4.13a and 4.13b we have a 2 species plasma, while magnetic field and density are $B_0 = 2.7 \text{ T}$ and $n_0 = 2 \times 10^{17} \text{ m}^{-3}$ for the L cutoff, and magnetic field and density are $B_0 = 0.1 \text{ mT}$ and $n_0 = 2 \times 10^{14} \text{ m}^{-3}$ for the R cutoff. Up to now, resonance conditions have not been analysed as the cold plasma permittivity tensor is not able to properly describe resonances and will be left for future work.

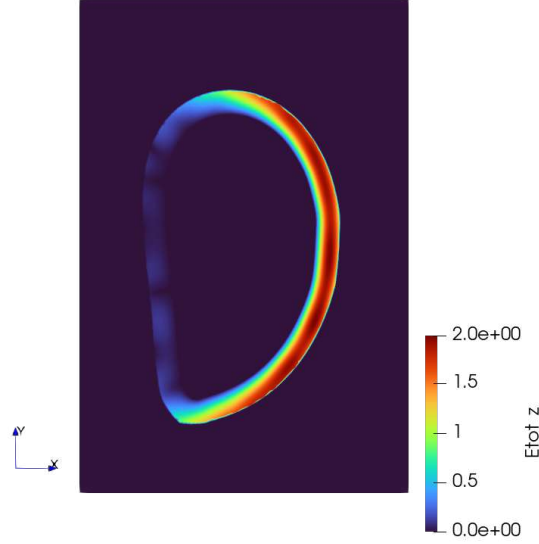


Figure 4.12: E_z^{tot} field for ordinary wave $\omega < \omega_p$ cutoff.

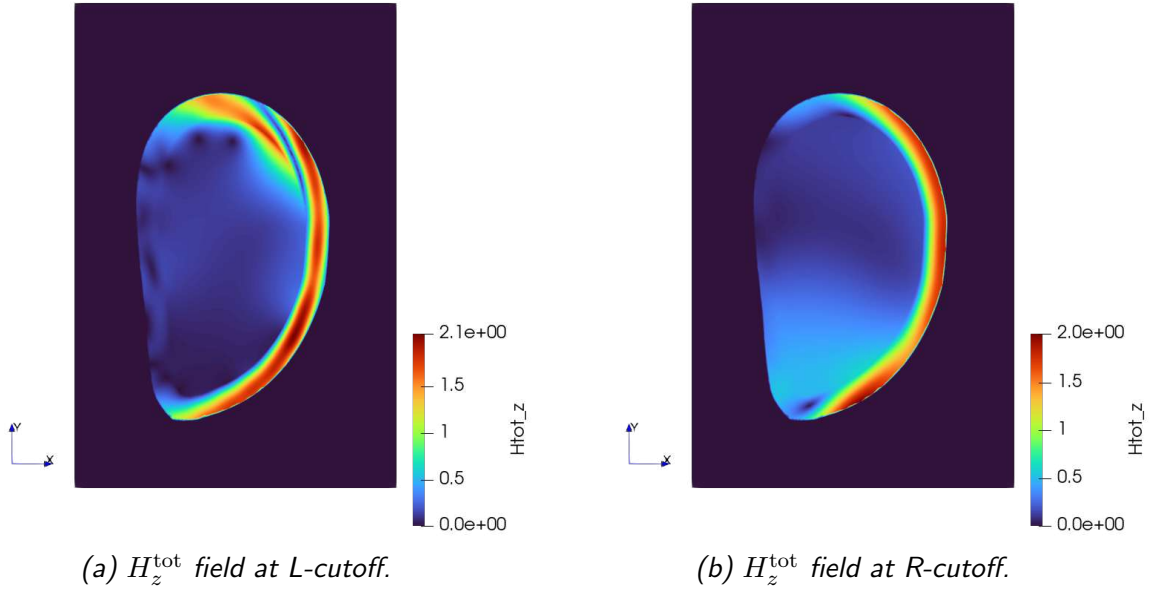


Figure 4.13: The extraordinary wave is unable to propagate when $n_X \sim 0$.

Chapter 5

Conclusions and Future Work

The development of the MAXWEL code has resulted in a first version of a FEM software being able to simulate the dispersion of a monochromatic plane wave in a dielectric or a cold plasma medium in a 2D domain. Several geometries have been used; such as circular domains, a particular tokamak cross-section configuration, and others. MAXWEL features the capability to process different types of elements, such as a combination of linear and quadratic order but also either triangular or quadrilateral shapes. The results have been successfully tested and benchmarked with software extracted from literature [21] and with ERMES [44]. Furthermore, we observe that the code captures the physics correctly for the cold plasma approximation, in addition to showing the cutoff behaviour as expected.

The capability of integrating the results of EQUILI, by generating the tokamak mesh in which some of the simulations took place, shows the potential compatibility between modules. This project aims towards an integrated modelling between all modules in fig. 1.6, in order to simulate a digital twin of a fully working tokamak reactor. One of the first steps to achieve that, and the main goal for the near future is to parallelise the code and migrate it to Alya [23, 24]. This would allow us to benefit from the huge computational capability of Marenostrum 5. Additionally, there is a strong interest in solving a more general vector wave equation, in which we would need 3D edge elements to represent correctly the new computational domain. Also, the cold plasma model may be valid as a first approximation to retrieve certain insights such as the wave accessibility; but is not accurate enough to simulate wave and wave-particle resonances, needed to account for wave damping and plasma power absorption. In the future, we would need to implement a hot plasma model, taking into account the heating effects and the particles' distribution function. Further development and case analysis will be performed during a future PhD program.

