Máster en Matemática Industrial

Academic year 2017 – Academic year 2018

Master's thesis

Optimization of an acoustic intensity P-U probe design using numerical methods

Coral Real Llamas

Fecha presentación:
Academic supervisor: Andrés Prieto
Company: Microflown Technologies
Company supervisor: Daniel Fernández Comesaña

Microflown Technologies

Acknowledgements

I am grateful for many people; not only for their contributions to scientific knowledge, but also for my personal benefit from the privilege of working alongside them. Without the mentorship and guidance of these colleagues this master thesis would not be done.

I would especially like to acknowledge the Microflown company in Arnhem, which gave me the opportunity to work with them and get deeply in the acoustic world. From them, I want to thank Dani Fernández and Grachi Carrillo for helping me and patiently supervise my work during the three months that I was working in the company. Likewise, to Michal Machnicki, who answered all my questions about `Python`. Thanks also to all my colleagues of Microflown, who make me feel like home in a foreign country. Even if I do not named all of them, without those people this project never come up.

Finally, but not less important, I want to acknowledge to my professor and supervisor Andrés Prieto for being there helping me through these months and solving my doubts, even before starting this master thesis.

Contents

Abstract

In this master thesis a mathematical model for an acoustic P-U probe is stated and solved numerically. Firstly, the acoustic problem stated in a bounded domain is studied, and then this analysis is extended to an unbounded domain. The acoustic governing equation used in all cases is the Helmholtz equation. Throughout this master thesis, the data associated to boundary conditions are given by a monopole solution and plane waves of different heading angles. Moreover, the Perfect Matched Layers technique is used to simulate the free field conditions. Any of the acoustic problems described in this work are discretized by means of the Finite Element Method, which is implemented in a computer code using the Python library, FEniCS. Finally, the optimization of the geometry of the P-U probe is made and the numerical results are shown in detail.

Keywords: Acoustic probe, Perfect Matched Layers, Finite Element Method, mathematical modelling, numerical simulation, parametric optimization.

Introduction

The Microflown device (see [8]) is the only Micro-Electro-Mechanical-System (MEMS) acoustic sensor that measures the particle velocity instead of the sound pressure, as conventional microphones usually do. Due to the heating of two microscopical wires placed in parallel, this sensor can quantify the velocity of air particles, which, combined with a pressure microphone, allows us to describe completely the sound field [7].

This characteristic sensor supports innovative products used in diverse acoustic applications, such as sound source location, airborne transfer path analysis and panel contribution analysis, in situ determination of acoustic properties, non-contact vibration measurements or micropore leak-testing [7].

The company

The enterprise that develops and markets these acoustic testing systems is Microflown Technologies. Located in Arnhem, a city situated in the eastern part of the Netherlands, it is a technological company that sells its products and services to an extensive range of market segments such as aerospace, automotive, appliances, manufacturing industries and defence industry. As it is said in [7] "*within the industry, Microflown based testing methods are used from the development of new prototypes till the end of line acoustic quality testing during manufacturing*".

The problem

One of the devices conceived by the company is a P-U probe [8] that means that both, the pressure and the particle velocity, are measured.

Experimentally, it was proven that its structure affects the transducers sensitivity, which is improved when a local increment of the gain associated to the particle velocity is produced. In order to upgrade it, the aim of this master thesis is the optimization of the geometry of the probe using a Finite Element Method (FEM) simulation framework. In addition, the behaviour of the device will be studied, namely, the frequency response of the velocity and the pressure field and their directivity patterns. The numerical results obtained in the simulations will be also compared with experimental measurements.

During the first chapter some basic acoustic concepts are described (see references [5] and [9]) under the assumption of time dependency and also in the frequency domain. Moreover, the description of the scattering problem (see reference [4]) is also introduced in this chapter. Then, in Chapter 2, the acoustic models stated are described in detail. First, an acoustic problem stated in a bounded domain is introduced for both cases, with and without viscosity. These are preliminary academic problems, easier to be solved than the acoustic problem stated in an unbounded domain. The technique used to deal with unbounded domains is the Perfect Matched Layers [1].

Once the acoustic models are stated, the following step is solving them using the Finite Element Method (see reference [1]), which is implemented by using a `FEniCS–Python` code [6]. This computer code has been validated with some test examples described in Chapter 5. As it has been mentioned above, the programming language used in this master thesis has been `Python`. It is a free interpreted software, which allows the users to implement easily and efficiently object-oriented codes. Another of their characteristics is the extension facility: it is easy to write new modules for `Python`, even using `C` or `C++`.

Furthermore, there are some modules included in `Python`, which can be loaded as libraries to obtain enriched capabilities. Some examples are `NumPy` [11], with advanced maths functions, `matplotlib` [3], used for data analysis or plotting, and `FEniCS`, which has been used to implement the finite element method.

Finally, thanks to the use of the free design software `Salome` (see reference [2]), different geometries of the P-U probe have been defined.

These geometrical configurations have been obtained varying the values of a number of discrete parameters. The optimization has been performed by using a sweeping procedure in a range of physical feasible values. The numerical results obtained in this parametric optimization are fully described in Chapter 6. Finally, some final remarks are included in the conclusion chapter.

Chapter 1

Acoustics background

According to [5], "*acoustics is the science of sound, that is, wave motion in gases, liquids and solids, and the effects of such wave motion.*" In this chapter, some basic concepts will be explained in order to easy the reading of this master thesis and its full comprehension. Most of the definitions included are based on [9]. First, the time dependent definitions will be introduced, and then, those ones which related to the frequency domain.

1.1 Time domain

Since the aim of this work is focused on the numerical simulation of the pressure and velocity fields for an acoustic intensity probe, which allows us improve its gain associated to the measurements of the particle velocity, it is necessary to define the acoustic concepts associated to that problem.

The first physical concept to be presented is the pressure one:

Definition 1.1.1. The **acoustic pressure** P is the disturbance with respect to the equilibrium state in the ambient, which depends on time and it is caused, usually by a sound wave. The unit in the International System (SI) are Pascals [Pa] (see Ref. [9]).

To describe completely the mathematical model associated to the acoustic propagation phenomena, it is also necessary to introduce the displacement and the velocity fields.

Definition 1.1.2. The **acoustic displacement** Ξ describes the displacement variations of a sound field, i.e., of a region where sound waves

are propagated, with respect to the equilibrium state. The SI units are meters [m] (see Ref. [9]).

The particle velocity is defined straightforwardly as the time-derivative of the displacement. It is denoted by $\boldsymbol{U}$, and the units in the SI are meters per seconds [m/s]. Another important concept to be introduced is the intensity of the sound field:

Definition 1.1.3. The **intensity** $\boldsymbol{I}$ is a magnitude which provides information about the energy flux associated with the acoustic wave propagation. If the pressure and the particle velocity is known, $\boldsymbol{I} = P\boldsymbol{U}$.

Furthermore, the viscosity concept is also introduced:

Definition 1.1.4. The **viscosity** ν of a fluid is the resistance that it is exerted to flow. The SI units are Pascals times second [Pa·s].

Taking into account the definitions written above, the linear motion equation of a compressible viscous fluid is given by:

$$\rho\, \partial_t \boldsymbol{U} - \nu\, \nabla(\operatorname{div} \boldsymbol{U}) = -\nabla P, \tag{1.1}$$

where ρ is the mass density in kilograms per cubic meter [kg/m^3] and ν denotes the viscosity. Moreover, the constitutive law for compressible fluids it is also necessary. It relates pressure and displacement as follows:

$$\partial_t\, P = -\rho c^2 \operatorname{div} \boldsymbol{U}, \tag{1.2}$$

where c is fluid sound speed [m/s].

1.2 Frequency domain

Once it is assumed a time-harmonic dependency of the acoustic fields, this is, supposing to work in the frequency domain, the first concepts to be introducing are frequencies and angular frequencies.

Definition 1.2.1. Frequency is a name used in physics to denote the number of waves that pass a fixed point in unit time. Its units are Hertz [Hz]. When this quantity is multiplied by 2π, it is named **angular frequency** ω, and its units are radians per second [rad/s] (see Ref. [9]).

Then, the wave number concept is described:

Definition 1.2.2. The **acoustic wave number** (k) provides the number of times that a wave is vibrating in a distance unit. Its units in the SI are m^{-1}. It is given by the formula

$$k = \sqrt{\frac{\omega^2 \rho}{\rho c^2 - i\omega\nu}} \in \mathbb{C}.$$

When the viscosity is not taking in account, the wave number expressions becomes a positive number: $k = \frac{\omega}{c} \in \mathbb{R}$.

It is possible to define pressure, displacement and velocity as follows:

$$\begin{aligned} P\left(x,y,z;t\right) &= \mathrm{Re}\left(p(x,y,z)e^{i\omega t}\right), \\ \boldsymbol{\Xi}\left(x,y,z;t\right) &= \mathrm{Re}\left(\boldsymbol{\xi}(x,y,z)e^{i\omega t}\right), \\ \boldsymbol{U}\left(x,y,z;t\right) &= \mathrm{Re}\left(\boldsymbol{u}(x,y,z)e^{i\omega t}\right), \end{aligned}$$

where p, $\boldsymbol{\xi}$ and $\boldsymbol{u}$ are complex-valued fields that denote pressure, displacement and velocity, respectively, in the frequency domain, i the imaginary unit; t, the time and ω, the angular frequency. From here is easy to deduce that:

$$\boldsymbol{U}(x,y,z;t) = \mathrm{Re}\left(i\omega\boldsymbol{\xi}(x,y,z;t)e^{i\omega t}\right)$$

and hence

$$i\,\omega\,\boldsymbol{u}(x,y,z) = -\omega^2\boldsymbol{\xi}.$$

Then, it is possible to write the Equation (1.1) in the frequency domain:

$$\begin{cases} \rho\, i\,\omega\boldsymbol{u} - \nu\,\nabla(\mathrm{div}\,\boldsymbol{u}) = -\nabla p, \\ \boldsymbol{u} = i\,\omega\boldsymbol{\xi}, \end{cases}$$

and writing it using the displacement and pressure fields:

$$-\rho\,\omega^2\boldsymbol{\xi} - \nu\, i\,\omega\,\nabla(\mathrm{div}\,\boldsymbol{\xi}) = -\nabla p.$$

If the divergence of first equation is computed, it holds

$$-\omega^2 \rho \, \mathrm{div}\boldsymbol{\xi} - \nu \; i\,\omega \, \mathrm{div}\left(\nabla\left(\mathrm{div}\boldsymbol{\xi}\right)\right) = -\mathrm{div}\left(\nabla p\right),$$

and using (1.2) in the frequency domain and replacing $\mathrm{div}\boldsymbol{\xi}$ by $\frac{-1}{\rho c^2}\, p$:

$$\begin{aligned} \omega^2 \rho \, p + \nu \; i\,\omega \, \mathrm{div}\left(\nabla p\right) + \rho c^2 \, \mathrm{div}\left(\nabla p\right) &= 0 \\ \Rightarrow \omega^2 \rho \, p + \left(\rho c^2 + \nu \, i\,\omega\right) \Delta p &= 0 \\ \Rightarrow -\omega^2 \rho \, p - \left(\rho c^2 + \nu \, i\,\omega\right) \Delta p &= 0. \end{aligned}$$

This last equation is known as the acoustic Helmholtz equation:

$$-\omega^2 \rho \, p - \left(\rho c^2 + i\omega\nu\right) \Delta p = 0, \tag{1.3}$$

and it can be rewritten as follows:

$$-\Delta p - k^2 p = 0, \tag{1.4}$$

where k denotes the wave number.

Finally, the pressure gain or the velocity gain is how is this magnitude increased with respect to a prescribed reference. They will be computed as the difference between numerical results and some known data. In this work the gain that is going to be computed is for the sound pressure level and for the particle velocity level, being a level the logarithm of the ratio of two physical quantities.

Definition 1.2.3. The **sound pressure level** (SPL) is the ratio between the pressure and its reference value ($p_{\mathrm{ref}} = 2 \times 10^{-5}\mathrm{Pa}$):

$$\mathrm{SPL} = 20 \log_{10}\left(\frac{p}{p_{\mathrm{ref}}}\right),$$

where p is the modulus of the pressure field at a fixed spatial point.

Definition 1.2.4. The **particle velocity level** (PVL) is the ratio between the velocity and its reference value ($u_{\mathrm{ref}} = 2 \times 10^{-8}\mathrm{m/s}$):

$$20\,\mathrm{PVL} = \log_{10}\left(\frac{u}{u_{\mathrm{ref}}}\right),$$

where u is the modulus of the velocity at a given spatial point.

Despite the fact that these two quantities are dimensionless, there is an unit in acoustics that expresses this amount: its name is Decibels [dB].

In addition, the concepts of monopole source and plane wave is needed to be described (the monopole equation is used in the validation test problems described in Chapter 5 and the plane wave is used to obtain the numerical results in Chapter 6).

Definition 1.2.5. A **monopole** is a sound generation mechanism which emits energy to all direction and it is given by the following Helmholtz solution with a Dirac's delta source term:

$$p(x,y,z) = \frac{A}{\sqrt{x^2+y^2+z^2}}\, e^{-i\,k\sqrt{x^2+y^2+z^2}}, \tag{1.5}$$

where A is the amplitude of the wave.

Definition 1.2.6. A **plane wave** is a solution of Helmholtz equation, which has the following expression:

$$p(x,y,z) = A\, e^{-i\,\boldsymbol{k}\cdot(x,y,z)}, \tag{1.6}$$

where $\boldsymbol{k}$ represents the wave number vector. This vector has the magnitude of the wave number, introduced in Definition (1.2.2), and shows the direction in which the plane wave if propagating.

Finally, the concept of directivity pattern has to be introduced:

Definition 1.2.7. The **directivity pattern** is a characteristic property of the sound waves which expresses the sound pressure level or the particle velocity level depending on an angle argument. Its value is computed by solving an scattering problem and the value of the angle argument corresponds to the angle of incidence of the plane wave involved in the scattering problem. It is usually plotted in polar coordinates.

In general, the shape of the directivity pattern is arbitrary. However, the numerical results obtained in this master thesis show omnidirectional ones, that means that the SPL values are similar in all directions, and the *"eight shape"* pattern, which receives this name because the shape is similar to a horizontally rotated number eight.

1.3 Scattering problem

The scattering problem is the name that receives a problem in which a plane wave impinges into a structure surrounded by a fluid. In order to introduce the plane wave as a boundary condition, it is necessary to apply a coupling condition between the fluid and the structure, which means that the normal displacements on the structure boundary are driven by the normal derivative of the total pressure field. Denoting by p_{inc} the pressure of the incident plane wave, which impinges on the structure, the following equation is obtained:

$$\rho\omega^2 \boldsymbol{\xi} \cdot \boldsymbol{n} = \frac{\partial \left(p + p_{\text{inc}}\right)}{\partial \boldsymbol{n}},$$

where, p is the scattering pressure field, $\boldsymbol{\xi}$ is the displacement field on the structure, and $\boldsymbol{n}$ is the outward unit normal vector on this boundary. Moreover, when the structure is considered as a rigid solid then $\boldsymbol{\xi} = \mathbf{0}$, so

$$\frac{\partial p}{\partial \boldsymbol{n}} = -\frac{\partial p_{\text{inc}}}{\partial \boldsymbol{n}}.$$

Consequently, taking into account the Helmholtz equation (see Eq. 1.4), the scattering problem will be:

$$\begin{cases} -\Delta p - k^2 p = 0 & \text{in } \Omega, \\ \frac{\partial p}{\partial \boldsymbol{n}} = -\frac{\partial p_{\text{inc}}}{\partial \boldsymbol{n}} & \text{on } \Gamma, \\ + \text{ Sommerfeld radiation condition.} \end{cases} \tag{1.7}$$

The pressure obtained when this problems is solved, is the scattering pressure, so to obtain the total pressure field it is necessary to add the one associated to the plane wave. Moreover, as it can be described in the numerical results of Chapter 6, to compute the gain associated to a structure it will be used the difference between the SPL and the PVL levels associated to the total pressure field p and those ones associated to the incident pressure field p_{inc}.

Chapter 2

Mathematical models

The acoustic models presented in this chapter involve the Helmholtz equation, this is it assumed that sound field have a time-harmonic dependency. For the sake of simplicity in the exposition, first, the acoustic problems will be stated in bounded domains, using with and without viscosity, which will be detailed in the first section. Then, since the industrial application of this master thesis involves an acoustic problem at free field conditions, the Helmholtz problem will be stated in an unbounded domain, where only the problem without viscosity will be considered.

Due to the fact that the pressure p is a complex-valued field, and due to restrictions imposed by the code implementation (described in Chapter 4), it will be necessary to split it in its real and imaginary parts, this is

$$p = p_{\mathrm{re}} + i\, p_{\mathrm{im}}.$$

2.1 Modelling acoustic propagation in bounded domains

Firstly, a three dimensional bounded computational domain is introduced. The considered consists in a solid structure surrounded by a fluid. Neumann conditions were imposed in the structure boundary Γ_{s}, while Dirichlet

conditions were placed in the outer fluid boundary Γ_{out}. Both cases, with and without viscosity, are fully described in the following subsections.

2.1.1 Model without viscosity

The first problem to be introduced is the one which does not take in account the viscosity. The equation that governs this problem is given by Eq. (1.3) with $\nu = 0$, so the model is:

$$\begin{cases} -\omega^2 \rho\, p - \rho c^2 \Delta p = 0 & \text{in } \Omega, \\ \frac{\partial p}{\partial \boldsymbol{n}} = g = g_{\text{re}} + i\, g_{\text{im}} & \text{on } \Gamma_{\text{s}}, \\ p = f = f_{\text{re}} + i\, f_{\text{im}} & \text{on } \Gamma_{\text{out}}. \end{cases} \tag{2.1}$$

Dividing it into its real and imaginary parts. It holds,

$$\begin{cases} -\omega^2 \rho\, p_{\text{re}} - \rho c^2 \Delta p_{\text{re}} = 0 & \text{in } \Omega, \\ -\omega^2 \rho\, p_{\text{im}} - \rho c^2 \Delta p_{\text{im}} = 0 & \text{in } \Omega, \\ \frac{\partial p_{\text{re}}}{\partial \boldsymbol{n}} = g_{\text{re}},\ \frac{\partial p_{\text{im}}}{\partial \boldsymbol{n}} = g_{\text{im}} & \text{on } \Gamma_{\text{s}}, \\ p_{\text{re}} = f_{\text{re}},\ p_{\text{im}} = f_{\text{im}} & \text{on } \Gamma_{\text{out}}. \end{cases} \tag{2.2}$$

In order to obtain the variational formulation of the problem (2.2), the complex-valued tests functions will be divided in its real and imaginary parts, denoted by φ_{re} and φ_{im}, respectively. The unknowns fields p_{re} and p_{im} will be assumed belonging to the following functional space:

$$\mathrm{H}^1(\Omega) = \left\{ f : \Omega \to \mathbb{C} : \int_\Omega |f|^2 \,\mathrm{d}V + \int_\Omega |\nabla f|^2 \,\mathrm{d}V < \infty \right\}$$

whereas the test functions are in $\mathrm{H}^1(\Omega)$ satisfying the restrictions $\varphi_{\text{re}|_{\Gamma_{\text{out}}}} = 0$ and $\varphi_{\text{im}|_{\Gamma_{\text{out}}}} = 0$.

Therefore, using the Green's Identity

$$\int_\Omega \psi \Delta \varphi \,\mathrm{d}V + \int_\Omega \nabla \psi \cdot \nabla \varphi \,\mathrm{d}V = \int_{\partial\Omega} \psi \left(\nabla \varphi \cdot \boldsymbol{n}\right) \,\mathrm{d}\gamma \;\forall \psi \in \mathcal{C}^1(\Omega),\, \varphi \in \mathcal{C}^2(\Omega) \tag{2.3}$$

the weak formulation of the model (2.2) can be written as follows:

$$\begin{cases} \int_\Omega -\omega^2 \rho\, p_{\mathrm{re}}\, \varphi_{\mathrm{re}}\, \mathrm{d}V + \int_\Omega \rho c^2\, \nabla p_{\mathrm{re}} \cdot \nabla \varphi_{\mathrm{re}}\, \mathrm{d}V - \int_{\Gamma_\mathrm{s}} \rho c^2\, g_{\mathrm{re}}\, \varphi_{\mathrm{re}}\, \mathrm{d}\gamma = 0, \\ \int_\Omega -\omega^2 \rho\, p_{\mathrm{im}}\, \varphi_{\mathrm{im}}\, \mathrm{d}V + \int_\Omega \rho c^2\, \nabla p_{\mathrm{im}} \cdot \nabla \varphi_{\mathrm{im}}\, \mathrm{d}V - \int_{\Gamma_\mathrm{s}} \rho c^2\, g_{\mathrm{im}}\, \varphi_{\mathrm{im}}\, \mathrm{d}\gamma = 0, \end{cases}$$

for all $\varphi_{\mathrm{re}},\, \varphi_{\mathrm{im}} \in \mathrm{H}^1(\Omega)$ such that $\varphi_{\mathrm{re}|_{\Gamma_{\mathrm{out}}}} = \varphi_{\mathrm{im}|_{\Gamma_{\mathrm{out}}}} = 0$.

