Title
Absorbing Boundary Conditions for Scalar and Elastic Waves in the Time-Domain

Author
Sagiyama, Koki

Publication Date
2013

Peer reviewed|Thesis/dissertation

Absorbing Boundary Conditions
for Scalar and Elastic Waves in the Time-Domain

by

Koki Sagiyama

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Engineering-Civil and Environmental Engineering

in the

Graduate Division

of the

University of California, BERKELEY

Committee in charge:

Professor Sanjay Govindjee, Chair
Professor James W. Rector
Professor Per-Olof Persson

Fall 2013

Absorbing Boundary Conditions
for Scalar and Elastic Waves in the Time-Domain

Abstract

Absorbing Boundary Conditions
for Scalar and Elastic Waves in the Time-Domain

by

Koki Sagiyama

Doctor of Philosophy in Engineering-Civil and Environmental Engineering

University of California, BERKELEY

Professor Sanjay Govindjee, Chair

Absorbing boundary conditions are a requisite element of many computational wave propagation problems. With our main motivation being the anchor loss simulations of Microelectromechanical Systems (MEMS) in three dimensions, efficient time-domain absorbing boundary conditions which do work well for elastodynamics are in demand. In this work we investigate three classes of absorbing boundary conditions which we believe are promising, viz., perfectly matched layers (PMLs), perfectly matched discrete layers (PMDLs), and high-order absorbing boundary conditions (HOABCs). We first devise a PML formulation on spherical domains which is particularly suited for the simulation of a large class of MEMS-resonator systems. What distinguishes our original PML formulation from most existing PML formulations is that it works with standard numerical solvers such as discontinuous Galerkin methods on unstructured meshes and that it allows for a natural application of Neumann boundary conditions on traction-free surfaces. It is also of significant importance in large three-dimensional problems that our formulation has fewer number of degrees of freedom than any existing PML formulations. We demonstrate the applicability of our spherical PML formulation to large problems via a simulation of a three-dimensional double-disk resonator in the time-domain using a discontinuous Galerkin method and an explicit fourth-order Runge-Kutta method. PMDL methods and HOABC methods are alternatives to PML methods, which in the context of the scalar wave equation surpass PML methods in their overall behavior. Unfortunately, their mathematical properties are not as well understood in the context of elastodynamics and, at least in a certain setting, they are known to result in unstable systems. Due to its involved nature, we focus in this work on the analysis of PMDLs/HOABCs for the scalar wave equation and prove several useful identities which will also be useful in the analysis of PMDLs/HOABCs for elastodynamics.

To my parents,

Contents

List of Figures

List of Tables

Acknowledgments

I would like to thank my advisor Professor Sanjay Govindjee for advising me for the entire period I studied for a Ph.D. in Berkeley. Without him, I would not have been able to achieve what I have. The research topic he chose was interesting and has given me great pleasure in my life here. He was patient even during periods I was not able to make progress and kept providing me invaluable pieces of idea with his perspective and sophisticated thought. I have always been proud of my being one of his students and I can not thank him too much.

I would like to thank Professor Per-Olof Persson for introducing me to the real world of numerical analysis. Without him, I would not have met one of the most powerful numerical method of discontinuous Galerkin in my life. But for his 3DG software and his warm assistance in my using various computational resources, I would not have been able to develop knowledge and skills to deal with large three-dimensional simulations. I would also like to appreciate his crucial help to resolve the issue of numerical instability I experienced.

I would like to thank Professor James W. Rector for reading my dissertation. He was willing to be one of my dissertation committee members when I asked him to be. I am grateful to him that he read this work from a slightly different perspective as an expert in seismic wave propagations.

I would like to thank Professor Tsuyoshi Koyama. He taught me so many things in my first semester of Ph.D. and I still appreciate all of them. Even though I was having difficulty in understanding things, he was very patient and did not hesitate to spend hours for instructing me. Had it not been for him, I would not have been able to jump into the field of research so smoothly.

I would like to thank Dr. Bradley M. Froehle. The Python interface for the 3DG software he developed helped me so much and, every time I had a trouble with it, he resolved the issue instantly. Without him, I would not have been able to work on the large numerical simulations so efficiently.

I would like to thank all my colleagues I met in Berkeley in my Master's and Ph.D. studies. It has been a great enjoyment for me to spend time with them.

Finally, I would like to thank my family and friends in Japan and in other countries who have kept supporting me in various aspects. But for them, I would not have been able to accomplish my study.

Chapter 1

Introduction

The analysis of wave propagation in an unbounded domain is of interest in many fields, such as geotechnical engineering and electromechanics. Among others, our main motivation is the anchor-loss simulations of Microelectromechanical Systems (MEMS), where a set of resonating bodies is attached to an unbounded substrate via cylindrical posts with small radii. Specifically, we are interested in an accurate evaluation of the quality factor of such three-dimensional devices. Chapter 2 introduces basic concepts of the anchor-loss mechanism and techniques to evaluate the quality factors. For a scalable evaluation, we employ, among others, a transient dynamical approach [22], which requires a numerical simulation of the resonator system.

Due to the unboundedness of the substrate, these problems do not allow for direct application of standard numerical schemes such as finite element methods which require a finite computational domain. We thus require new methodologies or a mapping of the problem to a finite domain.

Within the class of mapped methods, the so-called perfectly matched layer (PML) methods have become popular due to their simplicity of implementation, accuracy, and versatility. In the PML method, the unbounded domain is truncated and an artificial PML region is attached, resulting in a finite computational domain. In a continuous setting, provided that the PML region is of infinite depth, outgoing waves travel into the PML with no spurious reflections back into the physical domain and they rapidly attenuate within the PML domain. While reflections do arise in numerical computations due to discretization effects as well as finite depth effects, relatively shallow PMLs usually suffice to achieve a satisfactory level of accuracy, as has been widely seen in many fields over the last two decades. Chapter 3 focuses on an efficient PML formulation for time-domain elastodynamics on spherical domains in three dimensions, or axisymmetric domains in two dimensions. The above mentioned MEMS-resonator systems emit elastic waves almost spherically into the substrate from the bottom-end of the support posts/anchors. Therefore, it is natural to truncate the substrate spherically around the posts, resulting in a semi-half sphere, since PMLs exhibit best absorption for waves of normal incidence. While most existing PML formulations are developed for finite difference methods, our formulation is compatible with standard Galerkin finite el-

ement methods and discontinuous Galerkin methods on unstructured meshes, thus allowing for applications on complex geometries. Moreover, our formulation introduces fewer number of degrees of freedom than any existing PML formulations and so is computationally more efficient. Furthermore, application of Neumann boundary condition is natural in our formulation in contrast to in most of the existing formulations. Finally, it should also be noted that the absence of corners in the PML domains greatly simplifies the time-domain formulation.

Chapter 4 introduces perfectly matched discrete layers (PMDLs), which is a family of absorbing boundary conditions whose basic idea was formed in an attempt to optimize the parameters present in the PML formulations. It is known that PMDL methods in general achieve better accuracy than PML methods in terms of small reflection. It turned out that PMDL methods were equivalent to the later developed *complete radiation boundary conditions* (CRBCs), which is a focus of Chapter 5, when these formulations are written in first-order forms in terms of temporal and spatial derivatives. PMDLs have a *second-order* counterpart, which is more convenient, e.g., for the scalar wave equation. This second-order formulation of PMDLs is then identified with the second-order formulation of CRBCs to be proposed in this work. Thus all results we obtain for the second-order CRBCs for the scalar wave equation in Chapter 5 carry over to the second-order PMDLs.

High-order absorbing boundary conditions (HOABCs) form another class of mapped methods, which were originally developed for and have been applied to the scaler wave equation. In HOABC methods, the unbounded domain is truncated and special boundary conditions are applied *on* the resulting artificial boundaries. These boundary conditions are defined by a set of auxiliary functions and recursive relations which these auxiliary functions satisfy. These recursions are by nature first-order in time and space. The number of auxiliary functions controls the accuracy of HOABC methods in terms of small reflection from the artificial boundary as well as the computational cost required in numerical simulations. With our ultimate goal being the application of HOABCs to the standard elastodynamic equation, we study in Chapter 5 the application of the most general class of HOABCs, i.e. the CRBCs, to the scalar wave equation. The analysis and observations presented in this chapter is expected to aid one in understanding the behavior of the CRBCs applied to elastodynamics in the future.

HOABC formulations are written in first-order both in time and space in their original form and involve normal derivatives of the auxiliary functions on the boundary. For computational efficiency it is demanded that the auxiliary functions live only *on* the boundary, in which case representations of normal derivatives are not available, and one needs a reformulation of the original first-order systems into one that does not involve normal derivatives. It is known that for most classes of HOABCs one can rewrite the original first-order system as another system which only involves second-order temporal and tangential derivatives. We refer to those resultant formulations as *second-order formulations* in contrast to the original *first-order formulations*. In Chapter 5, we propose for the first time a second-order formulation of the most general class of HOABCs, the CRBCs.

In the context of the scalar wave equation, HOABC methods have been studied analyt-

ically and numerically and their behavior is relatively well understood. For instance it is well known that HOABC methods in general exhibit a sharper reduction of reflection error than PML methods with increasing computational effort. Well-posedness of several classes of HOABC methods have also been proved. In Chapter 5, we also provide several analytical results on the proposed second-order CRBCs.

Finally, due to its involved nature, numerical implementation of HOABC formulations is a formidable task especially when edges and corners exist, which keeps HOABC methods less popular than PML methods. In this work, we also apply our second-order CRBCs to a three-dimensional problem which involves edges and a corner. To our knowledge, this is the first application of HOABCs to this type of problem.

Chapter 2

Anchor loss simulation

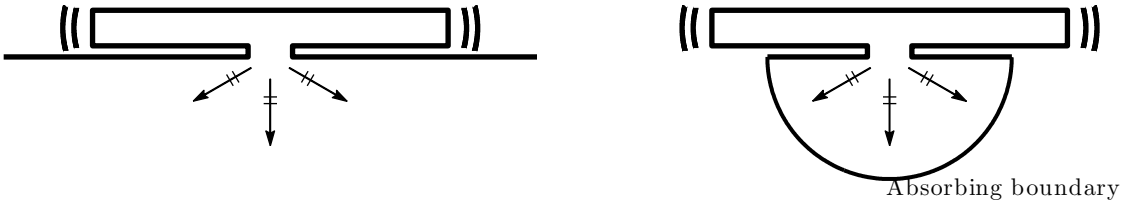
2.1 Introduction

The study of absorbing boundary conditions, perfectly matched layers (PMLs), perfectly matched discrete layers (PMDLs), and high-order absorbing boundary conditions (HOABCs), in this work is among others especially motivated by the *anchor-loss* simulation of high-frequency MEMS-resonator systems. The performance of such devices is standardized by the quality factor, Q, which is defined to be the ratio of the total stored energy to the energy dissipation out of the system per cycle. For an ideal system of no energy dissipation, $Q = +\infty$. In reality, however, there are mechanisms through which energy is lost and anchor-loss is known to be one of major sources of energy dissipation.

Specifically when a resonating disk is attached to an unbounded substrate through a cylindrical post of small radius, energy inside the disk escapes into the unbounded substrate through the post in a form of vibration, or wave energy; see Fig.2.1(a). This example characterizes the anchor-loss mechanism.

For a precise evaluation of the quality factor Q of a resonator, an accurate assessment of the anchor-loss dissipation is a must. In this work we devise and/or apply mathematical methodologies to achieve this goal. The essence of the quality-factor evaluation from a mathematical point of view is to find the eigenvalue of the vibration mode we are interested in. An obvious way to achieve this is to setup a model equation and solve the corresponding eigenvalue problem analytically. Linear elasticity models are often best suited for this type of problem. Unfortunately, analytical solutions for such models are only available for problems with simple geometries and boundary conditions and it is a rare event that we can find one in real problems.

An alternative approach is to solve the corresponding problems numerically, which will be our main focus in this work. A major issue that we encounter in numerically analyzing problems such as the one depicted in Fig.2.1(a) is that the domain is unbounded. Due to the limited computational resources, standard numerical solvers such as finite element methods require finite domains, and therefore they require a methodology to map an unbounded



(a) (b)

Figure 2.1: Sketches of an example resonator on (a) an original unbounded domain (b) a bounded domain with an absorbing boundary.

domain to a bounded one. PMLs, PMDLs, and HOABCs are major families devised for this purpose. These three methodology are all based on the idea of truncating the original unbounded domain into a finite one introducing artificial boundaries, or *absorbing boundaries*, and applying special conditions on or in the vicinity of the absorbing boundaries; see Fig.2.1(b). These special conditions are called *absorbing boundary conditions*. Our analytical model for linear elasticity with absorbing boundary conditions can then be discretized using standard numerical solvers.

One obvious way to find eigenvalues of such a numerical system is to solve the corresponding eigenvalue problem directly using a generic eigenvalue solver. This approach, however, is practical only for systems with a relatively small number of degrees of freedom due to its high memory requirements.

In this chapter, an alternative transient dynamical approach proposed in [22] is introduced. In this method, one extracts eigenvalues from time-series data obtained by directly solving a time-domain problem. This is done by using a harmonic inversion technique utilizing the *filter-diagonalization methods* [46]. Eigenvalues thus obtained are then used to evaluate the quality factor Q. This sequence of procedures, i.e. time-domain simulation followed by harmonic inversion, was demonstrated to scale well in three-dimensional examples in [22].

In Sec.2.2, we quickly introduce a mathematical expression for the quality factor in terms of the eigenvalue of the corresponding mode. In Sec.2.3, we consider a simple one-dimensional system which mimics MEMS-resonator systems and evaluate the quality factor of that system by analytically calculating the eigenvalues. In Sec.2.4, the theoretical background of a harmonic inversion technique to be used with the transient dynamical approach is described. We then revisit the one-dimensional system considered in Sec.2.3 and recompute its eigenvalues by two numerical approaches, i.e. direct computation by a generic eigenvalue solver and the transient dynamical approach. Sec.2.6 concludes.

2.2 Quality factors

There are several mathematical formulas available for computing the quality factor Q. We consistently use the form in which Q is directly related to the eigenvalue $\mathfrak{i}\omega$ corresponding to a vibrational mode of interest, that is:

$$Q = \frac{|\omega|}{2\mathrm{Im}\,(\omega)}. \tag{2.1}$$

Note that if $\mathrm{Im}\,(\omega)$ is positive, or if $\mathrm{Re}\,(\mathfrak{i}\omega)$ is negative, it represents energy dissipation, or damping, of the corresponding mode.

2.3 One-dimensional example: analytical approach

The simplest system that mimics MEMS-resonator systems such as the one sketched in Fig.2.1(a) is the one-dimensional spring-mass-spring-rod system depicted in Fig.2.2(a). In Fig.2.2(a), a mass m is attached to a rigid wall and to an elastic rod of semi-infinite length via springs of stiffness βk and k, respectively. The rod has a density per unit length of ρ and Young's modulus of E. The mass along with the rigid wall and the spring of stiffness βk represents the disk of the resonator in Fig.2.1(a). If we apply, say, impulsive load $f\,(t)$ to the mass, it starts oscillating, which represents the oscillation of the MEMS-disk. Was the mass detached from the elastic rod, there would be no damping in the wall-spring-mass system and, according to (2.1), the quality factor Q of this system would be $+\infty$. However, when it is attached to the semi-infinite elastic rod via the spring, just as a MEMS-disk is attached to an unbounded substrate via a small post, the energy in the wall-spring-mass system escapes into the elastic rod, resulting in a damping in the system. This example characterizes the mechanism of anchor-loss.

In this section, we analytically evaluate the quality factor of this system; we setup an eigenvalue problem for this system and directly compute the eigenvalues, and then Q using (2.1).

The complete time-domain problem for this system is given as:

$$\text{Find } u_r\,(t),\, u\,(x,t) \in L^2\,((0,+\infty)) \text{ such that}$$

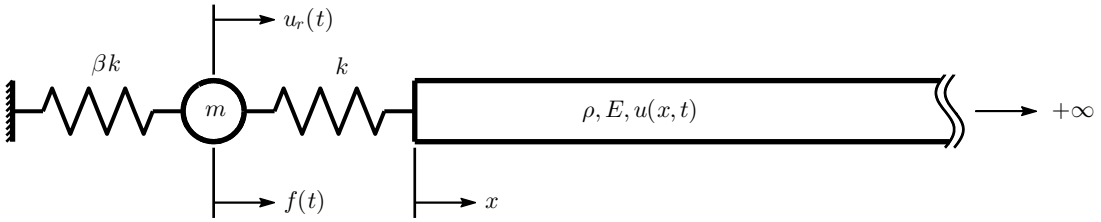
$$m\frac{\partial^2 u_r}{\partial t^2} + k\,((1+\beta)\,u_r - u\,(0,t)) = f\,(t)\,, \tag{2.2a}$$

$$\rho\frac{\partial^2 u}{\partial t^2} - E\frac{\partial^2 u}{\partial x^2} = 0, \quad \text{on } (0,+\infty)\,, \tag{2.2b}$$

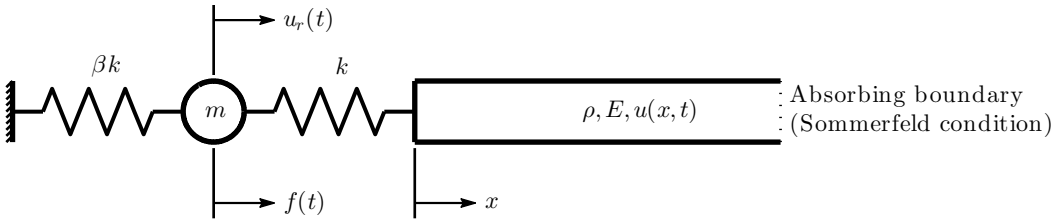
$$k\,(u_r - u\,(0,t)) + E\frac{\partial u}{\partial x}\,(0,t) = 0, \quad \text{on } x = 0, \tag{2.2c}$$

$$u_r\,(0) = 0, \quad \frac{\partial u_r}{\partial t}\,(0) = 0, \tag{2.2d}$$

$$u\,(x,0) = 0, \quad \frac{\partial u}{\partial t}\,(x,0) = 0, \quad \text{on } (0,+\infty)\,, \tag{2.2e}$$



(a)



(b)

Figure 2.2: Problem setup for a spring-mass-spring-rod system on (a) the original unbounded domain (a) a bounded domain with the Sommerfeld radiation boundary condition.

where $u_r(t)$ is the displacement of the mass, $u(x,t)$ is the displacement of the rod as a function of x, and $f(t)$ is a excitation applied to the mass. Eqns. (2.2a) and (2.2b) are equations of motion of the elastic rod and the point mass, respectively, (2.2c) is the interface condition between these two at $x = 0$, and (2.2d) and (2.2e) are the initial conditions. We also define β as:

$$\begin{aligned} \beta &:= 1 - \frac{\alpha^2}{4}, \\ \alpha &:= \sqrt{\frac{mk}{\rho E}}, \end{aligned} \tag{2.3}$$

where α is a dimensionless parameter representing the impedance mismatch between the dynamic systems. It is convenient to first nondimensionalize (2.2) by introducing changes of

variables as:

$$\tilde{x} := \sqrt{\frac{k\rho}{mE}}x, \tag{2.4a}$$

$$\tilde{t} := \sqrt{\frac{k}{m}}t. \tag{2.4b}$$

Using (2.4) in (2.2), one obtains:

$$\begin{aligned}
&\text{Find } \tilde{u}_r\left(\tilde{t}\right),\, \tilde{u}\left(\tilde{x},\tilde{t}\right) \in L^2\left((0,+\infty)\right) \text{ such that}\\
&\frac{\partial^2 \tilde{u}_r}{\partial \tilde{t}^2} + (1+\beta)\,\tilde{u}_r - \tilde{u}\left(0,\tilde{t}\right) = \tilde{f}\left(\tilde{t}\right), && \text{(2.5a)}\\
&\frac{\partial^2 \tilde{u}}{\partial \tilde{t}^2} - \frac{\partial^2 \tilde{u}}{\partial \tilde{x}^2} = 0, \quad \text{on } (0,+\infty), && \text{(2.5b)}\\
&\alpha\left(\tilde{u}_r - \tilde{u}\left(0,\tilde{t}\right)\right) + \frac{\partial \tilde{u}}{\partial \tilde{x}}\left(0,\tilde{t}\right) = 0, \quad \text{on } \tilde{x}=0, && \text{(2.5c)}\\
&\tilde{u}_r(0) = 0, \quad \frac{\partial \tilde{u}_r}{\partial \tilde{t}}(0) = 0, && \text{(2.5d)}\\
&\tilde{u}(\tilde{x},0) = 0, \quad \frac{\partial \tilde{u}}{\partial \tilde{t}}(\tilde{x},0) = 0, \quad \text{on } (0,+\infty), && \text{(2.5e)}
\end{aligned}$$

Now, the corresponding eigenvalue problem is obtained by setting:

$$u_r = u_r e^{\mathrm{i}\omega t} = \bar{u}_r e^{\mathrm{i}\tilde{\omega}\tilde{t}}, \tag{2.6a}$$

$$u = u(x)\, e^{\mathrm{i}\omega t} = \bar{u}(\tilde{x})\, e^{\mathrm{i}\tilde{\omega}\tilde{t}}, \tag{2.6b}$$

and $f = 0$ in (2.5a), (2.5b), and (2.5c), where according to (2.4b):

$$\mathrm{i}\tilde{\omega} := \sqrt{\frac{m}{k}}\mathrm{i}\omega.$$

This then gives us the following eigenvalue problem to be solved for $\mathrm{i}\omega$ and for a nontrivial set of solutions $(\bar{u}_r, \bar{u})$:

$$\begin{aligned}
&\text{Find } \bar{u}_r,\, \bar{u}(\tilde{x}) \in L^2\left((0,+\infty)\right) \text{ such that}\\
&\left(\mathrm{i}\tilde{\omega}\right)^2 \bar{u}_r + (1+\beta)\,\bar{u}_r - \bar{u}(0) = 0, && \text{(2.7a)}\\
&\left(\mathrm{i}\tilde{\omega}\right)^2 \bar{u} - \frac{\partial^2 \bar{u}}{\partial \tilde{x}^2} = 0, \quad \text{on } (0,+\infty), && \text{(2.7b)}\\
&\alpha\left(\bar{u}_r - \bar{u}(0)\right) + \frac{\partial \bar{u}}{\partial \tilde{x}}(0) = 0, \quad \text{on } \tilde{x}=0. && \text{(2.7c)}
\end{aligned}$$

Eqn. (2.7b) is a simple ODE, and one can readily solve it for a general solution:

$$\bar{u}(\tilde{x}) = c^{+} e^{-\mathrm{i}\tilde{\omega}\tilde{x}} + c^{-} e^{+\mathrm{i}\tilde{\omega}\tilde{x}}, \tag{2.8}$$

where c^+ and c^- are constant with respect to x. From (2.6b), one sees that the first and the second terms in (2.8) represent rightward and leftward propagating waves, respectively. Restricting ourselves to the rightward propagating wave, we set $c^- = 0$. We then substitute (2.8) for $\bar{u}$ in (2.7a) and solve for $\bar{u}_r$ to obtain:

$$\bar{u}_r = \frac{1}{(\mathrm{i}\tilde{\omega})^2 + (1+\beta)} c^+. \tag{2.9}$$

Finally we substitute (2.8) and (2.9) for $\bar{u}$ and $\bar{u}_r$ in (2.7c) and obtain a reduced form of the eigenvalue problem:

$$\left[(\mathrm{i}\tilde{\omega})^3 + \alpha\,(\mathrm{i}\tilde{\omega})^2 + (1+\beta)\,(\mathrm{i}\tilde{\omega}) + \alpha\beta\right] c^+ = 0.$$

To have a nontrivial solution, we set:

$$(\mathrm{i}\tilde{\omega})^3 + \alpha\,(\mathrm{i}\tilde{\omega})^2 + (1+\beta)\,(\mathrm{i}\tilde{\omega}) + \alpha\beta = 0, \tag{2.10}$$

and solve for the eigenvalues $\mathrm{i}\tilde{\omega}$. Noting (2.3), we have from (2.10) that:

$$\mathrm{i}\tilde{\omega} = -\frac{\alpha}{2}, \frac{-\alpha \pm \mathrm{i}\sqrt{32 - 9\alpha^2}}{4}. \tag{2.11}$$

The second set of solutions in (2.11) represents vibrating modes if $\alpha < \frac{4\sqrt{2}}{3}$ and non-vibrating modes, or purely damping/growing modes, otherwise. As an illustration of a MEMS-resonator, we are interested in the latter, which gives a meaningful quality factor from (2.1) as:

$$Q = \frac{\sqrt{8 - 2\alpha^2}}{\alpha},$$

which increases as α decreases. The corresponding mode is, from (2.6b) and (2.8), $u = c^+ e^{\mathrm{i}\tilde{\omega}\left(\tilde{t} - \tilde{x}\right)}$, which decays while oscillating with t at fixed x.

In the above, we were able to compute the eigenvalues and the corresponding quality factor analytically by solving the eigenvalue problem in a continuous setting. It will not be the case, however, for general systems in two and three dimensions. In Sec.2.4, we study a harmonic inversion technique with which one can extract eigenvalues from time-series data and, in Sec.2.5, we apply this technique to the same one-dimensional problem to compute the eigenvalues in a discrete setting.

2.4 Harmonic inversion: the filter-diagonalization method

In this section, we study the basic idea of a class of harmonic inversion techniques which employs the *filter-diagonalization method* introduced by Wall and Neuhauser [62] for continuous time-signals and then modified by Mandelshtam and Taylor [46] for discrete ones,

of which we put our focus on the latter. Though presented with minor modifications, our discussion in this section are essentially the same as one found in [46].

The purpose of this study is to extract complex eigenvalues $\mathrm{i}\omega_k$ from a given time-series recorded on an equidistant time grid $t_n = n\Delta t$, especially when the time-series data is rich in frequencies and we are only interested in modes in a certain narrow frequency band.

The time-series data is first assumed to be a superposition of fundamental modes as:

$$c\left(t_n\right) = \sum_{k=1}^{K} d_k e^{-\mathrm{i}\omega_k(\Delta t)n} = \sum_{k=1}^{K} d_k u_k^n, \tag{2.12}$$

where we defined $u_k := e^{-\mathrm{i}\omega_k \Delta t}$. In principle, if we have $2K$ points in the signal, we can solve (2.12) for $2K$ unknowns (d_k, ω_k), or equivalently, (d_k, u_k). In the sequel, we work on an extraction of u_k.

One can rewrite (2.12) in a matrix/vector form as:

$$c\left(t_n\right) = \boldsymbol{d}^{\mathrm{T}} \boldsymbol{\Omega}^n \boldsymbol{d}, \tag{2.13}$$

where

$$\boldsymbol{d} = \left[\sqrt{d_1}, \sqrt{d_2}, \cdots, \sqrt{d_K}\right]^{\mathrm{T}}, \tag{2.14}$$

$$\boldsymbol{\Omega} = \mathrm{diag}\left[u_1, u_2, \cdots, u_K\right], \tag{2.15}$$

Here we introduce an orthonormal basis $\boldsymbol{\Upsilon}$ of $K \times 1$ vectors. Noting that $\boldsymbol{\Upsilon}^{\mathrm{T}}\boldsymbol{\Upsilon} = \boldsymbol{I}$, one can rewrite (2.13) as:

$$c\left(t_n\right) = \boldsymbol{d}^{\mathrm{T}} \boldsymbol{\Upsilon}^{\mathrm{T}} \boldsymbol{\Upsilon} \boldsymbol{\Omega}^n \boldsymbol{\Upsilon}^{\mathrm{T}} \boldsymbol{\Upsilon} \boldsymbol{d} = \boldsymbol{\Phi_0}^{\mathrm{T}} \widehat{\boldsymbol{U}}^n \boldsymbol{\Phi_0}, \tag{2.16}$$

where

$$\boldsymbol{\Phi_0} := \boldsymbol{\Upsilon} \boldsymbol{d}, \tag{2.17}$$

$$\widehat{\boldsymbol{U}} := \boldsymbol{\Upsilon} \boldsymbol{\Omega} \boldsymbol{\Upsilon}^{\mathrm{T}}. \tag{2.18}$$

We apply $\boldsymbol{\Upsilon}_k$ to (2.18) from the right side. From (2.15) and noting orthonormality of $\boldsymbol{\Upsilon}$, one obtains:

$$\widehat{\boldsymbol{U}} \boldsymbol{\Upsilon}_k = u_k \boldsymbol{\Upsilon}_k. \tag{2.19}$$

We then introduce another set of basis vectors $\boldsymbol{\Psi}$, which is not necessarily orthogonal nor normal, such that $\boldsymbol{\Upsilon} = \boldsymbol{\Psi}\boldsymbol{B}$ ($\boldsymbol{\Upsilon}_k = \boldsymbol{\Psi}\boldsymbol{B}_k$) where $\boldsymbol{B}_k$ is a vector of coordinates of $\boldsymbol{\Upsilon}_k$ in the $\boldsymbol{\Psi}$ basis. We substitute this for $\boldsymbol{\Upsilon}$ in (2.19) and premultiply the resultant by $\boldsymbol{\Psi}^{\mathrm{T}}$ from the left side to obtain:

$$\boldsymbol{\Psi}^{\mathrm{T}} \widehat{\boldsymbol{U}} \boldsymbol{\Psi} \boldsymbol{B}_k = u_k \boldsymbol{\Psi}^{\mathrm{T}} \boldsymbol{\Psi} \boldsymbol{B}_k,$$

$$\Leftrightarrow \qquad \boldsymbol{U}\boldsymbol{B}_k = u_k \boldsymbol{S}\boldsymbol{B}_k, \tag{2.20}$$

where

$$\boldsymbol{U} := \boldsymbol{\Psi}^{\mathrm{T}}\widehat{\boldsymbol{U}}\boldsymbol{\Psi}, \tag{2.21}$$

$$\boldsymbol{S} := \boldsymbol{\Psi}^{\mathrm{T}}\boldsymbol{\Psi}. \tag{2.22}$$

Once $\boldsymbol{U}$ and $\boldsymbol{S}$ are known from (2.21) and (2.22) for a set of basis $\boldsymbol{\Psi}$, one can solve the generalized eigenvalue problem (2.20) for u_k (and $\boldsymbol{B}_k$). $\boldsymbol{U}$ and $\boldsymbol{S}$ turn out to be expressed merely in terms of the given time-series data $c\,(t_n)$ for special choices of basis $\boldsymbol{\Psi}$, which justifies the above performed series of manipulations.

Primitive Krylov basis set:
One possible choice of basis is the Krylov basis:

$$\boldsymbol{\Phi}_n = \widehat{\boldsymbol{U}}^n \boldsymbol{\Phi}_0, \quad n = 0, 1, \cdots, M. \tag{2.23}$$

Here for a moment we set $M = K - 1$ for simplicity. Substituting (2.23) for $\boldsymbol{\Psi}$ in (2.21) and (2.22), and using identity (2.16), one readily obtains:

$$\{U\}_{nn'} = \boldsymbol{\Phi}_n^{\mathrm{T}}\widehat{\boldsymbol{U}}\boldsymbol{\Phi}_{n'} = \boldsymbol{\Phi}_0^{\mathrm{T}}\widehat{\boldsymbol{U}}^n\widehat{\boldsymbol{U}}\widehat{\boldsymbol{U}}^{n'}\boldsymbol{\Phi}_0 = \boldsymbol{\Phi}_0^{\mathrm{T}}\widehat{\boldsymbol{U}}^{n+n'+1}\boldsymbol{\Phi}_0 = c\,(t_{n+n'+1})\,, \tag{2.24}$$

$$\{S\}_{nn'} = \boldsymbol{\Phi}_n^{\mathrm{T}}\boldsymbol{\Phi}_{n'} = \boldsymbol{\Phi}_0^{\mathrm{T}}\widehat{\boldsymbol{U}}^n\widehat{\boldsymbol{U}}^{n'}\boldsymbol{\Phi}_0 = \boldsymbol{\Phi}_0^{\mathrm{T}}\widehat{\boldsymbol{U}}^{n+n'}\boldsymbol{\Phi}_0 = c\,(t_{n+n'})\,. \tag{2.25}$$

In (2.24) and (2.25), $\boldsymbol{U}$ and $\boldsymbol{S}$ are given only in terms of the given time-series data. Though the problem statement is simple and one can in principle solve the $K \times K$ generalized eigenvalue problem (2.20), it requires a large amount of computational resources and is impractical when K is large.