Bibliography

- [1] California Independent System Operator. Library | Market reports | California ISO, 2024. URL <https://www.caiso.com/library/market-reports>
- [2] J. P. Freidberg. *Plasma Physics and Fusion Energy*. Cambridge University Press, Cambridge, 2007. ISBN 978-0-521-73317-5. URL <http://dx.doi.org/10.1017/CB09780511755705>
- [3] F. F. Chen. Plasma Applications. In F. F. Chen, editor, *Introduction to Plasma Physics and Controlled Fusion*, pp. 355–411. Springer International Publishing, Cham, 2016. ISBN 978-3-319-22309-4. URL http://dx.doi.org/10.1007/978-3-319-22309-4_10
- [4] M. B. Chadwick, M. Herman, P. Obložinský, M. E. Dunn, Y. Danon, A. C. Kahler, D. L. Smith, B. Pritychenko, G. Arbanas, R. Arcilla, R. Brewer, D. A. Brown, R. Capote, A. D. Carlson, Y. S. Cho, H. Derrien, K. Guber, G. M. Hale, S. Hoblit, S. Holloway, T. D. Johnson, T. Kawano, B. C. Kiedrowski, H. Kim, S. Kunieda, N. M. Larson, L. Leal, J. P. Lestone, R. C. Little, E. A. McCutchan, R. E. MacFarlane, M. MacInnes, C. M. Mattoon, R. D. McKnight, S. F. Mughabghab, G. P. A. Nobre, G. Palmiotti, A. Palumbo, M. T. Pigni, V. G. Pronyaev, R. O. Sayer, A. A. Sonzogni, N. C. Summers, P. Talou, I. J. Thompson, A. Trkov, R. L. Vogt, S. C. van der Marck, A. Wallner, M. C. White, D. Wiarda, and P. G. Young. ENDF/B-VII.1 Nuclear Data for Science and Technology: Cross Sections, Covariances, Fission Product Yields and Decay Data. *Nuclear Data Sheets*, **112** (2011) (12) 2887–2996. ISSN 0090-3752. URL <http://dx.doi.org/10.1016/j.nds.2011.11.002>
- [5] D. A. Brown, M. B. Chadwick, R. Capote, A. C. Kahler, A. Trkov, M. W. Herman, A. A. Sonzogni, Y. Danon, A. D. Carlson, M. Dunn, D. L. Smith, G. M. Hale, G. Arbanas, R. Arcilla, C. R. Bates, B. Beck, B. Becker, F. Brown, R. J. Casperson, J. Conlin, D. E. Cullen, M. A. Descalle, R. Firestone, T. Gaines, K. H. Guber, A. I. Hawari, J. Holmes, T. D. Johnson, T. Kawano, B. C. Kiedrowski, A. J. Koning, S. Kopecky, L. Leal, J. P. Lestone, C. Lubitz, J. I. Márquez Damián, C. M. Mattoon, E. A. McCutchan, S. Mughabghab, P. Navratil, D. Neudecker, G. P. A. Nobre, G. Noguere, M. Paris, M. T. Pigni, A. J. Plompen, B. Pritychenko, V. G. Pronyaev, D. Roubtsov, D. Rochman, P. Romano, P. Schillebeeckx, S. Simakov, M. Sin, I. Sirakov, B. Sleaford, V. Sobes, E. S. Soukhovitskii, I. Stetcu, P. Talou, I. Thompson, S. van der Marck, L. Welser-Sherrill, D. Wiarda, M. White, J. L. Wormald, R. Q. Wright, M. Zerkle, G. Žerovnik, and Y. Zhu. ENDF/B-VIII.0: The 8th Major Release of the Nuclear Reaction Data Library with CIELO-project Cross Sections, New Standards and Thermal Scattering Data. *Nuclear Data Sheets*, **148** (2018) 1–142. ISSN 0090-3752. URL <http://dx.doi.org/10.1016/j.nds.2018.02.001>

- [6] F. F. Chen. *Introduction to Plasma Physics and Controlled Fusion*. Springer International Publishing, Cham, 2016. ISBN 978-3-319-22308-7 978-3-319-22309-4. URL <http://dx.doi.org/10.1007/978-3-319-22309-4>
- [7] R. Fusion. Technology. URL <https://renfusion.eu/technology>
- [8] J. Proll. *Trapped-particle instabilities in quasi-isodynamic stellarators*. Ph.D. thesis, Ernst-Moritz-Arndt University Greifswald Faculty of Mathematics and Natural Sciences, 2014
- [9] L.-G. Eriksson, T. Hellsten, and U. Willen. Comparison of time dependent simulations with experiments in ion cyclotron heated plasmas. *Nuclear Fusion*, **33** (1993) (7) 1037. ISSN 0029-5515. URL <http://dx.doi.org/10.1088/0029-5515/33/7/I07>
- [10] M. Brambilla. A full wave code for ion cyclotron waves in toroidal plasmas. *Technical report*, Max-Planck-Institut für Plasmaphysik, Germany, 1996. IPP-5/66 INIS Reference Number: 27067158
- [11] M. Jucker, J. P. Graves, W. A. Cooper, N. Mellet, T. Johnson, and S. Brunner. Integrated modeling for ion cyclotron resonant heating in toroidal systems. *Computer Physics Communications*, **182** (2011) (4) 912–925. ISSN 0010-4655. URL <http://dx.doi.org/10.1016/j.cpc.2010.12.028>
- [12] J. Hillairet, P. Mollard, L. Colas, W. Helou, G. Urbanczyk, J.-M. Bernard, J.-M. Delaplanche, F. Durand, N. Faure, P. Garibaldi, G. Lombard, C. Bourdelle, C. Desgranges, E. Delmas, R. Dumont, A. Ekedahl, F. Ferlay, M. Goniche, C. Guillemaut, G. T. Hoang, P. Maget, R. Volpe, Y. Song, Q. Yang, Z. Chen, Y. Wang, H. Xu, S. Yuan, Y. Zhao, F. Durodie, E. Lerche, R. Ragona, N. Bertelli, M. Ono, S. Shiraiwa, V. Bobkov, C. Klepper, C. Lau, E. Martin, B. Lu, R. Maggiora, D. Milanesio, K. Vulliez, G. Wallace, and W. Team. WEST actively cooled load resilient ion cyclotron resonance heating system results. *Nuclear Fusion*, **61** (2021) (9) 096030. ISSN 0029-5515. Publisher: IOP Publishing, URL <http://dx.doi.org/10.1088/1741-4326/ac1759>
- [13] Max-Planck-Gesellschaft. ICRH. URL <https://www.ipp.mpg.de/3871973/ICRH>
- [14] J. Garcia, F. J. Casson, L. Frassinetti, D. Gallart, L. Garzotti, H.-T. Kim, M. Nocente, S. Saarelma, F. Auriemma, J. Ferreira, S. Gabriellini, A. Ho, P. Huynh, K. K. Kirov, E. Lerche, M. J. Mantsinen, V. K. Zotta, Z. Stancar, D. M. A. Taylor, D. V. Eester, C. D. Challis, and J. E. T. Contributors. Modelling performed for predictions of fusion power in JET DTE2: overview and lessons learnt. *Nuclear Fusion*, **63** (2023) (11) 112003. ISSN 0029-5515. Publisher: IOP Publishing, URL <http://dx.doi.org/10.1088/1741-4326/acedc0>
- [15] D. Gallart, M. J. Mantsinen, C. Challis, D. Frigione, J. Graves, E. Belonohy, F. Casson, A. Czarnecka, J. Eriksson, J. Garcia, M. Goniche, C. Hellesen, J. Hobirk, P. Jaquet, E. Joffrin, N. Krawczyk, D. King, M. Lennholm, E. Lerche, E. Pawelec, X. Sáez, M. Sertoli, G. Sips, E. Solano, M. Tsalas, P. Vallejos, M. Valisa, and J. E. T. Contributors. Modelling of JET hybrid plasmas with emphasis on performance of combined ICRF and NBI heating. *Nuclear Fusion*, **58** (2018) (10) 106037. ISSN 0029-5515. Publisher: IOP Publishing, URL <http://dx.doi.org/10.1088/1741-4326/aad9ad>
- [16] E. Lerche, D. Van Eester, P. Jaquet, F. Casson, Y. Baranov, P. Dumortier, D. Gallart, J. Graves, P. Huynh, T. Johnson, Y. Kazakov, V. Kiptily, K. Kirov, M. Machielsens,