Due φ_{re} and φ_{im} can be chosen separately, it was possible to sum both equations in order to get the final weak formulation:

$$\begin{cases} \text{Given functions } g_{\mathrm{re}}, g_{\mathrm{im}} \in \mathrm{L}^2(\Gamma_\mathrm{s}),\ f_{\mathrm{re}}, f_{\mathrm{im}} \in \mathrm{H}^{\frac{1}{2}}(\Gamma_{\mathrm{out}}) \text{ and } \omega, \rho, c > 0, \\ \text{find } p_{\mathrm{re}}, p_{\mathrm{im}} \in \mathrm{H}^1(\Omega) \text{ such that } p_{\mathrm{re}|_{\Gamma_{\mathrm{out}}}} = f_{\mathrm{re}},\ p_{\mathrm{im}|_{\Gamma_{\mathrm{out}}}} = f_{\mathrm{im}} \text{ and} \\ \int_\Omega -\omega^2 \rho\, p_{\mathrm{re}}\, \varphi_{\mathrm{re}}\, \mathrm{d}V + \int_\Omega \rho c^2\, \nabla p_{\mathrm{re}} \cdot \nabla \varphi_{\mathrm{re}}\, \mathrm{d}V + \int_\Omega -\omega^2 \rho\, p_{\mathrm{im}}\, \varphi_{\mathrm{im}}\, \mathrm{d}V + \\ \int_\Omega \rho c^2\, \nabla p_{\mathrm{im}} \cdot \nabla \varphi_{\mathrm{im}}\, \mathrm{d}V = \int_{\Gamma_\mathrm{s}} \rho c^2\, g_{\mathrm{re}}\, \varphi_{\mathrm{re}}\, \mathrm{d}\gamma + \int_{\Gamma_\mathrm{s}} \rho c^2\, g_{\mathrm{im}}\, \varphi_{\mathrm{im}}\, \mathrm{d}\gamma \\ \qquad\qquad \forall \varphi_{\mathrm{re}}, \varphi_{\mathrm{im}} \in \mathrm{H}^1(\Omega) \text{ with } \varphi_{\mathrm{re}|_{\Gamma_{\mathrm{out}}}} = 0,\ \varphi_{\mathrm{im}|_{\Gamma_{\mathrm{out}}}} = 0, \end{cases} \tag{2.4}$$

where $\mathrm{H}^{\frac{1}{2}}(\Gamma_{\mathrm{out}})$ is the trace space of H^1-functions on Γ_{out} and $\mathrm{L}^2(\Gamma_\mathrm{s})$ is the following functional space:

$$\mathrm{L}^2(\Gamma_\mathrm{s}) = \left\{ f : \Gamma_\mathrm{s} \to \mathbb{C} : \int_{\Gamma_\mathrm{s}} |f|^2\, \mathrm{d}\gamma < \infty \right\}.$$

The weak formulation described above can be rewritten as $\mathcal{A}\left((p_{\mathrm{re}},\, p_{\mathrm{im}}), (\varphi_{\mathrm{re}},\, \varphi_{\mathrm{im}})\right) = \mathcal{L}\left(\varphi_{\mathrm{re}},\, \varphi_{\mathrm{im}}\right)$, being

$$\begin{aligned} \mathcal{A}\left((p_{\mathrm{re}},\, p_{\mathrm{im}}), (\varphi_{\mathrm{re}},\, \varphi_{\mathrm{im}})\right) = &\int_\Omega -\omega^2 \rho\, p_{\mathrm{re}}\, \varphi_{\mathrm{re}}\, \mathrm{d}V + \int_\Omega \rho c^2\, \nabla p_{\mathrm{re}} \cdot \nabla \varphi_{\mathrm{re}}\, \mathrm{d}V \\ &+ \int_\Omega -\omega^2 \rho\, p_{\mathrm{im}}\, \varphi_{\mathrm{im}}\, \mathrm{d}V + \int_\Omega \rho c^2\, \nabla p_{\mathrm{im}} \cdot \nabla \varphi_{\mathrm{im}}\, \mathrm{d}V \end{aligned}$$

and

$$\mathcal{L}\left(\varphi_{\mathrm{re}},\, \varphi_{\mathrm{im}}\right) = \int_{\Gamma_{\mathrm{s}}} \rho c^2\, g_{\mathrm{re}}\, \varphi_{\mathrm{re}}\, \mathrm{d}\gamma + \int_{\Gamma_{\mathrm{s}}} \rho c^2\, g_{\mathrm{im}}\, \varphi_{\mathrm{im}}\, \mathrm{d}\gamma.$$

However, it is important to clarify that it is possible to solve (2.2) working with the real and the imaginary part of the variational problem separately, because they are not coupled.

2.1.2 Model with viscosity

Secondly, the viscosity coefficient was introduced in (2.1). In this case, the main equation of the model is formally equivalent to Eq. (1.3), with $\nu \neq 0$, but with different constant coefficients. Then, including in the model the boundary conditions, the completed problem is given as follows:

$$\begin{cases} -\omega^2 \rho\, p - \left(\rho c^2 + i\omega\nu\right) \Delta p = 0 & \text{in } \Omega, \\ \frac{\partial p}{\partial \boldsymbol{n}} = g = g_{\mathrm{re}} + i\, g_{\mathrm{im}} & \text{on } \Gamma_{\mathrm{s}}, \\ p = f = f_{\mathrm{re}} + i\, f_{\mathrm{im}} & \text{on } \Gamma_{\mathrm{out}}. \end{cases} \tag{2.5}$$

Separating the pressure in its real and imaginary part and rewriting the equations to have only real-valued coefficients it holds,

$$\begin{cases} -\omega^2 \rho\, p_{\mathrm{re}} - \left(\rho c^2\, \Delta p_{\mathrm{re}} - \omega\nu\, \Delta p_{\mathrm{im}}\right) = 0 & \text{in } \Omega, \\ -\omega^2 \rho\, p_{\mathrm{im}} - \left(\rho c^2\, \Delta p_{\mathrm{im}} + \omega\nu\, \Delta p_{\mathrm{re}}\right) = 0 & \text{in } \Omega, \\ \frac{\partial p_{\mathrm{re}}}{\partial \boldsymbol{n}} = g_{\mathrm{re}},\ \frac{\partial p_{\mathrm{im}}}{\partial \boldsymbol{n}} = g_{\mathrm{im}} & \text{on } \Gamma_{\mathrm{s}}, \\ p_{\mathrm{re}} = f_{\mathrm{re}},\ p_{\mathrm{im}} = f_{\mathrm{im}} & \text{on } \Gamma_{\mathrm{out}}. \end{cases} \tag{2.6}$$

In this problem, on the contrary that in the one proposed in Section 2.1, it was not possible to solve it as two separate different problems, the real and the imaginary one, since, both parts are coupled.

Once the problem (2.6) was written, the following step was to compute the variational formulation. As it has been shown in Subsection 2.1.1, the

solution, and the test function, should be divided into its real and imaginary parts. Again, the test functions, $\varphi_{\text{re}}, \varphi_{\text{im}}$ belong to $\text{H}^1(\Omega)$, satisfying they are null on Γ_{out}.

First, the variational formulation of the real and imaginary part of the problem was computed separately using the Green's Identity (Eq. (2.3)), as follows:

$$\int_\Omega -\omega^2\rho\, p_{\text{re}}\varphi_{\text{re}}\,\text{d}V + \int_\Omega \rho c^2 \nabla p_{\text{re}}\cdot\nabla\varphi_{\text{re}}\,\text{d}V + \int_\Omega (-\omega\nu)\nabla p_{\text{im}}\cdot\nabla\varphi_{\text{re}}\,\text{d}V = \int_{\Gamma_{\text{s}}} \rho c^2 g_{\text{re}}\varphi_{\text{re}}\,\text{d}\gamma + \int_{\Gamma_{\text{s}}} (-\omega\nu) g_{\text{im}}\varphi_{\text{re}}\,\text{d}\gamma \quad \forall\varphi_{\text{re}}\in \text{H}^1 \text{ with } \varphi_{\text{re}|_{\Gamma_{\text{out}}}} = 0,$$

$$\int_\Omega -\omega^2\rho\, p_{\text{im}}\varphi_{\text{im}}\,\text{d}V + \int_\Omega \rho c^2 \nabla p_{\text{im}}\cdot\nabla\varphi_{\text{im}}\,\text{d}V + \int_\Omega \omega\nu\nabla p_{\text{re}}\cdot\nabla\varphi_{\text{im}}\,\text{d}V = \int_{\Gamma_{\text{s}}} \rho c^2 g_{\text{im}}\varphi_{\text{im}}\,\text{d}\gamma + \int_{\Gamma_{\text{s}}} \omega\nu g_{\text{re}}\varphi_{\text{im}}\,\text{d}\gamma \quad \forall\varphi_{\text{im}}\in \text{H}^1 \text{ with } \varphi_{\text{im}|_{\Gamma_{\text{out}}}} = 0.$$

Finally, adding both expressions, the complete weak formulation was obtained:

$$\begin{cases} \text{Given functions } g_{\text{re}}, g_{\text{im}} \in \text{L}^2(\Gamma_{\text{s}}),\ f_{\text{re}}, f_{\text{im}} \in \text{H}^{\frac{1}{2}}(\Gamma_{\text{out}}) \text{ and } \omega,\rho,c,\nu>0, \\ \text{find } p_{\text{re}}, p_{\text{im}} \in \text{H}^1(\Omega) \text{ such that } p_{\text{re}|_{\Gamma_{\text{out}}}} = f_{\text{re}},\ p_{\text{im}|_{\Gamma_{\text{out}}}} = f_{\text{im}} \text{ and} \\ \int_\Omega (-\omega^2\rho)\, p_{\text{re}}\varphi_{\text{re}}\,\text{d}V + \int_\Omega (-\omega^2\rho)\, p_{\text{im}}\varphi_{\text{im}}\,\text{d}V + \int_\Omega \rho c^2\,\nabla p_{\text{re}}\cdot\nabla\varphi_{\text{re}}\,\text{d}V + \\ \int_\Omega \rho c^2\,\nabla p_{\text{im}}\cdot\nabla\varphi_{\text{im}}\,\text{d}V + \int_\Omega (-\omega\nu)\nabla p_{\text{im}}\cdot\nabla\varphi_{\text{re}} + \int_\Omega \omega\nu\nabla p_{\text{re}}\cdot\nabla\varphi_{\text{im}} = \\ \int_{\Gamma_{\text{s}}} \rho c^2\, g_{\text{re}}\,\varphi_{\text{re}}\,\text{d}\gamma + \int_{\Gamma_{\text{s}}} \rho c^2\, g_{\text{im}}\,\varphi_{\text{im}}\,\text{d}\gamma + \int_{\Gamma_{\text{s}}} (-\omega\nu)\, g_{\text{im}}\,\varphi_{\text{re}}\,\text{d}\gamma + \int_{\Gamma_{\text{s}}} \omega\nu\, g_{\text{re}}\,\varphi_{\text{im}}\,\text{d}\gamma \\ \qquad\qquad \forall\varphi_{\text{re}},\varphi_{\text{im}} \in \text{H}^1(\Omega) \text{ with } \varphi_{\text{re}|_{\Gamma_{\text{out}}}} = 0,\ \varphi_{\text{im}|_{\Gamma_{\text{out}}}} = 0. \end{cases} \tag{2.7}$$

2.2 Modelling acoustic propagation at free field conditions

In order to simulate the acoustic probe, an unbounded domain has to be introduced. So, the computational domain consisted in the expterior of a structure surrounded by a fluid occupying the entire three-dimensional space.

In this case, when the acoustic fields are computed in unbounded domains, it is called that the computations are performed at free field conditions. The governing equations involve the Helmholtz equation (see Eq. (1.4)) without viscosity ($k = \omega/c$). The boundary conditions include a Neumann condition settled on the boundary of the structure. However, this is not enough to guarantee that there are not reflection waves coming from the infinity. To avoid it, it is necessary to introduce the Sommerfeld radiation condition:

$$\lim_{r\to\infty} r\left(\frac{\partial p}{\partial r} - ikp\right) = 0, \tag{2.8}$$

where $r = \sqrt{x^2 + y^2 + z^2}$.

Therefore, the problem, which models the acoustic propagation in an unbounded domains with Neumann boundary conditions is described as follows:

$$\begin{cases} -\Delta p - k^2\, p = 0 & \text{in } \Omega, \\ \frac{\partial p}{\partial \boldsymbol{n}} = g & \text{on } \Gamma_{\mathrm{s}}, \\ \lim\limits_{r\to\infty} r\left(\dfrac{\partial p}{\partial r} - ikp\right) = 0. \end{cases} \tag{2.9}$$

However, the condition given by (2.8) is not easy to implement directly in a computer code. For this reason, the radiation conditions has been replaced by using the Perfect Matched Layer technique.

2.2.1 Perfect Matched Layers

Perfect Matched Layer (PML) (see [1]) is a method to build absorbing boundary conditions. The idea consists in introducing an outer layer to the computational domain in order to absorb completely the waves that come from the structure. This will avoid the spurious reflections coming from the outer boundary of a truncated computational domain. Consequently, the PML technique is used to simulate numerically wave propagation problems in unbounded domains (this is, to compute the pressure and the particle velocity fields at free field conditions).

In Figure 2.1 it is shown PML distribution for a Cartesian geometry in three dimensions. As it can be observed, the PML is divided in seven

subdomains, which surround the fluid domain. This division is made because it is necessary to absorb the waves in every direction. If j denotes the direction where the waves come from ($j \in \{x, y, z\}$), it is possible to denote by d_j the distance from the origin to the j-inner layer of the PML and by L_j the distance from the j-inner PML boundary to the j-outer boundary. In this manner, the different subdomains are defined as follows:

$$
\begin{cases}
\Omega_x = [d_x,\, d_x + L_x] \times [-d_y,\, d_y] \times [-d_z,\, d_z] \cup \\
\qquad [-d_x - L_x,\, -d_x] \times [-d_y,\, d_y] \times [-d_z,\, d_z]\,, \\
\\
\Omega_y = [-d_x,\, d_x] \times [d_y,\, d_y + L_y] \times [-d_z,\, d_z] \cup \\
\qquad [-d_x,\, d_x] \times [-d_y - L_y,\, -d_y] \times [-d_z,\, d_z]\,, \\
\\
\Omega_z = [-d_x,\, d_x] \times [-d_y,\, d_y] \times [d_z,\, d_z + L_z] \cup \\
\qquad [-d_x,\, d_x] \times [-d_y,\, d_y] \times [-d_z - L_z,\, -d_z]\,, \\
\\
\Omega_{xy} = [d_x,\, d_x + L_x] \times [-d_y - L_y,\, -d_y] \times [-d_z,\, d_z] \cup \\
\qquad [d_x,\, d_x + L_x] \times [d_y,\, d_y + L_y] \times [-d_z,\, d_z] \cup \\
\qquad [-d_x - L_x,\, -d_x] \times [d_y,\, d_y + L_y] \times [-d_z,\, d_z] \cup \\
\qquad [-d_x - L_x,\, -d_x] \times [-d_y - L_y,\, -d_y] \times [-d_z,\, d_z]\,, \\
\\
\Omega_{xz} = [d_x,\, d_x + L_x] \times [-d_y,\, d_y] \times [-d_z - L_z,\, -d_z] \cup \\
\qquad [d_x,\, d_x + L_x] \times [-d_y,\, d_y] \times [d_z,\, d_z + L_z] \cup \\
\qquad [-d_x - L_x,\, -d_x] \times [-d_y,\, d_y] \times [d_z,\, d_z + L_z] \cup \\
\qquad [-d_x - L_x,\, -d_x] \times [-d_y,\, d_y] \times [-d_z - L_z,\, -d_z]\,, \\
\\
\Omega_{yz} = [-d_x,\, d_x] \times [-d_y - L_y,\, -d_y] \times [-d_z - L_z,\, -d_z] \cup \\
\qquad [-d_x,\, d_x] \times [-d_y - L_y,\, -d_y] \times [d_z,\, d_z + L_z] \cup \\
\qquad [-d_x,\, d_x] \times [d_y,\, d_y + L_y] \times [d_z,\, d_z + L_z] \cup \\
\qquad [-d_x,\, d_x] \times [d_y,\, d_y + L_y] \times [-d_z - L_z,\, -d_z]\,, \\
\\
\Omega_{xyz} = [-d_x - L_x,\, -d_x] \times [-d_y - L_y,\, -d_y] \times [-d_z - L_z,\, -d_z] \cup \\
\qquad [-d_x - L_x,\, -d_x] \times [-d_y - L_y,\, -d_y] \times [d_z,\, d_z + L_z] \cup \\
\qquad [-d_x - L_x,\, -d_x] \times [d_y,\, d_y + L_y] \times [-d_z - L_z,\, -d_z] \cup \\
\qquad [-d_x - L_x,\, -d_x] \times [d_y,\, d_y + L_y] \times [d_z,\, d_z + L_z] \cup \\
\qquad [d_x,\, d_x + L_x] \times [-d_y - L_y,\, -d_y] \times [-d_z - L_z,\, -d_z] \cup \\
\qquad [d_x,\, d_x + L_x] \times [-d_y - L_y,\, -d_y] \times [d_z,\, d_z + L_z] \cup \\
\qquad [d_x,\, d_x + L_x] \times [d_y,\, d_y + L_y] \times [-d_z - L_z,\, -d_z] \cup \\
\qquad [d_x,\, d_x + L_x] \times [d_y,\, d_y + L_y] \times [d_z,\, d_z + L_z].
\end{cases}
\tag{2.10}
$$

Besides, Ω_0 denotes the fluid domain:

$$\Omega_0 = [-d_x, d_x] \times [-d_y, d_y] \times [-d_z, d_z]. \tag{2.11}$$

Each subdomain absorbs one different kind of waves: Ω_x, Ω_y, and Ω_z are the layers that absorb the waves coming only from the x, y and z directions, respectively. Ω_{xy}, Ω_{xz} and Ω_{yz} absorb the waves that come from two different directions, those are indicated by the subscripts in the notation. Finally, Ω_{xyz} absorbs the waves that come from the three Cartesian directions simultaneously.

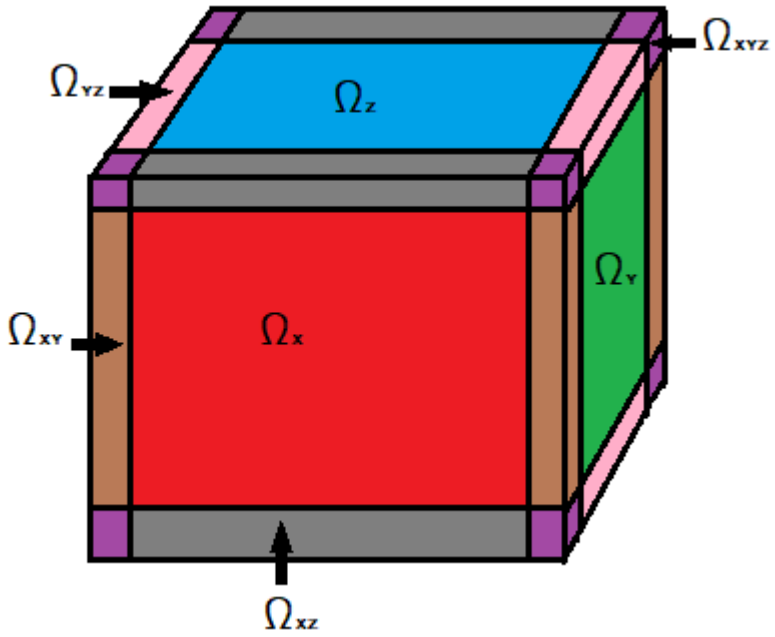


Figure 2.1: Sketch of the PML subdomain configuration in a three dimensional domain.

To obtain the partial differential equations associated to the PML, it is necessary to introduce a complex-valued change of variables in the Helmholtz equation (see Eq. (1.4)). This formal procedure consists in replacing the partial derivatives $\frac{\partial}{\partial x}$, $\frac{\partial}{\partial y}$, $\frac{\partial}{\partial z}$ by $\frac{1}{\gamma_x}\frac{\partial}{\partial x}$, $\frac{1}{\gamma_y}\frac{\partial}{\partial y}$ and $\frac{1}{\gamma_z}\frac{\partial}{\partial z}$, respectively, where γ_x, γ_y, γ_z are given by

$$\begin{cases} \gamma_x = 1 - i\frac{\sigma_x(x,y,z)}{\omega}, \\ \gamma_y = 1 - i\frac{\sigma_y(x,y,z)}{\omega}, \\ \gamma_z = 1 - i\frac{\sigma_z(x,y,z)}{\omega}, \end{cases} \tag{2.12}$$

being σ_x, σ_y and σ_z the absorption functions of the PML. Those functions depend on the PML subdomain setting. this is, they are defined by

$$\begin{aligned} \sigma_x(x,y,z) &= \begin{cases} \sigma_0 & \text{if } (x,y,z) \in \{\Omega_x,\, \Omega_{xy},\, \Omega_{xz},\, \Omega_{xyz}\}, \\ 0 & \text{in other case,} \end{cases} \\ \sigma_y(x,y,z) &= \begin{cases} \sigma_0 & \text{if } (x,y,z) \in \{\Omega_y,\, \Omega_{xy},\, \Omega_{yz},\, \Omega_{xyz}\}, \\ 0 & \text{in other case,} \end{cases} \\ \sigma_z(x,y,z) &= \begin{cases} \sigma_0 & \text{if } (x,y,z) \in \{\Omega_z,\, \Omega_{xz},\, \Omega_{yz},\, \Omega_{xyz}\}, \\ 0 & \text{in other case.} \end{cases} \end{aligned} \tag{2.13}$$

Notice that, throughout this master thesis, the absorbing functions in (2.13) will be assumed piecewise constant, and in particular, taking the same constant value σ_0 on every PML subdomain. However, the optimal value for σ_0 is strongly dependent on the frequency of the problem and on the thickness of the PML. Hence, it will be determined numerically (see Chapters 5 and 6 for a detailed discussion).