Fourier-type Krylov basis set:
As an alternative one can define $\boldsymbol{\Psi}_j$ as a linear combination of Krylov basis vectors defined in (2.23). M in (2.23) is now a parameter, which defines how many basis vectors we use in the linear combination. As coefficients of such linear superposition, we define z_j as the following:

$$z_j = e^{-\mathrm{i}\varphi_j \Delta t}, \quad j = 1, \cdots, J, \tag{2.26}$$

where J is the total number of basis vectors $\boldsymbol{\Psi}_j$ we use and

$$\omega_{min} < \varphi_1 < \cdots < \varphi_j < \cdots \varphi_J < \omega_{max}, \tag{2.27}$$

for minimum and maximum target eigenfrequencies ω_{min} and ω_{max}; we seek eigenvalues such that $\mathrm{Re}\,(\omega_k) \in [\omega_{min}, \omega_{max}]$. $\boldsymbol{\Psi}_j$ is then defined by:

$$\boldsymbol{\Psi}_j = \sum_{n=0}^{M} e^{\mathrm{i}n\varphi_j \Delta t}\boldsymbol{\Phi}_n = \sum_{n=0}^{M} \left(\frac{\widehat{\boldsymbol{U}}}{z_j}\right)^n \boldsymbol{\Phi}_0, \quad j = 1, \cdots, J. \tag{2.28}$$

Note that since $\mathbf{\Psi}_j$ given in (2.28) is a linear combination of the primitive Krylov base vectors, one can readily compute the components of the matrices $\boldsymbol{U}$ and $\boldsymbol{S}$ from (2.21) and (2.22) just as in (2.24) and (2.25).

The implication of the definition of $\mathbf{\Psi}_j$ (2.28) with (2.26) and (2.27) is given in the following. $\mathbf{\Psi}_j$ (2.28) can be written, using definitions of $\widehat{\boldsymbol{U}}$ (2.18) and z_j (2.26) and orthonormality of $\mathbf{\Upsilon}$, as:

$$\mathbf{\Psi}_j = \mathbf{\Upsilon} \boldsymbol{A}_j \mathbf{\Upsilon}^{\mathrm{T}} \mathbf{\Phi}_0, \tag{2.29}$$

where $\boldsymbol{A}_j$ is a diagonal matrix with diagonal entries given by:

$$\{A_j\}_{kk} := \sum_{n=0}^{M} e^{\mathrm{Im}(\omega_k)(\Delta t)n} \cdot e^{-\mathrm{i}(\mathrm{Re}(\omega_k)-\varphi_j)(\Delta t)n}. \tag{2.30}$$

Let us suppose for a moment that $\mathrm{Im}\,(\omega_k) = 0$, i.e. there is no damping in the kth mode. The phase angle appearing in the summation (2.30) is bounded as:

$$0 \leq (\mathrm{Re}\,(\omega_k) - \varphi_j)\,(\Delta t)\,n \leq (\mathrm{Re}\,(\omega_k) - \varphi_j)\,(\Delta t)\,M.$$

If φ_j is close enough to $\mathrm{Re}\,(\omega_k)$, specifically if $\mathrm{Re}\,(\omega_k) - \varphi_j \ll \frac{1}{(\Delta t)M}$, (2.30) gives $\{A_j\}_{kk} \sim M+1$. On the other hand, if $\mathrm{Re}\,(\omega_k) - \varphi_j \gg \frac{1}{(\Delta t)M}$, (2.30) gives $\{A_j\}_{kk} \sim 0$ or 1 on average. Then one has from (2.29):

$$\mathbf{\Psi}_j = (\mathbf{\Upsilon}\boldsymbol{A}_j)\left(\mathbf{\Upsilon}^{\mathrm{T}}\mathbf{\Phi}_0\right) \sim \sum_{k'} w_{k'} \mathbf{\Upsilon}_{k'}, \tag{2.31}$$

where the summation is over k' such that $\mathrm{Re}\,(\omega_{k'})$ is close enough to φ_j and $w_{k'}$ is some weight. This implies that the choice of the linear combination for $\mathbf{\Psi}_j$ in the form of (2.28) along with the definition of z_j (2.26) with φ_j densely chosen in the frequency window $[\omega_{min}, \omega_{max}]$ (2.27) effectively *filters out* the eigenmodes of no interest. In other words, it effectively projects the given time-series data (2.12) onto a signal subspace spanned by $\mathbf{\Upsilon}_{k'}$ with $k' \in \mathcal{S} = \{k \in [0, 1, \cdots, K] : \mathrm{Re}\,(\omega_k) \in [\omega_{min}, \omega_{max}]\}$, which can be seen form (2.17), (2.23), and (2.28) by setting $d_k = 0$ for $k \notin \mathcal{S}$ in (2.14). The sizes of $\boldsymbol{U}$ and $\boldsymbol{S}$ are now $J \times J$ ($J \ll K$) in stead of $K \times K$ as we would have if we used the primitive Krylov basis (2.23) and thus it allows for an efficient computation of the eigenvalues using (2.20).

If $\mathrm{Im}\,(\omega_k) \approx 0$, a similar argument still holds, but as $|\mathrm{Im}\,(\omega_k)|$ increases, elimination of irrelevant modes becomes less effective. Indeed the best accuracy is achieved when $\mathrm{Im}\,(\omega_k) = 0$. However, since the analysis of MEMS-resonator systems only involves those modes with extremely small damping, the use of filter-diagonalization methods is still justified.

Finally, since φ_j is ordered as in (2.27), (2.31) indicates that entries of $\boldsymbol{U}$ and $\boldsymbol{S}$ computed by (2.21) and (2.22) are large around the diagonals and decay out as it goes off the diagonals. Note also that $\boldsymbol{S}$ in practice tends to be close to singular, and standard techniques such as singular value decompositions have to be employed in actual computation of the eigenvalues.

We use in our application the harminv software [32] specifically developed for the harmonic inversion technique with filter-diagonalization methods discussed in this section.

2.5 One-dimensional example: numerical approach

In this section, we employ two numerical approaches to compute the eigenvalues for the one-dimensional system considered in Sec.2.3 and depicted in Fig.2.2(a). We set $\alpha = 1/8$ throughout this section. For both approaches, we first need to perform a spatial discretization of the problem (2.5). The unboundedness of the elastic rod is represented by the Sommerfeld radiation boundary condition:

$$\frac{\partial u}{\partial \tilde{t}} + \frac{\partial u}{\partial \tilde{x}} = 0,$$

on $\tilde{x} = 1$; see Fig.2.2(b). This boundary condition is *exact* for one-dimensional elasticity (2.5b) and it perfectly represents the original unboundedness; see Sec.5.2 for detail. Exact absorbing boundary conditions are not available for general two- and three-dimensional problems and thus approximate boundary conditions such as PMLs, PMDLs, and HOABCs, which are main topics in this work, are to be used instead. We discretize this resulting finite domain $\tilde{x} \in (0, 1)$ into 2^r $(r = 0, 1, 2, 3)$ elements and employ standard piecewise linear interpolation. We then obtain a semi-discrete system of equations:

$$\boldsymbol{M}\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{K}\boldsymbol{U} = \boldsymbol{F}, \tag{2.32}$$

where $\boldsymbol{U}$ is a solution vector, $\boldsymbol{M}$ is a mass matrix, $\boldsymbol{K}$ is a stiffness matrix, and $\boldsymbol{F}$ is a force vector resulting from the excitation $\tilde{f}\left(\tilde{t}\right)$ in (2.5a). To obtain (2.32), we rewrote (2.5) in first order in time by introducing velocity vectors $\tilde{v}_r = \partial \tilde{u}_r/\partial t$ and $\tilde{v} = \partial \tilde{u}/\partial t$.

2.5.1 Direct computation by generic eigenvalue solver

We first directly compute the eigenvalues of the system (2.32) by inverting the mass matrix. The eigenvalues of $-\boldsymbol{M}^{-1}\boldsymbol{K}$ are then computed by the MATLAB eig function for each refinement level r. The computed eigenvalues $\mathrm{i}\omega$ are then compared to the exact eigenvalue $\mathrm{i}\omega_{\mathrm{exact}}$ computed using the exact formula (2.11) with $\alpha = 1/8$. Note that when we numerically compute the eigenvalues of $-\boldsymbol{M}^{-1}\boldsymbol{K}$, we see many modes of no interest. Thus, we choose one whose eigenfrequency, or imaginary part of $\mathrm{i}\omega$, is closest to that of $\mathrm{i}\omega_{\mathrm{exact}}$. The results are summarized in Table 2.1 for each $r = 0, 1, 2, 3$, where we see good convergence of the computed eigenvalues to the exact one with increasing level of refinement. The direct eigenvalue computation, however, is not practical for large problems due to its high memory requirement.

2.5.2 Transient dynamical approach

In this section, we employ the dynamical transient approach proposed in [22] as an alternative which scales well for large three-dimensional problems.

r	$\mathrm{i}\omega$	$\log_{10}\lvert\mathrm{i}\omega - \mathrm{i}\omega_{\text{exact}}\rvert$
0	$-0.03670785 + 1.40830958\mathrm{i}$	-2.212
1	$-0.03253480 + 1.41062072\mathrm{i}$	-2.863
2	$-0.03156611 + 1.41099462\mathrm{i}$	-3.476
3	$-0.03132871 + 1.41107647\mathrm{i}$	-4.081
exact	$-0.03125000 + 1.41110274\mathrm{i}$	

Table 2.1: Eigenvalues computed by the MATLAB eig function for each $r = 0, 1, 2, 3$ and their absolute errors from the exact value.

The main idea is to numerically solve the problem (2.32) for some excitation $\tilde{f}\left(\tilde{t}\right)$ in the time-domain setting, record a time-series data in the phase of free vibration, and apply the harmonic inversion technique discussed in Sec.2.4 to extract eigenvalues of the system.

We define $\tilde{f}\left(\tilde{t}\right)$ as:

$$\tilde{f}\left(\tilde{t}\right) = \begin{cases} 1 - \cos\left(2\pi\tilde{t}\right), & \text{if } 0 < \tilde{t} \leq 1, \\ 0 & \text{if } 1 < \tilde{t}. \end{cases}$$

For each refinement level r, the time-integration of (2.32) is performed using the forward Euler method up to 16 unit-time with uniform time-step of $\Delta t = 1 \cdot 10^{-4}$ and at each time-step we record the mean value of the nodal displacements of the rod to produce a time-series data. A sample time-series is shown in Fig.2.3 for $r = 3$.

We then apply to these time-series the harmonic inversion technique with the filter-diagonalization method using harminv software [32] and extract eigenvalues of the system.

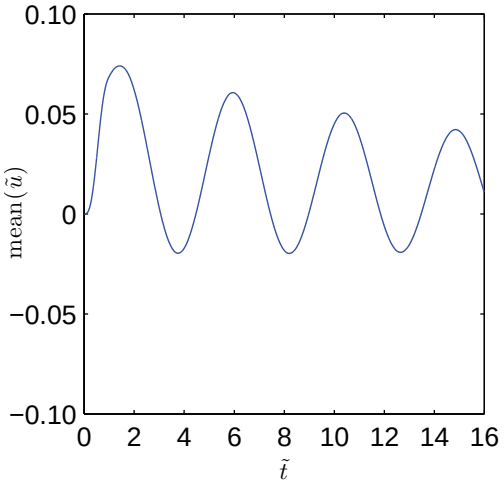


Figure 2.3: A sample time-series obtained for $r = 3$ for the numerical simulation of the one-dimensional system depicted in Fig.2.2(b).

r	$\mathrm{i}\omega$	$\log_{10} \lvert \mathrm{i}\omega - \mathrm{i}\omega_{\text{exact}} \rvert$
0	$-0.03661704 + 1.40831315\mathrm{i}$	-2.218
1	$-0.03242993 + 1.41063165\mathrm{i}$	-2.896
2	$-0.03146627 + 1.41099607\mathrm{i}$	-3.618
3	$-0.03122914 + 1.41107776\mathrm{i}$	-4.488
exact	$-0.03125000 + 1.41110274\mathrm{i}$	

Table 2.2: Eigenvalues extracted by the transient dynamical approach for each $r = 0, 1, 2, 3$ and their absolute errors from the exact value.

To ensure that we work on time-series in the phase of free-vibration, we cut off the first 4 unit-time of the time-series. In using harminv, we specify the frequency window of $[10^{-2}, 10^{2}]$. For each r, harminv extracted only a single eigenvalue. These extracted eigenvalues $\mathrm{i}\omega$ are compared to the exact eigenvalue $\mathrm{i}\omega_{\text{exact}}$ computed using the exact formula (2.11) with $\alpha = 1/8$.

The results are summarized in Table 2.2 for each $r = 0, 1, 2, 3$. Even though we used extremely low-order spatial and temporal approximations, we were still able to extract fairly accurate eigenvalues by the harmonic inversion technique.

2.6 Conclusion

We introduced the anchor-loss mechanism and illustrated three methodologies to evaluate the quality factors of MEMS-resonator systems using our simple one-dimensional model. These three methodologies, i.e. direct analytical approach, direct eigenvalue computation of discrete problems by generic eigenvalue solvers, and transient dynamical approach proposed in [22], demonstrated equally good performance in our one-dimensional problem, but the last one is the only one which is practically applicable to real three-dimensional problems. The transient dynamical approach will be applied for more realistic two- and three-dimensional models of MEMS-resonators with PMLs in Chapter 3.

Chapter 3

Perfectly matched layers

3.1 Introduction

This chapter focuses on an efficient perfectly matched layer (PML) formulation for time-domain elastodynamics on spherical domains in three dimensions, or axisymmetric domains in two dimensions. Our main motivation is the development of an efficient high-fidelity radiation boundary condition suitable for anchor-loss simulations of MEMS-resonator systems [63, 8, 41], where a set of resonating bodies is attached to an unbounded substrate via cylindrical posts with small radii. These types of structures emits elastic waves almost spherically into the substrate from the bottom-end of the support posts/anchors. Therefore, it is natural to truncate the substrate spherically around the posts, resulting in a semi-half sphere, since PMLs exhibit best absorption for waves of normal incidence. On the other hand, for applications such as seismic-wave propagation caused by a fault-rupture which often takes place along a plane, the radius of truncation must be large enough to include all the rupture-surface. Our spherical PML might then not give as good efficiency, since it will also include a large portion of the domain that is less relevant.

The PML method was developed by Berenger [6] for time-domain electromagnetics on an unbounded domain. The electromagnetic fields are decomposed into non-physical components according to their spatial derivatives and artificial damping terms are added to a set of so-called split equations outside the domain of interest such that outgoing waves are absorbed. The addition of artificial damping was later identified by Chew and Weedon as a complex coordinate transformation in the frequency-domain [13], where one could readily inverse-transform the resulting system into the time-domain without producing computationally expensive convolution integrals. The idea of field-splitting followed by a complex transformation of coordinate systems was soon adopted in time-domain elastodynamics written in the first-order velocity-stress form[1] [12, 43, 16, 47]; it was observed, however, that the

[1]It should be noted that the PML formulations thus obtained involve non-physical splitting of the equations and also involve the use of stresses as primary unknowns, both of which can make enforcement of boundary conditions quite difficult.

classical complex transformation functions in [13] tended to produce large spurious reflections into physical domains if waves were strongly evanescent and/or hit the PML with grazing incident angles. To resolve these issues, the Complex-Frequency-Shifted PML (CFS-PML) method was proposed in [40]. This is a class of PMLs that uses more general complex transformation functions than those in [13]. We will denote PML formulations that use the original transformation functions as 'Classical-PMLs' as opposed to CFS-PMLs.

The traditional PML formulation that employs splitting of the fields, however, is not suitable for CFS-PML since it requires convolutions for the inverse-transformation into the time-domain. Furthermore, splitting of the fields introduces two distinct sets of equations on the physical and the PML domains, requiring a special treatment of the interface between the two. To circumvent these issues and employ CFS-PML, unsplit convolution-PML (C-PML) was developed by Roden and Gedney [56] and it has been used as an alternative to the traditional field-splitting PML. C-PML is based on a direct inverse Fourier transform of the complex-transformed equations into the time-domain which produces convolutions. These convolutions are then approximated by the recursive convolution method [44], where they are evaluated by introducing additional unknowns and solving second-order recursions at each time-step. C-PML was first used for time-domain electromagnetics in [56] and since then it has been adopted for time-domain elastodynamics in [17, 34, 48]; it has been demonstrated that CFS-PML used with C-PML exhibits considerable improvement in accuracy against evanescent and grazing waves over Classical PML. Auxiliary-differential-equation PML (ADE-PML) uses differential equations instead of recursions and surpasses C-PML in the point that it can be arbitrarily high-order in time. It was first applied to electromagnetics with Classical PML in [55] and with CFS-PML in [19]. It has also been applied in elastodynamics with CFS-PML in [65, 49, 64].

It is worth noting that the ADE-PML formulations in [65, 49, 64] can be regarded as perturbations of the original velocity-stress formulation of elastodynamics; i.e., the physical domain is governed by the same set of equations as the PML domain with all auxiliary variables set to zero, which leaves the standard velocity-stress formulation in the physical domain and makes the implementation easier. It should also be noted that all these formulations, when applied to three-dimensional elastodynamics, have 9 degrees of freedom (DOFs) in the physical domain, 15 on the faces, 21 on the edges, and 27 at the corners. In addition, extra effort is required to apply traction-free boundary conditions on free-surface PML boundaries as mentioned in [64].

All of these PML formulations for elastodynamics are based on the first-order velocity-stress formulation, which is well-suited for discretization using finite difference methods. However, for computational domains with complex geometries, numerical methods based on fully unstructured meshes such as finite element methods are often preferred. Several efforts have been made to develop such PML formulations in the frequency-domain [4, 7] and in the time-domain [35, 15, 18, 5, 3, 42, 38, 39]. Among the time-domain formulations, those developed in [5, 3, 38, 39] possess a strong advantage over others including the ones based on the velocity-stress formulation. By not decomposing the divergence operator, they yield explicit boundary traction integrals in their weak formulations as a result of the divergence

theorem. Thus Neumann boundary conditions can be naturally applied. This makes the implementation considerably easier when an unbounded half-space is to be truncated by a PML and a traction-free boundary condition has to be applied on a surface of the PML. However, a common drawback of these formulations is that they require a large number of auxiliary nodal history/memory variables. For example, the formulation in [3] applied on a three-dimensional rectangular domain requires memory for displacement, velocity, strain, strain-history, stress-history, history of stress-history, and, at corners, displacement-history, for a total of 33 DOFs at a corner region in the PML and 30 DOFs elsewhere in the PML domain. Further, these formulations have two distinct structures in the physical and the PML domains, which necessitates a special treatment of the interface between the two.

Here, we present a PML formulation that is developed for three-dimensional domains which are truncated with a spherical boundary in which a complex-coordinate transformation is performed *solely* in the radial direction. Although transformations are usually done in directions parallel to the Cartesian coordinate axes, for many applications it is advantageous to do it along radial axes since this does not involve edges or corners which require specialized treatments in the time-domain. Our formulation is based on the frequency-domain formulation presented in [7] and it is compatible with standard finite element methods and discontinuous Galerkin methods on unstructured meshes. The formulation uses only 6 DOFs (displacements and velocities) in the physical domain and 12 DOFs (displacements, velocities, history of displacements, and auxiliary stress components) in the PML domain for three-dimensional elasticity written as a first-order system in time. The physical domain is governed by the same equation as in the PML domain with the auxiliary variables turned off, as is the case in [65, 49, 64]. Physical traction-free boundary conditions are naturally applied on PML surfaces as in [5, 3, 38, 39]. Moreover, using a discontinuous Galerkin method with second-derivatives treated based on the Compact Discontinuous Galerkin (CDG) scheme, the mass matrices can be explicitly inverted with small computational effort which enables the use of explicit time-integrators such as explicit Runge-Kutta methods without the loss of accuracy that typically accompanies explicit time stepping methods that employ traditional mass lumping schemes. The complex transformation functions used in our formulation lie between Classical-PML and CFS-PML. Although it is less general than CFS-PML, our examples show its high ability to absorb quite complex waves. In our numerical examples, we observe long time exponential error growth on coarse meshes, an advective instability, for the straight-forward Galerkin discretization using the CDG scheme. To rectify this, we propose an artificial viscosity based approach to stabilize the formulation and we show that this stabilization is only required on coarse meshes.

We derive our new formulations in Sec.3.2, explain the discontinuous Galerkin discretization procedure in Sec.3.3, and demonstrate its accuracy and versatility through examples in Sec.3.4.

3.2 PML for elastodynamics

3.2.1 Basic concept

In this section, we briefly introduce the concept of PMLs through a one-dimensional problem in the frequency-domain using the $e^{\mathrm{i}\omega t}$ convention ($\mathrm{i} = \sqrt{-1}$). Consider the vibration of a semi-infinite ($x \geq 0$) string on an elastic base [23] with a source located at $x = 0$ and suppose we are interested in the solution for $x \in [0, x_0]$. An approximate solution to this outgoing-wave problem is obtained by solving a PML system truncated at $x = x_{pml} > x_0$:

$$\begin{aligned}
&\text{Find } u(x) \text{ on } x \in (0, x_{pml}) \text{ such that:} \\
&(\mathrm{i}\omega)^2 \rho u - T\frac{d^2u}{d\tilde{x}^2} + ku = 0, && \text{(3.1a)} \\
&u(0) = 1, && \text{(3.1b)} \\
&u(x_{pml}) = 0, && \text{(3.1c)}
\end{aligned}$$

where the complex-valued coordinate

$$\tilde{x}(x) = x + \int_0^x f^e(s)\, ds + \frac{\omega_0}{\mathrm{i}\omega}\int_0^x f^p(s)\, ds, \quad x \in [0, x_{pml}], \tag{3.2}$$

and $f^p(x)$ and $f^e(x)$ are real functions of x defined such that

$$\begin{cases} f^p, f^e = 0 & \text{if } 0 \leq x \leq x_0, \\ f^p, f^e > 0 & \text{if } x > x_0, \end{cases} \tag{3.3}$$

for some constants $x_0 > 0$ and $\omega_0 > 0$. The constant ω_0 is introduced merely for non-dimensionalization. The complex transformation function (3.2) was also used as an alternative to CFS-PML in [5, 3, 38, 39]. There are two possible types of solutions to system (3.1) – a propagating wave solution and an evanescent wave solution depending on the sign of $k - \rho\omega^2$:

$$u(x) = \begin{cases} c^+ \exp\left[-\mathrm{i}\gamma x - \mathrm{i}\gamma\int_0^x f^e(s)ds - \gamma\frac{\omega_0}{\omega}\int_0^x f^p(s)ds\right] \\ \qquad + c^- \exp\left[+\mathrm{i}\gamma x + \mathrm{i}\gamma\int_0^x f^e(s)ds + \gamma\frac{\omega_0}{\omega}\int_0^x f^p(s)ds\right] & \text{if } k - \rho\omega^2 < 0, \\ c^+ \exp\left[-\bar{\gamma} x - \bar{\gamma}\int_0^x f^e(s)ds - \bar{\gamma}\frac{\omega_0}{\mathrm{i}\omega}\int_0^x f^p(s)ds\right] \\ \qquad + c^- \exp\left[+\bar{\gamma} x + \bar{\gamma}\int_0^x f^e(s)ds + \bar{\gamma}\frac{\omega_0}{\mathrm{i}\omega}\int_0^x f^p(s)ds\right] & \text{if } k - \rho\omega^2 > 0, \end{cases}$$

where we define γ and $\bar{\gamma}$ as:

$$\frac{k - \rho\omega^2}{T} := -\gamma^2 := \bar{\gamma}^2.$$

Note that because of definition (3.3), $u(x)$ on $0 \leq x < x_0$ coincides with the solution to the original half-space problem we are trying to model provided that $c^+ = 1$ and $c^- = 0$. In

reality, the Dirichlet boundary condition at $x = x_{pml}$ produces a small amount of spurious reflection which pollutes the solution on $0 \leq x < x_0$. The amount of reflection due to the termination of the PML is quantified by a reflection coefficient $r_{\text{termination}}$ defined as the ratio $|c^-/c^+|$:

$$r_{\text{termination}} = \begin{cases} \exp\left[-2\gamma\frac{\omega_0}{\omega}\int_0^{x_{pml}} f^p(s)ds\right] & \text{if } k - \rho\omega^2 < 0, \\ \exp\left[-2\bar{\gamma}x_{pml}\right] \cdot \exp\left[-2\bar{\gamma}\int_0^{x_{pml}} f^e(s)ds\right] & \text{if } k - \rho\omega^2 > 0. \end{cases} \tag{3.4}$$

From this it is clear that $f^p(x)$ and $f^e(x)$ control the absorption of propagating and evanescent waves, respectively, and $r_{\text{termination}} \to 0$, and thus $c^+ \to 1$ and $c^- \to 0$, as one increases $f^p(x)$, $f^e(x)$, and/or x_{pml}. In a continuous setting, the reflection due to termination can be made arbitrarily small with no additional computational effort by increasing $f^p(x)$ and/or $f^e(x)$, but in approximate numerical computations another type of reflection arises due to the spatial discretization, denoted by $r_{\text{discretization}}$. With rapid changes of $f^p(x)$ and $f^e(x)$ in space, $r_{\text{discretization}}$ increases. Therefore, one has to find a compromise between these competing effects in order to minimize the *total* reflection in an actual numerical solution.

For later application we rewrite Eqn. (3.1a) in terms of x to obtain a total system:

Find $u(x)$ on $x \in (0, x_{pml})$ such that:

$$(\mathrm{i}\omega)^2 \rho u - T\frac{1}{s}\frac{d}{dx}\left(\frac{1}{s}\frac{du}{dx}\right) + ku = 0, \tag{3.5a}$$

$$u(0) = 1, \tag{3.5b}$$

$$u(x_{pml}) = 0, \tag{3.5c}$$

where

$$s(x) = 1 + f^e(x) + \frac{\omega_0}{\mathrm{i}\omega} f^p(x). \tag{3.6}$$

This PML system is the actual form that one discretizes for numerical solutions.

3.2.2 General formulation

Next, we present a general formulation of PMLs for time-harmonic elasticity following [7]. This formulation will be the starting point for our new developments. We consider the problem of elasticity on an unbounded domain $(x_1, x_2, x_3) \in \Omega_\infty$ in which we are interested in the solution on $\Omega_0 \subset \Omega_\infty$, where $\text{vol}(\Omega_0) < \infty$. Ω_∞ is truncated and a PML region is attached to the artificial truncation boundary, producing a finite computational domain $\Omega \supset \Omega_0$. The problem statement is given as:

Find $\boldsymbol{u}(x_1, x_2, x_3)$ for $(x_1, x_2, x_3) \in \Omega$ such that:

$$(\mathrm{i}\omega)^2\rho\boldsymbol{u} - \tilde{\boldsymbol{\nabla}} \cdot \tilde{\boldsymbol{\sigma}}^T = \boldsymbol{f}, \tag{3.7a}$$

$$\tilde{\boldsymbol{\sigma}} = \mathsf{C} : \tilde{\boldsymbol{\epsilon}}, \tag{3.7b}$$

$$\tilde{\boldsymbol{\epsilon}} = \frac{1}{2}[\tilde{\boldsymbol{\nabla}}\boldsymbol{u} + \tilde{\boldsymbol{\nabla}}\boldsymbol{u}^T], \tag{3.7c}$$
$$\boldsymbol{u} = \bar{\boldsymbol{u}} \quad \text{on } \partial\Omega_u,$$
$$\tilde{\boldsymbol{\sigma}}^T\tilde{\boldsymbol{n}} = \tilde{\boldsymbol{t}} \quad \text{on } \partial\Omega_t,$$

where $\partial\Omega_u \cup \partial\Omega_t = \partial\Omega$ and $\partial\Omega_u \cap \partial\Omega_t = \phi$, $\tilde{\boldsymbol{n}}$ is the outward normal to $\partial\Omega$, C is the stiffness tensor, and $\tilde{\boldsymbol{\nabla}}$, $\tilde{\boldsymbol{\epsilon}}$ and $\tilde{\boldsymbol{\sigma}}$ are the gradient operator, the strain tensor, and the stress tensor in the $\tilde{x}$-coordinate system. Analogous to Eqn. (3.2), a complex-valued transformation of the coordinate system is defined with a set of functions χ_i as:

$$\tilde{x}_i = \chi_i(x_1, x_2, x_3), \quad i = 1, 2, 3, \tag{3.8}$$

where $\chi_i = x_i$ for $(x_1, x_2, x_3) \in \Omega_0$. We denote $\tilde{\Omega} = \{(\tilde{x}_1, \tilde{x}_2, \tilde{x}_3) : (x_1, x_2, x_3) \in \Omega\}$. The Jacobian of the transformation (3.8) is denoted by $\boldsymbol{\Lambda}$ so that

$$\tilde{\boldsymbol{V}} = \boldsymbol{\Lambda}\boldsymbol{V}, \tag{3.9}$$

where $\boldsymbol{V}$ is a tangent vector in $\mathbb{R}^3$ and $\tilde{\boldsymbol{V}}$ is its image under the coordinate transformation (3.8). $\boldsymbol{\Lambda}$ is assumed to be everywhere continuous and everywhere invertible for $(x_1, x_2, x_3) \in \Omega$.

As was done in Sec. 3.2.1, Eqn. (3.7a) is rewritten in terms of x_i. To this end, it is simpler to rewrite Eqn. (3.7a) in a weak form by applying a test function $\boldsymbol{w}$ and integrating the equation over $\tilde{\Omega}$. Applying the divergence theorem, this gives:

$$(\mathrm{i}\omega)^2 \int_{\tilde{\Omega}} \rho\boldsymbol{w} \cdot \boldsymbol{u} d\tilde{\Omega} + \int_{\tilde{\Omega}} \tilde{\boldsymbol{\nabla}}\boldsymbol{w} : \tilde{\boldsymbol{\sigma}}^T d\tilde{\Omega}$$
$$- \int_{\tilde{\Gamma}} \boldsymbol{w} \cdot \tilde{\boldsymbol{\sigma}}^T\tilde{\boldsymbol{n}} d\tilde{\Gamma} = \int_{\tilde{\Omega}} \boldsymbol{w} \cdot \boldsymbol{f} d\tilde{\Omega}, \quad \forall \boldsymbol{w}, \tag{3.10}$$

where $\tilde{\Gamma} = \partial\tilde{\Omega}$. Given the transformation rules for volumes, $d\tilde{\Omega} = \det\boldsymbol{\Lambda} d\Omega$, and for gradient operators,

$$\tilde{\boldsymbol{\nabla}}\boldsymbol{u} = \boldsymbol{\nabla}\boldsymbol{u} \cdot \boldsymbol{\Lambda}^{-1}, \tag{3.11}$$

Eqn. (3.10) can be transformed into the x-coordinate system as:

$$(\mathrm{i}\omega)^2 \int_{\Omega} \rho\boldsymbol{w} \cdot \boldsymbol{u} \det\boldsymbol{\Lambda} d\Omega + \int_{\Omega} \left(\boldsymbol{\nabla}\boldsymbol{w} \cdot \boldsymbol{\Lambda}^{-1}\right) : \tilde{\boldsymbol{\sigma}}^T \det\boldsymbol{\Lambda} d\Omega$$
$$- \int_{\Gamma} (\det\boldsymbol{\Lambda})\boldsymbol{w} \cdot \tilde{\boldsymbol{\sigma}}^T\boldsymbol{\Lambda}^{-T}\boldsymbol{n} d\Gamma = \int_{\Omega} \boldsymbol{w} \cdot \boldsymbol{f} \det\boldsymbol{\Lambda} d\Omega, \quad \forall \boldsymbol{w},$$

which by the localization theorem, yields

$$(\mathrm{i}\omega)^2\rho\boldsymbol{u} \det\boldsymbol{\Lambda} - \boldsymbol{\nabla} \cdot [(\det\boldsymbol{\Lambda})\boldsymbol{\Lambda}^{-1} \cdot \tilde{\boldsymbol{\sigma}}]^T = \boldsymbol{f} \det\boldsymbol{\Lambda}, \quad \boldsymbol{x} \in \Omega. \tag{3.12}$$

The coordinate transformation of Eqns. (3.7b) and (3.7c) is straightforward using Eqn. (3.11).

3.2.3 Spherical PML

We now specialize Eqn. (3.12) for a spherical coordinate system. Note that while we develop our formulation in a spherical coordinate system, the actual implementation is done in a standard Cartesian coordinate system. In other words, spherical coordinates are only used to facilitate the theoretical developments.