- M. Mantsinen, I. Monakhov, J. Ongena, and JET Contributors. ICRH options for JET-ILW DTE2 operation. *AIP Conference Proceedings*, **2254** (2020) (1) 030007. ISSN 0094-243X. URL <http://dx.doi.org/10.1063/5.0013530>
- [17] D. V. Eester, E. Lerche, P. Huynh, T. Johnson, D. Yadikin, Š. Štancar, S. Aleiferis, D. Frigione, L. Garzotti, P. Lomas, C. Lowry, M. Maslov, J. E. T. contributors, and E.-I. team. Maximising D - T fusion power by optimising the plasma composition and beam choice in JET. *Plasma Physics and Controlled Fusion*, **64** (2022) (5) 055014. ISSN 0741-3335. Publisher: IOP Publishing, URL <http://dx.doi.org/10.1088/1361-6587/ac5a09>
- [18] D. Gallart, M. J. Mantsinen, J. Manyer, E. Planas, D. M. A. Taylor, J. Garcia, D. Frigione, L. Garzotti, H.-T. Kim, M. Nocente, F. Rimini, D. V. Eester, and J. E. T. Contributors. Prediction of ICRF minority heating schemes for JET D-T experiments. *Plasma Physics and Controlled Fusion*, **64** (2022) (12) 125006. ISSN 0741-3335. Publisher: IOP Publishing, URL <http://dx.doi.org/10.1088/1361-6587/ac9925>
- [19] K. K. Kirov, Y. Kazakov, M. Nocente, J. Ongena, Y. Baranov, F. Casson, J. Eriksson, L. Giacomelli, C. Hellesen, V. Kiptily, R. Bilato, K. Crombe, R. Dumont, P. Jacquet, T. Johnson, E. Lerche, M. Mantsinen, D. V. Eester, J. Varje, H. Weisen, and JET Contributors. Synergistic ICRH and NBI heating for fast ion generation and maximising fusion rate in mixed plasmas at JET. *AIP Conference Proceedings*, **2254** (2020) (1) 030011. ISSN 0094-243X. URL <http://dx.doi.org/10.1063/5.0014235>
- [20] P. Huynh, E. A. Lerche, D. V. Eester, J. Garcia, T. Johnson, J. Ferreira, K. K. Kirov, D. Yadykin, P. Strand, J. E. T. Contributors, and t. E.-I. Team. European transport simulator modeling of JET-ILW baseline plasmas: predictive code validation and DTE2 predictions. *Nuclear Fusion*, **61** (2021) (9) 096019. ISSN 0029-5515. Publisher: IOP Publishing, URL <http://dx.doi.org/10.1088/1741-4326/ac0b34>
- [21] Ö. Özgün and M. Kuzuoğlu. *MATLAB-based Finite Element Programming in Electromagnetic Modeling*. CRC Press, 1st edition edition, 2018
- [22] J. Jin. *The Finite Element Method in Electromagnetics*. IEEE Press. Wiley, 2015. ISBN 9781118842027. URL <https://books.google.es/books?id=DFi-BgAAQBAJ>
- [23] BSC CASE department. bsc-alya · GitLab, 2022. URL <https://gitlab.com/bsc-alya>
- [24] M. Vázquez, G. Houzeaux, S. Koric, A. Artigues, J. Aguado-Sierra, R. Arís, D. Mira, H. Calmet, F. Cucchietti, H. Owen, A. Taha, E. D. Burness, J. M. Cela, and M. Valero. Alya: Multiphysics engineering simulation toward exascale. *Journal of Computational Science*, **14** (2016) 15–27. ISSN 1877-7503. URL <http://dx.doi.org/10.1016/j.jocs.2015.12.007>
- [25] A. Gutierrez-Milla, M. Mantsinen, M. Avila, G. Houzeaux, C. Riera-Auge, and X. Sáez. New high performance computing software for multiphysics simulations of fusion reactors. *Fusion Engineering and Design*, **136** (2018) 639–644. ISSN 0920-3796. URL <http://dx.doi.org/10.1016/j.fusengdes.2018.03.045>
- [26] E. Goldberg, M. C. i. Duxans, O. O. Gelabert, M. J. Mantsinen, and A. Soba. Validating NEUTRO, a deterministic finite element neutron transport solver for fusion applications, with literature tests, experimental benchmarks and other neutronic codes. *Plasma Physics and Controlled Fusion*, **64** (2022) (10) 104006. ISSN 0741-3335. Publisher: IOP Publishing, URL <http://dx.doi.org/10.1088/1361-6587/ac8acd>

- [27] A. Soba, O. Fernandez-Serracanta, J. Lorenzo, D. Garcín, G. Houzeaux, N. Lamas, X. Granados, and M. J. Mantsinen. A high-performance electromagnetic code to simulate high-temperature superconductors. *Fusion Engineering and Design*, **201** (2024) 114282. ISSN 0920-3796. URL <http://dx.doi.org/10.1016/j.fusengdes.2024.114282>
- [28] P. Manyer Fuertes, A. Soba Pascual, and D. Gallart Escolà. EQUILI module in ALYA: a free-boundary Grad-Shafranov equation solver using CutFEM. *11th International BSC Severo Ochoa Doctoral Symposium 2024*, (2024) 50–51. Publisher: Barcelona Supercomputing Center, URL https://www.bsc.es/sites/default/files/public/11thBSCDS_BoA.pdf
- [29] D. Gallart Escolà. *Computational analysis of ion cyclotron resonance frequency heating for JET experiments*. Doctoral thesis, Universitat Politècnica de Catalunya, 2019. Accepted: 2020-09-06T00:01:26Z Publication Title: TDX (Tesis Doctorals en Xarxa), URL <http://dx.doi.org/10.5821/dissertation-2117-328426>
- [30] D. G. Swanson. *Plasma Waves*. CRC Press, Boca Raton, 2 edition, 2003. ISBN 978-0-367-80272-1. URL <http://dx.doi.org/10.1201/9780367802721>
- [31] T. H. Stix. *The theory of plasma waves*. McGraw-Hill advanced physics monograph series. McGraw-Hill, New York, 1962. OCLC: 612645976, URL <http://books.google.com/books?id=qBYvAAAAIAAJ>
- [32] M. Mantsinen. *Development and experimental evaluation of theoretical models for ion cyclotron resonance frequency heating of tokamak plasmas*. Ph.D. thesis, Väitöskirja, Espoo, 1999. Tiivistelmä ja 6 erip., URL <https://www.finna.fi/Record/jykdok.805376>
- [33] M. Porkolab, A. Bécoulet, P. T. Bonoli, C. Gormezano, R. Koch, R. J. Majeski, A. Messina, J. M. Noterdaeme, C. Petty, R. Pinsky, D. Start, and R. Wilson. Recent progress in ICRF physics. *Plasma Physics and Controlled Fusion*, **40** (1998) (8A) A35. URL <http://dx.doi.org/10.1088/0741-3335/40/8A/004>
- [34] A. Hrennikoff. Solution of Problems of Elasticity by the Framework Method. *Journal of Applied Mechanics*, **8** (2021) (4) A169–A175. ISSN 0021-8936. URL <http://dx.doi.org/10.1115/1.4009129>
- [35] R. Courant. Variational methods for the solution of problems of equilibrium and vibrations. *Bulletin of the American Mathematical Society*, **49** (1943) (1) 1–23. ISSN 0002-9904, 1936-881X. URL <http://dx.doi.org/10.1090/S0002-9904-1943-07818-4>
- [36] The Efficient Engineer. *Understanding the Finite Element Method*, 2021. URL <https://efficientengineer.com/finite-element-method/>
- [37] K.-J. Bathe. *Finite Element Procedures*. Klaus-Jurgen Bathe, 2006. ISBN 978-0-9790049-0-2. Google-Books-ID: rWvefGICfO8C
- [38] J. N. Reddy. *Introduction to the Finite Element Method*. McGraw-Hill Education, 4th edition edition, 2019. ISBN 978-1-259-86190-1. URL <https://www.accessengineeringlibrary.com/content/book/9781259861901>
- [39] M. R. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *Journal of research of the National Bureau of Standards*, **49** (1952) 409–435. URL <https://api.semanticscholar.org/CorpusID:2207234>