When this complex-valued change of variables is introduced in the Helmholtz equation (1.4), the PML governing equation is obtained:

$$-\frac{1}{\gamma_x}\frac{\partial}{\partial x}\left(\frac{1}{\gamma_x}\frac{\partial}{\partial x}p\right) - \frac{1}{\gamma_y}\frac{\partial}{\partial y}\left(\frac{1}{\gamma_y}\frac{\partial}{\partial y}p\right) - \frac{1}{\gamma_z}\frac{\partial}{\partial z}\left(\frac{1}{\gamma_z}\frac{\partial}{\partial z}p\right) - k^2\, p = 0.$$

In order to get the variational formulation, the whole equation is multiplied by $\gamma_x\gamma_y\gamma_z$, and taking into account that each γ_j coefficient is constant in each PML subdomain, it leads to

$$-\frac{\partial}{\partial x}\left(\frac{\gamma_y\gamma_z}{\gamma_x}\frac{\partial p}{\partial x}\right) - \frac{\partial}{\partial y}\left(\frac{\gamma_x\gamma_z}{\gamma_y}\frac{\partial p}{\partial y}\right) - \frac{\partial}{\partial z}\left(\frac{\gamma_x\gamma_y}{\gamma_z}\frac{\partial p}{\partial z}\right) - k^2\,\gamma_x\gamma_y\gamma_z\, p = 0,$$

or equivalently, using a tensor form writing,

$$\begin{cases} -\mathrm{div}\left[\begin{pmatrix} \frac{\gamma_y\gamma_z}{\gamma_x} & 0 & 0 \\ 0 & \frac{\gamma_x\gamma_z}{\gamma_y} & 0 \\ 0 & 0 & \frac{\gamma_x\gamma_y}{\gamma_z}, \end{pmatrix} \nabla p\right] - k^2\,\gamma_x\gamma_y\gamma_z\,p = 0 & \text{in } \Omega, \\ \frac{\partial p}{\partial \boldsymbol{n}} = g = g_{\mathrm{re}} + i\,g_{\mathrm{im}} & \text{on } \Gamma_{\mathrm{s}}, \\ p = 0 & \text{on } \Gamma_{\mathrm{out}}. \end{cases} \tag{2.14}$$

In what follows, $C = \begin{pmatrix} \frac{\gamma_y\gamma_z}{\gamma_x} & 0 & 0 \\ 0 & \frac{\gamma_x\gamma_z}{\gamma_y} & 0 \\ 0 & 0 & \frac{\gamma_x\gamma_y}{\gamma_z} \end{pmatrix}$ and $M = \gamma_x\gamma_y\gamma_z$, which are described in detail, writing explicitly how the coefficients depend on the PML subdomains.

$\triangleright\,\Omega_0$**:** In the fluid subdomain, all the coefficients are equal to one, i.e., $\gamma_x = \gamma_y = \gamma_z = 1$. So, if the matrix are denoted by C_0 and M_0, it was obtained:

$$C_0 = \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\mathrm{Re}(C_0)}, \tag{2.15}$$

$$M_0 = \underbrace{1}_{\mathrm{Re}(M_0)}. \tag{2.16}$$

$\triangleright\,\Omega_x$**:** In this PML layer, $\gamma_x = 1 - i\frac{\sigma_x}{\omega}$ while $\gamma_y = \gamma_z = 1$. Then, the coefficients C_x and M_x are given by

$$C_x = \begin{pmatrix} \frac{1}{\gamma_x} & 0 & 0 \\ 0 & \gamma_x & 0 \\ 0 & 0 & \gamma_x \end{pmatrix} = \underbrace{\begin{pmatrix} \frac{\omega^2}{\sigma_x^2+\omega^2} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\mathrm{Re}(C_x)} + i\,\underbrace{\begin{pmatrix} \frac{\omega\sigma_x}{\sigma_x^2+\omega^2} & 0 & 0 \\ 0 & -\frac{\sigma_x}{\omega} & 0 \\ 0 & 0 & -\frac{\sigma_x}{\omega} \end{pmatrix}}_{\mathrm{Im}(C_x)}, \tag{2.17}$$

$$M_x = \gamma_x = \underbrace{1}_{\mathrm{Re}(M_x)} + i\,\underbrace{\frac{-\sigma_x}{\omega}}_{\mathrm{Im}(M_x)}. \tag{2.18}$$

▷ Ω_y **and** Ω_z**:** Those cases are similar to the previous one. In the first one $\gamma_y = 1 - i\frac{\sigma_y}{\omega}$ and $\gamma_x = \gamma_z = 1$, while in the second one, $\gamma_z = 1 - i\frac{\sigma_z}{\omega}$ and $\gamma_x = \gamma_y = 1$. Then,

$$C_y = \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{\omega^2}{\sigma_y^2+\omega^2} & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\mathrm{Re}(C_y)} + i \underbrace{\begin{pmatrix} -\frac{\sigma_y}{\omega} & 0 & 0 \\ 0 & \frac{\omega\sigma_y}{\sigma_y^2+\omega^2} & 0 \\ 0 & 0 & -\frac{\sigma_y}{\omega} \end{pmatrix}}_{\mathrm{Im}(C_y)}, \tag{2.19}$$

$$M_y = \underbrace{1}_{\mathrm{Re}(M_y)} + i \underbrace{\frac{-\sigma_y}{\omega}}_{\mathrm{Im}(M_y)}. \tag{2.20}$$

In the case of the PML subdomain Ω_z, it holds

$$C_z = \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{\omega^2}{\sigma_z^2+\omega^2} \end{pmatrix}}_{\mathrm{Re}(C_z)} + i \underbrace{\begin{pmatrix} -\frac{\sigma_y}{\omega} & 0 & 0 \\ 0 & -\frac{\sigma_z}{\omega} & 0 \\ 0 & 0 & \frac{\omega\sigma_z}{\sigma_z^2+\omega^2} \end{pmatrix}}_{\mathrm{Im}(C_z)}, \tag{2.21}$$

$$M_z = \underbrace{1}_{\mathrm{Re}(M_z)} + i \underbrace{(\frac{-\sigma_z}{\omega})}_{\mathrm{Im}(M_z)}. \tag{2.22}$$

▷ Ω_{xy}**:** In this PML subdomain $\gamma_x = 1 - i\frac{\sigma_x}{\omega}$, $\gamma_y = 1 - i\frac{\sigma_y}{\omega}$ and $\gamma_z = 1$, then, the matrix C_{xy} and the coefficient M_{xy} are given by

$$C_{xy} = \begin{pmatrix} \frac{\gamma_y}{\gamma_x} & 0 & 0 \\ 0 & \frac{\gamma_x}{\gamma_y} & 0 \\ 0 & 0 & \gamma_x\gamma_y \end{pmatrix} = \underbrace{\begin{pmatrix} \frac{\omega^2+\sigma_x\sigma_y}{\sigma_x^2+\omega^2} & 0 & 0 \\ 0 & \frac{\omega^2+\sigma_x\sigma_y}{\sigma_y^2+\omega^2} & 0 \\ 0 & 0 & 1-\frac{\sigma_x\sigma_y}{\omega^2} \end{pmatrix}}_{\mathrm{Re}(C_{xy})} + \underbrace{i \begin{pmatrix} \frac{-\omega(\sigma_y-\sigma_x)}{\sigma_x^2+\omega^2} & 0 & 0 \\ 0 & \frac{-\omega(\sigma_x-\sigma_y)}{\sigma_y^2+\omega^2} & 0 \\ 0 & 0 & \frac{-(\sigma_x+\sigma_y)}{\omega} \end{pmatrix}}_{\mathrm{Im}(C_{xy})}, \quad (2.23)$$

$$M_{xy} = \gamma_x\gamma_y = \underbrace{1-\frac{\sigma_x\sigma_y}{\omega^2}}_{\mathrm{Re}(M_{xy})} + i\underbrace{\frac{-(\sigma_x+\sigma_y)}{\omega}}_{\mathrm{Im}(M_{xy})}. \quad (2.24)$$

$\triangleright\, \Omega_{xz}$ **and** Ω_{yz}**:** In the PML subdomains where the waves are absorbed in the xz and/or yz direction respectively, the PML coefficients are given by

$$C_{xz} = \underbrace{\begin{pmatrix} \frac{\omega^2+\sigma_x\sigma_z}{\sigma_x^2+\omega^2} & 0 & 0 \\ 0 & 1-\frac{\sigma_x\sigma_z}{\omega^2} & 0 \\ 0 & 0 & \frac{\omega^2+\sigma_x\sigma_z}{\sigma_z^2+\omega^2} \end{pmatrix}}_{\mathrm{Re}(C_{xz})} + i\underbrace{\begin{pmatrix} \frac{-\omega(\sigma_z-\sigma_x)}{\sigma_x^2+\omega^2} & 0 & 0 \\ 0 & \frac{-(\sigma_x+\sigma_z)}{\omega} & 0 \\ 0 & 0 & \frac{-\omega(\sigma_x-\sigma_z)}{\sigma_z^2+\omega^2} \end{pmatrix}}_{\mathrm{Im}(C_{xz})}, \quad (2.25)$$

$$M_{xz} = \underbrace{1-\frac{\sigma_x\sigma_z}{\omega^2}}_{\mathrm{Re}(M_{xz})} + i\underbrace{\frac{-(\sigma_x+\sigma_z)}{\omega}}_{\mathrm{Im}(M_{xz})}. \quad (2.26)$$

Analogously, in the PML subdomain Ω_{yz}, it is obtained

$$C_{yz} = \underbrace{\begin{pmatrix} 1-\frac{\sigma_y\sigma_z}{\omega^2} & 0 & 0 \\ 0 & \frac{\omega^2+\sigma_y\sigma_z}{\sigma_y^2+\omega^2} & 0 \\ 0 & 0 & \frac{\omega^2+\sigma_y\sigma_z}{\sigma_z^2+\omega^2} \end{pmatrix}}_{\mathrm{Re}(C_{yz})} + i \underbrace{\begin{pmatrix} \frac{-(\sigma_y+\sigma_z)}{\omega} & 0 & 0 \\ 0 & \frac{-\omega(\sigma_z-\sigma_y)}{\sigma_y^2+\omega^2} & 0 \\ 0 & 0 & \frac{-\omega(\sigma_y-\sigma_z)}{\sigma_z^2+\omega^2} \end{pmatrix}}_{\mathrm{Im}(C_{yz})}, \tag{2.27}$$

$$M_{yz} = \underbrace{1-\frac{\sigma_y\sigma_z}{\omega^2}}_{\mathrm{Re}(M_{yz})} + i \underbrace{\frac{-(\sigma_y+\sigma_z)}{\omega}}_{\mathrm{Im}(M_{yz})}. \tag{2.28}$$

$\triangleright\, \Omega_{xyz}$**:** Finally, in the PML corner subdomain, where the waves are absorbed in all directions, the matrix denoted by C_{xyz} and the coefficient M_{xyz} are given by

$$C_{xyz} = \begin{pmatrix} \frac{\gamma_y\gamma_z}{\gamma_x} & 0 & 0 \\ 0 & \frac{\gamma_x\gamma_z}{\gamma_y} & 0 \\ 0 & 0 & \frac{\gamma_x\gamma_y}{\gamma_z} \end{pmatrix} \tag{2.29}$$

$$= \underbrace{\begin{pmatrix} \frac{\sigma_x\sigma_y+\sigma_x\sigma_z-\sigma_y\sigma_z+\omega^2}{\sigma_x^2+\omega^2} & 0 & 0 \\ 0 & \frac{\sigma_x\sigma_y-\sigma_x\sigma_z+\sigma_y\sigma_z+\omega^2}{\sigma_y^2+\omega^2} & 0 \\ 0 & 0 & \frac{-\sigma_x\sigma_y+\sigma_x\sigma_z+\sigma_y\sigma_z+\omega^2}{\sigma_z^2+\omega^2} \end{pmatrix}}_{\mathrm{Re}(C_{xyz})}$$

$$+ i \underbrace{\begin{pmatrix} -\frac{\sigma_x\sigma_y\sigma_z-\sigma_x\omega^2+\sigma_y\omega^2+\sigma_z\omega^2}{\omega(\sigma_x^2+\omega^2)} & 0 & 0 \\ 0 & -\frac{\sigma_x\sigma_y\sigma_z+\sigma_x\omega^2-\sigma_y\omega^2+\sigma_z\omega^2}{\omega(\sigma_y^2+\omega^2)} & 0 \\ 0 & 0 & -\frac{\sigma_x\sigma_y\sigma_z+\sigma_x\omega^2+\sigma_y\omega^2-\sigma_z\omega^2}{\omega(\sigma_z^2+\omega^2)} \end{pmatrix}}_{\mathrm{Im}(C_{xyz})},$$

$$M_{xyz} = \gamma_x\gamma_y\gamma_z = \underbrace{\frac{-\sigma_x\sigma_y - \sigma_x\sigma_z - \sigma_y\sigma_z + \omega^2}{\omega^2}}_{\mathrm{Re}(M_{xyz})} + i \left(\underbrace{\frac{\sigma_x\sigma_y\sigma_z}{w^3} - \frac{\sigma_x + \sigma_y + \sigma_z}{\omega}}_{\mathrm{Im}(M_{xyz})} \right). \tag{2.30}$$

After introducing explicitly those coefficients, Equation (2.14) is split into the different subdomain of the PML, so the following PML model is obtained:

$$\begin{cases} -\mathrm{div}\,[C_0\nabla p] - k^2\, M_0\, p = 0 & \text{in } \Omega_0, \\ -\mathrm{div}\,[C_x\nabla p] - k^2\, M_x\, p = 0 & \text{in } \Omega_x, \\ -\mathrm{div}\,[C_y\nabla p] - k^2\, M_y\, p = 0 & \text{in } \Omega_y, \\ -\mathrm{div}\,[C_z\nabla p] - k^2\, M_z\, p = 0 & \text{in } \Omega_z, \\ -\mathrm{div}\,[C_{xy}\nabla p] - k^2\, M_{xy}\, p = 0 & \text{in } \Omega_{xy}, \\ -\mathrm{div}\,[C_{xz}\nabla p] - k^2\, M_{xz}\, p = 0 & \text{in } \Omega_{xz}, \\ -\mathrm{div}\,[C_{yz}\nabla p] - k^2\, M_{yz}\, p = 0 & \text{in } \Omega_{yz}, \\ -\mathrm{div}\,[C_{xyz}\nabla p] - k^2\, M_{xyz}\, p = 0 & \text{in } \Omega_{xyz}, \\ \frac{\partial p}{\partial \boldsymbol{n}} = g = g_{\mathrm{re}} + i\, g_{\mathrm{im}} & \text{on } \Gamma_{\mathrm{s}}, \\ p = 0 & \text{on } \Gamma_{\mathrm{out}}, \\ [p] = 0 & \text{on } \Gamma_{\mathrm{I}}, \\ [\left(\frac{1}{\gamma_x}\frac{\partial p}{\partial x}, \frac{1}{\gamma_y}\frac{\partial p}{\partial y}, \frac{1}{\gamma_z}\frac{\partial p}{\partial z}\right) \cdot \boldsymbol{n}] = 0 & \text{on } \Gamma_{\mathrm{I}}, \end{cases}$$

where Γ_{I} is the coupling boundary among different subdomains and brackets $[\ldots]$ denote the jump among quantities at each side of Γ_{I}.

As in the bounded problems, the solution, as well as the test functions, should be divided into their real and imaginary parts. Again, the test functions, $\varphi_{\mathrm{re}}, \varphi_{\mathrm{im}}$ belong to $\mathrm{H}^1(\Omega)$ with $\varphi_{\mathrm{re}|_{\Gamma_{\mathrm{out}}}} = 0$ and $\varphi_{\mathrm{im}|_{\Gamma_{\mathrm{out}}}} = 0$. Firstly, the variational formulation of the real and imaginary part of the problem is computed separately using the Green's Identity (see Eq. (2.3)).

Finally, taking in account the different computational subdomains, the variational formulation of the problem was

$$\begin{cases} \text{Find } p_{\text{re}}, p_{\text{im}} \in \mathrm{H}^1(\Omega) \text{ such that } p_{\text{re}|_{\Gamma_{\text{out}}}} = 0,\ p_{\text{im}|_{\Gamma_{\text{out}}}} = 0 \text{ and} \\ \mathcal{A}\left((p_{\text{re}},\, p_{\text{im}}), (\varphi_{\text{re}},\, \varphi_{\text{im}})\right) = \mathcal{L}\left(\varphi_{\text{re}},\, \varphi_{\text{im}}\right) \\ \forall \varphi_{\text{re}}, \varphi_{\text{im}} \in \mathrm{H}^1(\Omega), \text{ with } \varphi_{\text{re}|_{\Gamma_{\text{out}}}} = 0 \text{ and } \varphi_{\text{im}|_{\Gamma_{\text{out}}}} = 0, \end{cases} \tag{2.31}$$

being $g_{\text{re}}, g_{\text{im}} \in \mathrm{L}^2(\Gamma_{\text{s}})$, $k = \omega/c > 0$, C_j, M_j with $j \in \{x, y, z, xy, xz, yz, xyz\}$ given by Eq. (2.15)–(2.30) and $\mathcal{A} = \mathcal{A}_1 + \mathcal{A}_2$, where $\mathcal{A}_1$, $\mathcal{A}_2$ and $\mathcal{L}$ are given by

$$\begin{aligned}
&\mathcal{A}_1\left((p_{\text{re}},\, p_{\text{im}}), (\varphi_{\text{re}},\, \varphi_{\text{im}})\right) = \int_{\Omega_0} \nabla p_{\text{re}} \cdot \nabla \varphi_{\text{re}}\, \mathrm{d}V + \int_{\Omega_0} \nabla p_{\text{im}} \cdot \nabla \varphi_{\text{im}}\, \mathrm{d}V \\
&+ \int_{\Omega_x} \left(C_{x_{\text{re}}} \nabla p_{\text{re}} - C_{x_{\text{im}}} \nabla p_{\text{im}}\right) \cdot \nabla \varphi_{\text{re}}\, \mathrm{d}V + \int_{\Omega_x} \left(C_{x_{\text{re}}} \nabla p_{\text{im}} + C_{x_{\text{im}}} \nabla p_{\text{re}}\right) \cdot \nabla \varphi_{\text{im}}\, \mathrm{d}V \\
&+ \int_{\Omega_y} \left(C_{y_{\text{re}}} \nabla p_{\text{re}} - C_{y_{\text{im}}} \nabla p_{\text{im}}\right) \cdot \nabla \varphi_{\text{re}}\, \mathrm{d}V + \int_{\Omega_y} \left(C_{y_{\text{re}}} \nabla p_{\text{im}} + C_{y_{\text{im}}} \nabla p_{\text{re}}\right) \cdot \nabla \varphi_{\text{im}}\, \mathrm{d}V \\
&+ \int_{\Omega_z} \left(C_{z_{\text{re}}} \nabla p_{\text{re}} - C_{z_{\text{im}}} \nabla p_{\text{im}}\right) \cdot \nabla \varphi_{\text{re}}\, \mathrm{d}V + \int_{\Omega_z} \left(C_{z_{\text{re}}} \nabla p_{\text{im}} + C_{z_{\text{im}}} \nabla p_{\text{re}}\right) \cdot \nabla \varphi_{\text{im}}\, \mathrm{d}V \\
&+ \int_{\Omega_{xy}} \left(C_{xy_{\text{re}}} \nabla p_{\text{re}} - C_{xy_{\text{im}}} \nabla p_{\text{im}}\right) \cdot \nabla \varphi_{\text{re}}\, \mathrm{d}V + \int_{\Omega_{xy}} \left(C_{xy_{\text{re}}} \nabla p_{\text{im}} + C_{xy_{\text{im}}} \nabla p_{\text{re}}\right) \cdot \nabla \varphi_{\text{im}}\, \mathrm{d}V \\
&+ \int_{\Omega_{xz}} \left(C_{xz_{\text{re}}} \nabla p_{\text{re}} - C_{xz_{\text{im}}} \nabla p_{\text{im}}\right) \cdot \nabla \varphi_{\text{re}}\, \mathrm{d}V + \int_{\Omega_{xz}} \left(C_{xz_{\text{re}}} \nabla p_{\text{im}} + C_{xz_{\text{im}}} \nabla p_{\text{re}}\right) \cdot \nabla \varphi_{\text{im}}\, \mathrm{d}V \\
&+ \int_{\Omega_{yz}} \left(C_{yz_{\text{re}}} \nabla p_{\text{re}} - C_{yz_{\text{im}}} \nabla p_{\text{im}}\right) \cdot \nabla \varphi_{\text{re}}\, \mathrm{d}V + \int_{\Omega_{yz}} \left(C_{yz_{\text{re}}} \nabla p_{\text{im}} + C_{yz_{\text{im}}} \nabla p_{\text{re}}\right) \cdot \nabla \varphi_{\text{im}}\, \mathrm{d}V \\
&+ \int_{\Omega_{xyz}} \left(C_{xyz_{\text{re}}} \nabla p_{\text{re}} - C_{xyz_{\text{im}}} \nabla p_{\text{im}}\right) \cdot \nabla \varphi_{\text{re}}\, \mathrm{d}V + \int_{\Omega_{xyz}} \left(C_{xyz_{\text{re}}} \nabla p_{\text{im}} + C_{xyz_{\text{im}}} \nabla p_{\text{re}}\right) \cdot \nabla \varphi_{\text{im}}\, \mathrm{d}V
\end{aligned} \tag{2.32}$$