Since we only apply the complex transformation in the radial direction, we let:

$$(x_1, x_2, x_3) = (r, \theta, \phi),$$
$$(\tilde{x}_1, \tilde{x}_2, \tilde{x}_3) = (\tilde{r}, \theta, \phi).$$

The complex transformation is defined as:

$$\tilde{r} = r + \int_0^r f^e(r)dr + \frac{\omega_0}{\mathrm{i}\omega}\int_0^r f^p(r)dr, \tag{3.13}$$

where

$$\begin{cases} f^p(r), f^e(r) = 0 & \text{if } 0 \le r \le r_0, \\ f^p(r), f^e(r) > 0 & \text{if } r_0 < r, \end{cases}$$

and r_0 is such that $\Omega_0 = \{(r, \theta, \phi) : 0 \le r < r_0\}$. For convenience, we will denote $\int_0^r f^e(r)dr$ and $\int_0^r f^p(r)dr$ as $F^e(r)$ and $F^p(r)$, respectively. We now express Eqns. (3.7b), (3.7c), and (3.12) in the standard orthonormal spherical basis $\{\boldsymbol{e}_r, \boldsymbol{e}_\theta, \boldsymbol{e}_\phi\}$. The simplest way to compute the Jacobian transformation $\boldsymbol{\Lambda}$ is taking the differentials of position vectors, $\boldsymbol{x} = r\boldsymbol{e}_r$ and $\tilde{\boldsymbol{x}} = \tilde{r}(r)\boldsymbol{e}_r$, using the same basis for $\tilde{\boldsymbol{x}}$. Simple differentiations give the relation:

$$d\tilde{\boldsymbol{x}} = \left[\frac{d\tilde{r}}{dr}\boldsymbol{e}_r \otimes \boldsymbol{e}_r + \frac{\tilde{r}}{r}(\boldsymbol{e}_\theta \otimes \boldsymbol{e}_\theta + \boldsymbol{e}_\phi \otimes \boldsymbol{e}_\phi)\right] d\boldsymbol{x},$$

from which one obtains the Jacobian by Def. (3.9) as:

$$\boldsymbol{\Lambda} = \frac{d\tilde{r}}{dr}\boldsymbol{e}_r \otimes \boldsymbol{e}_r + \frac{\tilde{r}}{r}(\boldsymbol{e}_\theta \otimes \boldsymbol{e}_\theta + \boldsymbol{e}_\phi \otimes \boldsymbol{e}_\phi). \tag{3.14}$$

Using relations (3.13), (3.14), and (3.11), Eqns. (3.12), (3.7b), and (3.7c) are combined into the compact equation:

$$(\mathrm{i}\omega)^2 \rho \boldsymbol{u} \det \boldsymbol{\Lambda} - \boldsymbol{\nabla} \cdot [\boldsymbol{\sigma} + \boldsymbol{\Sigma}]^T = \boldsymbol{f} \det \boldsymbol{\Lambda}, \tag{3.15}$$

where $\boldsymbol{\sigma}(\boldsymbol{\nabla u})$ is a conventional stress tensor for elasticity and $\boldsymbol{\Sigma}$ is an *unsymmetric* tensor whose components Σ_{ij} $(i, j = r, \theta, \phi)$ are functions of $\mathrm{i}\omega$ and $(\nabla u)_{kl}$ $(k, l = r, \theta, \phi)$. We note that on Ω_0, $\boldsymbol{\Sigma} = \boldsymbol{0}$ and $\boldsymbol{\Lambda} = \boldsymbol{I}$, the identity tensor. Thus Eqn. (3.15) reduces to the conventional elastic equation in the part of the domain where we desire the solution. For isotropy the stiffness tensor is given by:

$$\mathsf{C}_{ijkl} = \lambda\delta_{ij}\delta_{kl} + \mu\delta_{ik}\delta_{jl} + \mu\delta_{il}\delta_{jk} \quad i, j, k = r, \theta, \phi. \tag{3.16}$$

The components of $\boldsymbol{\Sigma}$ in the spherical basis for isotropy in the frequency-domain are given in Appendix A.1.1.

We now *inverse* transform Eqn. (3.15) into the time-domain. This requires transforming each component of $\boldsymbol{\Sigma}$ as well as the other $\imath\omega$-dependent terms in Eqn. (3.15), i.e., $\boldsymbol{u}\det\boldsymbol{\Lambda}$ and $\boldsymbol{f}\det\boldsymbol{\Lambda}$. We first note that, since $\boldsymbol{f}=\boldsymbol{0}$ in the PML domain, $\boldsymbol{f}\det\boldsymbol{\Lambda}=\boldsymbol{f}$ holds everywhere and we do not need any special treatment for this term. On the other hand, $\boldsymbol{u}\det\boldsymbol{\Lambda}$ contains $\frac{1}{\imath\omega}\boldsymbol{u}$ which results in an inverse transformation integral. This motivates the definition of a vector of auxiliary functions $\boldsymbol{h}$ as:

$$\boldsymbol{h}=\frac{1}{\imath\omega}\boldsymbol{u},$$

which permits the inverse transformation of $\Sigma_{ij}(i=\theta,\phi,\ j=r,\theta,\phi)$ as well as $\boldsymbol{u}\det\boldsymbol{\Lambda}$ without the explicit need for inverse transformation integrals. The expressions for $\Sigma_{rj}(j=r,\theta,\phi)$ have yet another factor $1/(\imath\omega+C_0)$ which produces convolution integrals upon inverse transformation. Among other possibilities, we define three *additional* auxiliary functions:

$$g_1=\left(C_2\frac{1}{\imath\omega}+C_3\frac{1}{\imath\omega+C_0}\frac{1}{\imath\omega}\right)(\nabla u)_{rr}, \tag{3.17a}$$

$$g_2=\left(C_2\frac{1}{\imath\omega}+C_3\frac{1}{\imath\omega+C_0}\frac{1}{\imath\omega}\right)(\nabla u)_{\theta r}, \tag{3.17b}$$

$$g_3=\left(C_2\frac{1}{\imath\omega}+C_3\frac{1}{\imath\omega+C_0}\frac{1}{\imath\omega}\right)(\nabla u)_{\phi r}, \tag{3.17c}$$

which result in a time-domain system to be solved for $\boldsymbol{u}$, $\boldsymbol{h}$, g_1, g_2, and g_3. Auxiliary equations corresponding to these three additional unknowns are obtained by multiplying both sides of Eqns. (3.17) by $\imath\omega+C_0$ and inverse-transforming. The resulting problem is summarized as follows. For convenience, we make the system first-order in time by introducing $\boldsymbol{v}=\dot{\boldsymbol{u}}$, resulting in a system of 12 equations for 12 unknowns:

Find $\boldsymbol{u}$, $\boldsymbol{v}$, $\boldsymbol{h}$, g_1, g_2, g_3 on Ω such that:

$$\dot{\boldsymbol{u}}=\boldsymbol{v}, \tag{3.18a}$$

$$\rho C_4\dot{\boldsymbol{v}}-\boldsymbol{\nabla}\cdot[\boldsymbol{\sigma}+\boldsymbol{\Sigma}]^T=-\rho\left(C_5\boldsymbol{v}+C_6\boldsymbol{u}+C_7\boldsymbol{h}\right)+\boldsymbol{f}, \tag{3.18b}$$

$$\dot{\boldsymbol{h}}=\boldsymbol{u}, \tag{3.18c}$$

$$\dot{g}_1=-C_0g_1+C_2(\nabla u)_{rr}+(C_0C_2+C_3)(\nabla h)_{rr}, \tag{3.18d}$$

$$\dot{g}_2=-C_0g_2+C_2(\nabla u)_{\theta r}+(C_0C_2+C_3)(\nabla h)_{\theta r}, \tag{3.18e}$$

$$\dot{g}_3=-C_0g_3+C_2(\nabla u)_{\phi r}+(C_0C_2+C_3)(\nabla h)_{\phi r}, \tag{3.18f}$$

where

$$\boldsymbol{u}=\bar{\boldsymbol{u}}\quad\text{on }\partial\Omega_u,$$
$$[\boldsymbol{\sigma}+\boldsymbol{\Sigma}]^T\boldsymbol{n}=\boldsymbol{t}\quad\text{on }\partial\Omega_t,$$

and

$$\boldsymbol{\sigma} = \mathsf{C} : \boldsymbol{\epsilon}, \quad (\mathsf{C} \text{ given in Eqn. (3.16)}),$$
$$\boldsymbol{\epsilon} = \frac{1}{2}\left[\boldsymbol{\nabla u} + \boldsymbol{\nabla u}^T\right].$$

$C_0, C_1, \cdots, C_7$ are functions of r but constant in time. Precise expressions are given in Eqns. (A.2) and (A.4) and the expressions for $\Sigma_{ij}(i,j = r,\theta,\phi)$, which are obtained by transforming Eqns. (A.1), are given in Eqns. (A.3); see Appendix A.1.2. As is often done in practice we set $\bar{\boldsymbol{u}} = \mathbf{0}$ on the outer boundary of the PML.

Since Eqn. (3.18b) inherits the structure of the conventional elasticity equation, it can be readily discretized by standard finite element methods or discontinuous Galerkin methods on unstructured meshes. A boundary integral $\int_\Gamma \boldsymbol{w}[\boldsymbol{\sigma} + \boldsymbol{\Sigma}]^T \boldsymbol{n} d\Gamma$ in a weak formulation of Eqn. (3.18b) is naturally treated; e.g. on a traction-free boundary of the PML, this term is simply set zero.

Since it is often more convenient to resolve system (3.18) in a standard Cartesian basis $\{\boldsymbol{e}_x, \boldsymbol{e}_y, \boldsymbol{e}_z\}$ for implementational purposes, the components of $\boldsymbol{\Sigma}$ in the spherical basis $\{\boldsymbol{e}_r, \boldsymbol{e}_\theta, \boldsymbol{e}_\phi\}$ should be transformed according to the basis transformation rules between the two frames as well as the components of $\boldsymbol{\nabla u}$, $\boldsymbol{\nabla v}$, and $\boldsymbol{\nabla h}$.

Note that by defining auxiliary functions as in Eqns. (3.17), one can ensure stability of the system (3.18) in the case of $f_p, F_p = 0$, since it keeps the structure of the standard second-order formulation of elasticity with real coordinate-stretching. Furthermore, since the auxiliary functions (3.17) allow for individual inversions of $(\tilde{\nabla} u)_{rr}$, $(\tilde{\nabla} u)_{\theta r}$, and $(\tilde{\nabla} u)_{\phi r}$, and thus $\tilde{\boldsymbol{\sigma}}$ according to Eqns. (3.7b) and (3.7c), our PML formulation may find possible applications to anisotropic problems without introducing any additional auxiliary functions.

For axisymmetric problems, the spherical PML formulation above can be reduced to an axisymmetric PML formulation; see Appendix A.2 for details.

3.3 Discretization

3.3.1 Spatial discretization

For the numerical discretization of Eqns. (3.18) or its axisymmetric counterpart (A.5), we closely follow the procedure presented in [22], emphasizing the differences due to our PML formulation.

For the spatial discretization of Eqns. (3.18), we use a second-order discontinuous Galerkin method [14] with numerical fluxes according to the Compact Discontinuous Galerkin (CDG) scheme [52]. Let the computational domain Ω be discretized by a set of non-overlapping elements $\mathcal{T}_h = \{K\}$. Introduce the piecewise polynomial finite element spaces V_h^p and Σ_h^p:

$$V_h^p = \{\boldsymbol{v} \in \left[L^2(\Omega)\right]^n : |\boldsymbol{v}|_K \in [\mathcal{P}_p(K)]^n \quad \forall K \in \mathcal{T}_h\},$$
$$\Sigma_h^p = \{\boldsymbol{\tau} \in \left[L^2(\Omega)\right]^{n\times m} : |\boldsymbol{\tau}|_K \in [\mathcal{P}_p(K)]^{n\times m} \quad \forall K \in \mathcal{T}_h\},$$

where $\mathcal{P}_p(K)$ is the space of polynomial functions of degree at most $p \geq 1$ on K, $n = 12$ is the number of solution components, and $m = 3$ is the space dimension. We first rewrite Eqns. (3.18) as a system of first-order equations:

$$\begin{aligned} \boldsymbol{m}\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{\nabla}\cdot\boldsymbol{F}\left(\boldsymbol{u},\boldsymbol{H}\right) &= \boldsymbol{S}\left(\boldsymbol{u},\boldsymbol{H}\right) + \boldsymbol{f}, \\ \boldsymbol{H} - \boldsymbol{\nabla}\boldsymbol{u} &= \boldsymbol{0}, \end{aligned}$$

where (with an abuse of notation) $\boldsymbol{u}$ now represents an array of the unknown functions $\boldsymbol{u}$, $\boldsymbol{v}$, $\boldsymbol{h}$, g_1, g_2, and g_3; $\boldsymbol{m}$, $\boldsymbol{F}$, $\boldsymbol{S}$, and $\boldsymbol{f}$ are tensors representing generalized mass, stresses, sources, and body forces. The finite element formulation is then given as:

Find $\boldsymbol{u}_h \in V_h^p$ and $\boldsymbol{H}_h \in \Sigma_h^p$ such that for all $K \in \mathcal{T}_h$,

$$\begin{aligned} \int_K \left(\boldsymbol{m}\frac{\partial \boldsymbol{u}_h}{\partial t} - \boldsymbol{S}\left(\boldsymbol{u_h},\boldsymbol{H}_h\right)\right)\cdot\boldsymbol{v}d\boldsymbol{x} - \int_K \boldsymbol{F}\left(\boldsymbol{u_h},\boldsymbol{H}_h\right) : \nabla\boldsymbol{v}d\boldsymbol{x} \\ + \int_{\partial K} \widehat{\boldsymbol{t}}_h\left(\boldsymbol{u}_h,\boldsymbol{H}_h\right)\cdot\boldsymbol{v}d\boldsymbol{s} = \int_K \boldsymbol{f}\cdot\boldsymbol{v}d\boldsymbol{x}, \quad \forall\boldsymbol{v}\in\left[\mathcal{P}_p\left(K\right)\right]^n, \quad (3.19) \end{aligned}$$

$$\int_K \boldsymbol{H}_h : \boldsymbol{\tau}d\boldsymbol{x} + \int_K \boldsymbol{u}_h\cdot\left(\boldsymbol{\nabla}\cdot\boldsymbol{\tau}\right)d\boldsymbol{x} - \int_{\partial K}\left(\widehat{\boldsymbol{u}}_h\otimes\boldsymbol{n}\right) : \boldsymbol{\tau}d\boldsymbol{s} = \boldsymbol{0}, \quad \forall\boldsymbol{\tau}\in\left[\mathcal{P}_p\left(K\right)\right]^{n\times m}. \quad (3.20)$$

The problem statement is complete when we specify the boundary fluxes $\widehat{\boldsymbol{t}}_h$ and $\widehat{\boldsymbol{u}}_h$.

To proceed, we first introduce a *switch function* $S_K^{K'} = \{-1, 1\}$ on each internal face e on element K shared by element K'. It is required that $S_K^{K'} = -S_{K'}^K$, and we choose a *natural switch* so that $S_K^{K'} = +1$ if global element number of K is greater than that of K' and $S_K^{K'} = -1$ otherwise.

We first solve Eqn. (3.20) on each element K for numerical gradients $\boldsymbol{H}_h$, which requires a representation of numerical fluxes $\widehat{\boldsymbol{u}}_h$ on each element face $e \in \partial K$. This is done by the standard *up-winding* defined by:

$$\widehat{\boldsymbol{u}}_h = \begin{cases} \boldsymbol{u}'_h & \text{if } S_K^{K'} = +1, \\ \boldsymbol{u}_h & \text{if } S_K^{K'} = -1, \end{cases} \quad (3.21)$$

where $\boldsymbol{u}'_h$ is the numerical solution of element K' on face e. Numerical gradients thus obtained are substituted for $\boldsymbol{H}_h$ in the first and second terms of Eqn. (3.19) and we are left with specifying the numerical tractions $\widehat{\boldsymbol{t}}_h$ in the third term of Eqn. (3.19) on each element face $e \in \partial K$. This is done by first defining a face gradient $\boldsymbol{H}_h^e$ for each face e using a slight modification of Eqn. (3.20) as:

$$\int_K \boldsymbol{H}_h^e : \boldsymbol{\tau}d\boldsymbol{x} + \int_K \boldsymbol{u}_h\cdot\left(\boldsymbol{\nabla}\cdot\boldsymbol{\tau}\right)d\boldsymbol{x} - \int_{\partial K}\left(\widehat{\boldsymbol{u}}_h^e\otimes\boldsymbol{n}\right) : \boldsymbol{\tau}d\boldsymbol{s} = \boldsymbol{0}, \quad \forall\boldsymbol{\tau}\in\left[\mathcal{P}_p\left(K\right)\right]^{n\times m},$$

where

$$\widehat{\boldsymbol{u}}_h^e = \begin{cases} \widehat{\boldsymbol{u}}_h & \text{on face } e \text{ from Eqn. (3.21)}, \\ \boldsymbol{u}_h & \text{otherwise.} \end{cases}$$

The numerical traction $\widehat{\boldsymbol{t}}_h$ on e is then defined as:

$$\widehat{\boldsymbol{t}}_h = C_{11}\left(\boldsymbol{u}'_h - \boldsymbol{u}_h\right) + \begin{cases} \boldsymbol{F}(\widehat{\boldsymbol{u}}_h, \boldsymbol{H}_h^e)\boldsymbol{n} & \text{if } S_K^{K'} = +1, \\ \boldsymbol{F}(\widehat{\boldsymbol{u}}_h, \boldsymbol{H}_h^{e'})\boldsymbol{n} & \text{if } S_K^{K'} = -1, \end{cases}$$

where $\boldsymbol{H}_h^{e'}$ is the face gradient of element K' on e. Note that these tractions can be seen as *down-winding*. The parameter C_{11} was introduced for added stability, which is to be of order $O\left(1/h\right)$ for some length scale h. We set $C_{11} = 200/h_{min}$ in this work, where h_{min} is the smallest edge length of a triangle/tetrahedron.

Finally, one needs to specify fluxes $\widehat{\boldsymbol{u}}_h$ and tractions $\widehat{\boldsymbol{t}}_h$ on the domain boundaries. On Dirichlet boundaries on which we have $\boldsymbol{u} = \bar{\boldsymbol{u}}$, we impose conditions weakly as:

$$\begin{aligned} \widehat{\boldsymbol{t}}_h &= \boldsymbol{F}(\boldsymbol{u}_h, \boldsymbol{H}_h)\boldsymbol{n} - C_{11}\left(\boldsymbol{u}_h - \bar{\boldsymbol{u}}\right), \\ \widehat{\boldsymbol{u}}_h &= \bar{\boldsymbol{u}}, \end{aligned}$$

where $C_{11} > 0$ and we use the same value as on the internal faces. On Neumann boundaries on which we have $\boldsymbol{F}\boldsymbol{n} = \bar{\boldsymbol{t}}$, we impose:

$$\begin{aligned} \widehat{\boldsymbol{t}}_h &= \bar{\boldsymbol{t}}, \\ \widehat{\boldsymbol{u}}_h &= \boldsymbol{u}_h. \end{aligned}$$

Note that the procedure described above can be regarded as a local elimination of the numerical gradients $\boldsymbol{H}_h$. We can now assemble a semi-discrete system of equations for $\boldsymbol{u}_h$ as:

$$\boldsymbol{M}\frac{d\boldsymbol{U}}{dt} = -\boldsymbol{K}\boldsymbol{U} + \boldsymbol{F}, \tag{3.22}$$

where $\boldsymbol{U}$ is a vector of nodal variables, $\boldsymbol{M}$ is the mass matrix, $\boldsymbol{K}$ is the stiffness matrix, and $\boldsymbol{F}$ is the force vector.

The mass matrix $\boldsymbol{M}$ in Eqn. (3.22) from the DG discretization is block-diagonal and can therefore be explicitly inverted to obtain:

$$\frac{d\boldsymbol{U}}{dt} = -\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{U} + \boldsymbol{M}^{-1}\boldsymbol{F}. \tag{3.23}$$

This system of ODEs can be integrated numerically for $\boldsymbol{U}$ using any time-integration scheme. Here we use the standard explicit fourth-order Runge-Kutta method.

3.3.2 Stabilization

We note here that Eqn. (3.18b) involves first-order spatial derivatives of g_1, g_2, and g_3, and Eqns. (3.18d), (3.18e), and (3.18f) involve first-order derivatives of $\boldsymbol{u}$ and $\boldsymbol{h}$. These terms add an advective character to the PML system, which requires special consideration for the numerical discretization. Existing PML formulations for elasticity have similar equations

(see e.g. [49]), but to our knowledge their advective character has attracted little attention. One symptom of advection is numerical instability; for coarse meshes we indeed observe exponential solution growth in time when the PML functions f^e and f^p increase rapidly with r. These instabilities can be removed by refining the mesh or by adding artificial diffusion in the form $-\boldsymbol{\nabla} \cdot \varepsilon \boldsymbol{\nabla} \boldsymbol{v}$ to the left-hand side of Eqn. (3.18b). Here ε is defined as:

$$\varepsilon = \begin{cases} \varepsilon_0 \cos^2\left(\frac{r-r_0}{2\kappa}\pi\right) & \text{if } r_0 - \kappa < r < r_0 + \kappa, \\ 0 & \text{otherwise,} \end{cases}$$

where $\varepsilon_0 \sim h\omega_0(f^e + f^p)_{max}$ and κ is chosen sufficiently large. Addition of artificial diffusion to the axisymmetric problem (A.5) is done in an obvious manner.

We observe in our numerical examples that PML functions which provide good accuracy in the sense of small reflection coefficients are long-time stable and artificial diffusion is not needed in such cases. However, on coarser meshes it is required for stability. Many other strategies have been proposed for stabilization of Galerkin formulations, such as upwinded Petrov-Galerkin schemes [10] or modified numerical fluxes in the discontinuous Galerkin formulation. However, we have found that our simple artificial diffusion approach is sufficient for our applications, and produces well-behaved computations on coarse discretizations.

3.4 Numerical examples

In this section we present several examples in two and three spatial dimensions. We consider an isotropic media with mass density 4.127[Mg/m^3], Young's modulus 139[GPa], and Poisson's ratio 0.28. The domain is discretized using the DistMesh mesh generator [53] for two-dimensional problems, which generates highly regular unstructured triangular meshes, and by netgen for three-dimensional problems utilizing unstructured tetrahedral meshes. For both the two- and the three-dimensional problems, we use the discontinuous Galerkin method with the CDG scheme implemented in the 3DG software package [51] (a general purpose software package for continuous/discontinuous Galerkin methods) for spatial discretization and a fourth order explicit Runge-Kutta method (RK4) for temporal discretization unless otherwise noted.

The PML-complex-transformation functions are assumed to have parabolic profiles:

$$f^p(r) = \beta^p \left(\frac{r - r_0}{r_{pml} - r_0}\right)^2, \quad r_0 < r \leq r_{pml},$$
$$f^e(r) = \beta^e \left(\frac{r - r_0}{r_{pml} - r_0}\right)^2, \quad r_0 < r \leq r_{pml},$$

where β^p and β^e are the values of $f^p(r)$ and $f^e(r)$ on the outer boundary of the PML.

3.4.1 Axisymmetric problems

Stability study

We first study the stability properties of our PML formulation using an axisymmetric test problem. The setup of the problem is given in Fig. 3.1(a). The edge $R = 0$ is the axis of symmetry. The surface $z = 0$ is traction-free, and a uniform Gaussian pressure pulse

$$f(t) = e^{-\left(\frac{t-\alpha}{w}\right)^2}, \quad \alpha = 6, \quad w = 0.01/f_0, \tag{3.24}$$

is applied along the inner hole, where we set $f_0 = 0.0613$[GHz] in this example. The average displacement over the entire domain in the R-direction $\bar{u}_R$ is measured up to $T = 10,000$[ns]. Note that the transit time for a P-wave traveling a distance r_0 is about 0.30[ns].

The domain is truncated at $r_0 = 2.0[\mu\text{m}]$ and surrounded by a PML of depth $r_{pml} - r_0 = 1.5[\mu\text{m}]$, modeling an unbounded domain which extends to $r \to +\infty$. The outer boundary of the PML is clamped. Discretization of the domain is performed by DistMesh; see Fig. 3.1(a). The smallest edge length of a triangle h_{min} is set to $h_{min} = 0.50[\mu\text{m}]$, and polynomials of degree $p = 4$ are used. In this example problem, we set $\omega_0 = 4\pi$ and $\beta^e = 0$ and use three different β^p: $4, 400$, and 4000. We note that in this setting the optimum β^p in the sense of reflection is estimated as $\beta^p \approx 4$ by our PML parameter choosing heuristics; see Appendix A.3. Artificial diffusion of $\varepsilon = 1$ is added as needed for stabilization. For simplicity in this example, we use a continuous Galerkin method in space and a trapezoidal method in time. The time-step Δt is set to 1[ns].

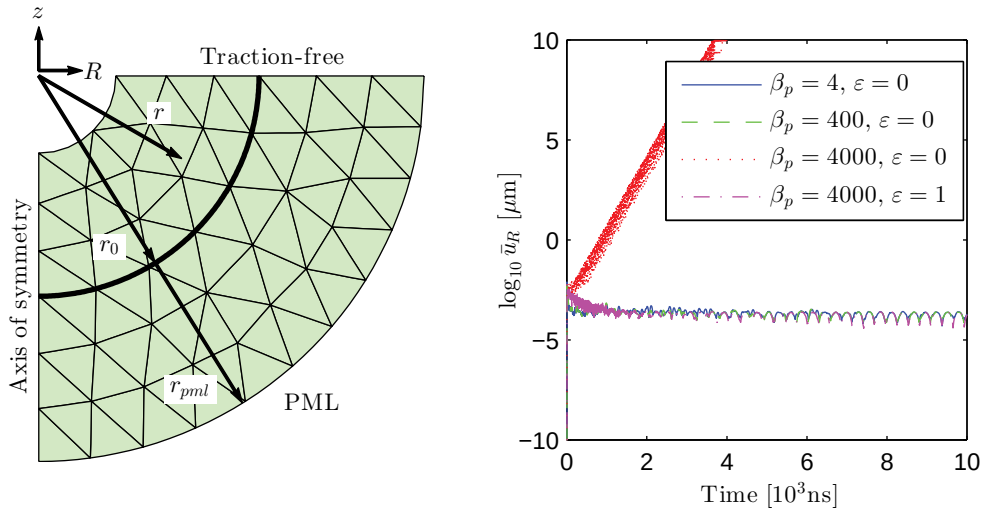
Fig. 3.1(b) shows plots of $\log_{10} \bar{u}_R$ versus time. Without stabilization, the scheme is stable for $\beta^p = 4$ and 400, but unstable for $\beta^p = 4000$. One can see, however, that addition of artificial diffusion removes this instability. This example demonstrates the ability of artificial diffusion to stabilize our numerical scheme when needed. Though it should be noted that $\beta^p = 4000$ is a very high value and the mesh is exceedingly coarse. When using mesh sizes that provide reasonable accuracy and using *optimal* PML parameters (see Appendix A.3), we find that stabilization is unnecessary.

Accuracy study

We next validate the accuracy of our proposed axisymmetric PML formulation. The setup of the problem is given in Fig. 3.2. The edge $R = 0$ is the axis of symmetry. The surface $z = 0$ is traction-free, and uniform Gaussian pressure pulses (3.24) with $f_0 = 0.0613$[GHz] are applied along the edges of the inner holes, generating a complex wave pattern. The domain is truncated at $r_0 = 8.0[\mu\text{m}]$ and surrounded by a PML of depth $r_{pml} - r_0 = 1.5[\mu\text{m}]$.

The domain is again discretized using DistMesh; see Fig. 3.2. The smallest edge length of a triangle is set to $h_{min} = 0.50[\mu\text{m}]$ and again fourth-order elements are used.

Considering the nature of the excitation and the discretization, we set $\omega_0 = 4\pi$ and $\beta^p = 4$ based on a one-dimensional parameter study; see Appendix A.3. We use two different values for β^e, i.e. $\beta^e = 0$ and $\beta^e = 4$, to see its effect on the accuracy.



(a) (b)

Figure 3.1: (a) Problem setup for a stability study. (b) Plots of computed $\bar{u}_R$ versus time on a semi-log scale.

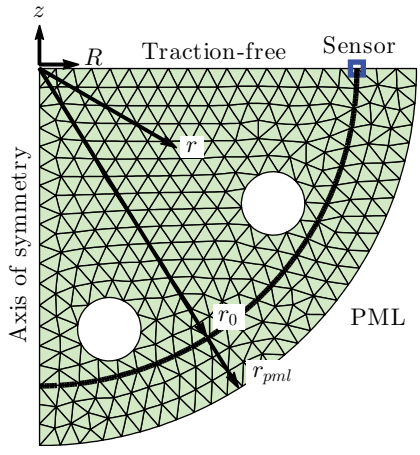


Figure 3.2: Problem setup for an accuracy study: Hole centers are located at $r = 0.85r_0$ and at angles $\pi/12$ and $\pi/3$ from the vertical.

Finally, we use a fixed time-step of $1 \cdot 10^{-3}$[ns] for time-integration. Displacements in the R- and z-directions are recorded up to 10[ns] at the sensor located right on the PML interface $(R, z) = (r_0, 0)$ as depicted in Fig. 3.2, which are compared with reference solutions $u_{\text{reference}}$ computed on an extended domain.

Figs. 3.3(a) and 3.3(b) show plots of computed u_R and u_z against time, respectively, each of which compares four solutions – a reference solution, PML solutions with $(\beta^p, \beta^e) = (4, 0)$ and $(\beta^p, \beta^e) = (4, 4)$, and a solution obtained by applying the classical Lysmer-Kuhlemeyer damper (LK-damper) [45] on $r = r_{pml}$ without any PML elements. The plots clearly demonstrates that both types of PML behave much better than the LK-damper, and show good agreement with the reference solution. Figs. 3.3(c) and 3.3(d) plot absolute differences between the PML solutions and the reference solutions, where a slight improvement can be observed due to the the additional parameter β^e. The relative errors of the PML solutions are about 0.5%, which we regard as satisfactory considering the complex nature of the problem. We note that the value of the parameter β^e is chosen rather arbitrarily and its optimization has yet to be studied.

Axisymmetric resonator

As a final two-dimensional example we test the use of our PML formulation for the computation of the quality factor Q of two MEMS resonators [7, 22]. Good behavior, notwithstanding method, will always rely on a high quality radiation (non-reflecting) boundary condition.

To assess the impact of our PML formulation on this type of problem we consider two axisymmetric resonators as shown in Fig. 3.4. Resonator A as shown in Fig. 3.4(a) has a disk of radius $R_0 = 32[\mu\text{m}]$ attached to a semi-infinite substrate by a cylindrical post of radius $1.0[\mu\text{m}]$ and height $0.70[\mu\text{m}]$. Resonator B as shown in Fig. 3.4(b) has an additional mushroom-like structure on top of the disk. Mushroom caps are an artifact of a popular manufacturing process for MEMS resonators [63, 41]. In our case, the modeled cap has radius $6.5[\mu\text{m}]$ and thickness $2.0[\mu\text{m}]$ and sits on a post of radius $1.0[\mu\text{m}]$ and height $1.0[\mu\text{m}]$. The thickness of the $32[\mu\text{m}]$ disk varies from $1.2[\mu\text{m}]$ to $1.8[\mu\text{m}]$ and the sensitivity of the quality factors to the thickness variation is studied. The surfaces of these resonators are assumed to be traction-free. Each semi-infinite substrate is truncated at radius $r_0 = 8.0[\mu\text{m}]$ and a PML of depth $r_{pml} - r_0 = 1.5[\mu\text{m}]$ is attached surrounding the resulting finite domain. As before, unstructured triangular meshes are generated by DistMesh with $h_{min} = 0.50[\mu\text{m}]$ and element orders of 4 as shown in Fig. 3.4. We set $\omega_0 = 4\pi$ and $(\beta^p, \beta^e) = (4, 4)$ for the PML parameters.