- [40] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes: the art of scientific computing*. Cambridge University Press, Cambridge [Cambridgeshire], 1986. ISBN 978-0-521-30811-3. OCLC: 1036772065, URL <https://archive.org/details/numericalrecipes00pres>
- [41] B. Delaunay. Sur la sphère vide. A la mémoire de Georges Voronoï. *Izvestija Rossijskoj akademii nauk. Serija matematičeskaja*, (1934) (6) 793–800
- [42] A. Melendo, A. Coll, M. Pasenau, E. Escolano, and A. Monros. www.gidhome.com, 2018. [Online; accessed Aug-2024], URL www.gidsimulation.com
- [43] Ö. Özgün and M. Kuzuoğlu. Non-Maxwellian Locally-Conformal PML Absorbers for Finite Element Mesh Truncation. *IEEE Transactions on Antennas and Propagation*, **55** (2007) (3) 931–937. ISSN 1558-2221. Conference Name: IEEE Transactions on Antennas and Propagation, URL <http://dx.doi.org/10.1109/TAP.2007.891865>
- [44] R. Otin. ERMES: A nodal-based finite element code for electromagnetic simulations in frequency domain. *Computer Physics Communications*, **184** (2013) (11) 2588–2595. ISSN 0010-4655. URL <http://dx.doi.org/10.1016/j.cpc.2013.06.010>
- [45] J. Ahrens, B. Geveci, and C. Law. ParaView: An End-User Tool for Large Data Visualization. In *Visualization Handbook*. Elsevier, 2005. ISBN 978-0123875822
- [46] M. Mantsinen, P. Jacquet, E. Lerche, D. Gallart, K. Kirov, P. Mantica, D. Taylor, D. V. Eester, M. Baruzzo, I. Carvalho, C. Challis, A. D. Molin, E. Delabie, E. D. L. Luna, R. Dumont, P. Dumortier, J. Eriksson, D. Frigione, J. Garcia, L. Garzotti, C. Giroud, R. Henriques, J. Hobirk, A. Kappatou, Y. Kazakov, D. Keeling, D. King, V. Kiptily, M. Lennholm, P. Lomas, C. Lowry, C. Maggi, J. Mailloux, M. Maslov, S. Menmuir, I. Monakhov, R. Morales, C. Noble, M. Nocente, A. Patel, G. Pucella, C. Reux, D. Rigamonti, F. Rimini, A. Sheikh, S. Silburn, P. Siren, E. Solano, Z. Stancar, M. Tardocchi, and J. Contributors. Experiments in high-performance JET plasmas in preparation of second harmonic ICRF heating of tritium in ITER. *Nuclear Fusion*, **63** (2023) (11) 112015. URL <http://dx.doi.org/10.1088/1741-4326/aceb08>
- [47] W. Zhang, P. Jacquet, E. Lerche, R. Bilato, V. Bobkov, D. Coster, Y. Feng, C. Guillemaut, M. Goniche, D. Harting, T. Lunt, J.-M. Noterdaeme, G. Szepesi, D. V. Eester, and t. J. Contributors. 3D simulations of gas puff effects on edge plasma and ICRF coupling in JET. *Nuclear Fusion*, **57** (2017) (5) 056042. ISSN 0029-5515. Publisher: IOP Publishing, URL <http://dx.doi.org/10.1088/1741-4326/aa6817>
- [48] Y. O. Kazakov, D. V. Eester, R. Dumont, and J. Ongena. On resonant ICRF absorption in three-ion component plasmas: a new promising tool for fast ion generation. *Nuclear Fusion*, **55** (2015) (3) 032001. ISSN 0029-5515. Publisher: IOP Publishing, URL <http://dx.doi.org/10.1088/0029-5515/55/3/032001>

Appendix A

Other cases and figures generated

Some of the figures mentioned in section 4.2 and others are shown here. All the cases tested present excellent accordance with the literature, but they are essentially similar to the ones included in the main body of the manuscript.

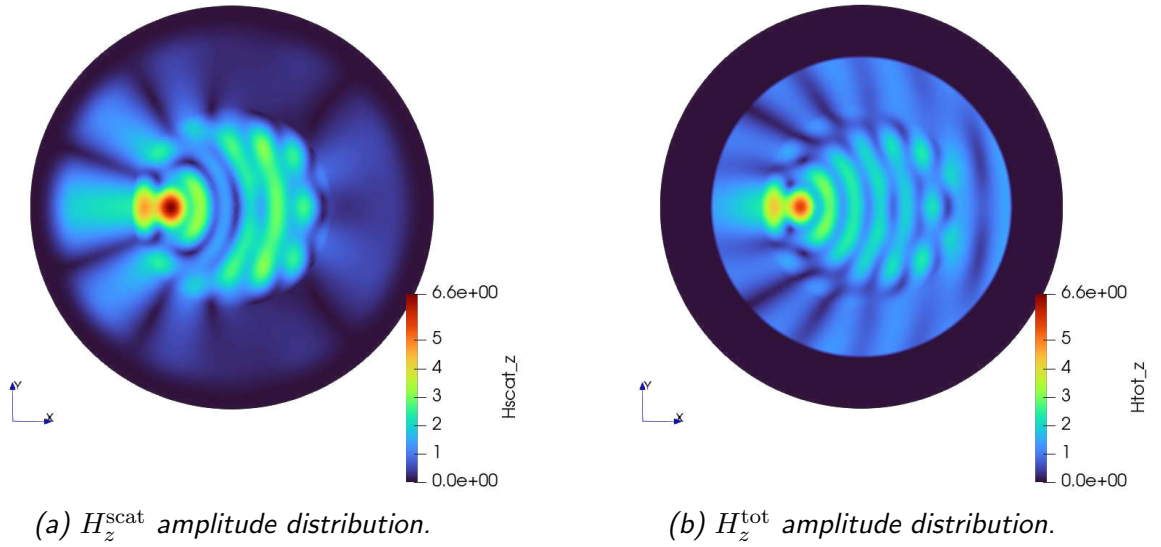


Figure A.1: TE polarisation with $\epsilon_{rc} = 4$, simulated with MAXWEL.