$$
\begin{aligned}
&\mathcal{A}_2\left((p_{\mathrm{re}},\, p_{\mathrm{im}}),(\varphi_{\mathrm{re}},\, \varphi_{\mathrm{im}})\right) = \\
&\qquad -\int_{\Omega_0}(k_{\mathrm{re}}^2 p_{\mathrm{re}} - k_{\mathrm{im}}^2 p_{\mathrm{im}})\varphi_{\mathrm{re}}\,\mathrm{d}V - \int_{\Omega_0}\left(k_{\mathrm{im}}^2 p_{\mathrm{re}} + k_{\mathrm{re}}^2 p_{\mathrm{im}}\right)\varphi_{\mathrm{im}}\,\mathrm{d}V \\
&\quad -\int_{\Omega_x}\left((k_{\mathrm{re}}^2 M_{x_{\mathrm{re}}} - k_{\mathrm{im}}^2 M_{x_{\mathrm{im}}})p_{\mathrm{re}} - (k_{\mathrm{im}}^2 M_{x_{\mathrm{re}}} + k_{\mathrm{re}}^2 M_{x_{\mathrm{im}}})p_{\mathrm{im}}\right)\varphi_{\mathrm{re}}\,\mathrm{d}V \\
&\quad -\int_{\Omega_x}\left((k_{\mathrm{im}}^2 M_{x_{\mathrm{re}}} + k_{\mathrm{re}}^2 M_{x_{\mathrm{im}}})p_{\mathrm{re}} + (k_{\mathrm{re}}^2 M_{x_{\mathrm{re}}} - k_{\mathrm{im}}^2 M_{x_{\mathrm{im}}})p_{\mathrm{im}}\right)\varphi_{\mathrm{im}}\,\mathrm{d}V \\
&\quad -\int_{\Omega_y}\left((k_{\mathrm{re}}^2 M_{y_{\mathrm{re}}} - k_{\mathrm{im}}^2 M_{y_{\mathrm{im}}})p_{\mathrm{re}} - (k_{\mathrm{im}}^2 M_{y_{\mathrm{re}}} + k_{\mathrm{re}}^2 M_{y_{\mathrm{im}}})p_{\mathrm{im}}\right)\varphi_{\mathrm{re}}\,\mathrm{d}V \\
&\quad -\int_{\Omega_y}\left((k_{\mathrm{im}}^2 M_{y_{\mathrm{re}}} + k_{\mathrm{re}}^2 M_{y_{\mathrm{im}}})p_{\mathrm{re}} + (k_{\mathrm{re}}^2 M_{y_{\mathrm{re}}} - k_{\mathrm{im}}^2 M_{y_{\mathrm{im}}})p_{\mathrm{im}}\right)\varphi_{\mathrm{im}}\,\mathrm{d}V \\
&\quad -\int_{\Omega_z}\left((k_{\mathrm{re}}^2 M_{z_{\mathrm{re}}} - k_{\mathrm{im}}^2 M_{z_{\mathrm{im}}})p_{\mathrm{re}} - (k_{\mathrm{im}}^2 M_{z_{\mathrm{re}}} + k_{\mathrm{re}}^2 M_{z_{\mathrm{im}}})p_{\mathrm{im}}\right)\varphi_{\mathrm{re}}\,\mathrm{d}V \\
&\quad -\int_{\Omega_z}\left((k_{\mathrm{im}}^2 M_{z_{\mathrm{re}}} + k_{\mathrm{re}}^2 M_{z_{\mathrm{im}}})p_{\mathrm{re}} + (k_{\mathrm{re}}^2 M_{z_{\mathrm{re}}} - k_{\mathrm{im}}^2 M_{z_{\mathrm{im}}})p_{\mathrm{im}}\right)\varphi_{\mathrm{im}}\,\mathrm{d}V \\
&\quad -\int_{\Omega_{xy}}\left((k_{\mathrm{re}}^2 M_{xy_{\mathrm{re}}} - k_{\mathrm{im}}^2 M_{xy_{\mathrm{im}}})p_{\mathrm{re}} - (k_{\mathrm{im}}^2 M_{xy_{\mathrm{re}}} + k_{\mathrm{re}}^2 M_{xy_{\mathrm{im}}})p_{\mathrm{im}}\right)\varphi_{\mathrm{re}}\,\mathrm{d}V \\
&\quad -\int_{\Omega_{xy}}\left((k_{\mathrm{im}}^2 M_{xy_{\mathrm{re}}} + k_{\mathrm{re}}^2 M_{xy_{\mathrm{im}}})p_{\mathrm{re}} + (k_{\mathrm{re}}^2 M_{xy_{\mathrm{re}}} - k_{\mathrm{im}}^2 M_{xy_{\mathrm{im}}})p_{\mathrm{im}}\right)\varphi_{\mathrm{im}}\,\mathrm{d}V \\
&\quad -\int_{\Omega_{xz}}\left((k_{\mathrm{re}}^2 M_{xz_{\mathrm{re}}} - k_{\mathrm{im}}^2 M_{xz_{\mathrm{im}}})p_{\mathrm{re}} - (k_{\mathrm{im}}^2 M_{xz_{\mathrm{re}}} + k_{\mathrm{re}}^2 M_{xz_{\mathrm{im}}})p_{\mathrm{im}}\right)\varphi_{\mathrm{re}}\,\mathrm{d}V \\
&\quad -\int_{\Omega_{xz}}\left((k_{\mathrm{im}}^2 M_{xz_{\mathrm{re}}} + k_{\mathrm{re}}^2 M_{xz_{\mathrm{im}}})p_{\mathrm{re}} + (k_{\mathrm{re}}^2 M_{xz_{\mathrm{re}}} - k_{\mathrm{im}}^2 M_{xz_{\mathrm{im}}})p_{\mathrm{im}}\right)\varphi_{\mathrm{im}}\,\mathrm{d}V \\
&\quad -\int_{\Omega_{yz}}\left((k_{\mathrm{re}}^2 M_{yz_{\mathrm{re}}} - k_{\mathrm{im}}^2 M_{yz_{\mathrm{im}}})p_{\mathrm{re}} - (k_{\mathrm{im}}^2 M_{yz_{\mathrm{re}}} + k_{\mathrm{re}}^2 M_{yz_{\mathrm{im}}})p_{\mathrm{im}}\right)\varphi_{\mathrm{re}}\,\mathrm{d}V \\
&\quad -\int_{\Omega_{yz}}\left((k_{\mathrm{im}}^2 M_{yz_{\mathrm{re}}} + k_{\mathrm{re}}^2 M_{yz_{\mathrm{im}}})p_{\mathrm{re}} + (k_{\mathrm{re}}^2 M_{yz_{\mathrm{re}}} - k_{\mathrm{im}}^2 M_{yz_{\mathrm{im}}})p_{\mathrm{im}}\right)\varphi_{\mathrm{im}}\,\mathrm{d}V \\
&\quad -\int_{\Omega_{xyz}}\left((k_{\mathrm{re}}^2 M_{xyz_{\mathrm{re}}} - k_{\mathrm{im}}^2 M_{xyz_{\mathrm{im}}})p_{\mathrm{re}} - (k_{\mathrm{im}}^2 M_{xyz_{\mathrm{re}}} + k_{\mathrm{re}}^2 M_{xyz_{\mathrm{im}}})p_{\mathrm{im}}\right)\varphi_{\mathrm{re}}\,\mathrm{d}V \\
&\quad -\int_{\Omega_{xyz}}\left((k_{\mathrm{im}}^2 M_{xyz_{\mathrm{re}}} + k_{\mathrm{re}}^2 M_{xyz_{\mathrm{im}}})p_{\mathrm{re}} + (k_{\mathrm{re}}^2 M_{xyz_{\mathrm{re}}} - k_{\mathrm{im}}^2 M_{xyz_{\mathrm{im}}})p_{\mathrm{im}}\right)\varphi_{\mathrm{im}}\,\mathrm{d}V
\end{aligned}
\tag{2.33}
$$

$$\mathcal{L}\left(\varphi_{\mathrm{re}},\, \varphi_{\mathrm{im}}\right) = \int_{\Gamma_{\mathrm{s}}} g_{\mathrm{re}}\, \varphi_{\mathrm{re}}\, \mathrm{d}\gamma + \int_{\Gamma_{\mathrm{s}}} g_{\mathrm{im}}\, \varphi_{\mathrm{im}} \mathrm{d}\gamma. \tag{2.34}$$

Notice that the wave number can be split in its real and imaginary parts $k = k_{\mathrm{re}} + i\, k_{\mathrm{im}}$. Since model (2.14) does not include the viscosity term, it holds:

$$k_{\mathrm{re}} = k, \quad k_{\mathrm{im}} = 0$$

Finally, it should be remarked that this master thesis does not include a proof of the existence and uniqueness of this variational problem.

Chapter 3

Finite Element Method

In this chapter the Finite Element Method is described, which is the numerical method used to approximate the solution of the models introduced in Chapter 2. First, the problem stated in a bounded domain, in both cases, with and without viscosity, is considered. Then, the solution of the free field model with the PML technique is approximated.

The most difficult task when solving numerically a variational problem (as those ones described in Chapter 2) is to approximate accurately an infinite functional space by means of a finite-dimensional discrete space. The Finite Element Method consists in approaching this functional Sobolev space by another one of finite dimension. Then, using this discrete space, an approximate problem is introduced and written as a linear matrix problem. Finally, once it is solved, an approximate solution of the original problem is obtained.

3.1 Acoustic problem stated in bounded domains

This section describes the Finite Element Method for acoustic problems stated in bounded domains. Both models, with and without viscosity (see respectively (2.2) and (2.6)) are approximated by the Finite Element Method in the following subsections.

Once the weak formulations of both problems are obtained (see (2.4) and

(2.7)), it is necessary to find a finite dimensional space to approach $\mathrm{H}^1(\Omega)$. Due to the computational domain is three-dimensional, Ω is split in a set of tetrahedra $\mathcal{T}_h(\Omega)$. It would be possible to divide it also in another geometric forms (such as hexaedra, prisms, ...), but, the finite element library, which is used in the computer implementation only admits this kind of partition (see [6] for further details). The set of all the tetrahedra and its connectivity data forms a mesh.

It will be assumed that $\bigcup_{T\in\mathcal{T}_h} T = \Omega$, and, in addition, there are some restrictions, which must be satisfied:

1. The interior of two different tetrahedra has to be disjoint.
2. The vertex of a tetrahedron cannot be placed on the edges or faces of another tetrahedra. In other words, the tetrahedra have to be adjacent.

Once the partition is given, the following step is to choose the type of elements of the discretization space. In the present work, Lagrangian elements that are continuous and piecewise linear have been chosen. This kind of elements are named: Lagrange $\mathbb{P}^1$ elements. To define the discrete basis associated to the Lagrange $\mathbb{P}^1$ discretization, it is necessary to enumerate all the vertex of the partition $\{v_j\}_{j=1}^n$. Then, a continuous and piecewise linear function ψ_j is associated to each vertex v_j. The value of this function is 1 in this vertex and 0 in the rest, i.e. $\psi_l(v_j) = \delta|_{lj}$, where $\delta_{|lj}$ is the Kronecker's delta. Those are the base functions: $\psi_l,\ l = 1, 2, \ldots n$, which form the discrete basis of the finite element space. Hence, the approaching space of $\mathrm{H}^1(\Omega)$ is built as follows:

$$\begin{aligned} \mathrm{V}_h(\Omega) = &\ \{\psi \in \mathrm{H}^1(\Omega) : \psi_{|_T} \circ F_T \in \mathbb{P}^1(\hat{T}), \forall\, T \in \mathcal{T}_h(\Omega)\} \\ = &\ \langle \psi_1, \psi_2, \ldots, \psi_n \rangle \subset \mathrm{H}^1(\Omega), \end{aligned} \tag{3.1}$$

where F_T is the affine function: $F_T : \hat{T} \to T$, which associates each tetrahedron $T \in \mathcal{T}_h(\Omega)$ with the reference tetrahedron $\hat{T}$, defined by the vertices: $(0,0,0), (1,0,0), (0,1,0), (0,0,1)$. Moreover, any $\psi \in \mathrm{V}_h(\Omega)$, is continuous in the whole domain Ω and, in each tetrahedron T, is a first order polynomial.

Consequently, the discrete variational problem is described as follows:

$$\begin{cases} \text{Given functions } g_{\text{re}}, g_{\text{im}} \in \mathrm{L}^2(\Gamma_{\text{s}}),\ f_{\text{re}}, f_{\text{im}} \in \mathrm{H}^{\frac{1}{2}}(\Gamma_{\text{out}}),\ \text{and } \omega, \rho, c > 0, \\ \text{find } p^h_{\text{re}}, p^h_{\text{im}} \in \mathrm{V}_h(\Omega) \text{ such that } p^h_{\text{re}|_{\Gamma_{\text{out}}}} = \mathcal{I}_h f_{\text{re}},\ p^h_{\text{im}|_{\Gamma_{\text{out}}}} = \mathcal{I}_h f_{\text{im}} \text{ and} \\ \int_\Omega -\omega^2 \rho\, p^h_{\text{re}}\, \varphi^h_{\text{re}}\, \mathrm{d}V + \int_\Omega \rho c^2\, \nabla p^h_{\text{re}} \cdot \nabla \varphi^h_{\text{re}}\, \mathrm{d}V + \int_\Omega -\omega^2 \rho\, p^h_{\text{im}}\, \varphi^h_{\text{im}}\, \mathrm{d}V + \\ \int_\Omega \rho c^2\, \nabla p^h_{\text{im}} \cdot \nabla \varphi^h_{\text{im}}\, \mathrm{d}V = \int_{\Gamma_{\text{s}}} \rho c^2\, g_{\text{re}}\, \varphi^h_{\text{re}}\, \mathrm{d}\gamma + \int_{\Gamma_{\text{s}}} \rho c^2\, g_{\text{im}}\, \varphi^h_{\text{im}}\, \mathrm{d}\gamma, \\ \forall\, \varphi^h_{\text{re}}, \varphi^h_{\text{im}} \in \mathrm{V}_h(\Omega) \text{ with } \varphi^h_{\text{re}|_{\Gamma_{\text{out}}}} = 0,\ \varphi^h_{\text{im}|_{\Gamma_{\text{out}}}} = 0, \end{cases} \tag{3.2}$$

where $\mathcal{I}_h$ is the pointwise interpolation function of the discrete space $\mathrm{V}_h(\Omega)$.

The Dirichlet boundary conditions, $p^h_{\text{re}|_{\Gamma_{\text{out}}}} = \mathcal{I}_h f_{\text{re}}$, $p^h_{\text{im}|_{\Gamma_{\text{out}}}} = \mathcal{I}_h f_{\text{im}}$ will not be taken into account at this point of the arguments. However, it will be imposed when the linear matrix system is computed, blocking the degrees of freedom associated to the vertices on the Dirichlet boundary, this is, imposing that $p^h_{\text{re}}(v_j) = f_{\text{re}}(v_j)$ and $p^h_{\text{im}}(v_j) = f_{\text{im}}(v_j)$ for all those vertices $v_j \in \Gamma_{\text{out}}$. Now, since

$$p^h_{\text{re}} \in \mathrm{V}_h(\Omega) \Rightarrow p^h_{\text{re}}(x,y,z) = \sum_{j=1}^{n} \mu_j \psi_j(x,y,z),$$

$$p^h_{\text{im}} \in \mathrm{V}_h(\Omega) \Rightarrow p^h_{\text{im}}(x,y,z) = \sum_{j=1}^{n} \lambda_j \psi_j(x,y,z),$$

and

$$(\varphi^h_{\text{re}}, \varphi^h_{\text{im}}) \in \mathrm{V}_h(\Omega) \times \mathrm{V}_h(\Omega) \Rightarrow (\varphi^h_{\text{re}}, \varphi^h_{\text{im}}) \in \langle (\psi_1, 0), \cdots, (\psi_n, 0), (0, \psi_1), \cdots, (0, \psi_n) \rangle,$$

then, the discrete problem (3.2) can be rewritten as follows:

$$\begin{cases} \displaystyle\sum_{j=1}^{n} \left[\left(\int_\Omega c^2 \rho\, \nabla \psi_j \cdot \nabla \psi_l\, \mathrm{d}V + \int_\Omega (-\omega^2 \rho)\, \psi_j \psi_l\, \mathrm{d}V \right) \mu_j \right] = \\ \qquad = \int_{\Gamma_{\text{s}}} \rho c^2\, g_{\text{re}}\, \psi_l\, \mathrm{d}\gamma \text{ with } l = 1, 2, \ldots, n, \\ \displaystyle\sum_{j=1}^{n} \left[\left(\int_\Omega c^2 \rho\, \nabla \psi_j \cdot \nabla \psi_l\, \mathrm{d}V + \int_\Omega (-\omega^2 \rho)\, \psi_j \psi_l\, \mathrm{d}V \right) \lambda_j \right] = \\ \qquad = \int_{\Gamma_{\text{s}}} \rho c^2\, g_{\text{im}}\, \psi_l\, \mathrm{d}\gamma, \text{ with } l = 1, 2, \ldots, n. \end{cases}$$

This problem can be written in matrix form as follows:

$$\left(\begin{array}{c|c} A & 0 \\ \hline 0 & A \end{array}\right) \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_n \\ \lambda_1 \\ \vdots \\ \lambda_n \end{pmatrix} = \left(\begin{array}{c} b_1 \\ \hline b_2 \end{array}\right),$$

being the matrix $A \in \mathcal{M}_{n\times n}(\mathbb{R})$ and vectors $b_1, b_2 \in \mathbb{R}^n$ given by

$$\begin{aligned} [A]_{lj} &= c^2\rho \int_\Omega \nabla\psi_j \cdot \nabla\psi_l \,\mathrm{d}V - \omega^2\rho \int_\Omega \psi_j\psi_l \,\mathrm{d}V, \\ [b_1]_l &= \rho\, c^2 \int_{\Gamma_\mathrm{s}} g_\mathrm{re}\, \varphi_l \,\mathrm{d}\gamma, \\ [b_2]_l &= \rho\, c^2 \int_{\Gamma_\mathrm{s}} g_\mathrm{im}\, \varphi_l \,\mathrm{d}\gamma, \end{aligned}$$

for $1 \leq l, j \leq n$.

One of the advantages of the finite element discretization consists in most of the matrix elements are null, this is, the matrix A of the linear system is sparse. In fact, $A_{lj} \neq 0$ if and only if the function supports of ψ_l and ψ_j are not disjoint, in other words, if and only if the vertex l and j belong to the same tetrahedron.

In order to impose the Dirichlet boundary conditions, $p_{h\mathrm{re}|\Gamma_\mathrm{out}} = \mathcal{I}_h f_\mathrm{re}$, $p_{h_\mathrm{im}|\Gamma_\mathrm{out}} = \mathcal{I}_h f_\mathrm{im}$, the degrees of freedom μ_j and λ_j which are associated to vertices v_j on the Dirichlet boundary of Ω are selected. Then, their values are replaced by the value of the Dirichlet conditions, this is, $\mu_j = f_\mathrm{re}(v_j)$ and $\lambda_j = f_\mathrm{im}(v_j)$ for all those vertices $v_j \in \Gamma_\mathrm{out}$

The integrals of the matrix components coefficients of matrix A, are computed as the sum of the integrals in the partition elements:

$$[A]_{lj} = c^2\rho \int_\Omega \nabla\psi_j \cdot \nabla\psi_l \,\mathrm{d}V - \omega^2\rho \int_\Omega \psi_j\psi_l \,\mathrm{d}V = \sum_{T\in\mathcal{T}_h(\Omega)} \left(c^2\rho \int_T \nabla\psi_j \cdot \nabla\psi_l \,\mathrm{d}V - \omega^2\rho \int_T \psi_j\psi_l \,\mathrm{d}V \right).$$

To perform these computations, all elements of the mesh are considered and for each tetrahedron, the integrals associated to the multiplication of the elemental functions ψ_j associated to the vertices at each tetrahedron, are computed. This process is known as assembly. Finally, the linear matrix system is solved with direct method.

After computing the weak formulation (see Eq. (2.7)) for the problem (2.5), the finite element discretization of the acoustic model with viscosity is analogous. More precisely, it is described as follows:

$$\begin{cases} \text{Given functions } g_{\mathrm{re}}, g_{\mathrm{im}} \in \mathrm{L}^2(\Gamma_{\mathrm{s}}),\ f_{\mathrm{re}}, f_{\mathrm{im}} \in \mathrm{H}^{\frac{1}{2}}(\Gamma_{\mathrm{out}}),\ \text{and } \omega, \rho, c, \nu > 0, \\ \text{find } p^h_{\mathrm{re}}, p^h_{\mathrm{im}} \in \mathrm{V}_h(\Omega) \text{ such that } p^h_{\mathrm{re}|_{\Gamma_{\mathrm{out}}}} = \mathcal{I}_h f_{\mathrm{re}},\ p^h_{\mathrm{im}|_{\Gamma_{\mathrm{out}}}} = \mathcal{I}_h f_{\mathrm{im}}, \text{ and} \\ \int_\Omega (-\omega^2\rho)\, p^h_{\mathrm{re}}\, \varphi^h_{\mathrm{re}} \,\mathrm{d}V + \int_\Omega \rho c^2\, \nabla p^h_{\mathrm{re}} \cdot \nabla\varphi^h_{\mathrm{re}} \,\mathrm{d}V + \int_\Omega (-\omega\nu)\, \nabla p^h_{\mathrm{im}} \cdot \nabla\varphi^h_{\mathrm{re}} \,\mathrm{d}V + \\ \int_\Omega (-\omega^2\rho)\, p^h_{\mathrm{im}}\, \varphi^h_{\mathrm{im}} \,\mathrm{d}V + \int_\Omega \rho c^2\, \nabla p^h_{\mathrm{im}} \cdot \nabla\varphi^h_{\mathrm{im}} \,\mathrm{d}V + \int_\Omega \omega\nu \nabla p^h_{\mathrm{re}} \cdot \nabla\varphi^h_{\mathrm{im}} \,\mathrm{d}V = \\ \int_{\Gamma_{\mathrm{s}}} \rho c^2 g_{\mathrm{re}}\, \varphi^h_{\mathrm{re}} \,\mathrm{d}\gamma + \int_{\Gamma_{\mathrm{s}}} \rho c^2 g_{\mathrm{im}}\, \varphi^h_{\mathrm{im}} \,\mathrm{d}\gamma + \int_{\Gamma_{\mathrm{s}}} (-\omega\nu) g_{\mathrm{im}}\, \varphi^h_{\mathrm{re}} \,\mathrm{d}\gamma + \int_{\Gamma_{\mathrm{s}}} \omega\nu g_{\mathrm{re}}\, \varphi^h_{\mathrm{im}} \,\mathrm{d}\gamma, \\ \forall\, \varphi^h_{\mathrm{re}}, \varphi^h_{\mathrm{re}} \in \mathrm{V}_h(\Omega) \text{ with } \varphi^h_{\mathrm{re}|_{\Gamma_{\mathrm{out}}}} = 0,\ \varphi^h_{\mathrm{im}|_{\Gamma_{\mathrm{out}}}} = 0. \end{cases} \tag{3.3}$$