The quality factor corresponding to a mode with eigenvalue $\mathrm{i}\omega$ is defined in Eqn.(2.1). The fundamental angular frequency ω^* of the disks can be estimated as [21]:

$$\omega^*_{\text{estimate}} = 2.04 \frac{c_0}{R_d} \tag{3.25}$$

where $c_0 = 6.045[\mu\text{m/ns}]$ is the plane-stress radial wave speed. We thus find an eigenvalue of $\boldsymbol{M}^{-1}\boldsymbol{K}$ in the semi-discrete system (3.23) whose imaginary part is closest to $\mathrm{i}\omega^*_{\text{estimate}}$

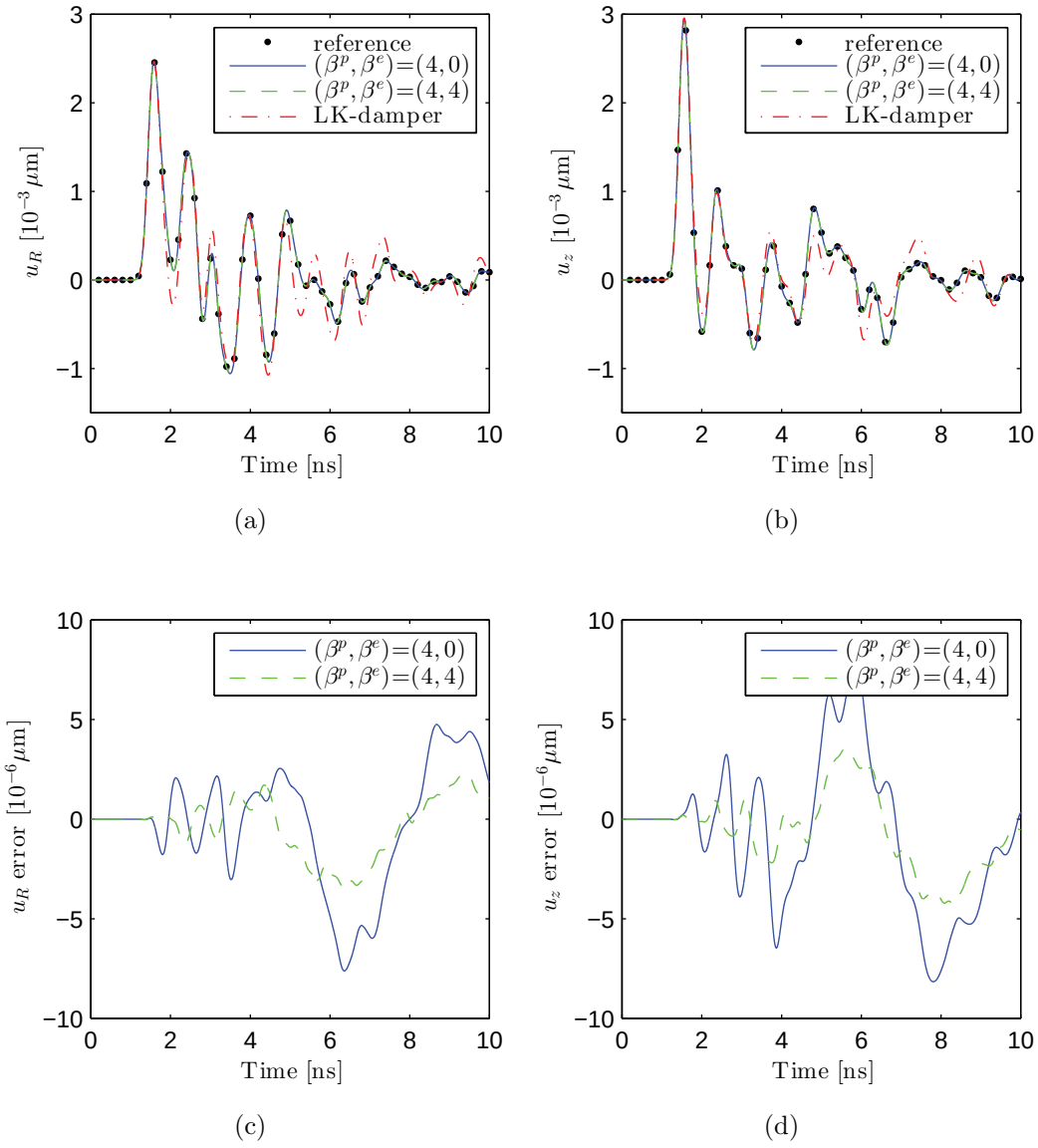
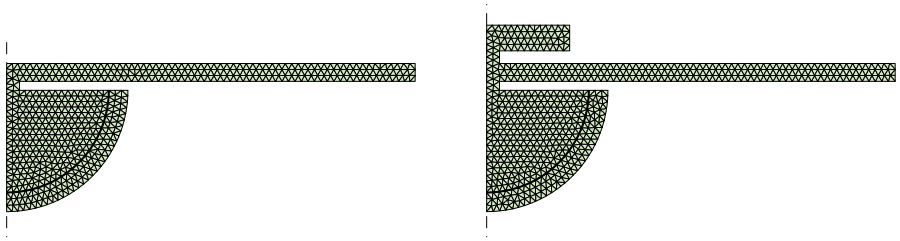


Figure 3.3: Plots of computed (a) u_R and (b) u_z versus time for four solutions and absolute differences of the PML solutions (c) u_R and (d) u_z from the reference solutions.

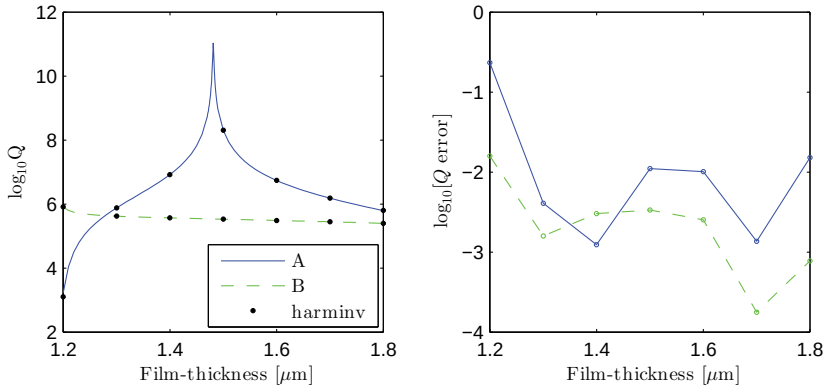


(a) Resonator A (b) Resonator B

Figure 3.4: Geometry and triangular mesh of (a) a resonator with flat top (b) a resonator with mushroom-like structure.

and compute the corresponding quality factor using Def. (2.1) for various film-thicknesses between 1.2 and 1.8[μm]. We first compute eigenvalues directly using a generic eigenvalue solver. Plots of quality factor versus film-thickness are shown in solid and dashed lines in Fig. 3.5(a) for resonators A and B, respectively. Note that resonator A exhibits a wild swing in Q around a thickness of 1.48[μm], while resonator B shows little sensitivity to the thickness of the film.

As mentioned in Chapter 2, however, a direct eigenvalue computation is only applicable to a small system due to its high memory requirements. In [22] an alternative transient methodology to compute the quality factors of the fundamental modes was proposed and its accuracy and scalability were demonstrated through two- and three-dimensional problems of resonators using LK-dampers. For later application of our PML formulation to full three-dimensional resonator problems, we also adopt the methodology presented in [22] to our axisymmetric resonator problems to compute the quality factors. Specifically, we apply a broadband Gaussian pulse (3.24) with $f_0 = \omega^*_{\text{estimate}}/2\pi$ radially on the edge of the disk, solve system (3.23) using a fourth-order explicit Runge-Kutta method with $\Delta t = 1 \cdot 10^{-3}$[ns] up to $5,000$[ns], and record a time-series of average radial displacement $\bar{u}_R$ over the entire domain. Next the filter-diagonalization method [46], discussed in Sec.2.4, is applied to this time-series data using the harminv software [32], which extracts for each excited mode its frequency, rate of damping, and quality factor. We pick the quality factor corresponding to a mode whose frequency is closest to $\omega^*_{\text{estimate}}$. In using harminv, we cut off the first 10[ns] of the time-series, store data at every 200th time-step, and specify a broad range of frequencies 10^{-4}-10^{4}[GHz] over which we expect to find the fundamental mode. Quality factors thus obtained are plotted in Fig. 3.5(a) as black dots, which show good agreement with values obtained by the eigenvalue solver. Fig. 3.5(b) shows relative errors of quality factors computed by harminv compared to the ones obtained by "exact" eigenvalue analysis. We note that experimental data shows such wild swings in flat top resonators [9] and that experience shows these swings are absent in resonators with mushroom caps [50].



(a) (b)

Figure 3.5: Plots of (a) $\log_{10} Q$ computed by an eigenvalue solver and harmonic inversion and (b) relative errors of Q computed by harmonic inversion compared to that computed by an eigenvalue solver for various film thicknesses for Resonator A and Resonator B.

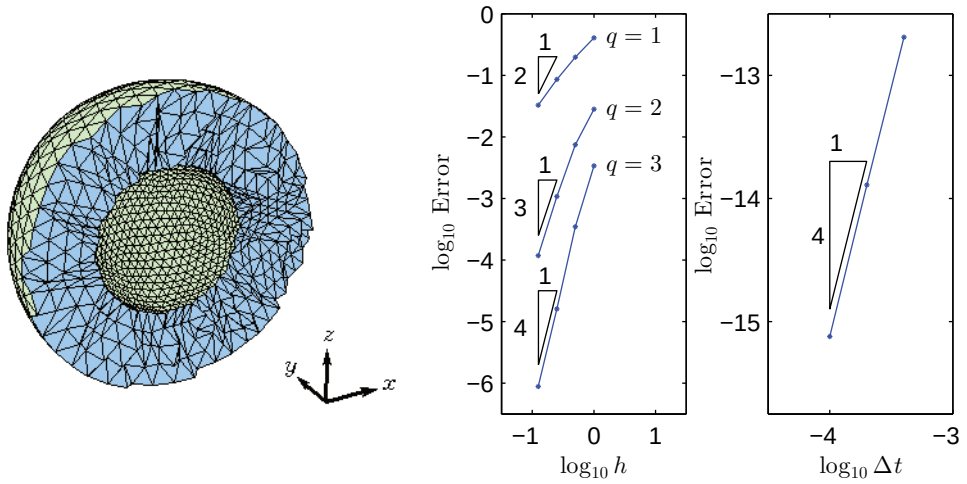
3.4.2 Three-dimensional problems

One of the main impetuses for the development of our spherical PML formulation was the accurate simulation of three-dimensional resonator systems where eigenvalue extraction is only feasible via time-series analysis coupled to filter diagonalization [46, 22]. In this setting computational cost reduction is of paramount importance. In this section we look at the numerical properties of our proposed method as well as demonstrate its use on a large scale problem.

Convergence study

First, we validate the spatial and temporal convergence rates of our DG formulation in conjunction with a RK4 time-integrator for three-dimensional problems. We consider a hollow sphere with inner and outer radii of 1.0 and 2.0, respectively, which has material properties of $\lambda = 1.0$, $\mu = 1.0$, and $\rho = 3.0$. The inner boundary is clamped and the outer boundary is traction-free. As output quantity for the error calculation, we study the average x-displacement $\bar{u}_x$ on the outer boundary.

To demonstrate the spatial convergence, we compute steady-state displacements of the hollow sphere subject to a body-force of $\boldsymbol{f} = (e^z, 0, 0)^T$. A sequence of uniformly refined unstructured meshes and polynomial degrees of $q = 1, ..., 4$ are used. The coarsest mesh has a single layer of tetrahedral elements across the thickness and a total of 381 tetrahedra. We refer to this element-size as $h = 1$. Each tetrahedron is then repeatedly split into 8 similar tetrahedra to produce a sequence of meshes of element-sizes $h_r = 1/2^r$, $r = 0, ..., 4$. Fig. 3.6a shows an example mesh corresponding to $r = 2$ generated by `netgen`. Considering



(a) (b) (c)

Figure 3.6: (a) Problem setup for a convergence study. Plots of error versus (b) h and (c) Δt in log-log scale.

the output quantity corresponding to $h_4 = 1/16$ and $q = 4$ as the "exact solution" $\bar{u}_{x,\text{exact}}$, errors $|\bar{u}_x - \bar{u}_{x,\text{exact}}|$ are computed for $q = 1, 2, 3$ and plotted in Fig. 3.6b against element-size h on a log-log scale. We note that the slopes are close to the expected $q + 1$ order of convergence.

To confirm the expected temporal convergence rate, we consider the mesh of $r = 4$ and $q = 4$. We multiply the body-forces by a smooth Gaussian profile in time:

$$\boldsymbol{b}(t) = \left(e^{-\left(\frac{t-0.1}{0.01}\right)^2} e^z, 0, 0\right)^T$$

and integrate until time $T = 1$ using four different time-steps $\Delta t = 4 \cdot 10^{-4}/2^s$ $s = 0, ..., 3$. The solution corresponding to the finest time-step is considered as "exact". Fig. 3.6c shows plots of errors at $T = 1$ against time-step Δt on a log-log scale and we can observe the expected fourth-order rate of convergence for the error.

Double-disk resonator

As our last example we compute the quality factor of a full three-dimensional double-disk resonator which is anchored to a semi-infinite substrate. Each disk has a radius of $R_d = 8.0[\mu\text{m}]$ and a thickness of $1.1[\mu\text{m}]$ and is anchored to the substrate by a cylindrical post which has a radius of $1.0[\mu\text{m}]$ and a height of $0.50[\mu\text{m}]$. These two disks are separated from each other by $20[\mu\text{m}]$ in the x-direction and connected by a bar with a width of $1.0[\mu\text{m}]$ and

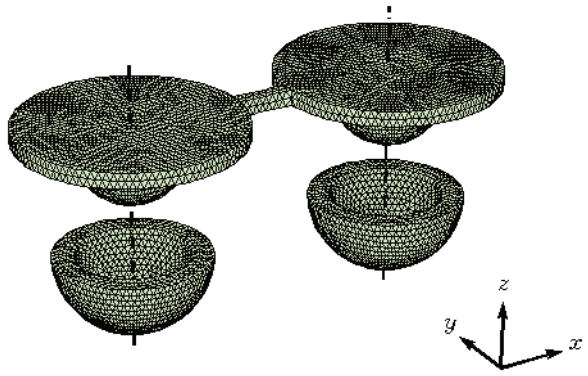


Figure 3.7: The double-disk resonator with its semi-infinite substrate replaced by the PML-bowls. A tetrahedral mesh is also shown.

a height of 1.1[μm]. The material properties are the same as were used for the axisymmetric resonator example.

To analyze this problem on a finite domain, we truncate the unbounded substrate and attach PMLs. In this type of problems, our spherical PML shows several advantages over other existing PMLs. First, the geometry does not allow for structured meshes, so PML formulations that work with finite element methods on unstructured meshes are required. Existing formulations which do work with finite element methods, however, introduce a large number of nodal variables, which adds significant computational cost for large three-dimensional problems. For instance, the formulation proposed in [3] requires about 2.5 times more memory storage for the PML domains than our formulation. Moreover, while most formulations are based on Cartesian PMLs, spherical PMLs better match this type of problems since resonating disks emit elastic waves almost spherically into the substrate via the small cylindrical posts. Finally, the existence of edges and corners as would be required for Cartesian based PMLs would add significant complexity to the implementation.

Here, the substrate is truncated to leave half spheres of radii 4.0[μm] and "*PML-bowls*" of thickness 1.5[μm] are attached on the surfaces of truncation; see Fig. 3.7. The outer boundaries of the PML-bowls are clamped and other boundaries are traction-free.

To compute the quality factor of this resonator, we again employ the transient dynamical approach proposed in [22] and also introduced in Chapter 2. The estimate of the fundamental frequency $\omega^*_{\text{estimate}}$ is computed using Eqn. (3.25) as 1.541×10^9[rad/s]. We apply a Gaussian pulse (3.24) with $f_0 = \omega^*_{\text{estimate}}/2\pi$ uniformly along the edge of the left disk and record a time-series of the average displacement in the x-direction over the entire domain $\bar{u}_x$ up to 250[ns]. We then apply harmonic-inversion via filter-diagonalization to this time-series via harminv

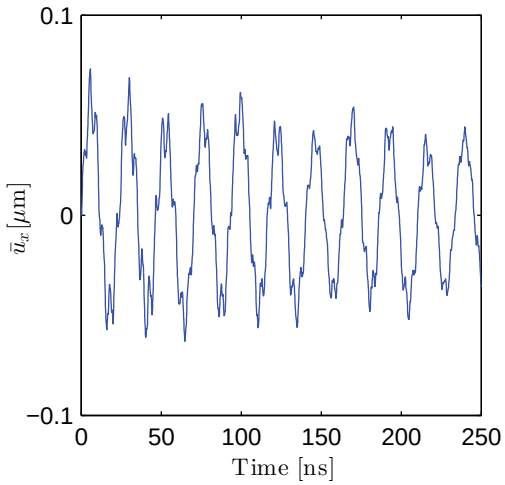


Figure 3.8: Plots of computed $\bar{u}_R$ versus time for the double-disk resonator problem.

and extract the eigenvalue corresponding to the fundamental radial mode of vibration.

We use netgen to construct an uniform tetrahedral mesh with $h \approx 0.50[\mu\text{m}]$; see Fig. 3.7. It produces a total of 55,644 elements among which 24,888 are in the PML, which for our polynomial degree of 3 gives about 9.7 million dofs. We set $\Delta t = 8 \cdot 10^{-4}$[ns] to satisfy the CFL-condition. The same set of PML-parameters is used as in Sec.3.4.1 since the nature of the problem is similar.

Fig. 3.8 shows a time-series of $\bar{u}_x$. In using harminv, we cut off the first 1[ns] of the time-series, store data at every 125th time-step, and specify a broad range of frequencies 10^{-4}-10^{4}[GHz]. The resulting eigenvalue $-0.003879 + 1.529\mathrm{i}$ corresponds to a quality factor of $Q = 197.0$. This example problem demonstrates the applicability of our spherical PML to a full real-world three-dimensional problem.

3.5 Conclusion

A new PML formulation was developed for time-domain analysis of elastic waves on three-dimensional spherical domains or two-dimensional axisymmetric domains. Since our spherical PML formulation is developed based on the regular second-order elastic equation instead of the first-order velocity-stress system, it readily works with standard finite element methods as well as discontinuous Galerkin methods on unstructured meshes. It is monolithic and simple to implement; it involves no edges or corners as in existing time-domain formulations of Cartesian PMLs which require special treatment. It also allows for a natural application of traction-free boundary conditions, taking advantage of the second-order formulation which is well-suited for elastodynamics. Furthermore, our formulation requires a smaller number of variables than other existing formulations, which is an advantage when solving

large three-dimensional problems where memory-usage can be demanding. The formulation was demonstrated using high-order Discontinuous Galerkin discretizations with a CDG scheme on unstructured meshes and a fourth-order explicit Runge-Kutta time-integrator, which showed the high accuracy of the method as well as its ability to solve large three-dimensional problems. Finally, we were able to successfully apply our methods to a large scale resonator problem and extract "damped" eigenvalues using explicit time-integration and a harmonic-inversion technique.

Chapter 4

Perfectly matched discrete layers

4.1 Introduction

The major shortcoming in using perfectly matched layers (PMLs) is that their nature is quite different in a continuous setting and in a discrete setting and an additional source of error arises upon discretization. Perfectly matched discrete layer (PMDL) is a family of absorbing boundary conditions whose basic idea formed in an attempt to optimize the PML parameters in a discrete setting [1]. This idea is then reinterpreted as a systematic procedure of constructing PMDL formulations for the scalar wave equation and a general vector equations in [24]. In [25], this construction technique was extended for the scalar wave equation on general polygonal domains in two dimensions and applied to various example problems in the time-domain. Note that in these numerical examples, only purely propagating waves were explicitly treated by adding a restriction to a set of parameters; the time-domain formulation is not known for general sets of parameters. The accuracy of PMDL formulations was studied for the scalar anisotropic wave equation in the frequency-domain in [57] and in the time-domain in [58]. Well-posedness of PMDL methods applied to the anisotropic wave equation was studied in [58], where again the PMDL formulation was specialized for the purely propagating wave case. PMDL formulations were then extended, in the frequency-domain, to tilted elliptic anisotropy in [59] and to untilted non-elliptic anisotropy in [60].

Later we will show that PMDL methods are essentially equivalent to the complete radiation boundary conditions (CRBCs) in Chapter 5 and in Appendix B.4. Thus, we limit ourselves in this chapter to introduce the basic construction procedure and the final formulation of PMDLs postponing detailed analysis and observations until we introduce CRBCs in Chapter5.

In Sec.4.2, we derive the PMDL formulation for a general vector equations following the procedure presented in [59] with slight change in the interpretation. Sec.4.3 poses several remarks.

4.2 Derivation

In this section we rederive the PMDL formulation. We closely follow the systematic procedure presented in [59] with several minor changes for clarification. The derivation is performed in two dimensions ($d = 2$), but extension to three dimensions is expected to be straightforward.

Consider a general two-dimensional vector equation given by:

$$\boldsymbol{u}_{,tt} - \boldsymbol{G}_{xx}\boldsymbol{u}_{,xx} - \left(\boldsymbol{G}_{xy} + \boldsymbol{G}_{xy}^{\mathrm{T}}\right)\boldsymbol{u}_{,xy} - \boldsymbol{G}_{yy}\boldsymbol{u}_{,yy} = \boldsymbol{f}, \tag{4.1}$$

where $\boldsymbol{u}$ is an unknown $d\times 1$ vector function, $\boldsymbol{G}_{xx}$, $\boldsymbol{G}_{xy}$, and $\boldsymbol{G}_{yy}$ are $d\times d$ constant coefficient matrices of $\boldsymbol{u}_{,xx}$, $\boldsymbol{u}_{,xy}$, and $\boldsymbol{u}_{,yy}$, respectively, and $\boldsymbol{f}$ is a $d\times 1$ source vector which is supported on $x < 0$. Initial data of $\boldsymbol{u}$ are also assumed to be supported on $x < 0$. We are interested in the solution to (4.1) on the full-space $\mathbb{R}^d$, but would like to solve for this solution on $x < 0$ with PMDLs on $x = 0$. PMDLs therefore have to provide a correct representation for the traction vector $\boldsymbol{t}$ given by:

$$\boldsymbol{t} = \boldsymbol{G}_{xx}\boldsymbol{u}_{,x} + \boldsymbol{G}_{\boldsymbol{xy}}\boldsymbol{u}_{,y}, \tag{4.2}$$

on the surface $x = 0$. Taking Laplace and Fourier transforms of (4.1) in the t- and y-directions ($t \leftrightarrow s$, $y \leftrightarrow k_y$), one obtains (with an abuse of notation):

$$s^2\boldsymbol{u} - \boldsymbol{G}_{xx}\boldsymbol{u}_{,xx} - \mathrm{i}k_y\left(\boldsymbol{G}_{xy} + \boldsymbol{G}_{xy}^{\mathrm{T}}\right)\boldsymbol{u}_{,x} - \left(\mathrm{i}k_y\right)^2\boldsymbol{G}_{yy}\boldsymbol{u} = \boldsymbol{f}, \tag{4.3}$$

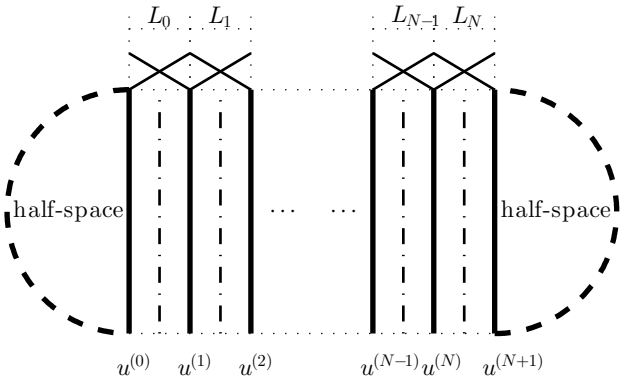
where $\boldsymbol{f}$ now incorporates the effect of the initial data. The corresponding traction vector is then given by:

$$\boldsymbol{t} = \boldsymbol{G}_{xx}\boldsymbol{u}_{,x} + \mathrm{i}k_y\boldsymbol{G}_{\boldsymbol{xy}}\boldsymbol{u}. \tag{4.4}$$

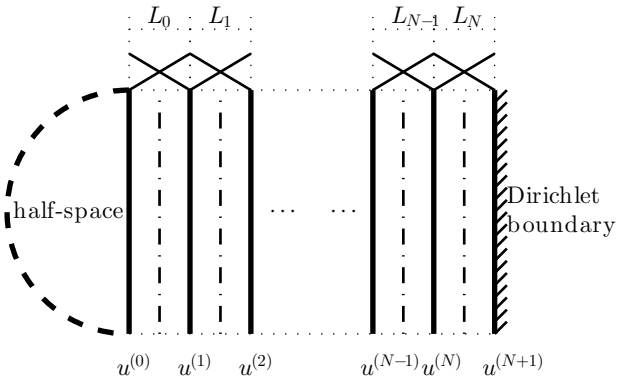
The PMDL formulation is derived in the frequency-domain using (4.3). We illustrate how the traction vector (4.4) is represented on $x = 0$ in PMDL methods.

We first quickly go through the whole derivation and later discuss some details. The half-space $x > 0$ is first divided into $N+1$ layers of length L_n ($n = 0, 1, \cdots, N$) and another half-space $x > \sum_{n'=0}^{N} L_{n'}$; see Fig.4.1(a). In Fig.4.1(a), $\boldsymbol{u}^{(0)}$ is a restriction of $\boldsymbol{u}$ to $x = x_0 = 0$ and $\boldsymbol{u}^{(n)}$ ($n = 1, \cdots, N$) are restrictions of $\boldsymbol{u}$ to $x = x_n = \sum_{n'=0}^{n-1} L_{n'}$. On each layer, (4.3) is discretized in the x-direction. To this end, we first require a weak form of (4.3). Noting that $\boldsymbol{f} = \boldsymbol{0}$ on $x \geq 0$, the weak form (per layer) of (4.3) is given by:

$$\begin{aligned}
&\int_{x_n}^{x_{n+1}} \boldsymbol{w}\cdot\left(s^2\boldsymbol{I} - \left(\mathrm{i}k_y\right)^2\boldsymbol{G}_{yy}\right)\boldsymbol{u}\mathrm{d}x - \int_{x_n}^{x_{n+1}} \mathrm{i}k_y\boldsymbol{w}\cdot\boldsymbol{G}_{xy}^{\mathrm{T}}\boldsymbol{u}_{,x}\mathrm{d}x \\
&+ \int_{x_n}^{x_{n+1}} \boldsymbol{w}_{,x}\cdot\left(\boldsymbol{G}_{xx}\boldsymbol{u}_{,x} + \mathrm{i}k_y\boldsymbol{G}_{xy}\boldsymbol{u}\right)\mathrm{d}x - \left[\boldsymbol{w}\cdot\boldsymbol{t}\right]_{x_n}^{x_{n+1}} = \boldsymbol{0}, \quad \forall\boldsymbol{w}
\end{aligned} \tag{4.5}$$



(a)



(b)

Figure 4.1: Illustration of the derivation of PMDL formulation. Dashed lines indicate *mid-points* on which integrals in the weak form (4.5) are evaluated.

where $\boldsymbol{w}$ is a test function and $\boldsymbol{I}$ is the $d \times d$ identity tensor. The divergence theorem was used in the derivation of (4.5). For both $\boldsymbol{u}$ and $\boldsymbol{w}$ in (4.5), we use *linear interpolations*, that is, we have on layer n $(n = 0, 1, \cdots, N)$:

$$\boldsymbol{u} = \boldsymbol{N}\left(T_n^{-1}(x)\right)\begin{pmatrix}\boldsymbol{u}^{(n)}\\ \boldsymbol{u}^{(n+1)}\end{pmatrix}, \quad \boldsymbol{w} = \boldsymbol{N}\left(T_n^{-1}(x)\right)\begin{pmatrix}\boldsymbol{w}^{(n)}\\ \boldsymbol{w}^{(n+1)}\end{pmatrix}, \tag{4.6}$$

where $\boldsymbol{w}^{(n)}$ and $\boldsymbol{w}^{(n+1)}$ are nodal values of $\boldsymbol{w}$ on $x = x_n$ and x_{n+1}, and

$$\begin{aligned} T_n : X \in [0,1] &\mapsto x = (1-X)\,x_n + X x_{n+1}, \\ \boldsymbol{N}(X) &= \begin{bmatrix} N_0 & 0 & N_1 & 0 \\ 0 & N_0 & 0 & N_1 \end{bmatrix}, \\ N_0(X) &= 1 - X, \end{aligned}$$

$$N_1(X) = X.$$

These shape functions are also depicted in Fig.4.1(a). We then substitute (4.6) for $\boldsymbol{u}$ and $\boldsymbol{w}$ in the weak form (4.5) and evaluate the integrals. Here, instead of evaluating them exactly, we use *midpoint integrations* on each layer; a midpoint on each layer are also depicted as a dashed line in Fig.4.1(a). This sequence of seemingly inaccurate approximations, *linear interpolation* followed by the *midpoint integrations*, characterizes the derivation of PMDLs. We then have, on layer n ($n = 0, 1, \cdots, N$):

$$\begin{aligned}
\begin{pmatrix} \boldsymbol{t}^{(n)} \\ \boldsymbol{0} \end{pmatrix} + \Bigg[& \frac{L_n}{4} \begin{pmatrix} s^2\boldsymbol{I} - (\mathrm{i}k_y)^2 \boldsymbol{G}_{yy} & s^2\boldsymbol{I} - (\mathrm{i}k_y)^2 \boldsymbol{G}_{yy} \\ s^2\boldsymbol{I} - (\mathrm{i}k_y)^2 \boldsymbol{G}_{yy} & s^2\boldsymbol{I} - (\mathrm{i}k_y)^2 \boldsymbol{G}_{yy} \end{pmatrix} \\
& + \frac{\mathrm{i}k_y}{2} \begin{pmatrix} -\boldsymbol{G}_{xy} & -\boldsymbol{G}_{xy} \\ \boldsymbol{G}_{xy} & \boldsymbol{G}_{xy} \end{pmatrix} \\
& + \frac{\mathrm{i}k_y}{2} \begin{pmatrix} \boldsymbol{G}_{xy}^{\mathrm{T}} & -\boldsymbol{G}_{xy}^{\mathrm{T}} \\ \boldsymbol{G}_{xy}^{\mathrm{T}} & -\boldsymbol{G}_{xy}^{\mathrm{T}} \end{pmatrix} \\
& + \frac{1}{L_n} \begin{pmatrix} \boldsymbol{G}_{xx} & -\boldsymbol{G}_{xx} \\ -\boldsymbol{G}_{xx} & \boldsymbol{G}_{xx} \end{pmatrix} \qquad \Bigg] \begin{pmatrix} \boldsymbol{u}^{(n)} \\ \boldsymbol{u}^{(n+1)} \end{pmatrix} - \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{t}^{(n+1)} \end{pmatrix} = \boldsymbol{0},
\end{aligned} \tag{4.7}$$

where

$$\boldsymbol{t}^{(n)} = \boldsymbol{G}_{xx}\boldsymbol{u}_{,x}^{(n)} + \mathrm{i}k_y \boldsymbol{G}_{\boldsymbol{xy}}\boldsymbol{u}^{(n)}. \tag{4.8}$$

$\boldsymbol{u}_{,x}^{(n)}$ in (4.8) is understood as a restriction of $\boldsymbol{u}_{,x}^{(n)}$ to $x = x_n$. A set of equations (4.7) ($n = 0, 1, \cdots, N$) is then assembled canceling out the traction vectors given in (4.8) between neighboring layers to produce a block tri-diagonal monolithic representation of $\boldsymbol{t}^{(0)}$ in a matrix form as:

$$\begin{pmatrix} \boldsymbol{t}^{(0)} \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} \boldsymbol{S}_{11}^{(0)} & \boldsymbol{S}_{12}^{(0)} & & & & & \\ \boldsymbol{S}_{21}^{(0)} & \boldsymbol{S}_{22}^{(0)} + \boldsymbol{S}_{11}^{(1)} & \boldsymbol{S}_{12}^{(1)} & & & & \\ & \boldsymbol{S}_{21}^{(1)} & \boldsymbol{S}_{22}^{(1)} + \boldsymbol{S}_{11}^{(2)} & \boldsymbol{S}_{12}^{(2)} & & & \\ & & \ddots & & & & \\ & & \boldsymbol{S}_{22}^{(N-2)} + \boldsymbol{S}_{11}^{(N-1)} & \boldsymbol{S}_{12}^{(N-1)} & & & \\ & & & \boldsymbol{S}_{21}^{(N-1)} & \boldsymbol{S}_{22}^{(N-1)} + \boldsymbol{S}_{11}^{(N)} & \boldsymbol{S}_{12}^{(N)} & \\ & & & & & \boldsymbol{S}_{21}^{(N)} & \boldsymbol{S}_{22}^{(N)} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^{(0)} \\ \boldsymbol{u}^{(1)} \\ \boldsymbol{u}^{(2)} \\ \vdots \\ \boldsymbol{u}^{(N-1)} \\ \boldsymbol{u}^{(N)} \\ \boldsymbol{u}^{(N+1)} \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ \boldsymbol{t}^{(N+1)} \end{pmatrix} = \boldsymbol{0}, \tag{4.9}$$

where

$$\boldsymbol{S}_{11}^{(n)} = \frac{L_n}{4}\left(s^2\boldsymbol{I} - (\mathrm{i}k_y)^2 \boldsymbol{G}_{yy}\right) - \frac{\mathrm{i}k_y}{2}\boldsymbol{G}_{xy} + \frac{\mathrm{i}k_y}{2}\boldsymbol{G}_{xy}^{\mathrm{T}} + \frac{1}{L_n}\boldsymbol{G}_{xx}, \tag{4.10a}$$

$$\boldsymbol{S}_{12}^{(n)} = \frac{L_n}{4}\left(s^2\boldsymbol{I} - (\mathrm{i}k_y)^2\boldsymbol{G}_{yy}\right) - \frac{\mathrm{i}k_y}{2}\boldsymbol{G}_{xy} - \frac{\mathrm{i}k_y}{2}\boldsymbol{G}_{xy}^{\mathrm{T}} - \frac{1}{L_n}\boldsymbol{G}_{xx}, \tag{4.10b}$$

$$\boldsymbol{S}_{21}^{(n)} = \frac{L_n}{4}\left(s^2\boldsymbol{I} - (\mathrm{i}k_y)^2\boldsymbol{G}_{yy}\right) + \frac{\mathrm{i}k_y}{2}\boldsymbol{G}_{xy} + \frac{\mathrm{i}k_y}{2}\boldsymbol{G}_{xy}^{\mathrm{T}} - \frac{1}{L_n}\boldsymbol{G}_{xx}, \tag{4.10c}$$

$$\boldsymbol{S}_{22}^{(n)} = \frac{L_n}{4}\left(s^2\boldsymbol{I} - (\mathrm{i}k_y)^2\boldsymbol{G}_{yy}\right) + \frac{\mathrm{i}k_y}{2}\boldsymbol{G}_{xy} - \frac{\mathrm{i}k_y}{2}\boldsymbol{G}_{xy}^{\mathrm{T}} + \frac{1}{L_n}\boldsymbol{G}_{xx}. \tag{4.10d}$$

The last step of the construction is to apply a termination condition:

$$\boldsymbol{u}^{(N+1)} = \boldsymbol{0}, \tag{4.11}$$

on $x = x_{N+1}$, which is depicted in Fig.4.1(b). Applying this termination condition (4.11) in (4.9) and removing the last block of equations involving $\boldsymbol{t}^{(N+1)}$, one obtains:

$$\begin{pmatrix} \boldsymbol{t}^{(0)} \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} \boldsymbol{S}_{11}^{(0)} & \boldsymbol{S}_{12}^{(0)} & & & & \\ \boldsymbol{S}_{21}^{(0)} & \boldsymbol{S}_{22}^{(0)}+\boldsymbol{S}_{11}^{(1)} & \boldsymbol{S}_{12}^{(1)} & & & \\ & \boldsymbol{S}_{21}^{(1)} & \boldsymbol{S}_{22}^{(1)}+\boldsymbol{S}_{11}^{(2)} & \boldsymbol{S}_{12}^{(2)} & & \\ & & \ddots & & & \\ & & & \boldsymbol{S}_{22}^{(N-2)}+\boldsymbol{S}_{11}^{(N-1)} & \boldsymbol{S}_{12}^{(N-1)} & \\ & & & \boldsymbol{S}_{21}^{(N-1)} & \boldsymbol{S}_{22}^{(N-1)}+\boldsymbol{S}_{11}^{(N)} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^{(0)} \\ \boldsymbol{u}^{(1)} \\ \boldsymbol{u}^{(2)} \\ \vdots \\ \boldsymbol{u}^{(N-1)} \\ \boldsymbol{u}^{(N)} \end{pmatrix} = \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{0} \\ \boldsymbol{0} \\ \vdots \\ \boldsymbol{0} \\ \boldsymbol{0} \end{pmatrix}, \tag{4.12}$$

where $\boldsymbol{S}_{11}^{(n)}$, $\boldsymbol{S}_{12}^{(n)}$, $\boldsymbol{S}_{21}^{(n)}$, and $\boldsymbol{S}_{22}^{(n)}$ are given in (4.10). As seen from (4.8), (4.12) provides a representation of $\boldsymbol{t}^{(0)}$ on $x = 0$. Equation (4.12) is the final form of the PMDL formulation. Note that (4.12) provides a representation of $\boldsymbol{t}^{(0)}$ merely in terms of $\boldsymbol{u}^{(n)}$ on $x = x_n$ ($n = 0, 1, \cdots, N$) and it does not involve any x-dependence, since we have already integrated in the x-direction. It implies that it is of no significance where on the x-axis $\boldsymbol{u}^{(n)}$ ($n = 1, \cdots, N+1$) are defined, or restricted, as long as they are defined on a plane orthogonal to the x-axis. Thus, for convenience, in the rest of this section, we assume that they are vectors defined on $x = 0$. We can assume so already in (4.7) and (4.8) and this change of interpretations does not affect the final form (4.12).