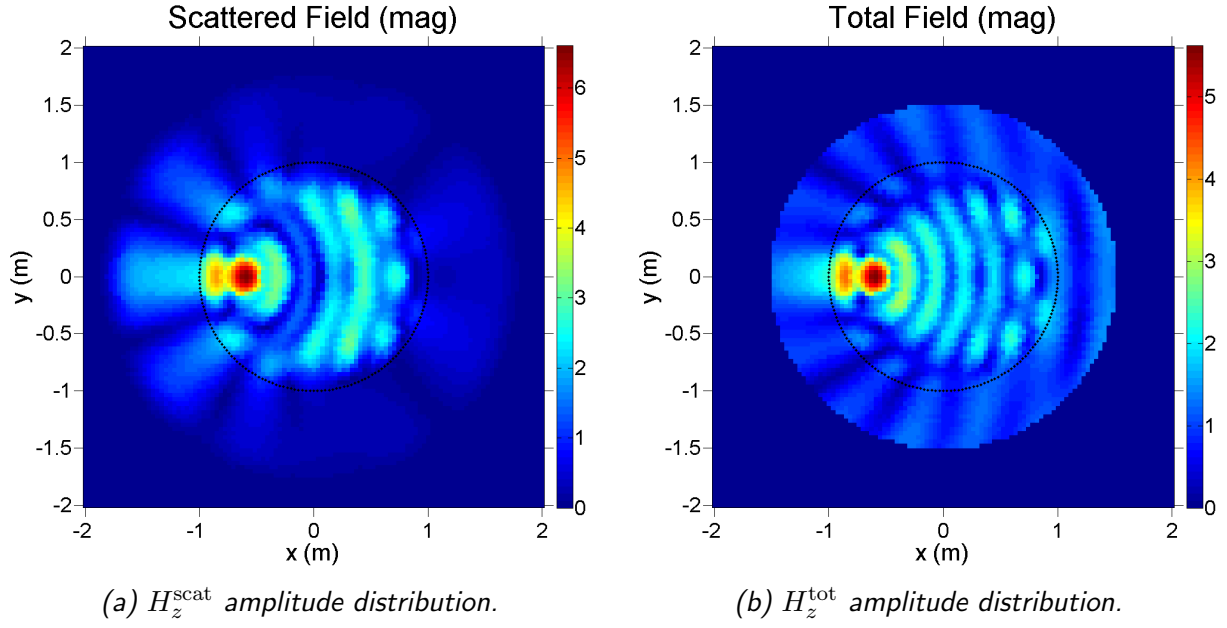


Figure A.2: TE polarisation with $\epsilon_{rc} = 4$, extracted from [21].

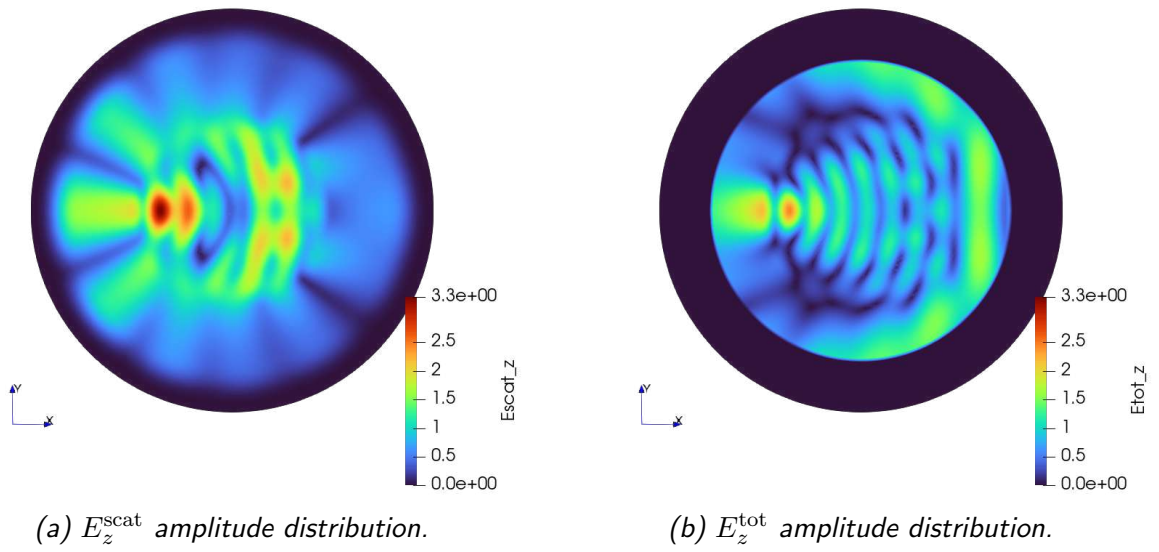
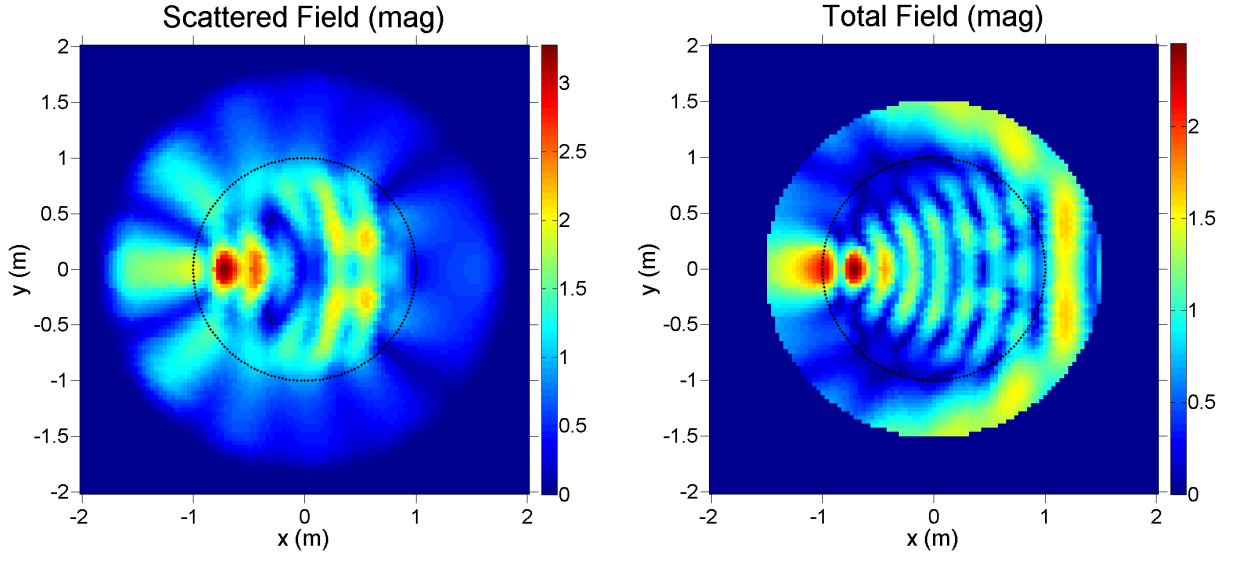