Using again the same discrete basis for the trial and test functions, the discrete variational formulation leads to

$$\begin{cases} \displaystyle\sum_{j=1}^{n}\sum_{m=1}^{n} \left[\left(\int_\Omega (-\omega^2\rho)\, \psi_j\, \psi_l \,\mathrm{d}V + \int_\Omega \rho c^2\, \nabla\psi_j \cdot \nabla\psi_l \,\mathrm{d}V \right) \mu_j + \left(\int_\Omega (-\omega\nu)\, \nabla\psi_m \cdot \nabla\psi_l \,\mathrm{d}V \right) \lambda_m \right] \\ \qquad = \int_{\Gamma_{\mathrm{s}}} \rho c^2 g_{\mathrm{re}}\, \psi_l \,\mathrm{d}\gamma + \int_{\Gamma_{\mathrm{s}}} (-\omega\nu) g_{\mathrm{im}}\, \psi_l \,\mathrm{d}\gamma, \qquad l = 1, 2, \ldots, n, \\ \displaystyle\sum_{j=1}^{n}\sum_{m=1}^{n} \left[\left(\int_\Omega \omega\nu \nabla\psi_j \cdot \nabla\psi_l \,\mathrm{d}V \right) \mu_j + \left(\int_\Omega (-\omega^2\rho)\, \psi_m\, \psi_l \,\mathrm{d}V + \int_\Omega \rho c^2\, \nabla\psi_m \cdot \nabla\psi_l \,\mathrm{d}V \right) \lambda_j \right] \\ \qquad = \int_{\Gamma_{\mathrm{s}}} \rho c^2 g_{\mathrm{im}}\, \psi_l \,\mathrm{d}\gamma + \int_{\Gamma_{\mathrm{s}}} \omega\nu g_{\mathrm{re}}\, \psi_l \,\mathrm{d}\gamma, \qquad l = 1, 2, \ldots, n. \end{cases}$$

This problem can be rewritten in matrix form as follows:

$$\left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array}\right] \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_n \\ \lambda_1 \\ \vdots \\ \lambda_n \end{pmatrix} = \left(\begin{array}{c} b_1 \\ \hline b_2 \end{array}\right),$$

where matrices $A_{11}, A_{12}, A_{21}, A_{22} \in \mathcal{M}_{n\times n}(\mathbb{R})$ and vectors $b_1, b_2 \in \mathbb{R}^n$ are given by

$$\begin{aligned}
[A_{11}]_{lj} &= \int_\Omega (-\omega^2\rho)\,\psi_j\,\psi_l\,\mathrm{d}V + \int_\Omega \rho c^2\,\nabla\psi_j\cdot\nabla\psi_l\,\mathrm{d}V,\\
[A_{12}]_{lj} &= \int_\Omega (-\omega\nu)\,\nabla\psi_j\cdot\nabla\psi_l\,\mathrm{d}V,\\
[A_{21}]_{lj} &= \int_\Omega \omega\nu\nabla\psi_j\cdot\nabla\psi_l\,\mathrm{d}V,\\
[A_{22}]_{lj} &= \int_\Omega (-\omega^2\rho)\,\psi_j\,\psi_l\,\mathrm{d}V + \int_\Omega \rho c^2\,\nabla\psi_j\cdot\nabla\psi_l\,\mathrm{d}V,\\
[b_1]_l &= \int_{\Gamma_\mathrm{s}} \rho c^2 g_\mathrm{re}\,\psi_l\,\mathrm{d}\gamma + \int_{\Gamma_\mathrm{s}} (-\omega\nu) g_\mathrm{im}\,\psi_l\,\mathrm{d}\gamma,\\
[b_2]_l &= \int_{\Gamma_\mathrm{s}} \rho c^2 g_\mathrm{im}\,\psi_l\,\mathrm{d}\gamma + \int_{\Gamma_\mathrm{s}} \omega\nu g_\mathrm{re}\,\psi_l\,\mathrm{d}\gamma,
\end{aligned}$$

for $1 \le j, l \le n$.

As it has been discussed previously, $[A_{\alpha\beta}]_{lj} \neq 0$ if and only if the supports of ψ_l and ψ_j are not disjoint; i.e., if and only if the vertex l and j belong to the same tetrahedron.

3.2 Acoustic problem stated in unbounded domains

This section describes the Finite Element Method for acoustic problems stated in unbounded domains for which the PML technique is used (see

(2.14)). After computing the weak formulation (Eq. (2.7)), a finite element method is used to approximate the solution of the variational problem. The finite element discretization is identical to that one described for the acoustic model stated in bounded domains.

Consequently, the variational discretized problem is:

$$\begin{cases} \text{Find } p^h_{\text{re}}, p^h_{\text{im}} \in \mathrm{V}_h(\Omega) \text{ such that } p^h_{\text{re}|\Gamma_{\text{out}}} = 0,\ p^h_{\text{im}|\Gamma_{\text{out}}} = 0, \text{ and} \\ \mathcal{A}\left(\left(p^h_{\text{re}},\, p^h_{\text{im}}\right), \left(\varphi^h_{\text{re}},\, \varphi^h_{\text{im}}\right)\right) = \mathcal{L}\left(\varphi^h_{\text{re}},\, \varphi^h_{\text{im}}\right) \\ \forall\, \varphi^h_{\text{re}},\, \varphi^h_{\text{im}} \in \mathrm{V}_h(\Omega) \text{ with } \varphi^h_{\text{re}|\Gamma_{\text{out}}} = 0,\ \varphi^h_{\text{im}|\Gamma_{\text{out}}} = 0. \end{cases} \tag{3.4}$$

where $\mathcal{A}$ and $\mathcal{L}$ are given by (2.32) and (2.34), respectively.

Using again the same discrete basis for the trial and test functions, the discrete variational formulation leads to

$$\sum_{j=1}^{n}\sum_{m=1}^{n}\left(\mathcal{A}\left((\psi_j,\,0),(\psi_l,\,0)\right)\mu_j + \mathcal{A}\left((0\,\psi_m),(\psi_l,\,0)\right)\lambda_m\right) = \mathcal{L}\left(\psi_l,\,0\right),$$
$$\sum_{j=1}^{n}\sum_{m=1}^{n}\left(\mathcal{A}\left((\psi_j,\,0),(0,\,\psi_l)\right)\mu_j + \mathcal{A}\left((0,\,\psi_m),(0,\,\psi_l)\right)\lambda_m\right) = \mathcal{L}\left(0,\,\psi_l\right),$$
$$\text{for } l = 1, 2, \ldots, n.$$

This problem can be rewritten in a matrix form as follows:

$$\left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array}\right] \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_n \\ \lambda_1 \\ \vdots \\ \lambda_n \end{pmatrix} = \left(\begin{array}{c} b_1 \\ \hline b_2 \end{array}\right),$$

where matrices $A_{11}, A_{12}, A_{21}, A_{22} \in \mathcal{M}_{n\times n}(\mathbb{R})$ and vectors $b_1, b_2 \in \mathbb{R}^n$ are

given by

$$\begin{aligned}
[A_{11}]_{lj} &= \mathcal{A}\left((\psi_j,\, 0)\,,(\psi_l,\, 0)\right), \\
[A_{12}]_{lj} &= \mathcal{A}\left((0,\, \psi_j)\,,(\psi_l,\, 0)\right), \\
[A_{21}]_{lj} &= \mathcal{A}\left((\psi_j,\, 0)\,,(0,\, \psi_l)\right), \\
[A_{22}]_{lj} &= \mathcal{A}\left((0,\, \psi_j)\,,(0,\, \psi_l)\right), \\
[b_1]_l &= \mathcal{L}\left(\psi_l,\, 0\right), \\
[b_2]_l &= \mathcal{L}\left(0,\, \psi_l\right),
\end{aligned}$$

for $1 \leq j, l \leq n$ and $\mathcal{A}$ and $\mathcal{L}$ are defined by (2.32)–(2.34).

Chapter 4

Code implementation

Once the mathematical models have been described (see Chapter 2), it was necessary to solve them by a Finite Element Method (see Chapter 3). In order to use them in a computer, they have been implemented using `FEniCS` (see [6]). `FEniCS` is an open-source `Python` library, which allows us to use a finite element discretizacion in a simple way. One of its greatest advantages is that the FEM implementation is done by setting the functional spaces, the type of finite element and the variational formulation. On the contrary, one of its main disadvantages is the inability of working with complex numbers. Consequently, any practitioner is enforced to work with the real and imaginary parts separately.

Before describing the codes, it is important to remark that, none of them are included in this report, however the following link contains all of them, including the input and the output of the test problems:

https://www.dropbox.com/sh/qavg1zbc7io450s/AACv9lZXYx1n1Hggjj2ANLGta?dl=0

4.1 Three dimensional acoustic problem stated in a bounded domain

The acoustic model proposed in Section 2.1 was computed in two different codes, one for the problem without viscosity, and another one for the acoustic problem with viscosity.

4.1.1 Model without viscosity

The first code solved the model (2.1). It is included in a file where a variety of functions have been implemented to compute the approximated solution. On one hand, the main program contains all the data and it calls the rest of the functions. It was run by the command:

```
python Helmholtz3D_boundedDomain.py
```

On the other hand, a test function was implemented. As the main program, it also contains all the data variables and calls the computation function. However, this program do not write any results to an external file. On the contrary, the relative error is computed for a numerical test where the exact solution is known in closed-form. To run the unit test it was necessary to type:

```
py.test -s -v Helmholtz3D_boundedDomain.py
```

The data variables used in the code are the following ones: mass density ρ, angular frequency ω, sound speed c and the error tolerance tol (used as threshold to decide the test fails).

Then, the `compute` function is called, where, first, the domain and the mesh were created. Because of the simple geometry used to solve this test, both, the geometry and the mesh have been created using the `mshr` library (a module of the `FEniCS` library).

The following step has been to identify the boundaries. To do it, two different classes were created, one for the rigid wall (outer boundary) and another one for the structure (inner boundary). Then, the mesh function for the boundary was initialized and saved in a `.pvd` file. This allows the user to check that not only the boundaries, but also the geometry, are well defined. Before setting the function space (Lagrange $\mathbb{P}^1$), the mesh function for the physical domain was defined.

Subsequently, the boundary conditions were implemented, besides the trial and the test functions. Finally, the variational formulation (2.4) was written as follows:

```
# Define variational problem

a = c**2*rho*inner(grad(u_re), grad(v_re))*dx
+ c**2*rho*inner(grad(u_im), grad(v_im))*dx
- omega**2*rho*(u_re*v_re)*dx
- omega**2*rho*(u_im*v_im)*dx

L = g_re*v_re*ds(2) + g_im*v_im*ds(2)
```

After assembling the matrix and the source vector and applying the boundary conditions to them, the problem was solved by the use of the `solve` command.

4.1.2 Model with viscosity

In this subsection, the code implemented to solve model (2.5) is described. As it has been explained previously (see Subsection 4.1.1), there were two ways to run the code, the main and the test one. To do this, it was necessary to type:

```
python Helmholtz3D_boundedDomain_viscous.py
```

or

```
py.test -s -v Helmholtz3D_boundedDomain_viscous.py
```

Again, the data variables (mass density ρ, angular frequency ω, sound speed c, viscosity ν and error tolerance tol) were settled in both of them. Moreover, in the second one, the relative error is computed for a numerical test, where the exact solution is known in closed-form. Then, the `compute` function is called, where firstly the domain and the mesh were created. Due to the geometry and the mesh are identical to the previous one, they were created using the `mshr` library.

The whole code was almost identical to that one without viscosity. Due to the viscosity (and its associated complex-valued coefficient), to split the boundary conditions into its real and imaginary part is more difficult. For this reason, the boundary conditions were implemented into a class instead in a expression as described above. The following code is an example that shows

how the Neumann condition were defined (an analogous implementation has been made for the Dirichlet conditions).

```
# Define Neumman boundary data for the real
# and the imaginary part

class RealNC(Expression):
def __init__(self, **kwargs):
self._omega = kwargs["omega"]
self._c = kwargs["c"]
self._rho = kwargs["rho"]
self._nu = kwargs["nu"]
def eval(self, value, x):
k = np.sqrt(self._omega**2*self._rho/(self._rho* \
self._c**2 - 1j*self._omega*self._nu))
r = np.sqrt(x[0]**2 + x[1]**2 + x[2]**2)
value[0] = -1*np.real(-1*np.exp(-1j * k * r) /r**2 \
- 1j * k * np.exp(-1j * k * r) / r)

g_re = RealNC(degree=4, omega=omega, c=c, rho=rho, nu=nu)

class ImagNC(Expression):
def __init__(self, **kwargs):
self._omega = kwargs["omega"]
self._c = kwargs["c"]
self._rho = kwargs["rho"]
self._nu = kwargs["nu"]
def eval(self, value, x):
k = np.sqrt(self._omega**2*self._rho/(self._rho*\
self._c**2 - 1j*self._omega*self._nu))
r = np.sqrt(x[0]**2 + x[1]**2 + x[2]**2)
value[0] = -1*np.imag(-1*np.exp(-1j * k * r) / r**2 \
- 1j * k * np.exp(-1j * k * r) / r)

g_im = ImagNC(degree=4, omega=omega, c=c, rho=rho, nu=nu)
```

4.2 Three dimensional acoustic problem stated in an unbounded domain

In order to solve the model (2.14), the PML program `Helmholtz_3D_PML` was introduced. It was composed by nine different files, which are detailed below.

4.2.1 Main program: `Helmholtz3D_PML_main.py`

This program reads the default settings and it calls any of the different options, which can be selected. Those ones are optimization, frequency response and computing the directivity pattern. To execute it, just type in the console:

```
python Helmholtz3D_PML_main.py
```

4.2.2 Default settings: `Defautl_settings.py`

In this file all the parameters can be changed in order to simulate a new geometry. Each one will be explained in detail.

The first parameter consists in an index that allows the user to choose the option that the program will execute depending on the number entered: 0, 1 or 3 for optimization, frequency response and directivity pattern, respectively. If another number is entered, there will be an error message and the program will not be executed. Then, the physical parameters, as mass density, sound speed and viscosity, as well as the external position of the PML are introduced. There are also options to refine the mesh or a part of it and data to modify the boundary conditions. Then, depending on the option that was chosen, input data can vary. For instance, for the optimization, only one frequency is needed, while for the other choices, a vector of frequency values has to be introduced. In all of them, the path and the name of the mesh are written as well as the name of the files, where the output results will be saved.

There are three boundary conditions implemented in this file. Two of them are related with the tests designed to check it, and the last one is used to simulate the probe in Chapter 6. The first boundary condition is

a Dirichlet one and consists in a monopole, while the second has the same exact solution but with a Neumann condition on the structure. Finally, the last one consists in a plane wave sent from the outside of the domain impinging the structure. Depending on which one must to be solve, it is necessary to introduce a number to indicate it: 0 for the first problem, 1 for the second one and 2 for the last one.

Furthermore, in the optimization code, values for the parameters to optimize must be entered, such as the number of meshes and the number of parameters to be optimized. Otherwise, for the frequency response, not only the frequency vector is needed but also whether the exact solution is known or not. Besides, in the directivity pattern, the number of angle values and the elevation angle from z-axis are required as input data.

Finally, a spline is computed to obtain the optimal values of the PML absorption coefficient, based on some fixed numbers of optimal values of this coefficient (see Chapters 5 and 6 for a detailed discussion).

4.2.3 Optimization: `Helmholtz_optimization.py`

In this file, the solution of model (2.14) is computed for different geometries. Because two different parameters can be optimizing at the same time, it is necessary to implement two loops, one for each parameter variation. In the inner loop, the module that contains all the data needed for the FEM simulation is called in addition to the computation one. Then, in the post processing, the sound pressure level as well as the particle velocity level are computed for a fixed measurement point. Moreover, the gain related with both of them is also obtained.

Finally, the results are written in `.txt` files and plotted. The type of graph that is used depends on the number of parameters to optimize. If there is only one, a line graph is drawn, if there are more, a 2D colour diagram is considered.

4.2.4 Frequency response: `Helmholtz_frf.py`

The frequency response consists in studying the solution for a sweep in frequencies, in other words, the solution will be obtained for a vector of

frequencies not only for one, as in the optimization. Moreover, only one loop is needed to change each frequency computation. While the computation function is called inside the loop, the data for the FEM simulation are pre-computed outside it.

In addition to the computation of pressure, velocity and gains in the measurement point, the pressure and the intensity fields are plotted for the highest frequency in the whole domain. Moreover, if the exact solution is known, the error will be also plotted there.

4.2.5 Directivity pattern: `Helmholtz_directivity.py`

The last choice consists in computing the directivity pattern. To do it, two loops are needed. The outer one changes the frequency in which it was computed, while the inner one varies the angle in which the plane wave is impinging on the structure.

In this case, to plot the results it is necessary to use polar coordinates. This is done as it is shown in what follows:

```
pplot = plt.figure('pres')
ap = pplot.add_axes([0.1,0.1,0.8,0.8], polar=True)
ap.set_ylim(-30,0)
ap.set_yticks(np.arange(-30,0,10))
ap.plot(phi,pressure[i] - np.amax(pressure), 'b-')
```

The maximum of the SPL and PVL levels is subtracted to normalize the directivity patter, so the maximum value will be null and the rest of entries will be negative.

4.2.6 Data for the FEM simulation: `Datfem.py`

This `Python` module initializes all the needed data for the FEM simulation. Firstly, the mesh and its subdomains are read from a `.txt.gz` file. Then, the mesh can be refined if the option is chosen in `Default_settings` and the domains are reorganized. After it, the boundaries are detected and the functional spaced defined. Again, it is used Lagrange $\mathbb{P}^1$. Finally, external unit normal vectors to the boundaries are computed as follows:

```
# Compute normal vectors
class NormalExpression(Expression):
def __init__(self, mesh, **kwargs):
self.msh = mesh

def eval_cell(self, values, x, ufc_cell):
# Find normal
cell = Cell(self.msh, ufc_cell.index)
n = cell.normal(ufc_cell.local_facet)
values[0] = n[0]
values[1] = n[1]
values[2] = n[2]

def value_shape(self):
return (3,)

normals=NormalExpression(mesh=self.mesh, degree=1)
self.n_vec = normals
```

4.2.7 Computation: `Helmholtz_computation.py`

This file contains the implementation of the computation of the problem. To do that, the function that contains boundary conditions has been called, and then, the obtained values are associated with the boundaries marked in `Datafem.py`.

Moreover, the wave number is computed, as well as the coefficients used in the variational formulation (2.32)−(2.34). Later, trial and test functions are initialized and the weak formulation is introduced. After assembling the matrix and the right-hand side vector, the boundary conditions are applied to the linear system. Finally, the linear matrix problem is solved.

Notice that, if the boundary condition is given by a plane wave, due to the fact that the acoustic model only computes the scattering pressure, the plane wave has to be added to the approximated solution.

```
p_re    =       Function(Q)
p_im    =       Function(Q)
uex_re_interp = interpolate(uex_re, Q)
uex_im_interp = interpolate(uex_im, Q)
p_re = project(u_re + uex_re, Q)
p_im = project(u_im + uex_im, Q)
```

4.2.8 Boundary conditions: `Boundary_conditions.py`

`Boundary_conditions.py` contains all the possibilities for the boundary conditions that has been explained in the previous sections. In order to do it, an `if` statement is created. The first option contains the monopole boundary conditions. Notice that the test problems stated in bounded domains are similar but the take into account different kinds of conditions: Dirichlet and Neumann.

With this purpose, four classes are written. Two of them contain the exact solution, one for the real part and the other one for the imaginary one. The other ones, are the real and the imaginary part of the gradient of the exact solution. Finally, the last option corresponds to the plane wave. Not only its expression is entered, but also its gradient for computing the Neumann conditions that has been imposed on the structure. It has also been implemented an error message that is reached when the number entered for choosing the boundary conditions is incorrect.

4.2.9 Computing the results: `Evaluations.py`

The last file contains some functions that were needed to post-process the numerical results. They are listed in what follows:

`Eval_pressure:` it contains the formula to compute the pressure at a point.

```
return u_re(p) + 1j*u_im(p)
```

`Eval_velocity:` it computes the gradient of the pressure and it is used to obtain the velocity at a point.

`Write:` This function is used to write the real and imaginary part of the pressure. Their values in the whole domain are considered (not only at a point). Moreover, the intensity is computed and also plotted in the whole computational domain.

`Write_error:` This function is only used when the exact solution is known and the error between it and the approximated solution is required to be plotted in the whole domain.

4.3 Computation of the optimal PML absorption coefficient

This program is a simplification of the previous one (Section 4.2). The aim of creating it was to obtain the optimal absorption coefficient for the PML, so any option is included in the code. There are only four files in this program, because the main program, the compute and the evaluation function are all included in one file.

There are several differences between this program and the equivalent described in the previous section. Due to no options are included, the loop changes the value of the absorption coefficient each time and compute the error for each PML absorption coefficient value. Moreover, the velocity is not computed, because it is not needed to check which one is the best PML absorption coefficient at each frequency. The computation of this optimal coefficient will be explained in Chapter 6.

According to the default settings, the computation of the optimal PML absorbing coefficient is performed at a single and fixed frequency value. To do it, a range of possible values of this coefficient is introduced.

4.4 Experimental comparison

This code was written to compare the numerical results obtained by the simulation with those ones obtained by the experimental measurements. In order to do that, the frequencies that were used in the simulations have to

be read from an external file, as well as the gain associated to the particle velocity level. Moreover, two different data from the measurements are also imported.

The following sept is comparing the calibration to check if the measurements were done correctly, checking if the position of the sensor was the same for both measurements. Then, the gain for the measurements are computed. It is important to notice, that the gain is included in one of the data, so it is necessary to subtract it before the comparison.

Finally, a spline is computed to obtain the values of the measurement gain for those frequencies of the numerical simulation. Additionally, the error between both gains is obtained.

Chapter 5

Validation tests

Once the finite element codes have been implemented (see Chapter 4), it was necessary to validate them. In order to do that, some academic test problems with a known solution in closed form have been considered. First, a problem stated in a bounded domain was introduced. This problem was solved with and without viscosity. Second, some problems were introduced in order to validate the Helmholtz equation stated in an unbounded domain, where the PML technique has been utilized.