Thus, $\boldsymbol{u}^{(n)}$ ($n = 1, \cdots, N$) can be regarded as a set of *auxiliary functions* defined on $x = 0$ and L_n ($n = 0, \cdots, N$) present in (4.12) together with (4.10) can be regarded as a set of parameters for PMDLs. Understanding this way, we do not need to restrict L_n to real numbers, i.e. they can be complex if so desired.

We can now discuss the implication of equation (4.7) together with (4.8). Suppose that $\tilde{\boldsymbol{u}}^{(0)}$ is an extension of $\boldsymbol{u}^{(0)}$, which is defined, not only on $x = 0$, but also on $x \in \mathbb{R}$, i.e., $\tilde{\boldsymbol{u}}^{(0)}\big|_{x=0} = \boldsymbol{u}^{(0)}$. Similarly, suppose that $\tilde{\boldsymbol{u}}^{(n)}$ ($n = 1, \cdots, N+1$) are extensions of $\boldsymbol{u}^{(n)}$ which are defined, not only on $x = 0$, but also on $x \geq 0$, i.e. $\tilde{\boldsymbol{u}}^{(n)}\big|_{x=0} = \boldsymbol{u}^{(n)}$. This slight difference in the definitions of $\tilde{\boldsymbol{u}}^{(0)}$ and $\tilde{\boldsymbol{u}}^{(n)}$ is for a technical reason. Suppose $\tilde{\boldsymbol{u}}^{(n)}$ ($n = 0, 1, \cdots, N+1$) satisfy (4.7) together with (4.8) on $x \geq 0$, that is:

$$\begin{pmatrix} \tilde{\boldsymbol{t}}^{(n)} \\ \boldsymbol{0} \end{pmatrix} + \left[\frac{L_n}{4} \begin{pmatrix} s^2\boldsymbol{I} - (\mathrm{i}k_y)^2\boldsymbol{G}_{yy} & s^2\boldsymbol{I} - (\mathrm{i}k_y)^2\boldsymbol{G}_{yy} \\ s^2\boldsymbol{I} - (\mathrm{i}k_y)^2\boldsymbol{G}_{yy} & s^2\boldsymbol{I} - (\mathrm{i}k_y)^2\boldsymbol{G}_{yy} \end{pmatrix} \right.$$

$$
\begin{aligned}
&+\frac{\mathrm{i}k_y}{2}\begin{pmatrix}-\boldsymbol{G}_{xy} & -\boldsymbol{G}_{xy}\\ \boldsymbol{G}_{xy} & \boldsymbol{G}_{xy}\end{pmatrix}\\
&+\frac{\mathrm{i}k_y}{2}\begin{pmatrix}\boldsymbol{G}_{xy}^{\mathrm{T}} & -\boldsymbol{G}_{xy}^{\mathrm{T}}\\ \boldsymbol{G}_{xy}^{\mathrm{T}} & -\boldsymbol{G}_{xy}^{\mathrm{T}}\end{pmatrix}\\
&+\frac{1}{L_n}\begin{pmatrix}\boldsymbol{G}_{xx} & -\boldsymbol{G}_{xx}\\ -\boldsymbol{G}_{xx} & \boldsymbol{G}_{xx}\end{pmatrix}
\end{aligned}
\Bigg]\begin{pmatrix}\tilde{\boldsymbol{u}}^{(n)}\\ \tilde{\boldsymbol{u}}^{(n+1)}\end{pmatrix}-\begin{pmatrix}\boldsymbol{0}\\ \tilde{\boldsymbol{t}}^{(n+1)}\end{pmatrix}=\boldsymbol{0},
\quad \text{on } x\geq 0, \tag{4.13}
$$

where

$$\tilde{\boldsymbol{t}}^{(n)}=\boldsymbol{G}_{xx}\tilde{\boldsymbol{u}}_{,x}^{(n)}+\mathrm{i}k_y\boldsymbol{G}_{\boldsymbol{xy}}\tilde{\boldsymbol{u}}^{(n)},$$

is a traction vector which depends on x and defined on $x\geq 0$. Equation (4.13) is a homogeneous system of ODEs in terms of x of double the size of (4.3), i.e. $2\times d$, to be solved for $\tilde{\boldsymbol{u}}^{(n)}$ and $\tilde{\boldsymbol{u}}^{(n+1)}$ on $x\geq 0$. We denote by I and II the first and second block-rows in (4.13); i.e., I represents equations of $\tilde{\boldsymbol{t}}^{(n)}$ and II represents those of $\tilde{\boldsymbol{t}}^{(n+1)}$. Then, one can show on $x\geq 0$ that $\mathrm{I},\mathrm{II}\Leftrightarrow$

$$\left(\frac{2}{L_n}+\partial_x\right)\tilde{\boldsymbol{u}}^{(n)}=\left(\frac{2}{L_n}-\partial_x\right)\tilde{\boldsymbol{u}}^{(n+1)}, \tag{4.14}$$

$$s^2\tilde{\boldsymbol{u}}^{(n)}-\boldsymbol{G}_{xx}\tilde{\boldsymbol{u}}_{,xx}^{(n)}-(\mathrm{i}k_y)\left(\boldsymbol{G}_{xy}+\boldsymbol{G}_{xy}^{\mathrm{T}}\right)\tilde{\boldsymbol{u}}_{,x}^{(n)}-(\mathrm{i}k_y)^2\boldsymbol{G}_{yy}\tilde{\boldsymbol{u}}^{(n)}=\boldsymbol{0}. \tag{4.15}$$

Equation (4.14) results from $\mathrm{I}-\mathrm{II}$ and (4.15) results from $\mathrm{I}+\mathrm{II}$ and (4.14). Equation (4.15) is nothing but the original vector equation (4.3) for $\tilde{\boldsymbol{u}}^{(n)}$ on $x\geq 0$. Equations (4.14) and (4.15) imply that $\tilde{\boldsymbol{u}}^{(n+1)}$ also satisfies (4.3).

Now suppose $\boldsymbol{u}^\infty$ is the exact solution for (4.3) on the entire domain $\mathbb{R}^d$ for some given force $\bar{\boldsymbol{f}}$ supported on $x<0$. Consider the following problem to be solved for $\tilde{\boldsymbol{u}}^{(0)}$, which is to be an approximation to $\boldsymbol{u}^\infty$ on $x<0$, given as:

Find $\tilde{\boldsymbol{u}}^{(0)}$ on $x<0$, $\boldsymbol{u}^{(n)}$ $(n=1,\cdots,N)$ on $x=0$, $\tilde{\boldsymbol{u}}^{(N+1)}$ on $x>0$, such that

$\tilde{\boldsymbol{u}^{(0)}}$ satisfies (4.3) with $\boldsymbol{f}=\bar{\boldsymbol{f}}$ on $x<0$, (4.16a)

(4.9) is satisfied on $x=0$, (4.16b)

$\tilde{\boldsymbol{u}}^{(N+1)}$ satisfies (4.3) with $\boldsymbol{f}=\boldsymbol{0}$ on $x>0$. (4.16c)

One can see that *a* solution to (4.16) can be constructed from $\boldsymbol{u}^\infty$ using (4.14) and (4.15) as:

$$\tilde{\boldsymbol{u}}^{(0)}=\boldsymbol{u}^\infty,\quad \text{on } x<0, \tag{4.17a}$$

$$\boldsymbol{u}^{(n)}=\prod_{n'=0}^{n-1}\left(\frac{\frac{2}{L_{n'}}+\partial_x}{\frac{2}{L_{n'}}-\partial_x}\right)\boldsymbol{u}^\infty\Bigg|_{x=0},\quad \text{for } n=1,\cdots,N, \tag{4.17b}$$

$$\tilde{\boldsymbol{u}}^{(N+1)} = \prod_{n'=0}^{N} \left(\frac{\frac{2}{L_{n'}} + \partial_x}{\frac{2}{L_{n'}} - \partial_x} \right) \boldsymbol{u}^{\infty}, \quad \text{on } x > 0. \tag{4.17c}$$

The solution $\boldsymbol{u}^{(0)}$ given in (4.17a) implies that the problem (4.16) is *consistent* with the exact solution $\boldsymbol{u}^{\infty}$ on $x < 0$. This indicates that equation (4.9) is some sort of absorbing boundary condition. If we know that $\tilde{\boldsymbol{u}}^{(N+1)}$ in (4.17c) is *small* enough, the use of the approximate termination condition (4.11) is also justified, which is at least the case for the scalar wave equation with a specific choice of L_n. This point will become clearer in Chapter 5.

It should be noted that from the vector equation (4.3) for $\tilde{\boldsymbol{u}}^{(0)}$ and (4.14) ($n = 0, 1, \cdots, N$) on $x \geq 0$, one can readily reconstruct (4.7) on $x \geq 0$. Therefore, one can construct PMDL formulation (4.12) defined on $x = 0$ from (4.14) ($n = 0, 1, \cdots, N$) on $x \geq 0$ together with the termination condition (4.11) on $x = 0$. Thus, even though it is not standard, we call (4.14) ($n = 0, 1, \cdots, N$) defined on $x \geq 0$ together with (4.11) on $x = 0$ a *first-order formulation* of PMDLs in contrast to the *second-order formulation* given by (4.12) on $x = 0$. The reverse process is not trivial, since (4.12) on $x = 0$ does not directly imply (4.7) on $x \geq 0$.

The above presented interpretation is mathematically not complete. For instance, to be precise, one would have to prove that the kernel of $\frac{2}{L_n} - \partial_x$ is null. Our objective in this chapter being to introduce the idea of PMDLs, we will not go into further detail here and postpone the detailed analysis and discussion until we introduce CRBCs in Chapter 5.

4.3 Remarks

A family of PMDL formulations as given in (4.12) has a favorable structure from a theoretical viewpoint and may be readily implemented in the time-domain provided that the parameters L_n are pure imaginary or pure real. When L_n is to be complex, say $L_n = s + 1$, however, factors of $\frac{1}{L_n}$ appearing in the expressions for $\boldsymbol{S}_{ij}^{(n)}$ ($i, j = 1, 2$) given in (4.10) pose trouble since they would produce expensive convolution operators upon inverse Laplace transform into the time-domain. Numerical example for the scalar wave equations presented in [25] and [57] indeed used pure imaginary values for L_n. This issue could be remedied, for instance, by introducing another set of auxiliary functions just as we have done in our PML formulation in Chapter 3 for a cost of increasing computational effort.

A better solution, however, becomes available when we realize the equivalence between PMDLs and CRBCs [27] discussed in Chapter 5. CRBCs are originally written in first-order form just as in (4.14). Indeed, the first-order CRBCs were identified as the first-order PMDLs in the original paper [27]. We will propose for the first time a second-order formulation of CRBCs for the scalar wave equation in Chapter 5, which can be efficiently implemented in the time-domain, and identify it as the second-order formulation of PMDLs (4.12) in Appendix B.4. It thus resolves the above mentioned issue. In Chapter 5, we will also provide several pieces of analysis on our second-order CRBCs, or equivalently the second-order PMDLs, for the scalar wave equation, where the compact structure of (4.12) is indeed greatly appreciated.

Chapter 5

High-order absorbing boundary conditions

5.1 Introduction

In this chapter, we examine the high-order absorbing boundary conditions (HOABCs). HOABC methods were originally developed for and have mainly been applied to the scalar wave equation. In this context, it is widely recognized that HOABC methods surpass PML methods discussed in Chapter 3. A major difficulty that the PML methods experience in general is that there are parameters whose optimal values are hard to predict *a priori* and the accuracy of PML methods is indeed very sensitive to the choice of these parameters. In Chapter 3, we estimated them systematically using existing PML parameter choosing heuristics proposed in [36] and repeated in Appendix A.3 and we obtained satisfactory accuracy in our examples. Note, however, that this optimization has to be performed in a different framework from actual numerical implementations and has to be done every time one attempts to improve the accuracy of the PMLs, i.e. every time one increases the depth of the PML layers or increases the overall refinement level. This complexity is ascribed to the difficulty in estimating the amount of spurious reflections which arises upon discretization of PML layers. HOABCs, on the other hand, are only defined *on* the boundary and no discretization is performed in the normal direction to the boundary; it is in a sense discrete by nature. Parameters in HOABC methods are in general well understood physically and mathematically, and one can often choose them in a rather arbitrary fashion with the only parameter of significant impact on the accuracy being the order N. HOABC methods in general provide sharp reduction of reflection error with increasing order, which is accompanied by a minor increase in the computational cost. The estimate of the reflection coefficients in a continuous setting is explicitly given and essentially carries over to the discrete setting.

The origin of HOABC methods is the classic Higdon boundary conditions [31] which are designed to absorb waves which can be written as a superposition of arbitrarily many number N of purely propagating plane waves. The Higdon condition, however, involves

high-order temporal and normal derivatives and is not suited for numerical implementation. The Givoli-Neta boundary conditions [20] are a re-writing of the classic Higdon conditions from a high-order differential operator into a recursive system of first-order differential operators using auxiliary functions. It greatly facilitates numerical computations due to the lowering of the differential order. Hagstrom and Warburton [29] modified the recursion form of Givoli-Neta conditions in a manner that squares the reflection coefficient, symmetrizes the formulation, and improves its overall behavior. Hagstrom and Warburton [27] removed the plane wave assumption and derived absorbing boundary conditions based on a complete analytical representation of the wave equation, which directly takes into account waves which decay while propagating. This is the so-called *complete radiation boundary condition* (CRBC).

These absorbing boundary conditions are written in first-order in their original form and involve normal derivatives of the auxiliary functions on the boundary. For computational efficiency it is demanded that the auxiliary functions live only *on* the boundary. In that case representations of normal derivatives are not available, and one needs a reformulation of the original first-order system into one that does not involve normal derivatives. This point is strongly aided by Givoli and Neta's transformation of the normal boundary derivatives into second-order tangential boundary derivatives. The resultant formulation is thus referred as a *second-order formulation* in contrast to the original *first-order formulation*. Second-order formulations are derived for the Givoli-Neta conditions in [20] and for the Hagstrom-Warburton conditions and their derivatives in [29, 26, 28, 11]. In [26, 28] a treatment of purely evanescent waves was separately added and in [11] more general anisotropic and convective media were considered. In this chapter we develop a second-order formulation of CRBCs, which has not appeared in the literature.

Analytical work on well-posedness of several classes of HOABCs has also been done. A theory of well-posedness was established by Kreiss [37] for a general hyperbolic IBVPs, in which existence, uniqueness, and continuous dependence to the excitation of the solution to the hyperbolic system was attributed to the nonexistence of ill-posed modes, or nonexistence of *eigenvalues* whose real part is nonnegative. Based on Kreiss's general theory, Trefethen and Halpern [61] derived necessary and sufficient conditions for well-posedness for the isotropic wave equation with absorbing boundary conditions. Their theory can in principle be applied to the second-order formulations of Hagstrom-Warburton conditions, but not to those of CRBCs to be proposed in this chapter. As is shown in Appendix B.4, the second-order formulation of Hagstrom-Warburton conditions are essentially equivalent to those of perfectly matched discrete layers (PMDLs) specialized for purely propagating waves. In [58], well-posedness of this type of PMDLs for the anisotropic wave equation was studied based on the theory by Trefethen and Halpern. Well-posedness of the first-order Hagstrom-Warburton conditions with added treatment of purely evanescent waves in more general anisotropic media was considered in [11] based on the Kreiss's criteria. Note, however, that this proof for the first-order formulation does not necessarily transfer to that for the second-order formulation, since their equivalence is not trivial. This is partially because, while the second-order form is derived from the first-order form by using the fact that each

auxiliary function also satisfies the wave equation, we do not explicitly impose this condition in the second-order form. Moreover, the second-order form is a restriction of the modified first-order form to the boundary, which makes it a necessary condition of the first-order form. In this chapter, we prove the existence of bounded solution and its uniqueness separately for our proposed second-order formulations of CRBCs, and thus for the second-order formulations of general PMDLs as shown in Appendix B.4. Though we take a different approach, the underlying concept we employ is the same as that used in existing theories.

The actual numerical implementation of HOABCs is involved. This is especially true when the computational domain involves edges and corners; the most involved application in literature is for a two-dimensional domain with corners. In this chapter we also demonstrate the applicability of the proposed second-order formulation in an example on a three-dimensional domain with edges and a corner. To our knowledge, this is the first application of HOABC methods to a three-dimensional problem as well as to a three-dimensional problems with edges and corners. It should also be noted that, in numerical experiments with standard finite element methods, existing second-order formulations have been used with explicit Newmark's methods with mass lumping for primary field variables and implicit Newmark's methods for auxiliary variables [26, 11]. Implicit Newmark's methods have been used because the mass matrices for the auxiliary variables are not standard and there has been no lumping technique to take them into account. Beyond the point that it is developed for a more general CRBCs, our proposed second-order formulation has an advantage over existing formulations in that the resulting mass matrix has a special structure which allows for efficient time-integrations with standard explicit time-integrators such as explicit fourth-order Runge-Kutta methods.

Finally, we note that HOABC methods have also been applied to elastodynamics. In [2], the Hagstrom-Warburton condition [29] was modified and applied to the elastodynamic equation written in the first-order velocity-stress form and shown to have a great overall accuracy. On the other hand, application of HOABCs to the standard elastodynamic equation still requires further analytical and numerical work. For example, for a certain reformulation of the Hagstrom-Warburton condition, it was observed to lead an unstable system of equations [54]. Since there exists great advantages in using the standard elastodynamic equation, better mathematical understanding on this application is desired. We believe that the analytical work presented in this chapter for the scalar wave equation leads us to the right direction in this sense.

In Sec.5.2, we give a quick overview of the existing HOABC formulations. In Sec.5.3, we re-derive the CRBCs proposed in [27] and, in Sec.5.4, its second-order formulation is proposed for the first time. We present some analytical work on our second-order formulation in Sec.5.5 and then present a three-dimensional example in Sec.5.6. Sec.5.7 concludes.

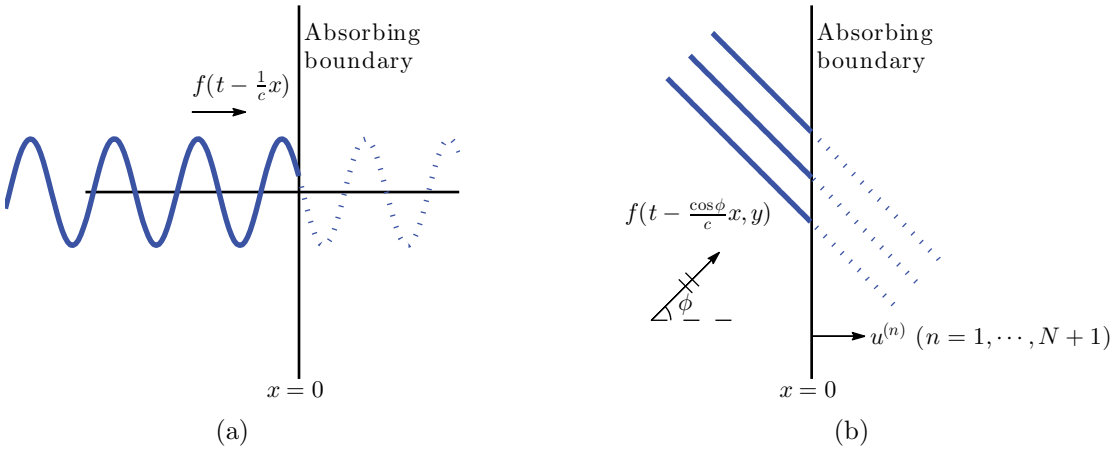


Figure 5.1: Schematics of harmonic waves in (a) one dimension (b) two dimensions. Auxiliary functions $u^{(n)}$ introduced in the Givoli-Neta conditions and Hagstrom-Warburton conditions are also shown in (b).

5.2 Motivation

In this section we consider a scalar wave propagation in d dimensions. This unbounded domain is truncated on $x = 0$ leaving a half-space $x < 0$ and we derive several classes of absorbing boundary conditions to be applied on $x = 0$ to represent the original unboundedness on $x > 0$. In other words we derive boundary conditions which absorb waves propagating in the positive x-direction. We refer to waves propagating in the positive and negative x-directions as rightward and leftward propagating waves. The absorbing boundary conditions introduced in this section can be derived by the harmonic assumption, $u = u(x, \boldsymbol{y})\, e^{\mathrm{i}\omega t}$, where $\mathrm{i} = \sqrt{-1}$ is the imaginary unit, $\boldsymbol{y} \in \mathbb{R}^{d-1}$ represents tangential coordinate variables, and ω is the frequency of the oscillation. A similar discussion is found in [11].

In one dimension ($d = 1$), the problem we consider is given by:

$$u_{,tt} - c^2 u_{,xx} = 0, \tag{5.1}$$

where c is a wave speed. Assuming a harmonic solution $u = u(x)\, e^{\mathrm{i}\omega t}$, (5.1) can be solved exactly on $x \in \mathbb{R}$ as:

$$u(x,t) = c^+ e^{\mathrm{i}\omega\left(t-\frac{1}{c}x\right)} + c^- e^{\mathrm{i}\omega\left(t+\frac{1}{c}x\right)}, \tag{5.2}$$

where c^+ and c^- are undetermined constants for rightward and leftward propagating waves. Since we are only interested in rightward propagating waves for the purpose of deriving absorbing boundary conditions on $x = 0$, we set $c^- = 0$. Then, the solution (5.2) can be rewritten in a simpler form as:

$$u(x,t) = f\left(t - \frac{1}{c}x\right), \tag{5.3}$$

for some function f; see Fig.5.1(a).

An *exact* absorbing boundary condition on $x = 0$ can be derived for this one-dimensional problem, by finding a condition that the exact solution (5.3) satisfies on $x = 0$, that is:

$$(\partial_t + c\partial_x)\, u = 0, \quad \text{on } x = 0. \tag{5.4}$$

Boundary condition (5.4) is the so-called *Sommerfeld radiation boundary condition.*

Equation (5.1) on $x < 0$ and the Sommerfeld condition (5.4) define the desired problem. The solution obtained by solving this set of equations is exact; one can readily check this by applying the Sommerfeld condition (5.4) to the general solution (5.2) and observing that c^- is indeed zero.

Exact absorbing boundary conditions, however, do not exist in higher spatial dimensions ($d \geq 2$). In d dimensions, we consider the following problem:

$$u_{,tt} - c^2 \left(u_{,xx} + \nabla^2_{\tan} u\right) = 0, \tag{5.5}$$

where $\nabla^2_{\tan}$ represents the tangential Laplacian operator on $\boldsymbol{y} \in \mathbb{R}^{d-1}$. A common way to find a solution to (5.5) is to assume the plane wave solution as:

$$u = u\,(x)\, e^{\mathrm{i}(\omega t + \boldsymbol{k}\cdot\boldsymbol{y})}. \tag{5.6}$$

Substituting (5.6) for u in (5.5) and solving the resultant ODE in terms of x, one obtains either one of the two types of solutions for each given ω and $\boldsymbol{k}$:

$$u = \begin{cases} c^+ \exp\left[\mathrm{i}\left(\omega t - k_x x + \boldsymbol{k}\cdot\boldsymbol{y}\right)\right] + c^- \exp\left[\mathrm{i}\left(\omega t + k_x x + \boldsymbol{k}\cdot\boldsymbol{y}\right)\right], & \text{if } \frac{\omega^2}{c^2} - k^2 > 0, \\ c^+ \exp\left[\mathrm{i}\omega t - \bar{k}_x x + \mathrm{i}\boldsymbol{k}\cdot\boldsymbol{y}\right] + c^- \exp\left[\mathrm{i}\omega t + \bar{k}_x x + \mathrm{i}\boldsymbol{k}\cdot\boldsymbol{y}\right], & \text{if } \frac{\omega^2}{c^2} - k^2 < 0, \end{cases} \tag{5.7}$$

where c^+ and c^- are undetermined constants, $k = \|\boldsymbol{k}\|$, and

$$\frac{\omega^2}{c^2} - k^2 := k_x^2 := -\bar{k}_x^2.$$

The first solution in (5.7) represents *purely propagating* waves, whose first and second terms represent rightward and leftward propagating waves. As in the one-dimensional problem leftward propagating waves are of no interest in this context, so we set $c^- = 0$. The second solution in (5.7) represents *purely evanescent* waves, which do not propagate in the x-direction. Its first term exponentially grows and second term exponentially decays into positive x-direction. For solutions to be bounded on $x > 0$, we set $c^- = 0$.

An important class of absorbing boundary conditions was developed in [30] based on the purely propagating wave solution in (5.7). For each given set of $(\omega, \boldsymbol{k})$, one can define an *incident* angle $\phi \in [0, \pi/2)$ by $\cos\phi = ck_x/\omega$. ϕ is an angle which the direction of wave propagation makes with respect to the positive x-direction. One can then rewrite the propagating wave solution in (5.7) in terms of ϕ as:

$$u = f\left(t - \frac{\cos\phi}{c}x, \boldsymbol{y}\right), \tag{5.8}$$

for some function f; see Fig.5.1(b). Note that we can find an exact absorbing boundary condition on $x = 0$ for the type of solutions given in (5.8) just as we have done in one-dimensional case, that is:

$$(\cos\phi\partial_t + c\partial_x)\, u = 0, \quad \text{on } x = 0. \tag{5.9}$$

This motivates one to write a solution to (5.5) in $\mathbb{R}^d$ as a superposition of plane waves of various incident angles $\phi_n \in [0, \pi/2)$ $(n = 0, 1, \cdots, N)$ for some integer N as:

$$u = \sum_{n=0}^{N} f_n \left(t - \frac{\cos\phi_n}{c} x, \boldsymbol{y} \right), \tag{5.10}$$

for some functions f_n $(n = 0, 1, \cdots, N)$.

Higdon boundary conditions are designed to absorb each component of the wave solution in (5.10) by successive application of the Sommerfeld-like boundary condition (5.9) as:

$$\left[\prod_{n=0}^{N} (\cos\phi_n\partial_t + c\partial_x) \right] u = 0, \quad \text{on } x = 0. \tag{5.11}$$

In (5.11), $\phi_n \in [0, \pi/2)$ $(n = 0, 1, \cdots, N)$ are regarded as a set of parameters of the Higdon conditions and N is the order.

The accuracy of an absorbing boundary condition as a tool to represent unboundedness is quantitatively measured by a *reflection coefficient*, which is defined to be the ratio of the size of the leftward propagating waves, or *reflection*, produced by the nonexactness of the absorbing boundary conditions to that of the rightward propagating waves, or the *incident waves*. The reflection coefficient for the Higdon conditions is obtained by applying (5.11) to the general propagating wave solution given in (5.7) and taking ratio of c^- to c^+, which gives:

$$\left| \frac{c^-}{c^+} \right| = \prod_{n=0}^{N} \left| \frac{\cos\phi_n - \cos\phi}{\cos\phi_n + \cos\phi} \right|. \tag{5.12}$$

Note that in (5.12), if one ϕ_n matches the actual incident angle of the propagating wave, the reflection coefficient is zero and the Higdon condition (5.11) is exact for such waves. Furthermore, since $|\cos\phi_n - \cos\phi| < |\cos\phi_n + \cos\phi|$ for $\phi, \phi_n \in [0, \pi/2)$, (5.12) states that the reflection coefficients are always less than unity and decreases with increasing N when Higdon conditions are applied to propagating waves.

Although increasing N improves accuracy of the Higdon conditions in terms of a small reflection coefficient, due to the presence of high-order derivatives, (5.11) can only be implemented at most up to $N = 2$ in practice. The Givoli-Neta boundary condition [20] fixed this point by introducing auxiliary functions and unfolding the product of the boundary operators in (5.11), which then reduce to a set of recursive relations satisfied by these auxiliary functions. This is given, for $n = 0, 1, \cdots, N$, by:

$$(\cos\phi_n\partial_t + c\partial_x)\, u^{(n)} = u^{(n+1)}, \quad \text{on } x \geq 0, \tag{5.13a}$$

$$u^{(N+1)} = 0, \quad \text{on } x = 0, \tag{5.13b}$$

where $u^{(n)}$ is a set of auxiliary functions with $u^{(0)} := u$, the primary unknown function; see Fig.5.1b. Equation (5.13b) is referred as a termination condition. Note that the recursions given in (5.13a) do not involve high-order derivatives unlike the Higdon conditions (5.11). One can readily show that the original Higdon conditions (5.11) can be constructed from the Givoli-Neta boundary condition (5.13). Equations (5.13) forms a model of HOABCs.