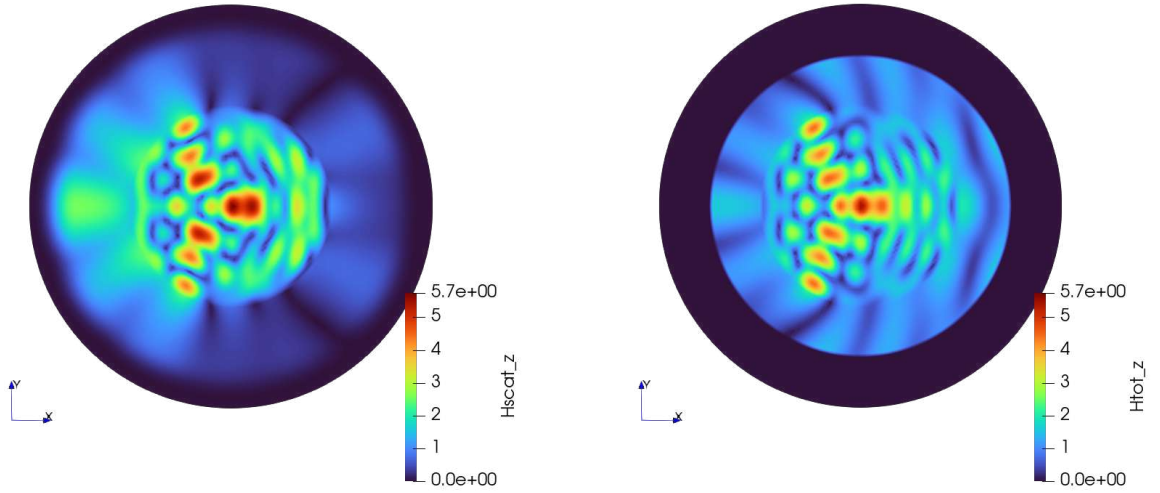
Figure A.3: TM polarisation with $\epsilon_{rc} = 4$, simulated with MAXWEL.



(a) E_z^{scat} amplitude distribution.

(b) E_z^{tot} amplitude distribution.

Figure A.4: TM polarisation with $\epsilon_{rc} = 4$, extracted from [21].



(a) H_z^{scat} amplitude distribution.

(b) H_z^{tot} amplitude distribution.

Figure A.5: TE polarisation with $\epsilon_{xx} = 4$ and $\epsilon_{yy} = 9$, simulated with MAXWEL.

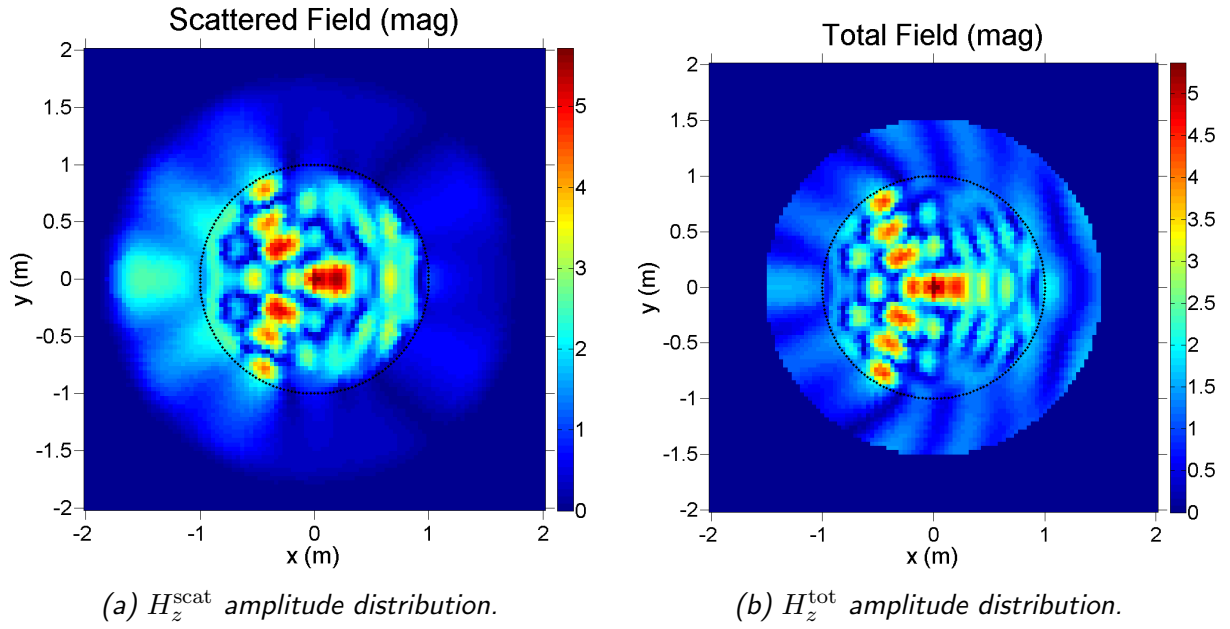
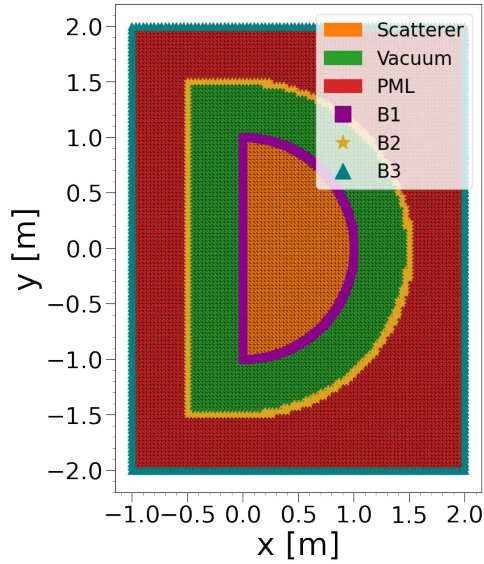


Figure A.6: TE polarisation with $\varepsilon_{xx} = 4$ and $\varepsilon_{yy} = 9$, extracted from [21].

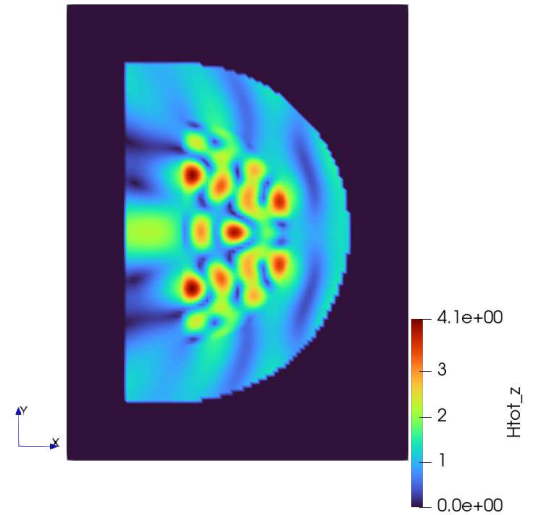
Appendix B

Different geometries

Here we present some different geometries that have been tested in MAXWEL. If any arbitrary mesh is generated correctly it can be fed to the program. These cases represents a 300 MHz wave in a dielectric material with $\varepsilon_{xx} = 4$ and $\varepsilon_{yy} = 9$.

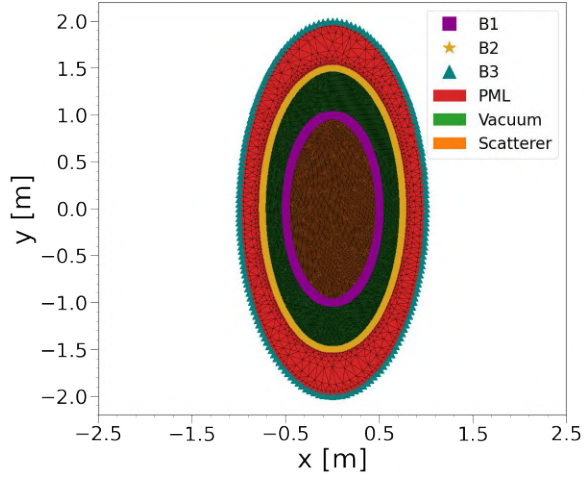


(a) Mesh for a semicircular shaped dielectric.