5.1 Three dimensional problem stated in a bounded domain

The model used to validate the implementation of the finite element codes involves a three dimensional problem stated in a bounded domain. It is analogous for the viscosity problem and for that one without viscous terms.

The computational domain is a cube occupied by the fluid, in which a sphere is embedded. The dimensions of the cube are $2 \times 2 \times 2$ meters, and the sphere is centred at origin with radius 0.5 m. Due to the sphere is considered as a rigid structure, the motion associated to the spherical surface is governed by a boundary condition. In both cases, the exact solution will be settled as an outgoing radial solution.

5.1.1 Problem without viscosity

The exact solution for the first test problem is given by the monopole equation (1.5), in which the intensity goes from the body to infinity. So, to obtain this exact solution, the boundary data have been computed taking into account the expression of the exact solution. Consequently, the test problem is given by (2.1), being the boundary data

$$f = \frac{\cos(-k\,r)}{r} + i\,\frac{\sin(-k\,r)}{r},$$

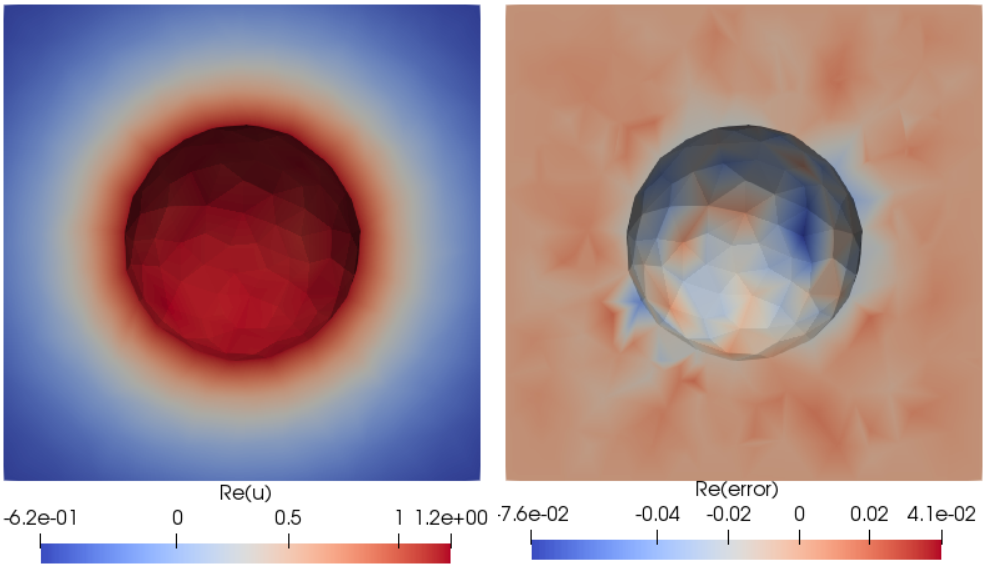
and

$$g = \left(\frac{-k\sin(-k\,r)}{r} + \frac{\cos(-k\,r)}{r^2}\right) + i\,\left(\frac{k\cos(-k\,r)}{r} + \frac{\sin(-k\,r)}{r^2}\right),$$

with $r = \sqrt{x^2+y^2+z^2}$.

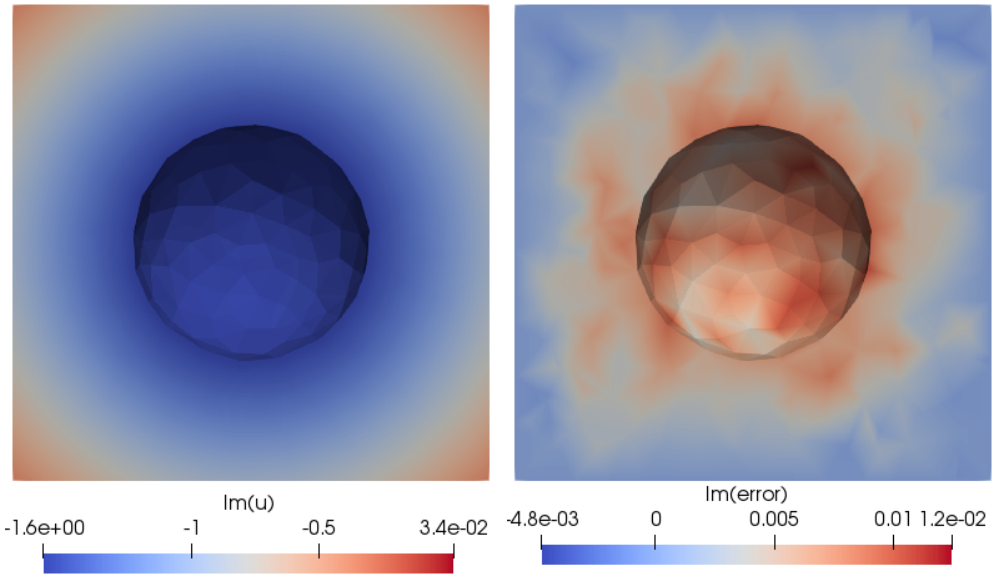
In addition, the default data settings are $\omega = 200\pi$ rad/s, $\rho = 1.2$ kg/m^3, $c = 340$ m/s, and $k = \frac{\omega}{c}$.

Using a regular mesh where the maximum mesh size is $h_{\max} = 0.08$, the L^2–relative error for the pressure field is given by 1.2%. The real part of the approximated pressure is shown in the left plot of Figure 5.1. The right plot of the same figure shows the absolute error between the finite element approximation and the exact solution. In Figure 5.2, analogous results are shown, but in this figure corresponding to the imaginary part of the pressure field.



(a) FEM solution (b) Error

Figure 5.1: Cross section at the origin point in the plane yz of the computational domain, where the numerical results associated to the inviscid acoustic problem stated in a bounded domain are plotted. On the left, the numerical solution of the real part of the pressure field is plotted. On the right plot, the relative error between the real parts of the approximated solution and the exact solution is plotted.



(a) FEM solution (b) Error

Figure 5.2: Cross section at the origin point in the plane yz of the computational domain, where the numerical results associated to the inviscid acoustic problem stated in a bounded domain are plotted. On the left, the numerical solution of the imaginary part of the pressure field is plotted. On the right plot, the relative error between the imaginary parts of the approximated solution and the exact solution is plotted.

5.1.2 Problem with viscosity

When the viscosity is introduced and the exact solution is the monopole equation (see Eq. (1.5)), the test problem is described by (2.5) with the following boundary data:

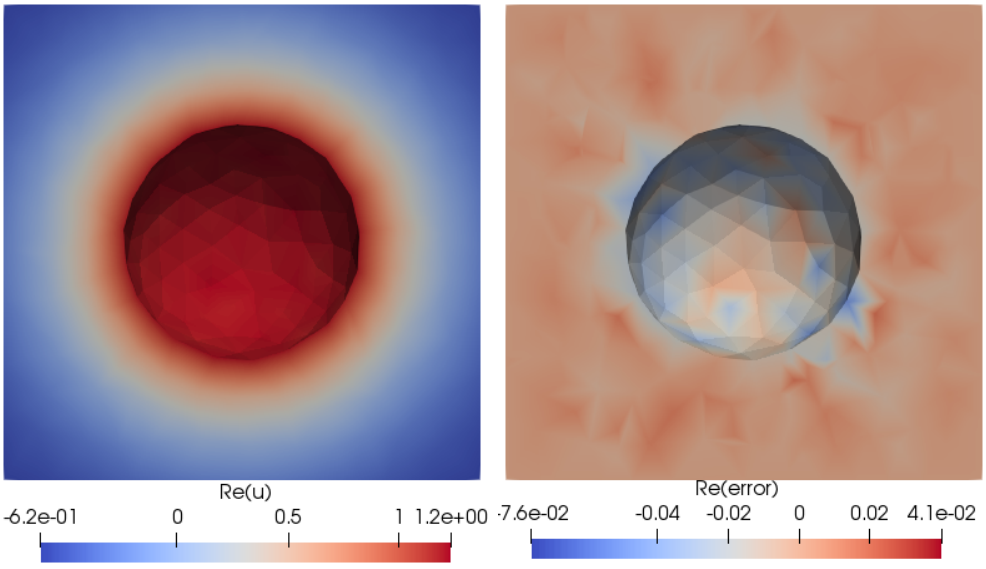
$$f = \frac{e^{-ikr}}{r},$$

$$g = \frac{e^{-ikr}}{r^2} + i\,\frac{k\,e^{-ikr}}{r},$$

where $r = \sqrt{x^2 + y^2 + z^2}$.

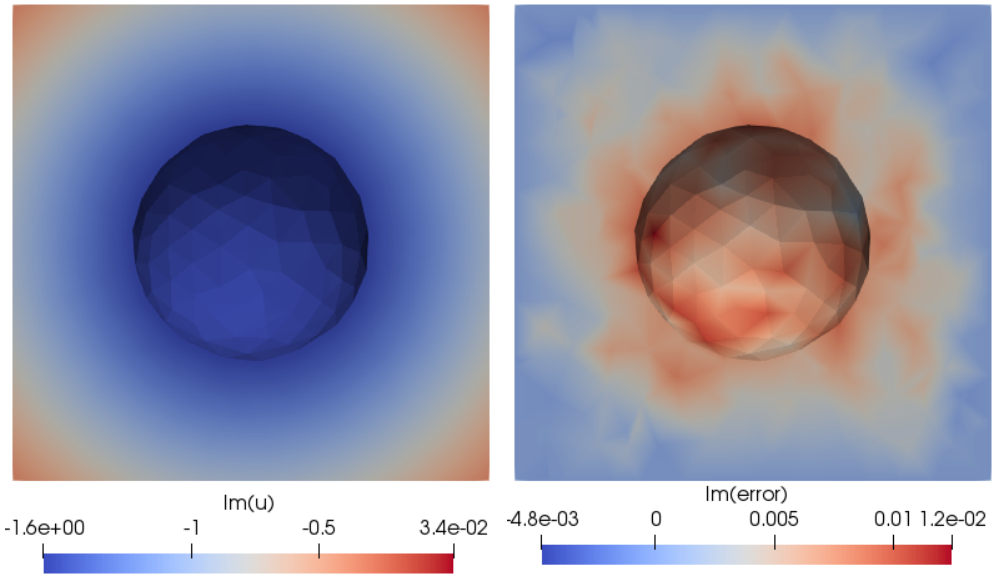
In addition, the default settings are $\omega = 200\pi$ rad/s, $\rho = 1.2$ kg/m^3, $c = 340$ m/s, $\nu = 1.8 \times 10^{-5}$ Pa·s and k is given by Definition 1.2.2.

Using a regular mesh where the maximum mesh size is $h_{\max} = 0.08$, the L^2–relative error for the pressure field is given by 1.2%. The real part of the approximated pressure is shown in the left plot of Figure 5.3. The right plot of the same figure shows the absolute error between the finite element approximation and the exact solution. In Figure 5.4, analogous results are shown, but in this figure corresponding to the imaginary part of the pressure field.



(a) FEM solution (b) Error

Figure 5.3: Cross section at the origin point in the plane yz of the computational domain, where the numerical results associated to the viscous acoustic problem stated in a bounded domain are plotted. On the left, the numerical solution of the real part of the pressure field is plotted. On the right plot, the relative error between the real parts of the approximated solution and the exact solution is plotted.



(a) FEM solution (b) Error

Figure 5.4: Cross section at the origin point in the plane yz of the computational domain, where the numerical results associated to the viscous acoustic problem stated in a bounded domain are plotted. On the left, the numerical solution of the imaginary part of the pressure field is plotted. On the right plot, the relative error between the imaginary parts of the approximated solution and the exact solution is plotted.

5.2 Three dimensional problem stated in an unbounded domain

The test problem used to validate the codes for a three dimensional problem stated in an unbounded domain has the same geometry as those ones considered for the problems stated in bounded domains. In this case, the sphere is centred at the origin with radius 0.05 m, the inner PML boundaries are located to 0.2 m from the origin and the PML thickness is 0.2 m. However, in order to check the accuracy of the PML technique, first, a Dirichlet condition was introduced on the structure. Then, Neumann boundary conditions will be considered on the spherical boundary.

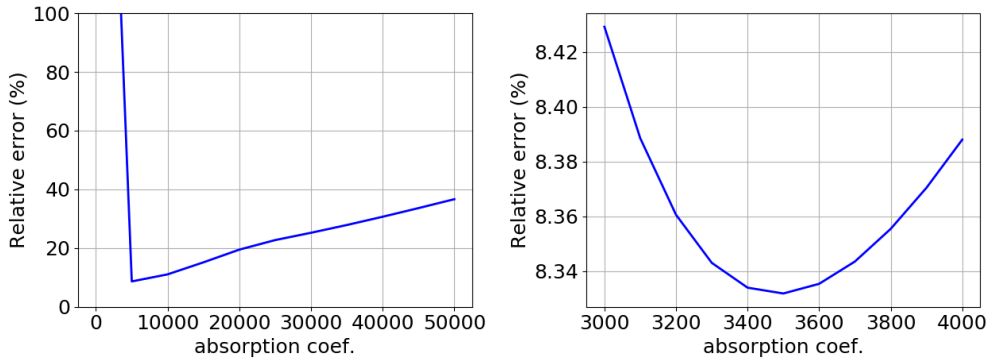
The exact solution for the first test problem is given by the monopole equation (1.5), in which the intensity goes from the body to infinity. So, to obtain this exact solution, the boundary data have been computed taking into account the expression of the exact solution. Notice that, despite Model (2.14) involves a Neumann boundary condition on the structure boundary, in this first testing problem, it was decided to introduce a Dirichlet condition in the spherical boundary with the following data:

$$p = \frac{e^{-i\,k\,r}}{r} \quad \text{on } \Gamma_{\mathrm{s}},$$

being $r = \sqrt{x^2 + y^2 + z^2}$.

However, to obtain an accurate approximated solution for this problem, it is necessary to compute the optimal value for PML absorption coefficient σ_0 in the definition of the PML coefficients. With this purpose, a sweeping procedure has been performed to identify the value for σ_0, which reaches the minimum error level in the test problem (2.14). Since the computation of the optimal value for σ_0 is strongly dependent on the frequency value, the sweeping procedure must be repeated for each frequency value.

In consequence, for each fixed frequency, a wide range of possible values for σ_0 are considered. A nested strategy is used to minimize the number of times that the test problem has to be solved: first, a large interval with only a reduced of number of σ_0 values are taking into account; once the location of the optimal value in this coarse grid is obtained, a more fine grid search is performed in a smaller interval. This nested search strategy is illustrated in the plots of Figure 5.5.



(a) (b)

Figure 5.5: L^2-relative error for the pressure field, computed for different values of the PML absorption coefficient σ_0, at frequency 1000 Hz. The error that is shown on the right plot does not have significant variations, so the optimal value for the absorption coefficient is considered to be 3500.

Once the optimal value is obtained for some frequencies (see Table 5.1), a spline is computed to obtain the optimal values for all the frequencies of interest. This spline is plotted in Figure 5.6.

Frequency [Hz]	50	150	205	500	750	1000
Optimal value	840	1900	2800	2700	3800	3500

Table 5.1: Optimal values for σ_0 [s/rad] at each frequency value of interest.

Taking into account these optimal values, the test problem is solved obtaining a L^2-relative error lower than 10% for all the frequencies considered. The absolute error for the real part of the pressure field is shown in Figure 5.7 (where the numerical results are not plotted inside the PML domain). Moreover, in Figure 5.8, analogous numerical results are shown for the imaginary part of the pressure field.

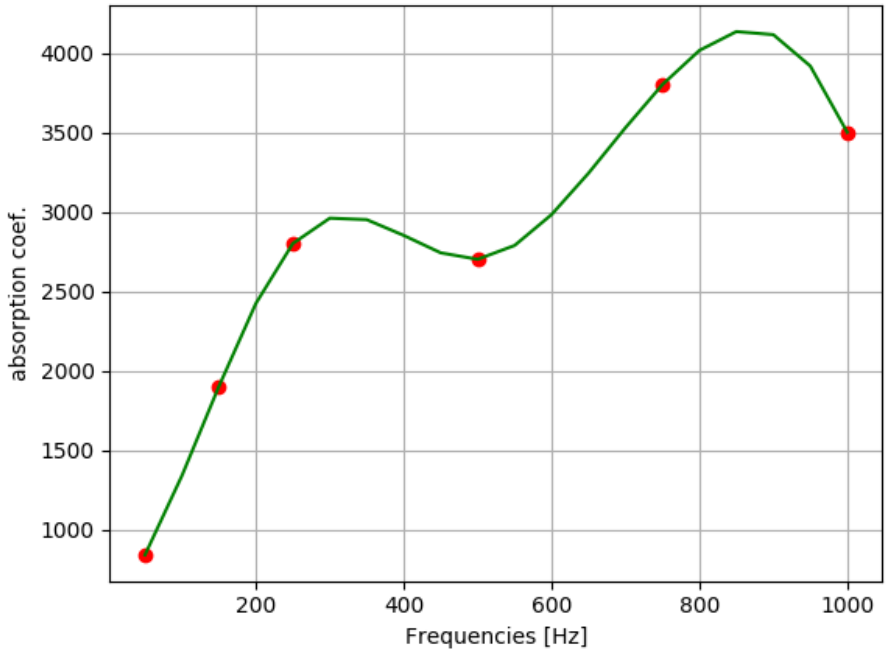
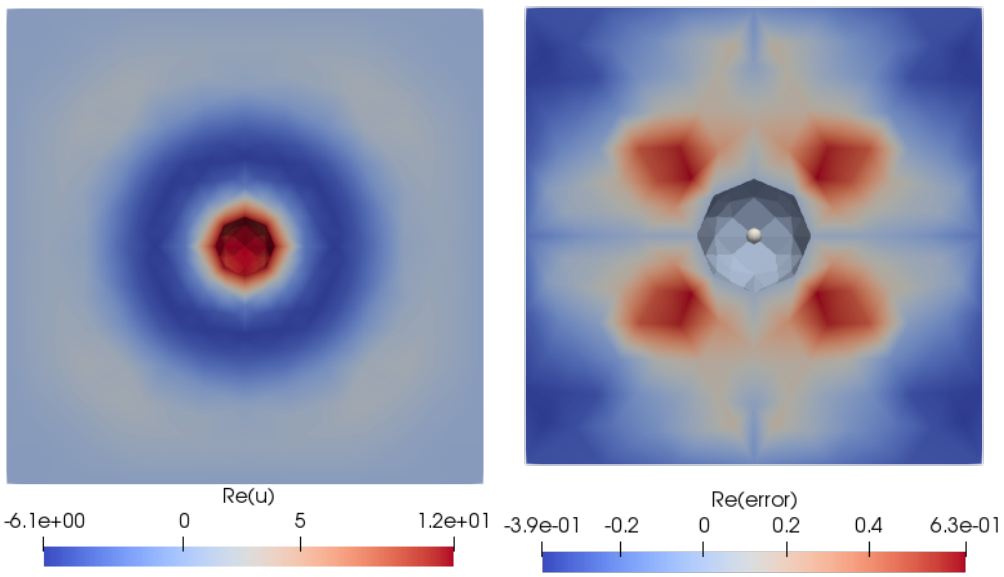
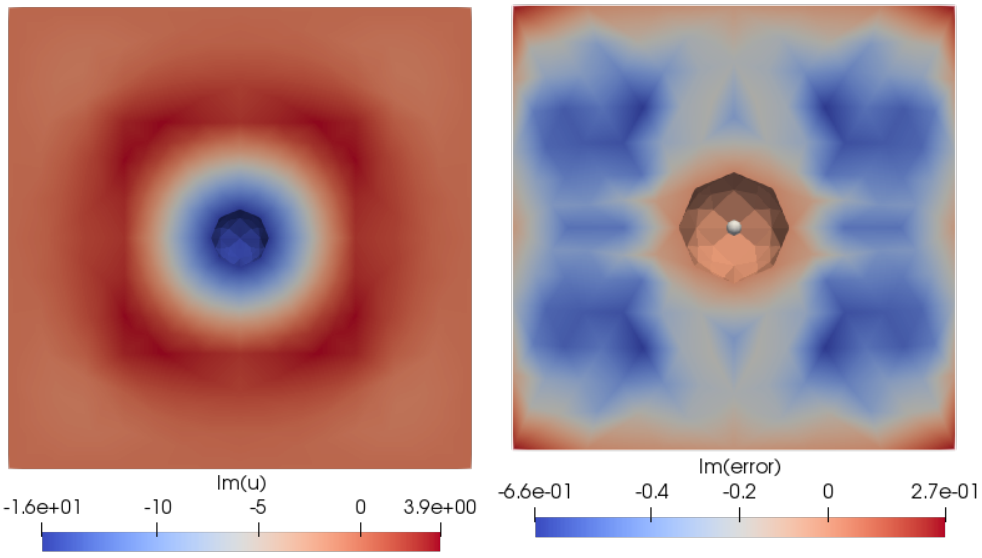


Figure 5.6: Values of the optimal absorption coefficient for each frequency computed by the spline, computed from the data of Table 5.1.



(a) Real pressure field (b) Real error

Figure 5.7: Cross section at the origin point in the plane yz of the computational domain, where the numerical results associated to the inviscid acoustic problem stated in an unbounded domain are plotted. On the left, the numerical solution of the real part of the pressure field is plotted (in the fluid and the PML domain). On the right plot, the relative error between the real parts of the approximated solution and the exact solution is plotted (only in the fluid domain).



(a) Imaginary pressure field (b) Imaginary error

Figure 5.8: Cross section at the origin point in the plane yz of the computational domain, where the numerical results associated to the inviscid acoustic problem stated in an unbounded domain are plotted. On the left, the numerical solution of the imaginary part of the pressure field is plotted (in the fluid and the PML domain). On the right plot, the relative error between the imaginary parts of the approximated solution and the exact solution is plotted (only in the fluid domain).