The Hagstrom-Warburton boundary conditions [29] are a modification of the Givoli-Neta conditions (5.13), which are defined, for $n = 1, \cdots, N$, by:

$$(\cos\phi_0\partial_t + c\partial_x)\, u^{(0)} = \partial_t u^{(1)}, \quad \text{on } x \geq 0, \tag{5.14a}$$

$$(\cos\phi_n\partial_t + c\partial_x)\, u^{(n)} = (\cos\phi_n\partial_t - c\partial_x)\, u^{(n+1)}, \quad x \geq 0, \tag{5.14b}$$

$$u^{(N+1)} = 0, \quad \text{on } x = 0, \tag{5.14c}$$

The symmetry of the Hagstrom-Warburton conditions (5.14) in terms of the orders of derivatives helps in numerical implementation. The irregularity of the first relation (5.14a) is merely for the implementation purpose. We now rewrite the Hagstrom-Warburton conditions in the form of the Higdon conditions (5.11). From (5.14a) and (5.14b), one readily obtains:

$$(\cos\phi_0\partial_t + c\partial_x)\left[\prod_{n=1}^{N}(\cos\phi_n\partial_t + c\partial_x)\right] u^{(0)} = \partial_t\left[\prod_{n=1}^{N}(\cos\phi_n\partial_t - c\partial_x)\right] u^{(N+1)}, \quad x \geq 0. \tag{5.15}$$

It is not immediate to rewrite (5.15) in the form of (5.11), since the termination condition (5.14c) is merely defined on $x = 0$ and normal derivatives of $u^{(N+1)}$ on $x = 0$ are unknown in general. Here, we arbitrarily apply the product term appearing on the left-hand side of (5.15) to (5.15) itself and obtain:

$$(\cos\phi_0\partial_t + c\partial_x)\left[\prod_{n=1}^{N}(\cos\phi_n\partial_t + c\partial_x)^2\right] u^{(0)} = \partial_t\left[\prod_{n=1}^{N}\left(\cos^2\phi_n\partial_t^2 - c^2\partial_x^2\right)\right] u^{(N+1)}, \quad x \geq 0, \tag{5.16}$$

As shown in [11] and also in Appendix B.1, each auxiliary function $u^{(n)}$ $(n = 1, \cdots, N+1)$ introduced in the Hagstrom-Warburton conditions (5.14) also satisfies the wave equation (5.5) on $x \geq 0$, which allows one to rewrite high-order x-derivatives of $u^{(N+1)}$ in (5.16) merely in terms of t- and y-derivatives. Finally applying the termination condition (5.14c) on $x = 0$, we obtain an equivalent Higdon condition as:

$$(\cos\phi_0\partial_t + c\partial_x)\left[\prod_{n=1}^{N}(\cos\phi_n\partial_t + c\partial_x)^2\right] u^{(0)} = 0, \quad \text{on } x = 0. \tag{5.17}$$

Note that the Hagstrom-Warburton conditions (5.17) involve squared operators in the product term in (5.17) and thus the reflection coefficient in (5.12). This is another advantage of using the Hagstrom-Warburton conditions (5.14) over the Givoli-Neta conditions (5.13).

The Givoli-Neta conditions (5.13) and the Hagstrom-Warburton conditions (5.14) are based on the Higdon conditions (5.11), and thus do not take into account the effect of purely evanescent waves given in (5.7). In [28] another set of recursions was added to the Hagstrom-Warburton conditions (5.14) to separately treat purely evanescent waves. In [11] the same correction was made, but for waves in a more general anisotropic media.

All absorbing boundary conditions mentioned so far are developed based on the plane wave solution (5.6). The most general HOABCs for the scalar wave equation was developed in [27] which directly takes into account the treatment of waves which decay while propagating. This HOABC formulation, the so-called *complete radiation boundary conditions* (CRBCs), is the main focus in the rest of this chapter.

5.3 Complete radiation boundary conditions

In this section we rederive complete radiation boundary conditions (CRBCs) proposed in [27]. We consider the wave equation on $\mathbb{R}^d$ with all sources and initial data located on $x < 0$ and derive absorbing boundary conditions to be applied on $x = 0$ upon truncation of the domain into a half-space $x < 0$. Explicitly writing the exact representation of the wave solution on $\mathbb{R}^d$, the CRBCs are derived as the conditions that this exact solution approximately satisfies on $x = 0$.

In real problems, multiple absorbing boundaries might intersect with each other on *edges* and at *corners*, which requires special treatments. These cases are discussed in Appendix B.2.

We consider the following IVP on $\mathbb{R}^d$:

$$\begin{aligned}
&\text{Find } u^\infty(x, \boldsymbol{y}, t) \in L^2\left(\mathbb{R}^d\right) \text{ such that} \\
&u^\infty_{,tt} - c^2\left(u^\infty_{,xx} + \nabla^2_{\tan} u^\infty\right) = f, \quad \text{on } \mathbb{R}^d, && \text{(5.18a)} \\
&u^\infty(x, \boldsymbol{y}, 0) = u_0, && \text{(5.18b)} \\
&u^\infty_{,t}(x, \boldsymbol{y}, 0) = u_1, && \text{(5.18c)}
\end{aligned}$$

where $f, u_0, u_1 \in L^2\left(\mathbb{R}^d\right)$ are a source term, an initial displacement, and an initial velocity, and are supported on $x < 0$. Assuming that u^∞ has exponential order $\eta_0 > 0$ in t, we take Laplace and Fourier transforms of (5.18) in the t- and $\boldsymbol{y}$-directions ($t \leftrightarrow s$, $\boldsymbol{y} \leftrightarrow \boldsymbol{k}$; $\mathrm{Re}(s) = \eta > \eta_0, \boldsymbol{y} \in \mathbb{R}^{d-1}$) to obtain (with an abuse of notation):

$$\begin{aligned}
&\text{Find } u^\infty(x, \boldsymbol{k}, s) \in L^2(x \in \mathbb{R}) \text{ such that} \\
&u^\infty_{,xx} - \left(s_c^2 + k^2\right) u^\infty = f, \quad \text{on } x \in \mathbb{R}, && \text{(5.19)}
\end{aligned}$$

where f represents the source term and initial data in the transformed domain and $k = \|\boldsymbol{k}\|$. Here and henceforth a subscript c represents division by c, e.g., $s_c = s/c$. Also, we define γ as

a branch of $\sqrt{s_c^2+k^2}$ such that $\gamma=+k$ at $s=0$ with branch cut on $(-\mathfrak{i}\infty,-\mathfrak{i}k)\cup(+\mathfrak{i}k,+\mathfrak{i}\infty)$; see Fig.5.2. The other branch of $\sqrt{s_c^2+k^2}$ is then represented as $-\gamma$. It is important to note that $\mathrm{Re}(\gamma)>0$ if $\mathrm{Re}(s)>0$.

The solution to Problem (5.19) on $x\geq 0$ can be represented by $u\left(0,\boldsymbol{k},s\right)e^{-\gamma x}$; see (5.58) for the precise expression. Inverse Fourier and Laplace transforms give the solution to the original problem (5.18) as:

$$u^{\infty}\left(x,\boldsymbol{y},t\right)=\frac{1}{(2\pi)^{d-1}}\frac{1}{2\pi\mathfrak{i}}\int\limits_{\mathbb{R}^{d-1}}\int\limits_{\eta-\mathfrak{i}\infty}^{\eta+\mathfrak{i}\infty}e^{st}e^{\mathfrak{i}\boldsymbol{k}\cdot\boldsymbol{y}}u(0,\boldsymbol{k},s)e^{-\gamma x}\mathrm{d}s\mathrm{d}\boldsymbol{k}. \tag{5.20}$$

We seek here another representation for γ which is amenable to further manipulations. Let $s=\eta+\mathfrak{i}\xi$ ($\eta>\eta_0$, $\xi\in\mathbb{R}$) and set:

$$\gamma=a+\mathfrak{i}b, \tag{5.21}$$

where $a,b\in\mathbb{R}$. Since $\eta=\mathrm{Re}(s)>0$ we have $a=\mathrm{Re}(\gamma)>0$; see Fig.5.2. Squaring both sides of (5.21) and comparing real and imaginary parts, one obtains:

$$b=\frac{\eta_c\xi_c}{a}, \tag{5.22a}$$

$$a^2-\frac{\eta_c^2\xi_c^2}{a^2}=\eta_c^2-\xi_c^2+k^2. \tag{5.22b}$$

Since the left-hand side of (5.22b) is an increasing function of a^2 and the left-hand side is less than or equal to the right-hand side at $a=\eta_c$, we have $a\geq\eta_c$. Therefore, there is $\phi=\phi\left(k,\xi;\eta\right)\in[0,\pi/2)$ such that:

$$a=\frac{\eta_c}{\cos\phi}. \tag{5.23}$$

Using (5.23), (5.22b) reduces to:

$$k^2-\left(\xi_c\sin\phi\right)^2=\left(\eta_c\tan\phi\right)^2,$$

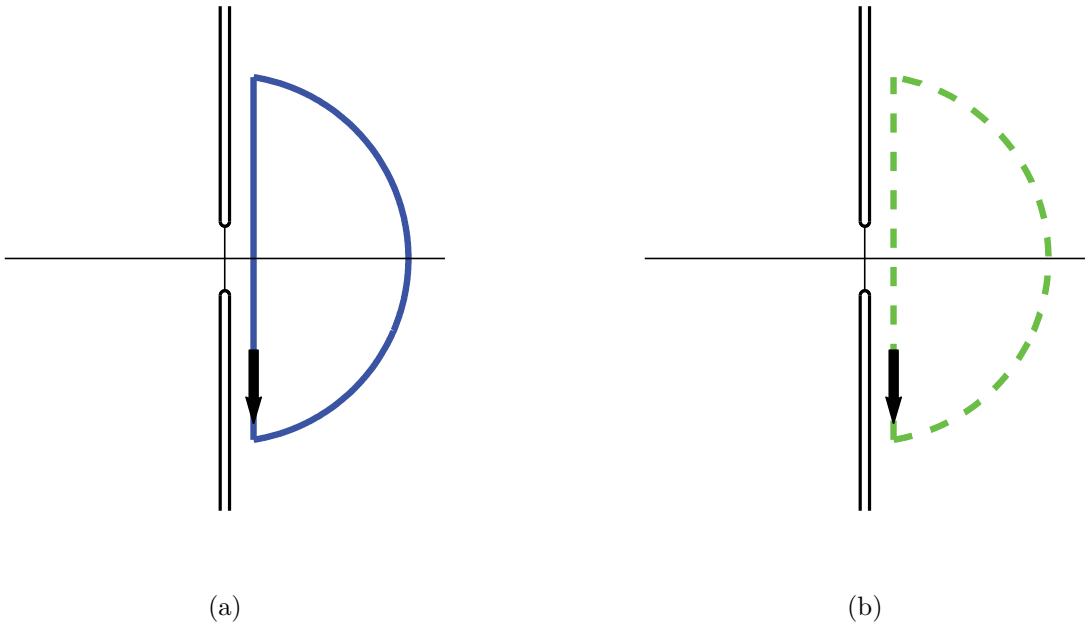
which can be parametrized as:

$$k=\eta_c\frac{\tan\phi}{\cos\psi}, \tag{5.24a}$$

$$\xi_c=\eta_c\frac{\tan\psi}{\cos\phi}, \tag{5.24b}$$

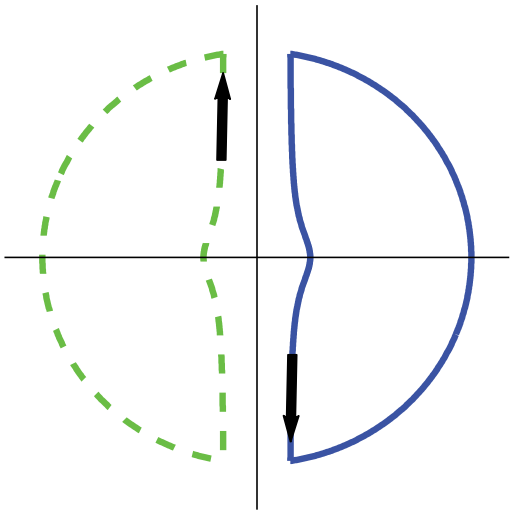
where $\psi=\psi\left(k,\xi;\eta\right)\in(-\pi/2,\pi/2)$. Using (5.23), (5.22a), and (5.24b), γ in (5.21) can be written as:

$$\gamma=\eta_c\frac{1}{\cos\phi}+\mathfrak{i}\xi_c\cos\phi,$$



(a)

(b)



(c)

Figure 5.2: Two sheets that compose a Riemann surface for $\sqrt{s_c^2 + k^2}$ with branch cut on $(-\mathrm{i}\infty, -\mathrm{i}k) \cup (+\mathrm{i}k, +\mathrm{i}\infty)$ corresponding to (a) $+\gamma$ and (b) $-\gamma$. (c) Value of $\sqrt{s_c^2 + k^2}$ on the contours depicted in (a) and (b).

or in terms of ϕ and ψ instead of ϕ and ξ as:

$$\gamma = \eta_c \left(\frac{1}{\cos\phi} + \mathfrak{i} \tan\psi \right). \tag{5.25}$$

Using (5.25) and appropriate changes of variables, (5.20) can be rewritten as:

$$\begin{aligned}
u^\infty(x, \boldsymbol{y}, t) &= \frac{1}{(2\pi)^d} \frac{1}{\mathfrak{i}} \int_{\eta - \mathfrak{i}\infty}^{\eta + \mathfrak{i}\infty} \int_{\mathbb{R}^{d-1}} e^{st} e^{-\gamma x} e^{\mathfrak{i}\boldsymbol{k}\cdot\boldsymbol{y}} u(0, \boldsymbol{k}, s) \mathrm{d}\boldsymbol{k} \mathrm{d}s \\
&= \frac{1}{(2\pi)^d} \int_{-\infty}^{\infty} \int_{\mathbb{R}^{d-1}} e^{st} e^{-\gamma x} e^{\mathfrak{i}\boldsymbol{k}\cdot\boldsymbol{y}} u(0, \boldsymbol{k}, s) \mathrm{d}\boldsymbol{k} \mathrm{d}\xi \\
&= \frac{1}{(2\pi)^d} \int_{-\infty}^{\infty} \int_{0}^{\infty} \int_{\mathbb{S}^{d-2}} e^{st} e^{-\gamma x} e^{\mathfrak{i}\boldsymbol{k}\cdot\boldsymbol{y}} u(0, \boldsymbol{k}, s) k^{d-2} \mathrm{d}\boldsymbol{\theta} \mathrm{d}k \mathrm{d}\xi \\
&= \frac{1}{(2\pi)^d} \int_{0}^{\frac{\pi}{2}} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \int_{\mathbb{S}^{d-2}} e^{\eta(1+\mathfrak{i}\frac{\tan\psi}{\cos\phi})\left(t - \frac{\cos\phi}{c}x\right)} e^{-\frac{\eta}{c}\frac{\sin^2\phi}{\cos\phi}x} e^{\mathfrak{i}\boldsymbol{k}\cdot\boldsymbol{y}} u(0, \boldsymbol{k}, s) k^{d-2} \det\boldsymbol{J} \mathrm{d}\boldsymbol{\theta} \mathrm{d}\psi \mathrm{d}\phi \\
&= \int_0^{\frac{\pi}{2}} \Phi\left(t - \frac{\cos\phi}{c}x, \boldsymbol{y}, \phi\right) e^{-\frac{\eta}{c}\frac{\sin^2\phi}{\cos\phi}x} \mathrm{d}\phi,
\end{aligned} \tag{5.26}$$

where $\boldsymbol{\theta} = \boldsymbol{k}/k$, $\boldsymbol{J}$ is the Jacobian of the transformation $(k, \xi) \leftrightarrow (\psi, \phi)$, and Φ is a function defined as:

$$\Phi(\tau, \boldsymbol{y}, \phi) := \frac{1}{(2\pi)^d} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \int_{\mathbb{S}^{d-2}} e^{\eta(1+\mathfrak{i}\frac{\tan\psi}{\cos\phi})\tau} e^{\mathfrak{i}\boldsymbol{k}\cdot\boldsymbol{y}} u(0, \boldsymbol{k}, s) k^{d-2} \det\boldsymbol{J} \mathrm{d}\boldsymbol{\theta} \mathrm{d}\psi.$$

The expression for $\det\boldsymbol{J}$ is given by:

$$\det\boldsymbol{J} = \frac{\eta^2}{c} \frac{1 - \sin^2\phi \sin^2\psi}{\cos^3\phi \cos^3\psi}.$$

Finally, we approximate the integral in (5.26) by quadrature as:

$$u^\infty(x, \boldsymbol{y}, t) \approx \sum_{l=0}^{N} w_l \Phi\left(t - \frac{\cos\phi_l}{c}x, \boldsymbol{y}, \phi_l\right) e^{-\frac{\eta}{c}\frac{\sin^2\phi_l}{\cos\phi_l}x}, \tag{5.27}$$

where ϕ_l and w_l $(l = 0, 1, \cdots, N)$ are sets of quadrature points and weights, respectively. Note that terms in the summation (5.27) represents waves which decay while propagating. The CRBCs are designed so that they are exact for wave components present in the summation (5.27), and thus given by, for $n = 0, 1, \cdots, N$:

$$\left(\frac{\cos\phi_n}{c}\partial_t + \partial_x + \frac{1}{cT}\frac{\sin^2\phi_n}{\cos\phi_n} \right) u^{(n)} = \left(\frac{\cos\bar{\phi}_n}{c}\partial_t - \partial_x + \frac{1}{cT}\frac{\sin^2\bar{\phi}_n}{\cos\bar{\phi}_n} \right) u^{(n+1)}, \quad \text{on } x \geq 0, \tag{5.28a}$$

$$u^{(N+1)} = 0, \quad \text{on } x = 0, \tag{5.28b}$$

where $u^{(0)} := u$ is an approximation to u^∞ and $u^{(n)}$ $(n = 1, \cdots, N+1)$ is a set of auxiliary functions defined on $x \geq 0$. The ϕ_n $(n = 0, 1, \cdots, N)$ can now be interpreted as a set of parameters and the $\bar{\phi}_n$ $(n = 0, 1, \cdots, N)$ is another set of parameters.

Initial conditions for $u^{(n)}$ are naturally given by:

$$u^{(n)}(x, \boldsymbol{y}, 0) = 0, \quad \text{on } x \geq 0, \tag{5.29a}$$

$$\partial_t u^{(n)}(x, \boldsymbol{y}, 0) = 0, \quad \text{on } x \geq 0. \tag{5.29b}$$

We then aim to solve the wave equation on $x < 0$ with the CRBCs (5.28) and (5.29) on $x = 0$.

5.4 Second-order formulation

Although auxiliary functions in (5.28) are defined on $x \geq 0$, for computational efficiency it is demanded that these auxiliary functions live only on the absorbing boundary $x = 0$. In that case, however, the normal derivatives on the boundary $x = 0$ present in (5.28) can not be represented, and one needs a reformulation of the original first-order systems (5.28) into a second-order form that only involves temporal and tangential derivatives. To our knowledge, second-order formulations of the CRBCs (5.28) have not appeared. In this section, we will systematically derive a family of second-order formulations for (5.28), which enables direct treatment of propagating *and* evanescent waves. A major advantage of our current second-order formulation over existing second-order formulations beyond the point that it is more general is that the resulting mass matrix has a special structure which allows for an efficient time-integrations. If continuous Galerkin methods are used for spatial discretization, one can employ explicit time-integrations in conjunction with conjugate gradient methods, even though the mass matrix is not perfectly symmetric. If discontinuous Galerkin methods are used, the mass matrix can be explicitly inverted.

In this section we derive a second-order formulation of the CRBCs (5.28) which is to be applied on the *face* $x = 0$. Second-order formulations on *edges* and *corners* are also required if the computational volume involves intersections of absorbing boundaries. These two cases are considered in Appendix B.3.

5.4.1 Face

All equations as well as functions in this section are defined on $x \geq 0$ unless otherwise noted. The first-order system of the CRBCs (5.28) can be written in a matrix form as:

$$\boldsymbol{M}_1 \boldsymbol{U}_{,x} = \frac{1}{c}\boldsymbol{M}_2 \boldsymbol{U}_{,t} + \frac{1}{cT}\boldsymbol{M}_3 \boldsymbol{U} - \boldsymbol{m}_1 u^{(0)}_{,x} - \frac{1}{c}\boldsymbol{m}_2 u^{(0)}_{,t} - \frac{1}{cT}\boldsymbol{m}_3 u^{(0)}, \tag{5.30}$$

where $\boldsymbol{M}_1$, $\boldsymbol{M}_2$, and $\boldsymbol{M}_3$ are $(N+1)\times(N+1)$ matrices of collections of coefficients of $u^{(i)}_{,x}$, $u^{(i)}_{,t}$, and $u^{(i)}$ $(i=1,...,N+1)$, $\boldsymbol{m}_1$, $\boldsymbol{m}_2$, and $\boldsymbol{m}_3$ are $(N+1)\times 1$ vectors of coefficients of $u^{(0)}_{,x}$, $u^{(0)}_{,t}$, and $u^{(0)}$, and $\boldsymbol{U}=[u^{(1)},...,u^{(N+1)}]^{\mathrm{T}}$ is a collection of auxiliary functions. $\boldsymbol{M}_1$ being invertible, (5.30) can be written as:

$$\boldsymbol{U}_{,x}=\frac{1}{c}\boldsymbol{M}_1^{-1}\boldsymbol{M}_2\boldsymbol{U}_{,t}+\frac{1}{cT}\boldsymbol{M}_1^{-1}\boldsymbol{M}_3\boldsymbol{U}-\boldsymbol{M}_1^{-1}\boldsymbol{m}_1u^{(0)}_{,x}-\frac{1}{c}\boldsymbol{M}_1^{-1}\boldsymbol{m}_2u^{(0)}_{,t}-\frac{1}{cT}\boldsymbol{M}_1^{-1}\boldsymbol{m}_3u^{(0)}. \tag{5.31}$$

Now taking another x-derivative in (5.31), one obtains:

$$\boldsymbol{U}_{,xx}=\frac{1}{c}\boldsymbol{M}_1^{-1}\boldsymbol{M}_2\boldsymbol{U}_{,tx}+\frac{1}{cT}\boldsymbol{M}_1^{-1}\boldsymbol{M}_3\boldsymbol{U}_{,x}-\boldsymbol{M}_1^{-1}\boldsymbol{m}_1u^{(0)}_{,xx}-\frac{1}{c}\boldsymbol{M}_1^{-1}\boldsymbol{m}_2u^{(0)}_{,tx}-\frac{1}{cT}\boldsymbol{M}_1^{-1}\boldsymbol{m}_3u^{(0)}_{,x}. \tag{5.32}$$

First spatial derivatives in the x-direction $\boldsymbol{U}_{,tx}$ and $\boldsymbol{U}_{,x}$ appearing in (5.32) which would not be represented if $\boldsymbol{U}$ is defined only on the boundary $x=0$ can be removed by successive use of (5.31) and we obtain:

$$\begin{aligned}
\boldsymbol{U}_{,xx}=&\frac{1}{c^2}\boldsymbol{M}_1^{-1}\boldsymbol{M}_2\boldsymbol{M}_1^{-1}\boldsymbol{M}_2\boldsymbol{U}_{,tt}\\
&+\frac{1}{c^2T}\left[\boldsymbol{M}_1^{-1}\boldsymbol{M}_2\boldsymbol{M}_1^{-1}\boldsymbol{M}_3+\boldsymbol{M}_1^{-1}\boldsymbol{M}_3\boldsymbol{M}_1^{-1}\boldsymbol{M}_2\right]\boldsymbol{U}_t\\
&+\frac{1}{c^2T^2}\boldsymbol{M}_1^{-1}\boldsymbol{M}_3\boldsymbol{M}_1^{-1}\boldsymbol{M}_3\boldsymbol{U}\\
&-\boldsymbol{M}_1^{-1}\boldsymbol{m}_1u^{(0)}_{,xx}\\
&-\frac{1}{c^2}\boldsymbol{M}_1^{-1}\boldsymbol{M}_2\boldsymbol{M}_1^{-1}\boldsymbol{m}_2u^{(0)}_{,tt}\\
&-\frac{1}{c^2T}\left[\boldsymbol{M}_1^{-1}\boldsymbol{M}_2\boldsymbol{M}_1^{-1}\boldsymbol{m}_3+\boldsymbol{M}_1^{-1}\boldsymbol{M}_3\boldsymbol{M}_1^{-1}\boldsymbol{m}_2\right]u^{(0)}_{,t}\\
&-\frac{1}{c^2T^2}\boldsymbol{M}_1^{-1}\boldsymbol{M}_3\boldsymbol{M}_1^{-1}\boldsymbol{m}_3u^{(0)}\\
&-\frac{1}{c}\left[\boldsymbol{M}_1^{-1}\boldsymbol{M}_2\boldsymbol{M}_1^{-1}\boldsymbol{m}_1+\boldsymbol{M}_1^{-1}\boldsymbol{m}_2\right]u^{(0)}_{,tx}-\frac{1}{cT}\left[\boldsymbol{M}_1^{-1}\boldsymbol{M}_3\boldsymbol{M}_1^{-1}\boldsymbol{m}_1+\boldsymbol{M}_1^{-1}\boldsymbol{m}_3\right]u^{(0)}_{,x}.
\end{aligned} \tag{5.33}$$

As shown in [11] and also in Appendix B.1, each component of $\boldsymbol{U}$ defined in (5.28a) also satisfies the wave equation (5.5) on $x\geq 0$, i.e. :

$$\boldsymbol{U}_{,tt}-c^2\left(\boldsymbol{U}_{,xx}+\nabla^2_{\tan}\boldsymbol{U}\right)=\boldsymbol{0}. \tag{5.34}$$

Substituting (5.33) for $\boldsymbol{U}_{,xx}$ in the wave equation (5.34), one obtains a system of equations which does not involve x-derivatives:

$$\left[\boldsymbol{M}_1^{-1}\boldsymbol{M}_2\boldsymbol{M}_1^{-1}\boldsymbol{m}_2+\boldsymbol{M}_1^{-1}\boldsymbol{m}_1\right]u^{(0)}_{,tt}+\left[\boldsymbol{I}-\boldsymbol{M}_1^{-1}\boldsymbol{M}_2\boldsymbol{M}_1^{-1}\boldsymbol{M}_2\right]\boldsymbol{U}_{,tt}$$

$$
\begin{aligned}
&+\frac{1}{T}\left[\boldsymbol{M}_1^{-1}\boldsymbol{M}_2\boldsymbol{M}_1^{-1}\boldsymbol{m}_3+\boldsymbol{M}_1^{-1}\boldsymbol{M}_3\boldsymbol{M}_1^{-1}\boldsymbol{m}_2\right]u^{(0)}_{,t}\\
&-\frac{1}{T}\left[\boldsymbol{M}_1^{-1}\boldsymbol{M}_2\boldsymbol{M}_1^{-1}\boldsymbol{M}_3+\boldsymbol{M}_1^{-1}\boldsymbol{M}_3\boldsymbol{M}_1^{-1}\boldsymbol{M}_2\right]\boldsymbol{U}_{,t}\\
&+\frac{1}{T^2}\boldsymbol{M}_1^{-1}\boldsymbol{M}_3\boldsymbol{M}_1^{-1}\boldsymbol{m}_3u^{(0)}-\frac{1}{T^2}\boldsymbol{M}_1^{-1}\boldsymbol{M}_3\boldsymbol{M}_1^{-1}\boldsymbol{M}_3\boldsymbol{U}\\
&-c^2\boldsymbol{M}_1^{-1}\boldsymbol{m}_1\nabla^2_{\tan}u^{(0)}-c^2\nabla^2_{\tan}\boldsymbol{U}\\
&+\frac{1}{T}\left[\boldsymbol{M}_1^{-1}\boldsymbol{M}_2\boldsymbol{M}_1^{-1}\boldsymbol{m}_1+\boldsymbol{M}_1^{-1}\boldsymbol{m}_2\right]u^{(a)}_{,t}+\frac{1}{T^2}\left[\boldsymbol{M}_1^{-1}\boldsymbol{M}_3\boldsymbol{M}_1^{-1}\boldsymbol{m}_1+\boldsymbol{M}_1^{-1}\boldsymbol{m}_3\right]u^{(a)}=\boldsymbol{0},
\end{aligned}
\tag{5.35}
$$

where $u^{(0)}_{,xx}$ was replaced by $u^{(0)}_{,tt}$ and $\nabla^2_{\tan}u^{(0)}$ using the wave equation for $u^{(0)}$ and we introduced new function $u^{(a)}=(cT)\,u^{(0)}_{,x}$, which is a key idea of our second-order formulation.

Equation (5.35) holds on $x\geq 0$, but we here restrict it to $x=0$ so that auxiliary functions only live on $x=0$, which is essential for real applications. Applying the termination condition (5.28b) by setting $u^{(N+1)}=0$ in (5.35) evaluated on $x=0$, one obtains:

$$
\begin{aligned}
\boldsymbol{M}'_{tt}\begin{pmatrix}u^{(0)}\\ \boldsymbol{U}\end{pmatrix}_{,tt}=&-\frac{1}{T}\boldsymbol{M}'_t\begin{pmatrix}u^{(0)}\\ \boldsymbol{U}\end{pmatrix}_{,t}-\frac{1}{T^2}\boldsymbol{M}_0\begin{pmatrix}u^{(0)}\\ \boldsymbol{U}\end{pmatrix}-c^2\boldsymbol{M}_{\tan}\nabla^2_{\tan}\begin{pmatrix}u^{(0)}\\ \boldsymbol{U}\end{pmatrix}\\
&-\frac{1}{T}\boldsymbol{m}'_tu^{(a)}_{,t}-\frac{1}{T^2}\boldsymbol{m}'_0u^{(a)},
\end{aligned}
\tag{5.36}
$$

where (with an abuse of notation), $\boldsymbol{U}=\left[u^{(1)},...,u^{(N)}\right]^{\mathrm{T}}$, $\boldsymbol{M}'_{tt}$, $\boldsymbol{M}'_t$, $\boldsymbol{M}_0$, and $\boldsymbol{M}_{\tan}$ are $(N+1)\times(N+1)$ matrices, and $\boldsymbol{m}'_t$ and $\boldsymbol{m}'_0$ are $(N+1)\times 1$ vectors. Note that we have up to second-order temporal derivatives of $u^{(0)}$ and $\boldsymbol{U}$ and first-order temporal derivative of $u^{(a)}$. We now rewrite (5.36) in first order in time as:

$$
\begin{pmatrix}u^{(0)}\\ \boldsymbol{U}\end{pmatrix}_{,t}=\begin{pmatrix}v^{(0)}\\ \boldsymbol{V}\end{pmatrix},
\tag{5.37a}
$$

$$
\begin{aligned}
\boldsymbol{M}_{tt}\begin{pmatrix}\frac{1}{T}u^{(a)}\\ \boldsymbol{V}\end{pmatrix}_{,t}=&-\frac{1}{T}\boldsymbol{M}_t\begin{pmatrix}\frac{1}{T}u^{(a)}\\ \boldsymbol{V}\end{pmatrix}-\frac{1}{T^2}\boldsymbol{M}_0\begin{pmatrix}u^{(0)}\\ \boldsymbol{U}\end{pmatrix}-c^2\boldsymbol{M}_{\tan}\nabla^2_{\tan}\begin{pmatrix}u^{(0)}\\ \boldsymbol{U}\end{pmatrix}\\
&-\boldsymbol{m}_tv^{(0)}_{,t}-\frac{1}{T}\boldsymbol{m}_0v^{(0)},
\end{aligned}
\tag{5.37b}
$$

where $\boldsymbol{M}_{tt}$ and $\boldsymbol{M}_t$ are new $(N+1)\times(N+1)$ matrices and $\boldsymbol{m}_t$ and $\boldsymbol{m}_0$ are new $(N+1)\times 1$ vectors. $\boldsymbol{M}_0$ and $\boldsymbol{M}_{\tan}$ are unchanged from (5.36). $\boldsymbol{M}_{tt}$ being invertible, one can rewrite (5.37) to obtain the final form of our second-order formulation as:

$$
\begin{pmatrix}u^{(0)}\\ \boldsymbol{U}\end{pmatrix}_{,t}=\begin{pmatrix}v^{(0)}\\ \boldsymbol{V}\end{pmatrix},
\tag{5.38a}
$$

$$
\begin{pmatrix}\frac{1}{T}u^{(a)}\\ \boldsymbol{V}\end{pmatrix}_{,t}=-\frac{1}{T}\boldsymbol{M}_t\begin{pmatrix}\frac{1}{T}u^{(a)}\\ \boldsymbol{V}\end{pmatrix}-\frac{1}{T^2}\boldsymbol{M}_0\begin{pmatrix}u^{(0)}\\ \boldsymbol{U}\end{pmatrix}-c^2\boldsymbol{M}_{\tan}\nabla^2_{\tan}\begin{pmatrix}u^{(0)}\\ \boldsymbol{U}\end{pmatrix}-\boldsymbol{m}_tv^{(0)}_{,t}-\frac{1}{T}\boldsymbol{m}_0v^{(0)},
$$

(5.38b)

abusing notation in an obvious way. The second-order formulation (5.38) is solved on $x = 0$ in conjunction with the wave equation for $u^{(0)}$ on $x < 0$ written also in first order in time form. The IBVP that we solve is thus given by:

Find $u^{(0)}$, $v^{(0)}$ on $x < 0$and $u^{(a)}$, $\boldsymbol{U}$, and $\boldsymbol{V}$ on $x = 0$ such that

$$u^{(0)}_{,t} = v^{(0)}, \quad \text{on } x < 0, \tag{5.39a}$$

$$v^{(0)}_{,t} = c^2\nabla^2 u^{(0)}, \quad \text{on } x < 0, \tag{5.39b}$$

$$u^{(0)}\,(x, \boldsymbol{y}, 0) = u_0, \quad \text{on } x < 0, \tag{5.39c}$$

$$u^{(0)}_{,t}\,(x, \boldsymbol{y}, 0) = u_1, \quad \text{on } x < 0, \tag{5.39d}$$

$$(cT)\,u_{,x} = u^{(a)}, \quad \text{on } x = 0. \tag{5.39e}$$

together with the absorbing boundary conditions (5.38) on $x = 0$ and homogeneous initial data for $u^{(n)}$ $(n = 1, \cdot, N)$.