(b) H_z^{tot} field amplitude.

Figure B.1: Semicircular geometry.



(a) Mesh for an elliptical shaped dielectric.

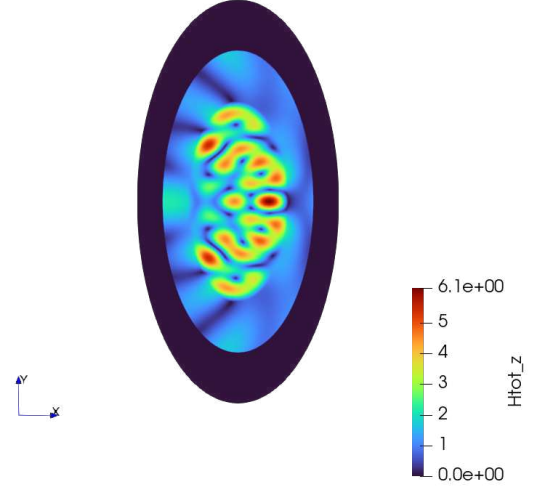
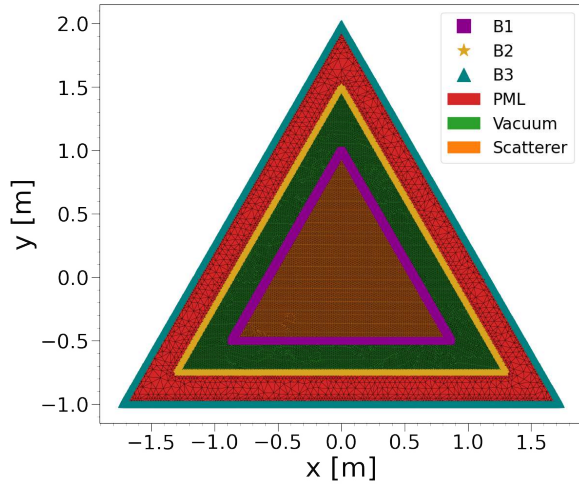
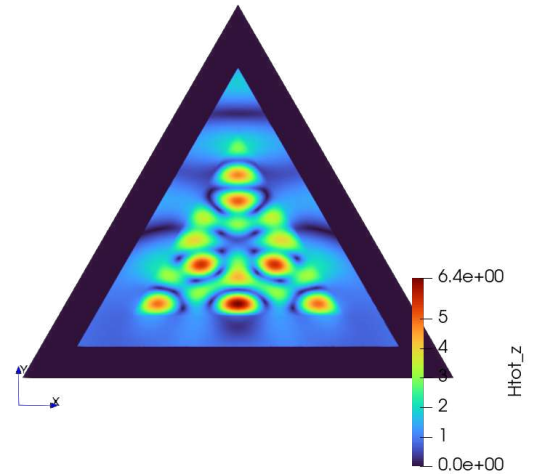
(b) H_z^{tot} field amplitude.

Figure B.2: Elliptical geometry.



(a) Mesh for a triangular-shaped dielectric.

(b) H_z^{tot} field amplitude.Figure B.3: Triangular geometry. In this case, the angle of incidence was $\varphi = \frac{\pi}{2}$.

Appendix C

In-plane fields

As mentioned in section 4.1.4, we are able to compute the first derivatives of our solution. Consequently, the complementary field in the xy plane can be derived using eqs. (4.2) and (4.3). In figs. C.1a and C.1b it is shown the $\text{Re}(E_{\perp}^{\text{tot}})$ and $\text{Im}(E_{\perp}^{\text{tot}})$ respectively. The case correspond to a TE polarised plane wave of 300 MHz and a dielectric of $\varepsilon_{xx} = 9$ and $\varepsilon_{yy} = 4$. In fig. C.2 it is presented the same case as in section 4.2.2, since it was of great relevance to benchmark our results.

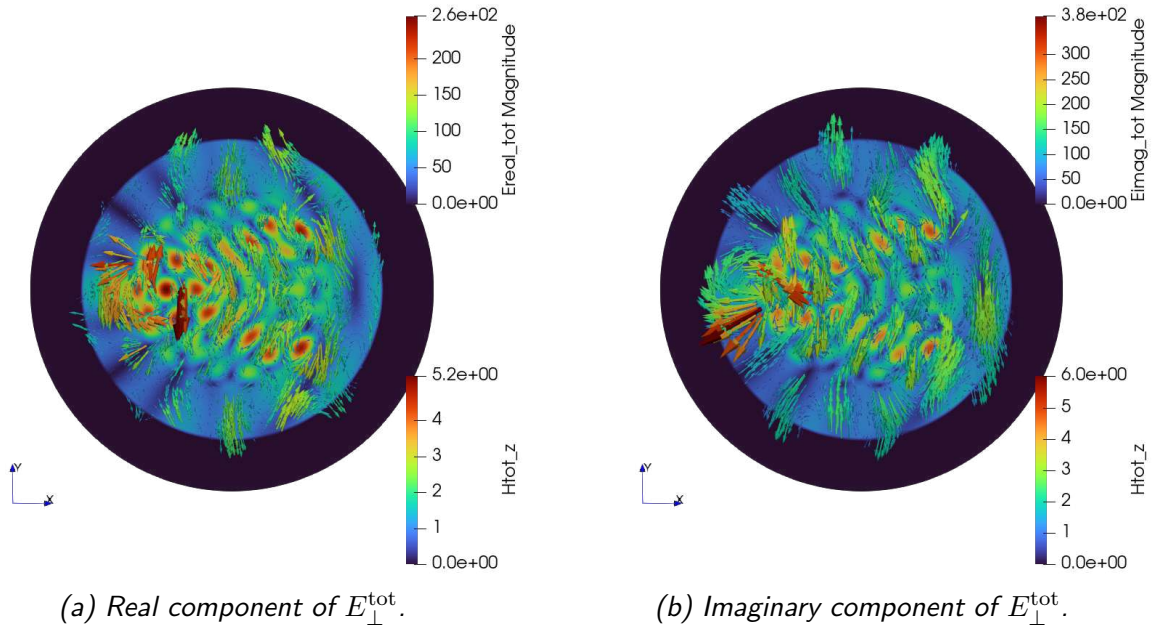


Figure C.1: $E_{\perp}^{\text{tot}}$ field superposed with the H_z^{tot} component. The in-plane field is represented as a vector, indicating the magnitude and orientation in the plane.

In fig. C.3, the case corresponds to a TE polarised plane wave of 400 MHz propagating through a cold plasma of $B_0 = 0.1$ T and $n_0 = 10^{16} \text{ m}^{-3}$. It is shown only the derivatives from eq. (4.3), without the constants and permittivity tensor multiplication, due to normalisation reasons.