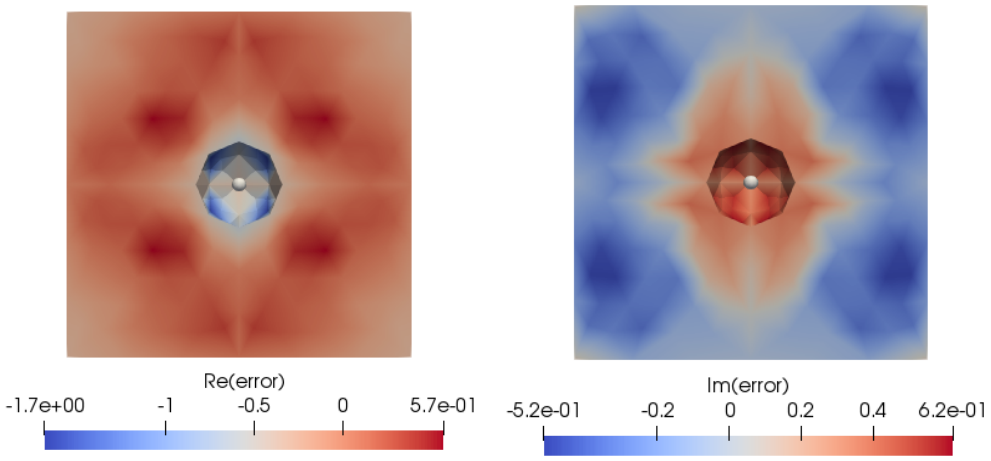
Finally, the Dirichlet boundary condition is replaced by a Neumann one, considering model (2.14) with following boundary data:

$$\frac{\partial p}{\partial \boldsymbol{n}} = \left(\frac{-k \sin(-k\, r)}{r} + \frac{\cos(-k\, r)}{r^2} \right) + i \left(\frac{k \cos(-k\, r)}{r} + \frac{\sin(-k\, r)}{r^2} \right),$$

being $r = \sqrt{x^2 + y^2 + z^2}$.

Due to the computational domain and the mesh are exactly the same as the previous one, the values for the PML absorption coefficient are not modified and hence they are given by Table 5.1. Again, the problem is approximated obtaining an error lower than 7% for all the frequencies

considered. The absolute error for the fluid without taking into account the PML layer is shown in Figure 5.9. On the left plot, it is shown the real error, while on the left one, it is plotted the imaginary error for the pressure field.



(a) Real error (b) Imaginary error

Figure 5.9: Cross section at the origin point in the plane yz of the computational domain, where the numerical results associated to the inviscid acoustic problem stated in an unbounded domain are plotted. On the left, the relative error between the real parts of the approximated solution and the exact solution is plotted (only in the fluid domain). On the right plot, the relative error between the imaginary parts of the approximated solution and the exact solution is plotted (only in the fluid domain).

Chapter 6

Optimization and numerical results

The aim of this master thesis has been focused on the optimization of a P-U probe. Therefore, after testing and validating the finite element code, the final step has consisted in modelling the current design of the probe and trying to improve it. The parametric optimization of the probe structure relies on a sweeping procedure to identify that geometry for which the highest value for the velocity gain was reached at the spatial point where the sensor is located. Once the optimal parameters have been computed, it has been also necessary to check whether the frequency response of the SPL and PVL levels remains flat for a wide frequency band and whether the directivity patterns have not been changed from those ones associated to the original shape of the probe structure. Furthermore, the numerical results obtained with the current design have been compared with respect to experimental measurements.

The software suite used to create the geometrical designs was `Salome` (see reference [2]). This free CAD software allows the users to create a `Python` script and to execute it in the console as an usual `Python` code. This feature eases the parametrization of the geometry and hence, with a simple loop, it is possible to create different parametric settings leading to a wide number of geometric configurations.

6.1 The original design

The first design to be introduced was the current one. The idea was to verify the model and the numerical simulation comparing the numerical results obtained by the implemented computer code with respect to the experimental measurements, which are provided by the company. In addition, this original CAD design has been used as the reference to define further parametric modifications. In Figure 5.7 the original design of the P-U probe is shown. In this design, the velocity sensor it is not included because of confidential restrictions imposed in the model by the company. Notice that the size of the P-U probe is quite small: its length is 17 mm and its diameter is 0.5 inches.

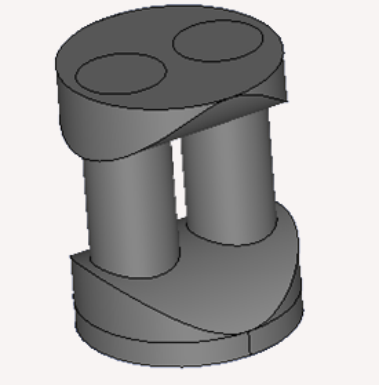


Figure 6.1: Original design of the P-U probe.

Firstly, as it has been discussed in Chapter 5, the PML absorption coefficient has to be optimized. Following the steps described in Section 5.2, the optimal values have been computed for a discrete set of 11 different frequencies (see Table 6.1).

Taking into account this computational domain, the scattering problem is solved by assuming that a plane wave is impinging the probe. The goal in the numerical simulation consists in computing the gain of the SPL and the PVL levels.

Figure 6.2 shows both gains, which have a flat frequency response. While

Frequency [Hz]	Optimal values [s/rad]
50	1280
150	3708
250	5900
500	10700
750	14800
1000	18300
2000	29700
4000	43100
6000	44500
8000	49000
10000	48500

Table 6.1: Optimal values of the PML absorption coefficient for 11 different frequencies. The computational domain is the exterior of the original P-U probe, surrounded by a PML of thickness 13 mm located at 13 mm from the probe.

the sound pressure level gain is around one decibel, the gain associated to particle velocity level is approximately equal to 10 dB.

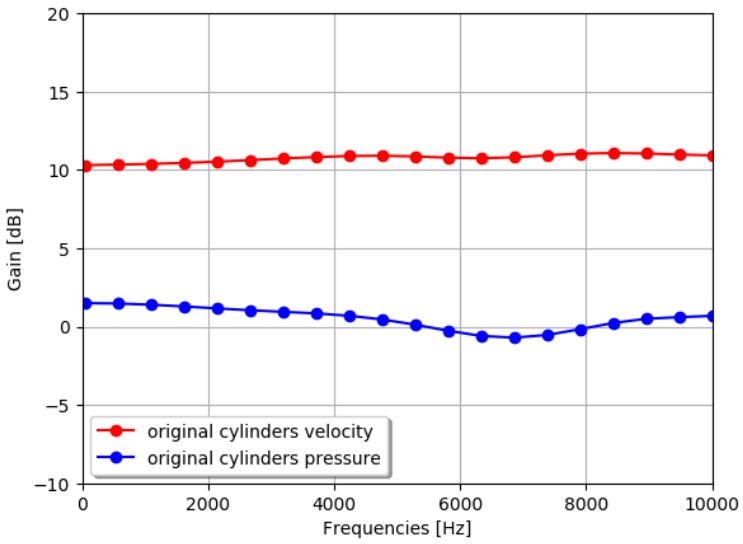
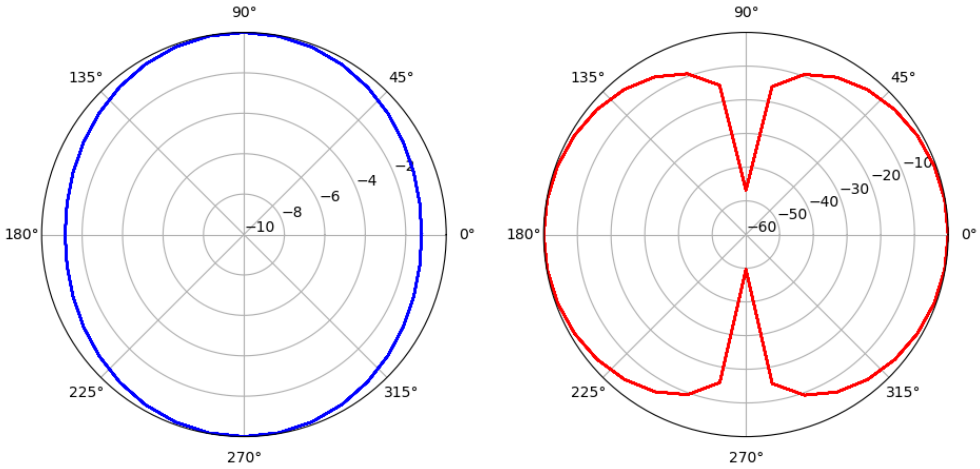


Figure 6.2: Gains associated to SPL and PVL, which are computing by using the original P-U probe design.

Finally, the directivity pattern is studied and the numerical results are shown in Figure 6.3. It can be observed that the directivity pattern associated to the pressure field is omnidirectional. However, the pattern associated to velocity field has an eight shape rotated horizontally.



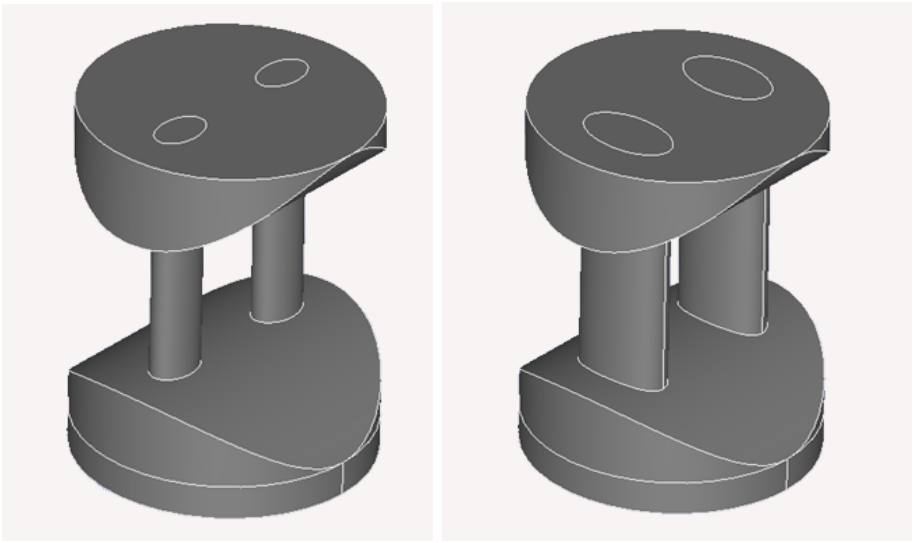
(a) Pressure in dB

(b) Velocity in dB

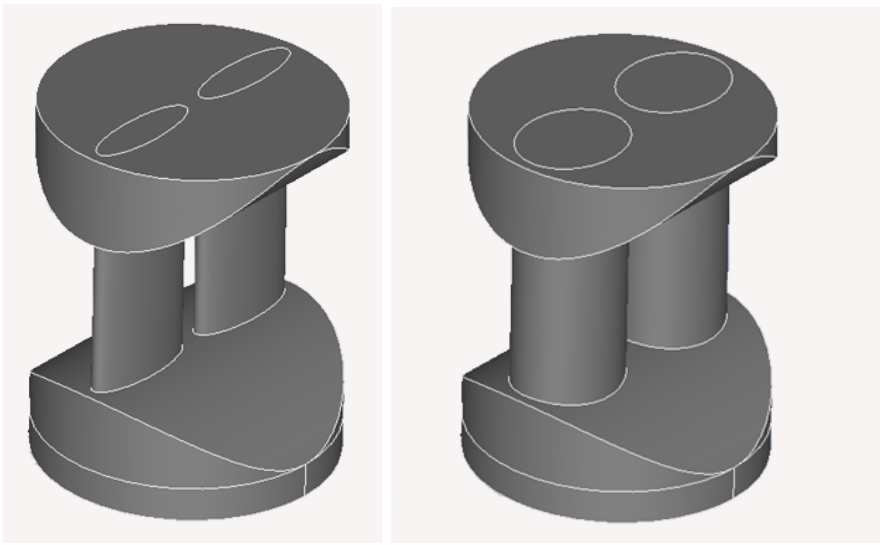
Figure 6.3: Directivity pattern for the SPL and PVL levels computed with the original P-U probe design for a frequency of 10000 Hz.

6.2 Modifying the eccentricity of the pillars

The first part of the geometry to be changed was the pillars. In the company, the first goal was focused on the gain response in the case of the eccentricity of the pillars had been changed. With this purpose, the radius of the elliptic cross-sections of the pillars were modified for a feasible range of values (see different geometries associated to different values in Figure 6.4).



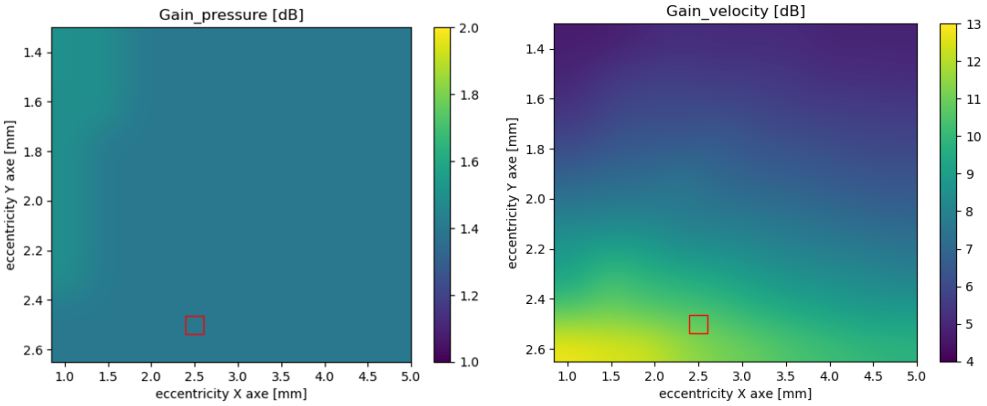
(a) (b)



(c) (d)

Figure 6.4: Different examples of probe designs when the eccentricity of the pillars is modified. In plot (a), both, the radius in the x- and y-axis are the smallest ones. In plot (b), the radius in the x-axis is larger than in the y-axis, while in plot (c), the relation between radius of x- and y- is reciprocal. Finally in panel (d), the radius are the largest ones that have been considered.

The optimization results (see Figure 6.5) show that the sound level pressure never changes no matter how the eccentricity does. However, the particle velocity level is higher when the radius in the x-axis (perpendicular axis to the blank space between pillars) is smaller and the radius of the y-axis is larger. In the plots of Figure 6.5, the red squares mark the computed values associated to the original shape of the cylindrical pillars.



(a) Pressure in dB

(b) Velocity in dB

Figure 6.5: Gain results computed for the optimization of the pillars. On the left, the gain associated to the sound pressure level variations are plotted, while on the right it is the plotted the gain associated to the particle velocity level. The red squares in both cases mark the value computed with the original design of the probe.

Then, for the optimal configuration, which is shown in Figure 6.6, the frequency response of the gains is studied numerically. As it can be observed in Figure 6.7, the gain of the SPL is pretty similar for the optimized geometry and for the original one. However, the gain of the PVL level is increased two decibels with the new pillars design.

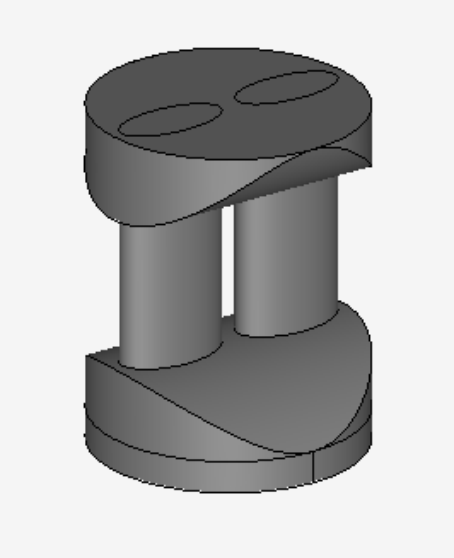


Figure 6.6: Optimal design for the parametric pillars varying the radius in the x- and y-axis.

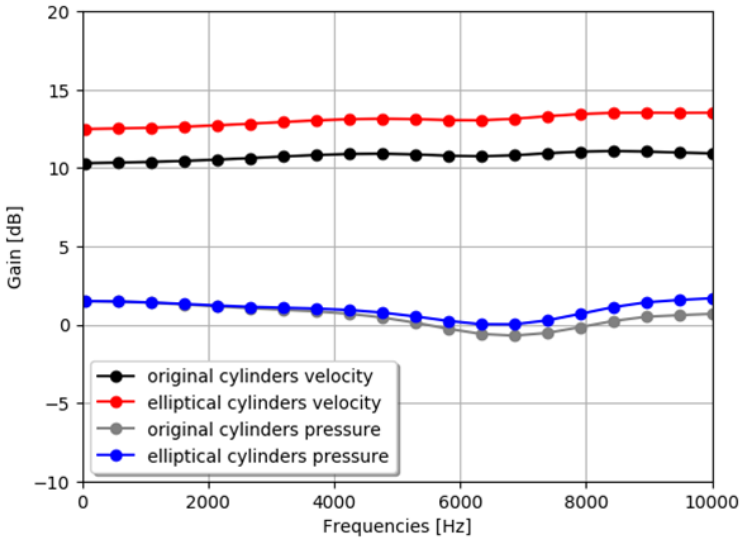
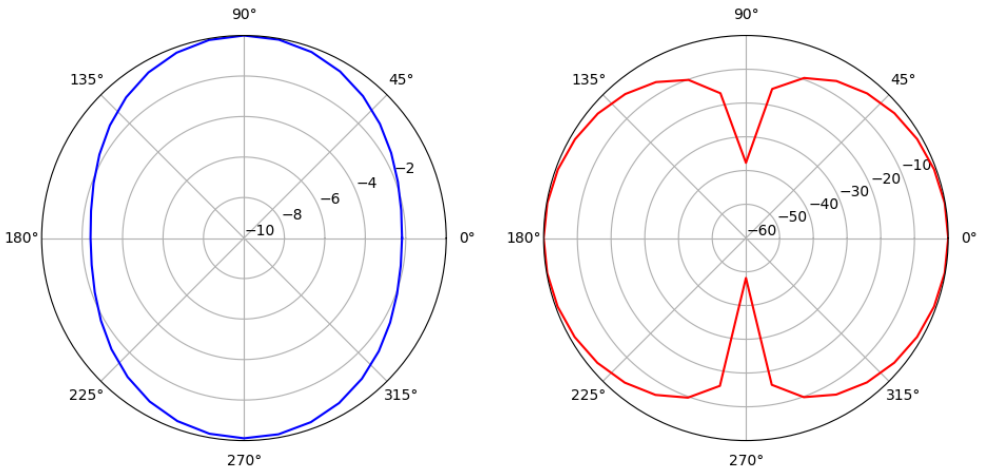


Figure 6.7: Frequency response of the SPL and PVL gains computed with the original P-U geometry and with the optimal design of the pillars.

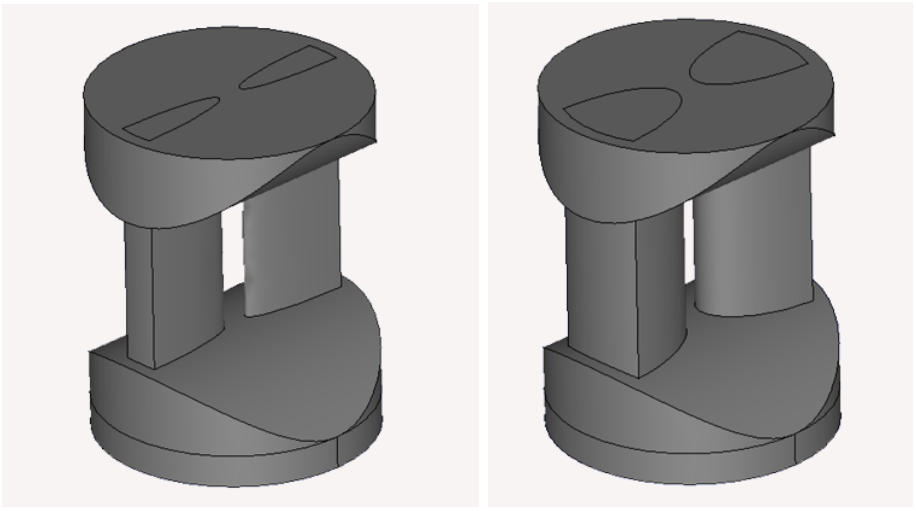
Finally, the directivity pattern is studied. In Figure 6.8, the patterns of the sound pressure level and particle velocity level are plotted at a frequency of 1000 Hz. Both patterns are almost identical to those ones obtained for the original design of the acoustic probe. Due to this fact and taking into account that the frequency response of the gains are flat, it is possible to admit this new configuration as feasible and optimal.



(a) Pressure in dB (b) Velocity in dB

Figure 6.8: Directivity patterns for SPL and PVL levels computed with the optimal design of the pillars at a frequency of 10000 Hz.

In view of the numerical results obtained for the optimization of the pillars, the following goal in the parametric optimization was focused on maximizing the acoustic intensity through the pillars, where the sensor should be placed. With this purpose, the elliptical cylinders are made larger and symmetrically truncated, as it is shown in Figure 6.9. In this figure, two different examples of this kind of geometry are plotted with opposite configurations. On the left, the width of the truncated part is the smallest one, while on the right, the diameter of the truncated cylinders is as large as possible.



(a) (b)

Figure 6.9: Two different examples of probe design when the elliptical pillars are truncated symmetrically. On plot (a), the width of the truncated part is the smallest one. In plot (b), the diameter of the truncated cylinders is as large as possible.

The obtained results are plotted in Figure 6.10, where it can be checked that the largest is the truncated part, the highest is the gain associated with the particle velocity level. Again, the sound pressure level is not affected by the changes in the geometry.

Once the optimal configuration has been identified (see Figure 6.11), the frequency response of the gains is studied. In Figure 6.12, it is shown that the SPL and PVL gains are almost flat. Moreover, despite the fact that the gain associated to the sound pressure level is almost identical in the original P-U design, the gain of the particle velocity level is higher in this new design.