A strong advantage of the second-order formulation (5.38), beyond the fact that it uses the CRBCs, is that it allows for an explicit time-integration in numerical simulations. Explicit time-integration involves evaluation of residuals, i.e. the right-hand side of equations (5.39a), (5.39b), and (5.38), which requires inversion of mass matrices. In existing second-order formulations, normal derivatives on $x = 0$ are represented by time-derivatives of auxiliary functions, which adds couplings in the mass matrix. In our second-order formulation, on the other hand, normal derivatives on $x = 0$ are explicitly represented by $u^{(a)}$, which only adds elements into the stiffness matrix. As a result, the mass matrix is only one-way coupled; time-derivatives of auxiliary functions are absent in the equations for $v^{(0)}$ as mentioned above, but those of $v^{(0)}$ still appear in the equations for the auxiliary functions as seen in the fourth term on the right-hand side of (5.38b). This allows one to first solve for $v^{(0)}_{,t}$ and then the residuals for the auxiliary functions. Similar arguments hold when edges and corners are involved. This point is discussed in detail in Sec.5.6 with a concrete example.

In Appendix B.4 we show equivalence of our second-order formulation of CRBCs developed in this section (with $\phi_n = \bar{\phi}_n$) and the second-order formulation of Perfectly Matched Discrete Layers (PMDLs) for the more general system of equations (4.1) considered in Chapter 4. The proof is based on a two-dimensional problems, but extension to three-dimensional problems should be straightforward.

5.5 Analysis

In this section we study the existence and the uniqueness of the solution to the isotropic wave equation to which our proposed second-order formulations of CRBCs (5.38) with $\phi_n = \bar{\phi}_n$ are applied.

We consider the wave equation on a half-space $x < 0$ in three dimensions ($d = 3$) and apply our second-order formulation of the CRBCs (5.28) on $x = 0$. The IBVP that we solve is given by:

$$
\begin{aligned}
&\text{Find } u\left(x, \boldsymbol{y}, t\right) \in L^2\left(x < 0\right) \text{ such that} \\
&u^{(0)}_{,tt} - c^2\left(u^{(0)}_{,xx} + \nabla^2_{\tan} u^{(0)}\right) = f, \quad \text{on } \mathbb{R}^d_-, && \text{(5.40a)} \\
&u^{(0)}\left(x, \boldsymbol{y}, 0\right) = u_0, && \text{(5.40b)} \\
&u^{(0)}_{,t}\left(x, \boldsymbol{y}, 0\right) = u_1, && \text{(5.40c)} \\
&\mathcal{B}^N_H u^{(0)} = 0, \quad \text{on } x = 0, && \text{(5.40d)}
\end{aligned}
$$

where f, u_0, and u_1 are body-force, initial displacement, and initial velocity, respectively, which are supported on $x < 0$ and $\mathcal{B}^N_H$ represents the second-order CRBCs (5.38) of order N applied on $x = 0$. Assuming that the solution has exponential order $\eta_0 > 0$ in t, we take Laplace and Fourier transforms in the t- and y-directions ($t \leftrightarrow s$, $\boldsymbol{y} \leftrightarrow \boldsymbol{k}$; $s = \eta + \mathrm{i}\xi$, $\eta > \eta_0$, $\xi \in \mathbb{R}$, $\boldsymbol{k} \in \mathbb{R}^{d-1}$) to obtain a transformed problem:

$$
\begin{aligned}
&\text{Find } u\left(x, \boldsymbol{k}, s\right) \in L^2\left(x < 0\right) \text{ such that} \\
&u^{(0)}_{,xx} - \left(s_c^2 + k^2\right) u^{(0)} = f, \quad \text{on } x < 0, && \text{(5.41a)} \\
&\mathcal{B}^N_H u^{(0)} = 0, \quad \text{on } x = 0, && \text{(5.41b)}
\end{aligned}
$$

where (with an abuse of notation) f represents transformed body-force and initial data, $\mathcal{B}^N_H$ is a transformed boundary operator, and $k = \|\boldsymbol{k}\|$.

In Appendix B.4 we show that our second-order formulation (5.38) with $\phi_n = \bar{\phi}_n$ is equivalent to the second-order formulation of PMDLs (4.12) for a more general equation including the scalar wave equation. Since the latter has a simpler structure and is suited for analysis, we restrict (4.12) for the wave equation and use it in this section instead of directly working on (5.38). The specialized form of (4.12) we analyze is:

$$
\begin{pmatrix} u^{(0)}_{,x} \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}
+
\begin{pmatrix}
S^{(0)}_{11} & S^{(0)}_{12} & & & & \\
S^{(0)}_{21} & S^{(0)}_{22} + S^{(1)}_{11} & S^{(1)}_{12} & & & \\
& S^{(1)}_{21} & S^{(1)}_{22} + S^{(2)}_{11} & S^{(2)}_{12} & & \\
& & \ddots & & & \\
& & & S^{(N-2)}_{22} + S^{(N-1)}_{11} & S^{(N-1)}_{12} & \\
& & & S^{(N-1)}_{21} & S^{(N-1)}_{22} + S^{(N)}_{11}
\end{pmatrix}
\begin{pmatrix} u^{(0)} \\ u^{(1)} \\ u^{(2)} \\ \vdots \\ u^{(N-1)} \\ u^{(N)} \end{pmatrix}
=
\begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix},
\tag{5.42}
$$

where

$$
S^{(n)}_{11} = S^{(n)}_{22} = \frac{1}{2}\left(\frac{\gamma^2}{\gamma_n} + \gamma_n\right), \tag{5.43a}
$$

$$S_{12}^{(n)} = S_{21}^{(n)} = \frac{1}{2}\left(\frac{\gamma^2}{\gamma_n} - \gamma_n\right), \tag{5.43b}$$

where $\gamma_n = \frac{\cos\phi_n}{c}s + \frac{\sin^2\phi_n}{\cos\phi_n}$ as defined in (B.23). In (5.42), one can recursively eliminate $u^{(n)}$ in descending order in n to represent $u_{,x}^{(0)}$ merely in terms of $u^{(0)}$. Specifically, we define α_n as:

$$u^{(n+1)} := \alpha_n u^{(n)}, \quad \text{(no sum on } n\text{)}. \tag{5.44}$$

Since $u^{(N+1)} = 0$ by definition, we have $\alpha_N := 0$. Note that the n-th row of equation $(n = 1, \cdots, N)$ in (5.42) can be written as:

$$S_{21}^{(n-1)}u^{(n-1)} + \left(S_{22}^{(n-1)} + S_{11}^{(n)}\right)u^{(n)} + S_{12}^{(n)}u^{(n+1)} = 0. \tag{5.45}$$

We substitute (5.44) for $u^{(n+1)}$ in (5.45) and solve for $u^{(n)}$ to obtain:

$$u^{(n)} = \frac{-S_{21}^{(n-1)}}{S_{22}^{(n-1)} + \left(S_{11}^{(n)} + \alpha_n S_{12}^{(n)}\right)}u^{(n-1)} := \alpha_{n-1}u^{(n-1)}. \tag{5.46}$$

Equation (5.46) defines a recursion for α_n as:

$$\alpha_{n-1} = \frac{-S_{21}^{(n-1)}}{S_{22}^{(n-1)} + \left(S_{11}^{(n)} + \alpha_n S_{12}^{(n)}\right)}. \tag{5.47}$$

Knowing that $\alpha_N = 0$, one can recursively compute α_n $(n = N-1, N-2, \cdots, 1)$ using (5.47). Then, the 0-th row of equation in (5.42) can be written merely in terms of $u^{(0)}$ as:

$$u_{,x}^{(0)} + r^N u^{(0)} = 0, \quad \text{on } x = 0, \tag{5.48}$$

where

$$r^N\left(k, s; \gamma_0, \gamma_1, \cdots, \gamma_N\right) = S_{11}^{(0)} + \alpha_1 S_{12}^{(0)}. \tag{5.49}$$

Equation (5.48) defines the boundary operator $\mathcal{B}_H^N$ in (5.41b). Note that the exact representation for the rightward propagating wave on $x = 0$ is given by $u\left(x, \boldsymbol{k}, s\right) = u\left(0, \boldsymbol{k}, s\right)e^{-\gamma x}$ as used in Sec.5.3. Substituting this for $u^{(0)}$ in (5.48), we have:

$$\left(-\gamma + r^N\right)u^{(0)} = 0,$$

which implies that if $r^N = \gamma$ for all k and s, the N-th order HOABCs/PMDLs is exact; they produce no reflection. This will not be the case, however, since γ is irrational and r^N is rational in terms of k and s. Thus, r^N can be regarded as a rational approximation of γ,

and $\gamma - r^N$ represents the error due to this approximation. In Sec.5.5.1, we present a closed form expression for $\gamma - r^N$.

We obtained (5.48) by recursively eliminating auxiliary functions $u^{(n)}$ $(n = N, N-1, \cdots, 1)$ in the matrix equation (5.42). By symmetry of this process, if we eliminate only up to $u^{(n)}$ $(n = N, N-1, \cdots, 2)$, we are left with equations involving $u^{(0)}$ and $u^{(1)}$ given by:

$$\begin{pmatrix} u_{,x}^{(0)} \\ 0 \end{pmatrix} + \begin{bmatrix} S_{11}^0 & S_{12}^0 \\ S_{21}^0 & S_{22}^0 + r^{N-1}\left(k, s; \gamma_1, \cdots, \gamma_N\right) \end{bmatrix} \begin{pmatrix} u^{(0)} \\ u^{(1)} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{5.50}$$

Noting that $u_{,x}^{(0)}$ is given in (5.48), one obtains from (5.50) a relation between r^N and r^{N+1} as:

$$r^N\left(k, s; \gamma_0, \gamma_1, \cdots, \gamma_N\right) = S_{11}^{(0)} - \frac{S_{12}^{(0)} S_{21}^{(0)}}{S_{22}^{(0)} + r^{N-1}\left(k, s; \gamma_1, \cdots, \gamma_N\right)}. \tag{5.51}$$

In the sequel, we adopt a shorthand notation of $r^N\left(k, s\right)$ for $r^N\left(k, s; \gamma_0, \gamma_1, \cdots, \gamma_N\right)$ for simplicity, but it should cause no confusion.

5.5.1 Preliminaries

We first prove the following identity which plays an important role in our analysis:

$$\gamma - r^N = \frac{p^N\left(\gamma\right)}{q^N\left(\gamma\right)}, \quad \forall\{\gamma_0, \gamma_1, \cdots, \gamma_N\} \tag{5.52}$$

where

$$p^N\left(\gamma\right) = \prod_{n=0}^{N} \left(\gamma - \gamma_n\right)^2 \in \mathbb{P}_{2N+2}\left(\gamma\right),$$
$$q^N\left(\gamma\right) \in \mathbb{P}_{2N}\left(\gamma\right).$$

The identity (5.52) gives an explicit expression for the error due to the rational approximation r^N to the irrational function γ.

Proof. By induction.
I: $N = 0$. We set $\alpha_1 = 0$ in (5.49) to obtain $r^{N=0} = S_{11}^0 = \frac{1}{2}\left(\frac{\gamma^2}{\gamma_0} + \gamma_0\right)$. We then have:

$$\gamma - r^{N=0} = \frac{\left(\gamma - \gamma_0\right)^2}{-2\gamma_0}.$$

II: Assume that identity (5.52) holds for $N = K$ and consider a general CRBCs of order $K+1$ with a set of parameters $\{\gamma_0, \gamma_1, \cdots, \gamma_{K+1}\}$. Since identity (5.52) holds for any set of parameters for $N = K$, one can write by assumption:

$$r^K = \gamma - \frac{p^K\left(\gamma\right)}{q^K\left(\gamma\right)}, \quad \text{for } \{\gamma_1, \cdots, \gamma_{K+1}\}, \tag{5.54}$$

where

$$p^K(\gamma) = \prod_{\kappa=1}^{K+1} (\gamma - \gamma_\kappa)^2 \in \mathbb{P}_{2K+2}(\gamma), \tag{5.55a}$$

$$q^K(\gamma) \in \mathbb{P}_{2K}(\gamma). \tag{5.55b}$$

r^{K+1} can be expressed in terms of r^K using (5.51). Together with the expressions for $S_{ij}^{(0)}$ given in (5.43) and an expression for r^K in (5.54), one obtains:

$$\gamma - r^{K+1} = \frac{(\gamma - \gamma_0)^2 p^K}{(\gamma + \gamma_0)^2 q^K - 2\gamma_0 p^K} := \frac{p^{K+1}}{q^{K+1}}. \tag{5.56}$$

Using (5.55), one obtains:

$$p^{K+1}(\gamma) = \prod_{\kappa=0}^{K+1} (\gamma - \gamma_\kappa)^2 \in \mathbb{P}_{2(K+1)+2}(\gamma),$$

$$q^{K+1}(\gamma) \in \mathbb{P}_{2(K+1)}(\gamma). \qquad (q.e.d.)$$

Note that from (5.43), one can see that $r^N(\gamma) = r^N(-\gamma)$. Then from the identity (5.52) one also has:

$$\gamma + r^N(\gamma) = -\frac{p^N(-\gamma)}{q^N(-\gamma)}, \tag{5.57}$$

where

$$p^N(-\gamma) = \prod_{\kappa=0}^{n} (\gamma + \gamma_\kappa)^2 \in \mathbb{P}_{2N+2}(\gamma),$$

$$q^N(-\gamma) \in \mathbb{P}_{2N}(\gamma).$$

5.5.2 Existence

In this section we prove existence of a weakly bounded solution to the half-space problem (5.40). To this end, we construct a solution based on the exact solution u^∞ to the unbounded-domain problem (5.18). It is convenient to work on the corresponding transformed equations (5.19) on $x \in \mathbb{R}$ and (5.41) on $x < 0$.

The solution to the problem (5.19) is given by:

$$u^\infty = c_1(x, \boldsymbol{k}, s)\, e^{+\gamma x} + c_2(x, \boldsymbol{k}, s)\, e^{-\gamma x},$$

where

$$c_{1,x} = \frac{e^{-\gamma x} f(x, \boldsymbol{k}, s)}{2\gamma}, \quad c_{2,x} = -\frac{e^{+\gamma x} f(x, \boldsymbol{k}, s)}{2\gamma}.$$

$u^{\infty}(x, \boldsymbol{k}, s)$ being $L^2(x \in \mathbb{R})$, or applying boundary conditions at $\pm\infty$, one obtains:

$$u^{\infty} = -\int_{-\infty}^{x} \frac{e^{-\gamma(x-\tau)} f(\tau, \boldsymbol{k}, s)}{2\gamma} \mathrm{d}\tau - \int_{x}^{+\infty} \frac{e^{+\gamma(x-\tau)} f(\tau, \boldsymbol{k}, s)}{2\gamma} \mathrm{d}\tau. \tag{5.58}$$

One way to construct a solution $u^{(0)}$ to the half-space problem (5.41) is to add to (5.58) a homogeneous solution to (5.41a) which is $L^2(x < 0)$ as:

$$u^{(0)} = u^{\infty} + c^{-}(\boldsymbol{k}, s) e^{+\gamma x}, \tag{5.59}$$

which clearly satisfies equation (5.41a). Applying the boundary condition (5.41b), or (5.48), one can solve for c^{-} as:

$$c^{-}(\boldsymbol{k}, s) = -\left(\frac{\gamma - r^N}{\gamma + r^N}\right) \int_{-\infty}^{0} \frac{e^{+\gamma\tau} f(\tau, \boldsymbol{k}, s)}{2\gamma} \mathrm{d}\tau + \int_{0}^{+\infty} \frac{e^{-\gamma\tau} f(\tau, \boldsymbol{k}, s)}{2\gamma} \mathrm{d}\tau.$$

Noting that $f(x, \boldsymbol{k}, s) = 0$ on $x \geq 0$,

$$c^{-}(\boldsymbol{k}, s) = -\left(\frac{\gamma - r^N}{\gamma + r^N}\right) \int_{-\infty}^{0} \frac{e^{+\gamma\tau} f(\tau, \boldsymbol{k}, s)}{2\gamma} \mathrm{d}\tau. \tag{5.60}$$

Then, from (5.59) and (5.60), a solution to the half-space problem (5.41) is found to be:

$$u^{(0)} = u^{\infty} - \left(\frac{\gamma - r^N}{\gamma + r^N}\right) \int_{-\infty}^{0} \frac{e^{+\gamma\tau} f(\tau, \boldsymbol{k}, s)}{2\gamma} \mathrm{d}\tau \cdot e^{+\gamma x},$$

which can be rewritten using (5.58) as:

$$u^{(0)}(x, \boldsymbol{k}, s) = u^{\infty}(x, \boldsymbol{k}, s) - \left(\frac{\gamma - r^N}{\gamma + r^N}\right) u^{\infty}(-x, \boldsymbol{k}, s), \quad \text{on } x < 0. \tag{5.61}$$

Now we prove the following identity:

$$\frac{\gamma - r^N}{\gamma + r^N} = -\prod_{n=0}^{N} \left(\frac{\gamma - \gamma_n}{\gamma + \gamma_n}\right)^2, \quad \forall\{\gamma_0, \gamma_1, \cdots, \gamma_N\}. \tag{5.62}$$

Proof. By induction.
I: $N = 0$. We have $\gamma - r^{N=0} = \frac{(\gamma-\gamma_0)^2}{-2\gamma_0}$, from which one obtains,

$$-\left(\gamma + r^{N=0}\right) = -2\gamma + \left(\gamma - r^{N=0}\right) = \frac{(\gamma + \gamma_0)^2}{-2\gamma_0}.$$

Then,

$$\frac{\gamma - r^{N=0}}{\gamma + r^{N=0}} = -\left(\frac{\gamma - \gamma_0}{\gamma + \gamma_0}\right)^2.$$

II: Suppose the identity (5.62) holds when $N = K$ and consider a general CRBCs of order $K+1$ with a set of parameters $\{\gamma_0, \gamma_1, \cdots, \gamma_{K+1}\}$. Since identity (5.62) holds for any set of parameters for $N = K$, one can write by assumption:

$$\frac{\gamma - r^K}{\gamma + r^K} = -\prod_{\kappa=1}^{K+1} \left(\frac{\gamma - \gamma_\kappa}{\gamma + \gamma_\kappa}\right)^2.$$

From the identity (5.56) one has:

$$\gamma - r^{K+1} = \frac{(\gamma - \gamma_0)^2 p^K}{(\gamma + \gamma_0)^2 q^K - 2\gamma_0 p^K},$$

from which one obtains:

$$-\left(\gamma + r^{K+1}\right) = -2\gamma + \left(\gamma - r^{K+1}\right) = \frac{(\gamma + \gamma_0)^2 \left(-2\gamma q^K + p^K\right)}{(\gamma + \gamma_0)^2 q^K - 2\gamma_0 p^K}.$$

Therefore, one finally has:

$$\frac{\gamma - r^{K+1}}{\gamma + r^{K+1}} = -\left(\frac{\gamma - \gamma_0}{\gamma + \gamma_0}\right)^2 \frac{\frac{p^K}{q^K}}{-2\gamma + \frac{p^K}{q^K}} = \left(\frac{\gamma - \gamma_0}{\gamma + \gamma_0}\right)^2 \frac{\gamma - r^K}{\gamma + r^K} = -\prod_{\kappa=0}^{K+1} \left(\frac{\gamma - \gamma_\kappa}{\gamma + \gamma_\kappa}\right)^2. \quad (q.e.d.)$$

Using (5.62), the solution (5.61) can be rewritten as:

$$u^{(0)}(x, \boldsymbol{k}, s) = u^\infty(x, \boldsymbol{k}, s) + u_{\text{refl}}(x, \boldsymbol{k}, s), \quad \text{on } x < 0.$$

where

$$u_{\text{refl}}(x, \boldsymbol{k}, s) := R(k, s)\, u^\infty(-x, \boldsymbol{k}, s), \tag{5.63}$$

$$R(k, s) := \prod_{n=0}^{N} \left(\frac{\gamma - \gamma_n}{\gamma + \gamma_n}\right)^2. \tag{5.64}$$

Equation (5.64) is a well-known form of a reflection coefficient, which is often obtained by analyzing first-order formulations; see e.g. [27, 58]. By Definition (B.23), we have:

$$\gamma_n = \frac{\cos\phi_n}{c} s + \frac{\sin^2\phi_n}{\cos\phi_n}, \quad \phi_n \in \left[0, \frac{\pi}{2}\right),$$

Then, $\text{Re}(s) > 0$ implies:

$$\text{Re}(\gamma) > 0, \quad \text{Re}(\gamma_n) > 0, \quad \text{Im}(\gamma) \cdot \text{Im}(\gamma_n) > 0;$$

see also Fig.5.2. By a distance argument on the complex plane, one immediately sees that $|\gamma - \gamma_n| < |\gamma + \gamma_n|$. Therefore, we have $R(k, s) < 1$. Specifically, for each n and each fixed k,

there exists $\alpha < 1$ such that $\sup_{\xi\in\mathbb{R}} \left|\frac{\gamma-\gamma_n}{\gamma+\gamma_n}\right| < \alpha$. This can be seen by observing that $\left|\frac{\gamma-\gamma_n}{\gamma+\gamma_n}\right| \to 1$ is possible only when γ or γ_n takes an extreme value. That is, for a fixed k, it is possible only when $|\xi| \to +\infty$. Simple calculation shows, however, that:

$$\lim_{|\xi|\to+\infty} \left|\frac{\gamma-\gamma_n}{\gamma+\gamma_n}\right| = \left|\frac{1-\cos\phi_n}{1+\cos\phi_n}\right| < 1.$$

Therefore, we have $R(k,s) < \alpha^{2N} < 1$ and the amount of reflection decreases as a power of $2N$.

We now investigate what (5.63) implies in the original $(x, \boldsymbol{y}, t)$-space. To this end, note that the Laplace transform of some function $f(t)$ ($t \leftrightarrow s = \eta + \mathfrak{i}\xi$) is merely the Fourier transform of $e^{-\eta t} f(t)$. Thus $u_{\text{refl}}(x, \boldsymbol{k}, s)$ can be seen as the d-dimensional Fourier transform of $e^{-\eta t} u_{\text{refl}}(x, \boldsymbol{y}, t)$ ($\{\boldsymbol{y}, t\} \leftrightarrow \boldsymbol{\zeta} = \{\boldsymbol{k}, \xi\}$). We then apply to (5.63) the Parseval's identity for multi-dimensional space (C.4); see Appendix C. From (5.63) and $|R(k,s)| < 1$, one has:

$$\int_{\mathbb{R}^d} |u_{\text{refl}}(x, \boldsymbol{k}, s)|^2 \,\mathrm{d}\boldsymbol{\zeta} = \int_{\mathbb{R}^d} |R(k,s)|^2 \, |u^\infty(-x, \boldsymbol{k}, s)|^2 \,\mathrm{d}\boldsymbol{\zeta} < \int_{\mathbb{R}^d} |u^\infty(-x, \boldsymbol{k}, s)|^2 \,\mathrm{d}\boldsymbol{\zeta}, \quad (5.65)$$

Applying the Parseval's identity (C.4) to both sides of (5.65), one obtains:

$$\int_{-\infty}^{+\infty} \int_{\mathbb{R}^{d-1}} \left|e^{-\eta t} u_{\text{refl}}(x, \boldsymbol{y}, t)\right|^2 \mathrm{d}\boldsymbol{y}\mathrm{d}t < \int_{-\infty}^{+\infty} \int_{\mathbb{R}^{d-1}} \left|e^{-\eta t} u^\infty(-x, \boldsymbol{y}, t)\right|^2 \mathrm{d}\boldsymbol{y}\mathrm{d}t,$$
$$\int_{-\infty}^{+\infty} e^{-2\eta t} \left\|u_{\text{refl}}(x, \boldsymbol{y}, t)\right\|^2_{L^2\left(\mathbb{R}^{d-1}\right)} \mathrm{d}t < \int_{-\infty}^{+\infty} e^{-2\eta t} \left\|u^\infty(-x, \boldsymbol{y}, t)\right\|^2_{L^2\left(\mathbb{R}^{d-1}\right)} \mathrm{d}t.$$

Finally, noting that $u^\infty(-x, \boldsymbol{y}, t) = 0$ and $u_{\text{refl}}(x, \boldsymbol{y}, t) = 0$ on $t < 0$,

$$\int_0^{+\infty} e^{-2\eta t} \left\|u_{\text{refl}}(x, \boldsymbol{y}, t)\right\|^2_{L^2\left(\mathbb{R}^{d-1}\right)} \mathrm{d}t < \int_0^{+\infty} e^{-2\eta t} \left\|u^\infty(-x, \boldsymbol{y}, t)\right\|^2_{L^2\left(\mathbb{R}^{d-1}\right)} \mathrm{d}t, \quad (5.66)$$

Since $u^\infty(-x, \boldsymbol{y}, t)$ has exponential order $\eta_0 < \eta$ in t, (5.66) implies that $u_{\text{refl}}(x, \boldsymbol{y}, t)$ has exponential order at most η_0. Specifically, it is often the case that $\left\|u^\infty(-x, \cdot, t)\right\|^2_{L^2\left(\mathbb{R}^{d-1}\right)}$ has exponential order η_0 for *any* $\eta_0 > 0$ (i.e., it can grow polynomially but not exponentially) and the right-hand side of (5.66) is finite for any $\eta > 0$. In such cases, $u_{\text{refl}}(x, \boldsymbol{k}, t)$ can not have exponential order $\eta_1 > 0$ (with η_1 constant), or one can choose $\eta = \eta_1/2 > 0$ so that the left-hand side of (5.66) is unbounded while the right-hand side is bounded. It should be noted, however, that the inequality (5.66) still allows $u_{\text{refl}}(x, \boldsymbol{y}, t)$ to grow, say, polynomially. Therefore, (5.66) states that the reflection error is bounded by some function which grows slower than any exponential functions.

5.5.3 Uniqueness

To prove the uniqueness of the solution to the half-space problem (5.40), we show that the solution to its homogeneous counterpart must be zero; i.e. we set in Problem (5.40), $f = 0$,

$u_0 = 0$, and $u_1 = 0$ and prove that $u^{(0)} = 0$. Laplace and Fourier transforms of this problem is then given by (5.41) with $f = 0$.

Proof. The general solution to the homogeneous counterpart of Problem (5.41) is given by:

$$u^{(0)}(x,k,s) = c^+(k,s)\, e^{-\gamma x} + c^-(k,s)\, e^{+\gamma x},$$

Since $u^{(0)}(x, \boldsymbol{y}, s) \in L^2(x < 0)$, we set $c^+ = 0$. Applying the boundary condition (5.41b), or (5.48), we reduce the problem to proving the following:

$$\left(\gamma + r^N(k,s)\right) u^{(0)} = 0 \quad \text{on } x = 0 \quad \Rightarrow \quad c^-(k,s) = 0.$$

That is, $\gamma + r^N(k,s) = 0$ has no solution s for $\mathrm{Re}(s) = \eta$ for all $k > 0$. The identity (5.57) indicates that:

$$\gamma + r^N = 0 \quad \Rightarrow \quad \gamma + \gamma_n = 0 \text{ for some } n \in \{0, 1, \cdots, N\},$$

so it is sufficient to prove the following:

$$\forall n \in \{0, 1, \cdots, N\}, \quad (\forall k > 0, \quad \gamma + \gamma_n = 0 \text{ has no solution } s \text{ for } \mathrm{Re}(s) > 0),$$

which is clear since $\mathrm{Re}(\gamma + \gamma_n) > 0$ always holds for $\mathrm{Re}(s) > 0$. (*q.e.d.*)

5.6 Numerical examples

In this section we present a numerical example in three dimensions to illustrate the features of our second-order formulation of CRBCs discussed so far in this section; i.e. ability to use explicit time-integrators in conjunction with a conjugate gradient method and spectral convergence of the approximate solutions to the exact solution in the limit of large N. To our knowledge, this is the first application of HOABCs to three-dimensional problems as well as to three-dimensional problems with edges and corners.

We consider a scaler wave propagation on an unbounded three-dimensional domain $\Omega_\infty = \{(x,y,z) \in \mathbb{R}^3 : x > -2, y > -1, z > -1\}$ with homogeneous Neumann boundary conditions on $x = -2, y = -1, z = -1$. The wave speed in this media is $c = 1$. We apply a point source:

$$f(t) = \begin{cases} -4\pi \sin^7\left(2\pi \frac{t}{3}\right), & \text{if } 0 \le t < 3, \\ 0, & \text{otherwise,} \end{cases}$$

at $(x,y,z) = (-2,-1,-1)$. The exact solution $u^\infty(x,y,z,t)$ to this problem is readily obtained as:

$$u^\infty(x,y,z,t) = -\frac{f\left(t - \frac{\rho}{c}\right)}{4\pi\rho},$$

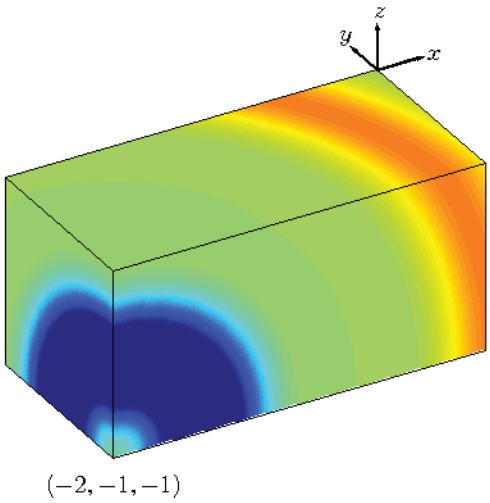


Figure 5.3: Problem setup for a numerical analysis of the wave equation in three dimensions. The second-order CRBCs are applied on three boundaries, $x = 0$, $y = 0$, and $z = 0$. Solution at $t = 2.75$ unit-time is plotted.

where ρ is the distance from the point source to (x, y, z).