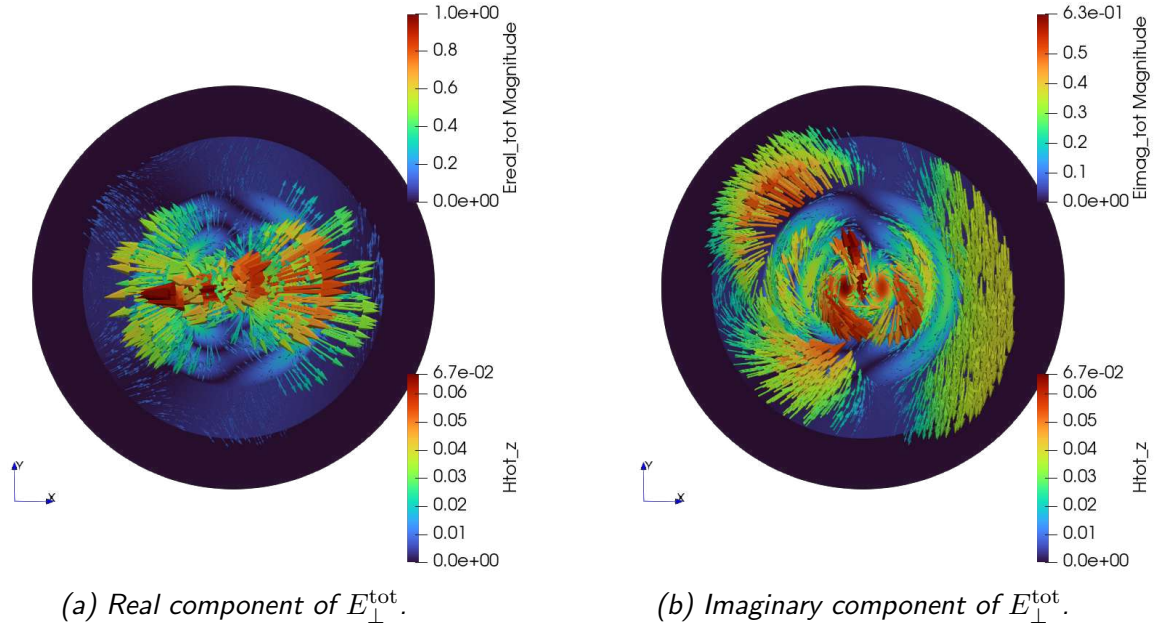


Figure C.2: Cold plasma case as in section 4.2.2, used to benchmark our code with the results provided by ERMES.

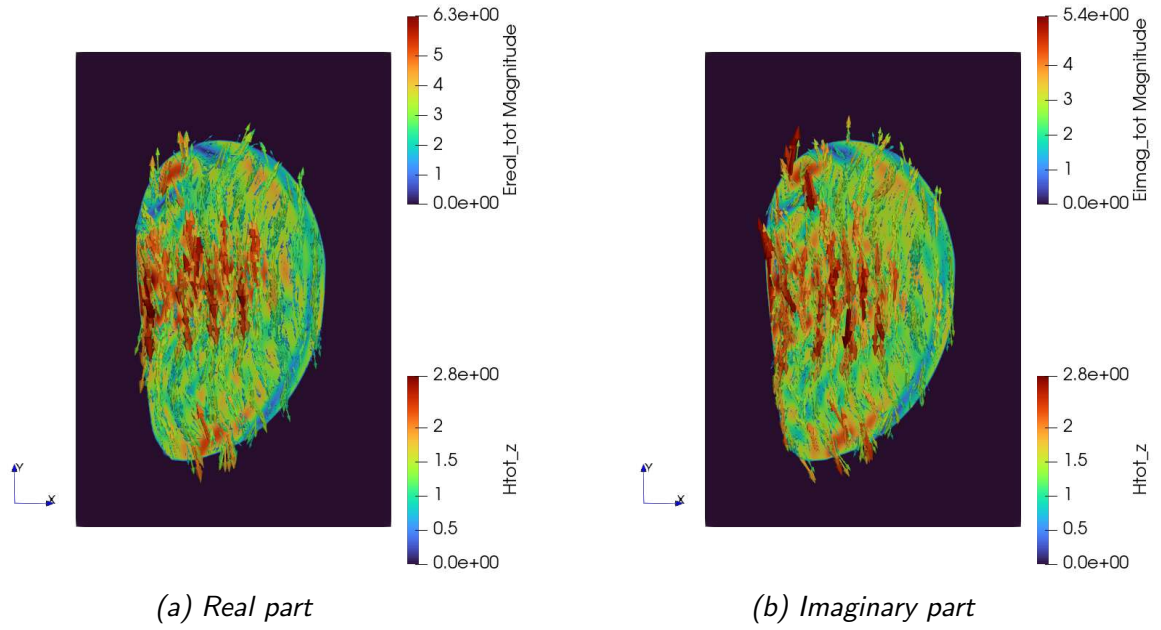


Figure C.3: Re-normalised in-plane fields.

Acknowledgements

I would like to thank my supervisors Dr. Daniel Gallart and Dr. Alejandro Soba, for their incredible help and support throughout this thesis. They were exceptionally attentive from the very beginning and really considerate in every aspect of this project, both academically and personally. Their ability to create such a comfortable yet intellectually stimulating environment has allowed me to learn extensively about numerical methods and plasma physics. Thankfully, I will be able to continue under their mentorship for some more years as a Ph.D. student, for which I am eager and excited to begin. Additionally, I would like to thank the Barcelona Supercomputing Center, the Fusion Group, and especially the group leader, Prof. Mervi Johanna Mantsinen, for hosting me during these months and providing the academic opportunity to pursue a doctoral program. I also wish to acknowledge the financial support provided by the FuseNet association, which funded this mobility scheme. My gratitude extends to Pau Manyer and Marina Maroto for their collaboration, and especially to Dr. Ruben Otin, who generously contributed his software and personal time to help us benchmark some of our results.

This work has been carried out within the EUROfusion Consortium framework, funded by the European Union via the Euratom Research and Training Programme (Grant Agreement No 101052200—EUROfusion). Views and opinions expressed are however those of the author only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the European Commission can be held responsible for them.

This master's program has taught me a lot. The academic knowledge is key to a successful career and to open professional opportunities in the future. But what I will treasure the most is the incredible people that I have met on the way. The experiences, anecdotes, and friendships forged in these two years are already worth the long hours seated in front of a whiteboard, the stressful weeks of exams, and the late-night study sessions. I am happy to believe that many of us will keep in touch and keep seeing each other in the Fusion community. And for the ones who may not, I am sure that life will bring us together again, at some point.

Last but not least, I can only express words of gratitude to my family and friends, which have been an invaluable source of support and sometimes even an escape valve. I would like to dedicate to them not only this work but also all of these years of effort and the ones to come. To the people who have been by my side, celebrating with me when finding success and cheering me up during harder times. These words are specially dedicated to *Júlia, mamá, papá, la Ro y la abu*.

Gracias por estar ahí siempre y sin esperar nada a cambio. En los momentos alegres y en los momentos difíciles. Son un verdadero tesoro, el qual nunca podré compensar el increíble valor que tiene. Juntos somos uno.

Declaration in lieu of oath

I am aware that plagiarism is not consistent with academic and research ethics. I declare in lieu of oath that all the work presented in this report, which is not cited in the text or by references, is my own work.

Barcelona, Spain; August 25, 2024

Hernán Domingo Ramos

Copyright Agreement

I hereby grant the FUSION-EP consortium the non-exclusive right to publish this work.

I declare that this work is free of copyright claims of third parties.

Barcelona, Spain; August 25, 2024

Hernán Domingo Ramos