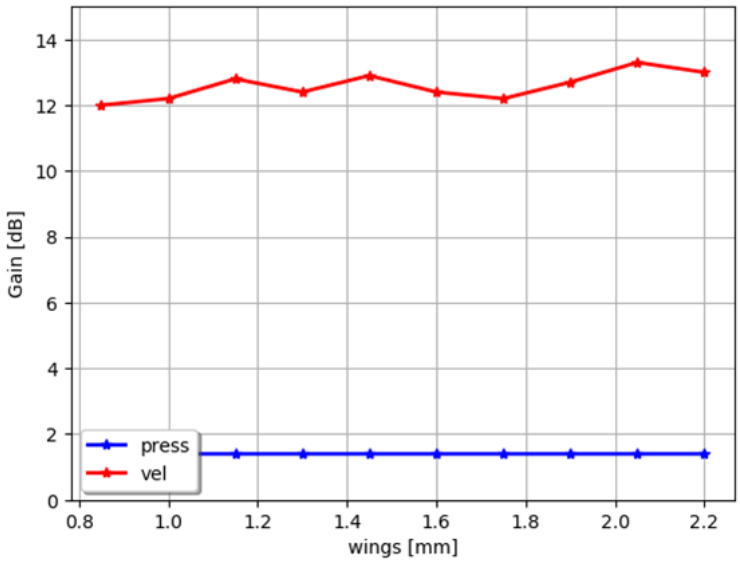


Figure 6.10: Gains associated to SPL and PVL, which have been computed by using different truncated pillars configurations.

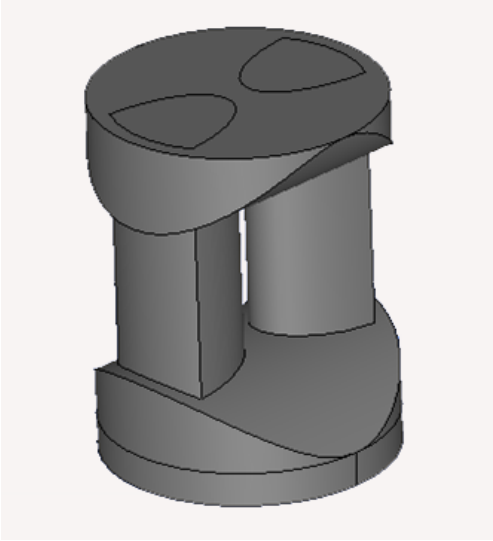


Figure 6.11: Optimal design for the parametric truncated pillars.

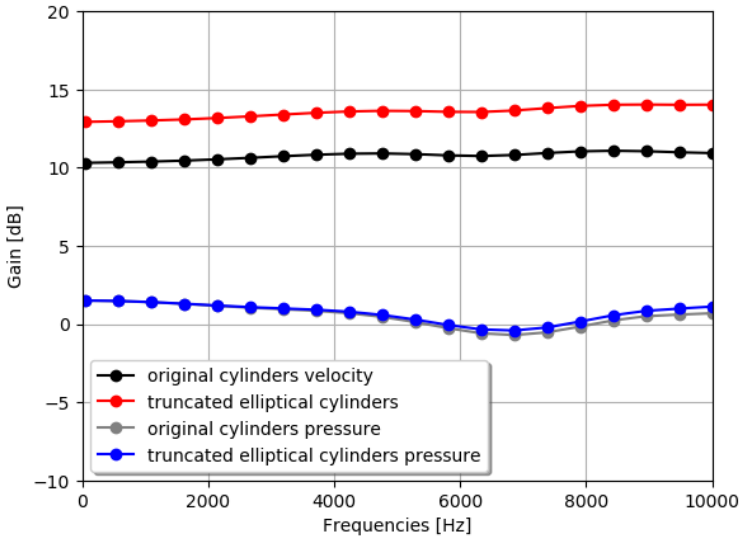
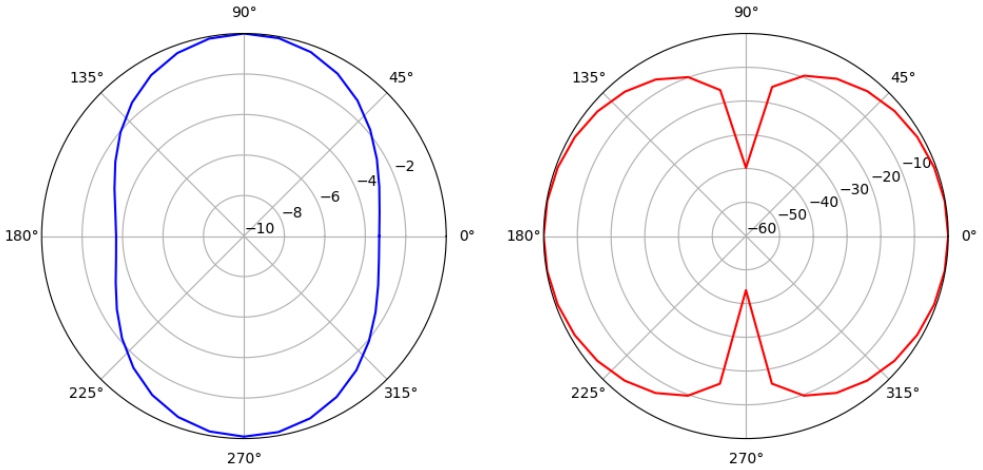


Figure 6.12: Frequency response of the SPL and PVL gains computed with the original P-U geometry and with the optimal design of the truncated pillars.



(a) Pressure in dB (b) Velocity in dB

Figure 6.13: Directivity patterns for the SPL and PVL levels computed with the optimal design of the truncated pillars at a frequency of 10000 Hz.

The numerical study of the directivity pattern confirms that this

geometry is not only optimal in terms of increasing the PVL gain but also feasible. Again, in Figure 6.13, it is shown that the SPL directivity pattern is omnidirectional and the PVL pattern is described by an horizontally rotated *"eight–shape"* angle distribution.

Finally, a comparative plot among the frequency response of the gains for the original probe and the two new optimized geometries is made. Due to the sound pressure level was almost identical in the three cases, only the gain associated to the particle velocity level is plotted in Figure 6.14. In this plot, it is easy to check that the geometry with the largest PVL gain is that one computed with the optimal truncated pillars. This configuration provides a PVL gain of 2.5 dB larger than the original one, and 0.5 dB larger than the geometry with the optimal elliptical cylinders. Hence, taking into account the parametric optimization of the pillars, it is possible to conclude that the optimal truncated elliptical cylinders configuration is the optimal one among all the probe designs tested in this master thesis.

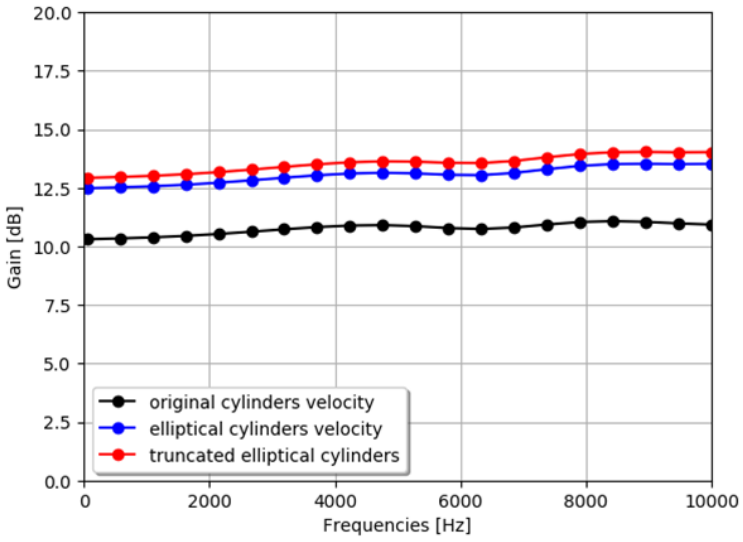


Figure 6.14: Frequency response of the gain associated to the PVL using three different designs: the original one, that one with optimal elliptical pillars and that one with optimal symmetrically truncated pillars.

6.3 Modifying the width of the probe

Another completely different approach has been also considered: it has consisted in enlarging the probe, re-scaling its geometrical dimensions but without changing the ratios between lengths, thickness or radius of its different parts. The diameter of the original probe is 0.5 inches, so the idea is to increase it first, to 0.75 inch and second, to 1 inch.

In this case, no sweeping procedure is required to identify the optimal geometric parameters, since the aim of the numerical study is just to obtain the frequency response of a new probe with a fixed set of parameters. Due to this fact, only the frequency response of the gains is analysed.

However, the modification of the width of the probe implies to vary the position of the PML, since the dimensions of the fluid domain surrounding the probe are also modified. For this reason, it is necessary to optimize the PML absorption coefficient for these two new geometries.

6.3.1 Diameter of 0.75 inch

The optimal values for the absorption coefficient of the PML are written in Table 6.2, while the geometry is plotted in Figure 6.15.

Frequency [Hz]	Optimal values [s/rad]
50	1400
150	4000
250	6400
500	11600
750	16000
1000	19600
2000	31000
4000	45000
6000	47000
8000	51000
10000	48000

Table 6.2: Optimal values of the PML absorption coefficient for 11 different frequencies. The computational domain is the exterior of the P-U probe with a diameter of 0.75 inches, surrounded by a PML of thickness 13 mm located at 13 mm from the probe.

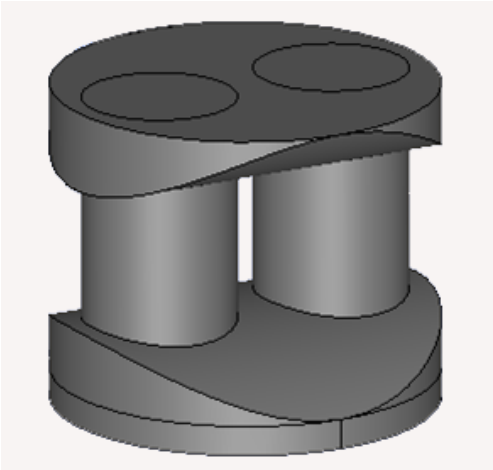


Figure 6.15: Geometry of the P-U probe with a diameter of 0.75 inches.

The numerical results obtained for the gains associated to this geometry are plotted in Figure 6.16. Despite the gain associated to the particle velocity level is almost flat, the gain associated to the sound pressure level varies in a range between 1 to 10 dB, which becomes infeasible to consider this geometrical design for experimental measurements.

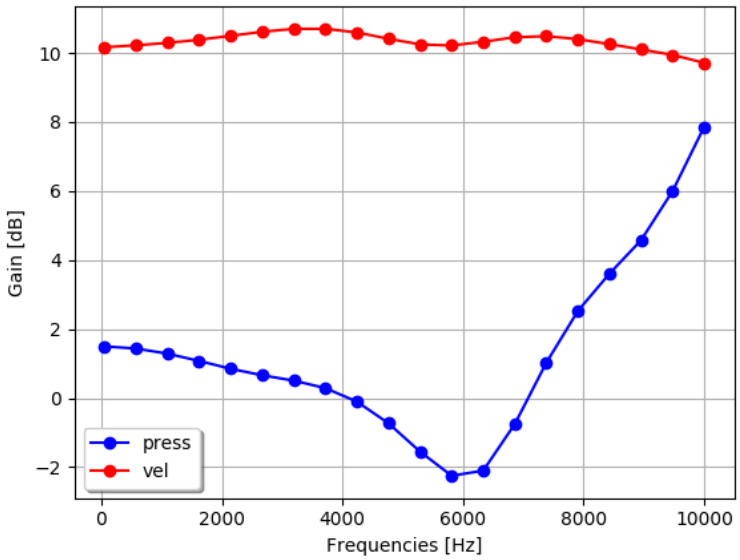


Figure 6.16: Frequency response of the SPL and PVL gains, which have been computed with the P-U probe with a diameter of 0.75 inch.

6.3.2 Diameter of one inch

The optimal values for the absorption coefficient of the PML are written in Table 6.3, while the geometry is plotted in Figure 6.17.

Frequency [Hz]	Optimal value [s/rad]
50	1600
150	4500
250	7100
500	12700
750	17300
1000	20900
2000	32400
4000	44700
6000	48500
8000	51600
10000	48400

Table 6.3: Optimal values of the PML absorption coefficient for 11 different frequencies. The computational domain is the exterior of the P-U probe with a diameter of 1 inch, surrounded by a PML of thickness 13 mm located at 13 mm from the probe.

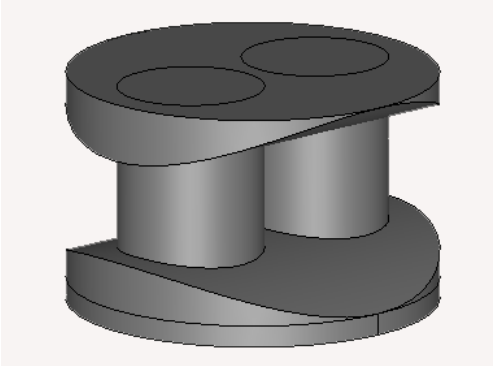


Figure 6.17: Geometry of the P-U probe with a diameter of one inch.

As it was expected from the numerical results obtained with the 0.75 inch probe, the results are far from being optimal or feasible. In Figure 6.18, not only the gain associated to the sound pressure level, but also the gain associated with the particle velocity level are not flat. For this reason, the probe with this geometry is neither optimal nor feasible to perform reliable measurements.

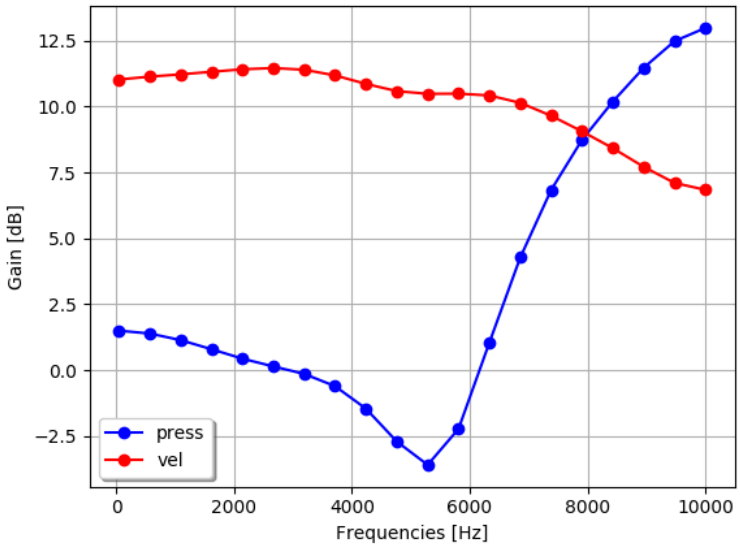


Figure 6.18: Frequency response of the SPL and PVL gains, which have been computed with the P-U probe with a diameter of 1 inch.

Taking into account the numerical results obtained for the 0.75 and 1 inch designs, it could be concluded that it is not feasible to design a P-U probe with larger diameter than the original one.

6.4 Comparison with experimental measurements

Finally, the numerical results are compared with respect to the experimental measurements obtained with the original probe. With this purpose, two identical sensors have been used. The first sensor measures the PVL without being surrounded by an structure, while the second one is introduced into a P-U probe. The use of a pair of sensors makes possible to compute the gain of the PVL level. The computation of the PVL gain from the experimental measurements is performed by using an in-house Microflown software. This software allows the user to made acoustic measurements and to analyse the obtained results. Then, to compare these experimental results with respect to the numerical ones, the `Python` code described in Section 4.4 is used. In

what follows, it was described the steps to compare both results.

The first step consists in checking the calibration of both measurements, this is, to check that both sensors are exactly at the same locations. In Figure 6.19 it can be observed that the signals of both sensors are quite similar and the calibration error is almost null in the frequency band of interest, lower than 10000 Hz.

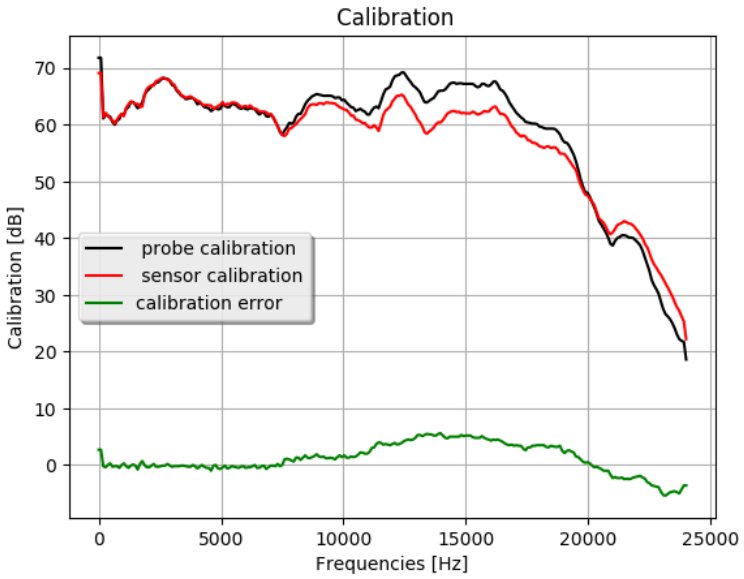


Figure 6.19: Comparison of the calibration of the experimental measurements.

Secondly, the PVL gain of the experimental measurements is computed. Due to the gain associated to the sensor packing is included in the calibration, it has been necessary to subtract it from the probe to obtain an accurate gain estimation. Both, numerical and experimental PVL gains are plotted in Figure 6.20. Finally, the error between both gains is computed. It is shown in Figure 6.20, that the error is around 2 dB. These different values on the PVL gain could be explained thanks to the fact that the probe design used in the numerical simulations does not include the geometrical details regarding the sensor location. In fact, it is well-known by the company that the sensor geometry increases the PVL gain in 2 dB approximately.

Figure 6.20: Comparison of the PVL gains of the experimental measurements and the the PVL gains obtained from numerical simulations.

Chapter 7

Conclusions

Nowadays, the so-called new industry 4.0 is rising. More and more technological companies are including modelling strategies and simulation procedures in their work chains, and not only in the industry. Furthermore, the company goals are focused on designing novel products faster, cheaper and safer. Consequently, mathematical modelling and numerical simulation tools are becoming important key tools in a variety of applications (see for instance [10]). Currently, it is possible to understand the behaviour of a product or a complex process without building it or letting it happens.

Throughout the present work, it was demonstrated how the numerical simulation can describe accurately the vibro-acoustic behaviour of P-U probe. More precisely, the comparison of the numerical results and the experimental measurements shows that the numerical simulation techniques used in this master thesis provide an accurate procedure.

Regarding the conclusions with respect to the optimization of the P-U probe, it is also possible to infer the smaller the space between pillars, the higher the particle velocity is. Moreover, if the pressure and velocity fields are focused on the blank space between the pillars, the increment is higher, as it was expected by the company.

Three different mathematical models were described in this master thesis. Two of them (with and without viscosity) were stated in a bounded domain. However, the third one was stated in an unbounded domain. Comparing the size of the probe with respect to the surrounding fluid

domain in the experimental measurements, the fluid seems to be unbounded. For this reason, even if the viscosity is not taking into account in this last model, the problem stated in an unbounded domain is a suitable choice to simulate the behaviour of a P-U probe. In addition, due to the facts that the numerical simulation involves a reduced number of wavelengths (below 10000 Hz in a computational domain size around ten millimetres) and that the viscous coefficient in the air is quite small, it is possible to conclude that the numerical results obtained in an inviscid mathematical model will be accurate.

Furthermore, the model stated in an unbounded domain involves the Sommerfeld radiation condition, which describes the free field conditions. However, in Finite Element Methods it is quite hard to implement this kind of conditions. Therefore it has to be replaced for a new one. The technique used to simulate an unbounded domain was the Perfect Matched Layer technique. The PML method mimics accurately the free field conditions, by surrounding the domain with some sponge layers, which absorb the reflected waves that go to the infinity.

Finally, the implementation and validation of the Finite Element Method has to be done. The numerical results of the test problems (see Chapter 5) show that the numerical results obtained with the `FEniCS` computer code are reliable and accurate. In all cases, the relative errors between the exact solution and the approximated solutions was lower than 10% for all the frequency range of interest, which illustrate the accuracy and validate the implementation of the FEM code.

Bibliography

[1] A. Bermúdez, L. M. Hervella-Nieto, and A. Prieto. *Métodos Matemáticos en Acústica*. Lectures notes from Acoustic course, Master in Industrial Mathematics, 2016.

[2] CEA, EDF, and OpenCascade. Salome: The open source integration platform for numerical simulation. http://www.salome-platform.org/user-section/salome-tutorials. [Online; last access: 20 November 2017].

[3] J. D. Hunter. Matplotlib: A 2D graphics environment. *Computing In Science & Engineering*, 9(3):90–95, 2007.

[4] F. Ihlenburg. *Finite Element Analysis of Acoustic Scattering*. Springer-Verlag, New York, 1998.

[5] F. Jacobsen and P. M. Juhl. *Fundamentals of General Linear Acoustics*. Wiley, United Kingdom, 2013.

[6] A. Logg, K.-A. Mardal, G. N. Wells, et al. *Automated Solution of Differential Equations by the Finite Element Method*. Springer, 2012.

[7] Microflown-Technologies. Corporate profile. http://www.microflown.com/about-us/corporate-profile.html, . [Online; last access: 3 January 2018].

[8] Microflown-Technologies. The Microflown E-Book. http://www.microflown.com/library/books/the-microflown-e-book.htm, . [Online; last access: 3 January 2018].

[9] C. Morfey. *Dictionary of Acoustics*. Academic Press, United Kingdom, 2001.

[10] P. Quintela, P. Barral, D. Gómez, F. Pena, J. Rodríguez, P. Salgado, and M. Vázquez-Méndez, editors. *Progress in Industrial Mathematics at ECMI 2016*, 2017. Springer.

[11] S. van der Walt, S. C. Colbert, and G. Varoquaux. The NumPy Array: A Structure for Efficient Numerical Computation. *Computing in Science & Engineering*, 13(2):22–30, 2011.