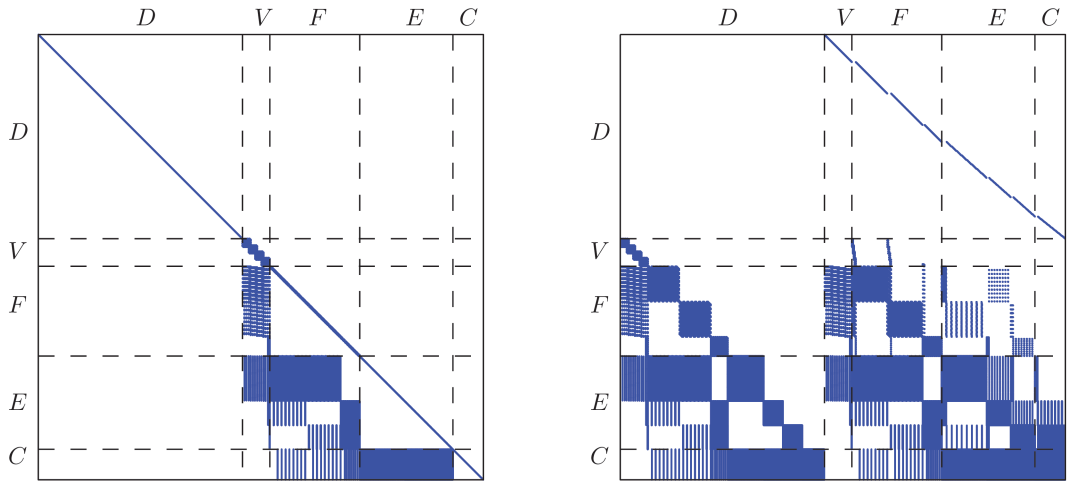
We now truncate Ω_∞ by three orthogonal planes $x = 0$, $y = 0$, and $z = 0$ to produce a bounded domain Ω (see Fig.5.3) and apply our second-order formulation of CRBCs on these three artificial boundaries to compute the numerical solution u on Ω. Specifically, we employ our face, edge, and corner formulations of equal order N proposed in Sec.5.4.1, Appendix B.3.1, and Appendix B.3.2. Here, we set $\phi_n = \bar{\phi}_n$ for the parameters and choose ϕ_n according to the Legendre-Gauss quadrature points on the interval $[0, \pi/2)$.

We discretize the resulting bounded domain Ω by uniform cubic elements with various edge-lengths of $h_r = 2^{-r}$ with $r = 1, 2, 3, 4$ and employ $\mathbb{Q}_q$ Lagrange finite elements of orders $q = 1, 2, 3$ to produce a semi-discrete system:

$$\boldsymbol{M}\boldsymbol{U}_{,t} = -\boldsymbol{K}\boldsymbol{U} + \boldsymbol{F}, \tag{5.67}$$

where $\boldsymbol{U}$ is a solution vector, $\boldsymbol{M}$ is a mass matrix, $\boldsymbol{K}$ is a stiffness matrix, and $\boldsymbol{F}$ is a force vector. $\boldsymbol{U}$ includes all nodal auxiliary variables of CRBCs as well as nodal variables on Ω. Time-integration is performed by the standard explicit fourth-order Runge-Kutta method in conjunction with a conjugate gradient (CG) method using an uniform time-step of $\Delta t = 1 \cdot 10^{-3}$ up to 10 unit-time for accuracy studies. For each refinement level r and element-order q, we compute a numerical solution u and compute the average of $|u - u^\infty|$ over the absorbing boundaries $x = 0$, $y = 0$, and $z = 0$ at each time-step, which is then averaged over a time-interval $t \in [5, 10]$. This quantity is used as a measure of approximation error produced by the nonexactness of the CRBCs.

Here we quickly present an overview of the features of the mass matrix discussed in Sec.5.4.1. Fig.5.4 shows sparsity patterns of an example mass $\boldsymbol{M}$ and stiffness $\boldsymbol{K}$ matrices



(a) Mass matrix, $\boldsymbol{M}$. (b) Stiffness matrix, $\boldsymbol{K}$.

Figure 5.4: Sparsity patterns of example (a) mass matrix $\boldsymbol{M}$ and (b) stiffness matrix $\boldsymbol{K}$ for $r = 1$, $q = 3$, and $N = 8$ along with partitions.

for $r = 1$, $q = 3$, and $N = 8$. These matrices are partitioned for convenience into 5×5 submatrices, where V, F, E, and C represent *volume* (or Ω), *face*, *edge*, and *corner*. We also partition the solution vector $\boldsymbol{U}$ accordingly, i.e. $\boldsymbol{U} = [\boldsymbol{U}_D, \boldsymbol{U}_V, \boldsymbol{U}_F, \boldsymbol{U}_E, \boldsymbol{U}_C,]^{\mathrm{T}}$. We then define, for instance, $\boldsymbol{M}_{FV}$ as a submatrix which represents the inertia effects from $\boldsymbol{U}_V$ to the equations for $\boldsymbol{U}_F$. The ODEs introduced to write the system in first order in time, such as (5.38a), are all included in rows in partition D, i.e. $\boldsymbol{M}_{DD}$ is an identity matrix. Rows in partition V represent the scalar wave equation on Ω, rows in partition F represent equations for the CRBCs (5.38b) on three faces $x = 0$, $y = 0$, $z = 0$, and rows in partitions E and C represent equations for the CRBCs on edges and corners discussed in detail in Appendix B.3. All diagonal blocks, $\boldsymbol{M}_{VV}$, $\boldsymbol{M}_{FF}$, $\boldsymbol{M}_{EE}$, $\boldsymbol{M}_{CC}$ are symmetric by construction. Note first that equations in the partition D are ODEs and one can readily compute $\boldsymbol{U}_{D,t}$ at each time-step/time-stage. Since, in the equations for $\boldsymbol{U}_V$, $\boldsymbol{M}_{VF}$, $\boldsymbol{M}_{VE}$, and $\boldsymbol{M}_{VC}$ are all zero, $\boldsymbol{U}_{V,t}$ can be computed by implicitly inverting the symmetric matrix $\boldsymbol{M}_{VV}$ using the CG method. Further, in the equations for $\boldsymbol{U}_F$, $\boldsymbol{M}_{FE}$ and $\boldsymbol{M}_{FC}$ are zero. Though $\boldsymbol{M}_{FV}$ is nonzero, we can evaluate $\boldsymbol{M}_{FV}\boldsymbol{U}_{V,t}$ since we already know $\boldsymbol{U}_{V,t}$ from previous step. One can then compute $\boldsymbol{U}_{F,t}$ by inverting $\boldsymbol{M}_{FF}$ using the CG method. $\boldsymbol{U}_{E,t}$ and $\boldsymbol{U}_{C,t}$ are computed at each time-step/time-stage in the same manner. Note that Fig.5.4 was produced for illustrative purposes using a coarse mesh with $r = 1$. On sufficiently fine meshes, the size of the partition V surpasses those of partitions F, E, and C, so evaluation of $\boldsymbol{M}_{FV}\boldsymbol{U}_{V,t}$, say, would only add a minor computational effort.

Fig.5.5(a)-(c) show $\log_{10}$ of the errors versus order N of the CRBCs for three different orders of element, $q = 1, 2, 3$. In each plot (except possibly the one for $q = 1$ and $r = 1$)

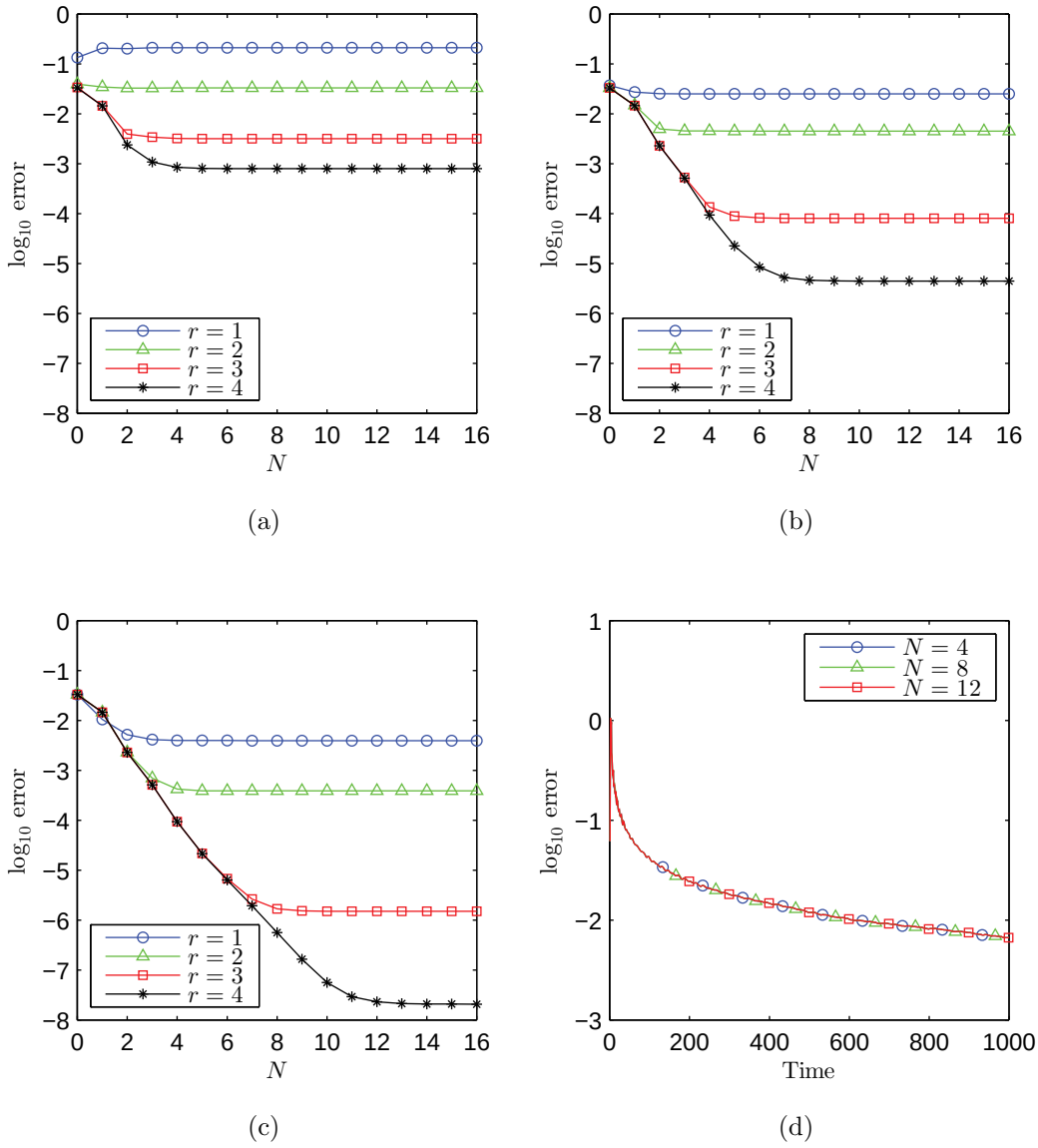


Figure 5.5: Plots of computed error against N for $r = 1, 2, 3, 4$ for (a) $q = 1$, (b) $q = 2$, and (c) $q = 3$. (d) Plots of computed error against time up to 1000 unit-time for $r = 3$, $q = 3$, and select values of $N = 4, 8, 12$.

one can see that the error decreases very sharply as N increases to a certain level, which is recognized as the underlying discretization error.

Finally long-time stability of the proposed second-order formulation is demonstrated. For $r = 3$, $q = 3$, and $N = 4, 8, 12$ we numerically integrate (5.67) in time using an implicit-midpoint method with uniform time-step of 1 unit-time up to $1 \cdot 10^3$ unit-time. At each time-step the average of $|u - u^\infty|$ over the absorbing boundaries are recorded as an error. Fig.5.5(d) shows the error versus time in a semi-log scale, where we do not see any sign of instability.

5.7 Conclusion and future works

We derived a new family of second-order formulations for the CRBCs proposed in [27]. In addition to that it was developed for the most general class of HOABCs which directly deal with waves which decay while propagating, our second-order formulation allows for the use of standard explicit time-integrators such as explicit fourth-order Runge-Kutta methods in conjunction with the conjugate gradient method when used with the standard continuous Galerkin finite element methods. We also proved analytically the existence, weak boundedness, and uniqueness of the solution to the scalar wave equations with the proposed second-order CRBCs by directly representing the error due to the rational approximation of an irrational function. We believe that application of the sequence of procedures introduced in these proofs to the analytical study of HOABCs for elastodynamics is promising. Finally we have demonstrated the accuracy and stability of our second-order formulation in a three-dimensional example problem. To our knowledge, it is the first application of the HOABCs to a three-dimensional problem as well as to a three-dimensional problem with edges and corners. Extension of this sequence of study to elastodynamics is left for future work.

Bibliography

[1] S. Asvadurov et al. "On optimal finite-difference approximation of PML". In: *SIAM Journal on Numerical Analysis* 41 (2003), pp. 287–305.

[2] D. Baffet et al. "Long-time stable high-order absorbing boundary conditions for elastodynamics". In: *Computer Methods in Applied Mechanics and Engineering* 241-244 (2012), pp. 20–37.

[3] U. Basu. "Explicit finite element perfectly matched layer for transient three-dimensional elastic waves". In: *International Journal for Numerical Methods in Engineering* 77 (2009), pp. 151–176.

[4] U. Basu and A.K. Chopra. "Perfectly matched layers for time-harmonic elastodynamics of unbounded domains: theory and finite-element implementation". In: *Computer Methods in Applied Mechanics and Engineering* 192 (2003), pp. 1337–1375.

[5] U. Basu and A.K. Chopra. "Perfectly matched layers for transient elastodynamics of unboundeddomains". In: *International Journal for Numerical Methods in Engineering* 59 (2004), pp. 1039–1074.

[6] J.-P. Berenger. "A perfectly matched layer for the absorption of electromagnetic waves". In: *Journal of Computational Physics* 114 (1994), pp. 185–200.

[7] D.S. Bindel and S. Govindjee. "Elastic PMLs for resonator anchor loss simulation". In: *International Journal for Numerical Methods in Engineering* 64 (2005), pp. 789–818.

[8] D.S. Bindel et al. "Anchor loss simulation in resonators". In: *Proceedings of MEMS 2005, IEEE* (2005), pp. 133–136.

[9] D.S. Bindel et al. "Anchor loss simulation in resonators". In: *Proceedings of MEMS 2005, IEEE* (2005), pp. 133–136.

[10] A.N. Brooks and T.J.R. Hughes. "Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations". In: *Computer Methods in Applied Mechanics and Engineering* 32 (1982), pp. 199–259.

[11] E. Bcache, D. Givoli, and T. Hagstrom. "High-order Absorbing Boundary Conditions for anisotropic and convective wave equations". In: *Journal of Computational Physics* 229 (2010), pp. 1099–1129.

[12] W.C. Chew and Q.H. Liu. "Perfectly matched layers for elastodynamics: A new absorbing boundary condition". In: *Journal of Computational Acoustics* 4 (1996), pp. 341–359.

[13] W.C. Chew and W.H. Weedon. "A 3D perfectly matched medium from modified Maxwell's equations with stretched coordinates". In: *Microwave and Optical Technology Letters* 7 (1994), pp. 599–604.

[14] B. Cockburn and C.W. Shu. "Runge-Kutta discontinuous Galerkin methods for convection-dominated problems". In: *Journal of Scientific Computing* 16 (2001), pp. 173–261.

[15] G. Cohen and S. Fauqueux. "Mixed Spectral Finite Elements for the Linear Elasticity System in Unbounded Domains". In: *SIAM Journal on Scientific Computing* 26 (2005), pp. 864–884.

[16] F. Collino and C. Tsogka. "Application of the perfectly matched absorbing layer model to the linear elastodynamic problem in anisotropic heterogeneous media". In: *Geophysics* 66 (2001), pp. 294–307.

[17] F.H. Drossaert and A. Giannopoulos. "Complex frequency shifted convolution PML for FDTD modelling of elastic waves". In: *Wave Motion* 44 (2007), pp. 593–604.

[18] G. Festa and J.-P. Vilotte. "The Newmark scheme as velocity-stress time-staggering: an efficient PML implementation for spectral element simulations of elastodynamics". In: *Geophysical Journal International* 161 (2005), pp. 789–812.

[19] S.D. Gedney and B. Zhao. "An Auxiliary Differential Equation Formulation for the Complex-Frequency Shifted PML". In: *IEEE Transactions on Antennas and Propagation* 58 (2010), pp. 838–847.

[20] D. Givoli and B. Neta. "High-order non-reflecting boundary scheme for time-dependent waves". In: *Journal of Computational Physics* 186 (2003), pp. 24–46.

[21] S. Govindjee. "Estimation of the Fundamental Mode of a Radial-Disk Resonator". In: *Technical Report UCB/SEMM-2011/02, University of California Berkeley, Department of Civil Engineering* (2011).

[22] S. Govindjee and P.-O. Persson. "A time-domain Discontinuous Galerkin method for mechanical resonator quality factor computations". In: *Journal of Computational Physics* 231 (2012), pp. 6380–6392.

[23] K.F. Graff. *Wave Motion In Elastic Solids*. Dover Publications, Inc., New York, 1991.

[24] M.N. Guddati. "Arbitrarily wide-angle wave equations for complex media". In: *Computer Methods in Applied Mechanics and Engineering* 195 (2006), pp. 65–93.

[25] M.N. Guddati and K.-W. Lim. "Continued fraction absorbing boundary conditions for convex polygonal domains". In: *International Journal for Numerical Methods in Engineering* 66 (2006), pp. 949–977.

[26] T. Hagstrom, A. Mar-Or, and D. Givoli. "High-order local absorbing conditions for the wave equation: extensions and improvements". In: *Journal of Computational Physics* 227 (2008), pp. 3322–3357.

[27] T. Hagstrom and T. Warburton. "Complete Radiation Boundary Conditions: Minimizing the Long Time Error Growth of Local Methods". In: *SIAM Journal on Numerical Analysis* 47 (2009), pp. 3678–3704.

[28] T. Hagstrom, T. Warburton, and D. Givoli. "Radiation boundary conditions for time-dependent waves based on complete plane wave expansions". In: *Journal of Computational and Applied Mathematics* 234 (2010), pp. 1988–1995.

[29] Thomas Hagstrom and Timothy Warburton. "A new auxiliary variable formulation of high-order local radiation boundary conditions: corner compatibility conditions and extensions to first-order systems". In: *Wave Motion* 39 (2004), pp. 327 –338.

[30] R.L. Higdon. "Absorbing Boundary-Conditions for Difference Approximations to the Multidimensional Wave-Equation". In: *Mathematics of Computation* 47 (1986), pp. 437–459.

[31] R.L. Higdon. "Initial-Boundary Value Problems for Linear Hyperbolic Systems". In: *SIAM Review* 28 (1986), pp. 177–217.

[32] S. Johnson. "Harminv (ver. 1.3.1)". In: (2004).

[33] A.W. Knapp. *Basic Real Analysis*. Birkhauser, 2005.

[34] D. Komatitsch and R. Martin. "An unsplit convolutional perfectly matched layer improved at grazing incidence for the seismic wave equation". In: *Geophysics* 72 (2007), SM155–SM167.

[35] D. Komatitsch and J. Tromp. "A perfectly matched layer absorbing boundary condition for the second-order seismic wave equation". In: *Geophysical Journal International* 154 (2003), pp. 146–153.

[36] T. Koyama. "Efficient Evaluation of Damping in Resonant MEMS". In: *Ph.D. thesis, University of California, Berkeley* (2008).

[37] H.-O. Kreiss. "Initial boundary value problems for hyperbolic systems". In: *Communications on Pure and Applied Mathematics* 23 (1970), pp. 277–298.

[38] S. Kucukcoban and L.F. Kallivokas. "A Mixed Perfectly-Matched-Layer for Transient Wave Simulations in Axisymmetric Elastic Media". In: *Computer Modeling in Engineering & Sciences* 64 (2010), pp. 109–145.

[39] S. Kucukcoban and L.F. Kallivokas. "Mixed perfectly-matched-layers for direct transient analysis in 2D elastic heterogeneous media". In: *Computer Methods in Applied Mechanics and Engineering* 200 (2011), pp. 57–76.

[40] M. Kuzuoglu and R. Mittra. "Frequency dependence of the constitutive parameters of causal perfectly matched anisotropic absorbers". In: *IEEE Microwave and Guided Wave Letters* 6 (1996), pp. 447–449.

[41] S.S. Li et al. "An MSI micromechanical differential disk-array filter". In: *Digest of Technical Papers, the 14th International Conference on Solid-State Sensors & Actuators (Transducers '07)* (2007), pp. 307–311.

[42] Y. Li and O.B. Matar. "Convolutional perfectly matched layer for elastic second-order wave equation". In: *Journal of the Acoustical Society of America* 127 (2010), pp. 1318–1327.

[43] Q.H. Liu. "Perfectly matched layers for elastic waves in cylindrical and spherical coordinates". In: *Journal of the Acoustical Society of America* 105 (1999), pp. 2075–2084.

[44] R.J. Luebbers and F. Hunsberger. "FDTD for Nth-order dispersive media". In: *IEEE Transactions on Antennas and Propagation* 40 (1992), pp. 1297–1301.

[45] J. Lysmer and R.L. Kuhlmeyer. "Finite dynamic model for infinite media". In: *J Eng Mech Div ASCE* 95 (1969), pp. 859–877.

[46] V. Mandelshtam and H. Taylor. "Harmonic inversion of time signals and its applications". In: *Journal of Chemical Physics* 107 (1997), pp. 6756–6769.

[47] C. Marcinkovich and K. Olsen. "On the implementation of perfectly matched layers in a three-dimensional fourth-order velocity-stress finite difference scheme". In: *Journal of Geophysical Research* 108 (2003).

[48] R. Martin and D. Komatitsch. "An unsplit convolutional perfectly matched layer technique improved at grazing incidence for the viscoelastic wave equation". In: *Geophysical Journal International* 179 (2009), pp. 333–344.

[49] R. Martin et al. "A High-Order Time and Space Formulation of the Unsplit Perfectly Matched Layer for the Seismic Wave Equation Using Auxiliary Differential Equations (ADE-PML)". In: *Computer Modeling in Engineering & Sciences* 56 (2010), pp. 17–41.

[50] C.T.C. Nguyen. In: *Private communication.* (2013).

[51] J. Peraire and P.-O. Persson. *Adaptive High-order methods in computational fluid dynamics.* Vol. 2. World Scientific Publishing Co., 2011.

[52] J. Peraire and P.-O. Persson. "The Compact Discontinuous Galerkin (CDG) Method for Elliptic Problems". In: *SIAM Journal on Scientific Computing* 30 (2008), pp. 1806–1824.

[53] P.-O. Persson and G. Strang. "A Simple Mesh Generator in Matlab". In: *SIAM Review* 46 (2004), pp. 329–345.

[54] D. Rabinovich et al. "A finite element scheme with a high order absorbing boundary condition for elastodynamics". In: *Computer Methods in Applied Mechanics and Engineering* 200 (2011), pp. 2048–2066.

[55] O. Ramadan. "Auxiliary differential equation formulation: an efficient implementation of the perfectly matched layer". In: *IEEE Microwave and Wireless Components Letters* 13 (2003), pp. 69–71.

[56] J.A. Roden and S.D. Gedney. "Convolution PML (CPML): An efficient FDTD implementation of the CFS-PML for arbitrary media". In: *Microwave and Optical Technology Letters* 27 (2000), pp. 334–339.

[57] S. Savadatti and M.N. Guddati. "Absorbing boundary conditions for scalar waves in anisotropic media. Part 1: Time harmonic modeling". In: *Journal of Computational Physics* 229 (2010), pp. 6696–6714.

[58] S. Savadatti and M.N. Guddati. "Absorbing boundary conditions for scalar waves in anisotropic media. Part 2: Time-dependent modeling". In: *Journal of Computational Physics* 229 (2010), pp. 6644–6662.

[59] S. Savadatti and M.N. Guddati. "Accurate absorbing boundary conditions for anisotropic elastic media. Part 1: Elliptic anisotropy". In: *Journal of Computational Physics* 231 (2012), pp. 7584–7607.

[60] S. Savadatti and M.N. Guddati. "Accurate absorbing boundary conditions for anisotropic elastic media. Part 2: Untilted non-elliptic anisotropy". In: *Journal of Computational Physics* 231 (2012), pp. 7608–7625.

[61] L.N. Trefethen and L. Halpern. "Well-Posedness of One-Way Wave Equations and Absorbing Boundary Conditions". In: *Mathematics of Computation* 47 (1986), pp. 421–435.

[62] M. Wall and D. Neuhauser. "Extraction through filter-diagonalization of general quantum eigenvalues or classical normal mode frequencies from a small number of residues or a short-time segment of a signal. I. Theory and application to a quantum-dynamics model". In: *Journal of Chemical Physics* 102 (1995), pp. 8011–8022.

[63] J. Wang et al. "1.51-GHz polydiamond micromechanical disk resonator with impedance-mismatched isolating support". In: *Proceedings IEEE International MEMS Conference* (2004), pp. 641–644.

[64] W. Zhang and Y. Shen. "Unsplit complex frequency-shifted PML implementation using auxiliary differential equations for seismic wave modeling". In: *Geophysics* 75 (2010), T141–T154.

[65] Q. Zhen et al. "The implementation of an improved NPML absorbing boundary condition in elastic wave modeling". In: *Applied Geophysics* 6 (2009), pp. 113–121.

Appendix A

Perfectly matched layers

A.1 Tensor components and constants

A.1.1 Frequency-domain formulation

In the frequency-domain, the components of $\mathbf{\Sigma}$ in Eqn. (3.15) in the spherical basis $\{\boldsymbol{e}_r, \boldsymbol{e}_\theta, \boldsymbol{e}_\phi\}$ are given by:

$$\begin{aligned}
\Sigma_{rr} &= (\lambda + 2\mu)\left(C_1 + C_2\frac{1}{\mathrm{i}\omega} + C_3\frac{1}{\mathrm{i}\omega + C_0}\frac{1}{\mathrm{i}\omega}\right)(\nabla u)_{rr} \\
&\quad + \lambda\left(\frac{F^e}{r} + \frac{\omega_0 F^p}{r}\frac{1}{\mathrm{i}\omega}\right)\left[(\nabla u)_{\theta\theta} + (\nabla u)_{\phi\phi}\right], \qquad \text{(A.1a)} \\
\Sigma_{r\theta} &= \mu\left(C_1 + C_2\frac{1}{\mathrm{i}\omega} + C_3\frac{1}{\mathrm{i}\omega + C_0}\frac{1}{\mathrm{i}\omega}\right)(\nabla u)_{\theta r} \\
&\quad + \mu\left(\frac{F^e}{r} + \frac{\omega_0 F^p}{r}\frac{1}{\mathrm{i}\omega}\right)(\nabla u)_{r\theta}, \qquad \text{(A.1b)} \\
\Sigma_{r\phi} &= \mu\left(C_1 + C_2\frac{1}{\mathrm{i}\omega} + C_3\frac{1}{\mathrm{i}\omega + C_0}\frac{1}{\mathrm{i}\omega}\right)(\nabla u)_{\phi r} \\
&\quad + \mu\left(\frac{F^e}{r} + \frac{\omega_0 F^p}{r}\frac{1}{\mathrm{i}\omega}\right)(\nabla u)_{r\phi}, \qquad \text{(A.1c)} \\
\Sigma_{\theta r} &= \mu\left(f^e + \omega_0 f^p\frac{1}{\mathrm{i}\omega}\right)(\nabla u)_{r\theta} + \mu\left(\frac{F^e}{r} + \frac{\omega_0 F^p}{r}\frac{1}{\mathrm{i}\omega}\right)(\nabla u)_{\theta r}, \qquad \text{(A.1d)} \\
\Sigma_{\theta\theta} &= \lambda\left(\frac{F^e}{r} + \frac{\omega_0 F^p}{r}\frac{1}{\mathrm{i}\omega}\right)(\nabla u)_{rr} + \lambda\left(f^e + \omega_0 f^p\frac{1}{\mathrm{i}\omega}\right)(\nabla u)_{\phi\phi} \\
&\quad + (\lambda + 2\mu)\left(f^e + \omega_0 f^p\frac{1}{\mathrm{i}\omega}\right)(\nabla u)_{\theta\theta}, \qquad \text{(A.1e)} \\
\Sigma_{\theta\phi} &= \mu\left(f^e + \omega_0 f^p\frac{1}{\mathrm{i}\omega}\right)\left[(\nabla u)_{\theta\phi} + (\nabla u)_{\phi\theta}\right], \qquad \text{(A.1f)}
\end{aligned}$$

$$\Sigma_{\phi r} = \mu\left(f^e + \omega_0 f^p \frac{1}{\mathrm{i}\omega}\right)(\nabla u)_{r\phi} + \mu\left(\frac{F^e}{r} + \frac{\omega_0 F^p}{r}\frac{1}{\mathrm{i}\omega}\right)(\nabla u)_{\phi r}, \tag{A.1g}$$

$$\Sigma_{\phi\theta} = \mu\left(f^e + \omega_0 f^p \frac{1}{\mathrm{i}\omega}\right)\left[(\nabla u)_{\theta\phi} + (\nabla u)_{\phi\theta}\right], \tag{A.1h}$$

$$\begin{aligned}\Sigma_{\phi\phi} = {} & \lambda\left(\frac{F^e}{r} + \frac{\omega_0 F^p}{r}\frac{1}{\mathrm{i}\omega}\right)(\nabla u)_{rr} + \lambda\left(f^e + \omega_0 f^p \frac{1}{\mathrm{i}\omega}\right)(\nabla u)_{\theta\theta} \\ & + (\lambda + 2\mu)\left(f^e + \omega_0 f^p \frac{1}{\mathrm{i}\omega}\right)(\nabla u)_{\phi\phi},\end{aligned} \tag{A.1i}$$

where C_0, C_1, C_2, and C_3 are temporally constant and defined as:

$$C_0 = \frac{\omega_0 f^p}{1 + f^e}, \tag{A.2a}$$

$$C_1 = \frac{\left(1 + \frac{F^e}{r}\right)^2}{1 + f^e} - 1, \tag{A.2b}$$

$$C_2 = \frac{\left(1 + \frac{F^e}{r}\right)^2}{1 + f^e}\left(\frac{2\frac{\omega_0 F^p}{r}}{1 + \frac{F^e}{r}} - C_0\right), \tag{A.2c}$$

$$C_3 = \frac{\left(1 + \frac{F^e}{r}\right)^2}{1 + f^e}\left(\frac{\frac{\omega_0 F^p}{r}}{1 + \frac{F^e}{r}} - C_0\right)^2. \tag{A.2d}$$

A.1.2 Time-domain formulation

In the time-domain, the components of $\boldsymbol{\Sigma}$ in Eqn. (3.18b) in the spherical basis $\{\boldsymbol{e}_r, \boldsymbol{e}_\theta, \boldsymbol{e}_\phi\}$ are given by:

$$\begin{aligned}\Sigma_{rr} = {} & (\lambda + 2\mu)C_1(\nabla u)_{rr} + (\lambda + 2\mu)g_1 \\ & + \lambda\left(\frac{F^e}{r}\left[(\nabla u)_{\theta\theta} + (\nabla u)_{\phi\phi}\right] + \frac{\omega_0 F^p}{r}\left[(\nabla h)_{\theta\theta} + (\nabla h)_{\phi\phi}\right]\right),\end{aligned} \tag{A.3a}$$

$$\Sigma_{r\theta} = \mu C_1(\nabla u)_{\theta r} + \mu\left(\frac{F^e}{r}(\nabla u)_{r\theta} + \frac{\omega_0 F^p}{r}(\nabla h)_{r\theta}\right) + \mu g_2, \tag{A.3b}$$

$$\Sigma_{r\phi} = \mu C_1(\nabla u)_{\phi r} + \mu\left(\frac{F^e}{r}(\nabla u)_{r\phi} + \frac{\omega_0 F^p}{r}(\nabla h)_{r\phi}\right) + \mu g_3, \tag{A.3c}$$

$$\Sigma_{\theta r} = \mu\left(f^e(\nabla u)_{r\theta} + \omega_0 f^p(\nabla h)_{r\theta} + \frac{F^e}{r}(\nabla u)_{\theta r} + \frac{\omega_0 F^p}{r}(\nabla h)_{\theta r}\right), \tag{A.3d}$$

$$\begin{aligned}\Sigma_{\theta\theta} = {} & \lambda\left(\frac{F^e}{r}(\nabla u)_{rr} + \frac{\omega_0 F^p}{r}(\nabla h)_{rr} + f^e(\nabla u)_{\phi\phi} + \omega_0 f^p(\nabla h)_{\phi\phi}\right) \\ & + (\lambda + 2\mu)\left(f^e(\nabla u)_{\theta\theta} + \omega_0 f^p(\nabla h)_{\theta\theta}\right),\end{aligned} \tag{A.3e}$$

$$\Sigma_{\theta\phi} = \mu\left(f^e\left[(\nabla u)_{\theta\phi} + (\nabla u)_{\phi\theta}\right] + \omega_0 f^p\left[(\nabla h)_{\theta\phi} + (\nabla h)_{\phi\theta}\right]\right), \tag{A.3f}$$

$$\Sigma_{\phi r} = \mu\left(f^e(\nabla u)_{r\phi} + \omega_0 f^p(\nabla h)_{r\phi}\right) + \mu\left(\frac{F^e}{r}(\nabla u)_{\phi r} + \frac{\omega_0 F^p}{r}(\nabla h)_{\phi r}\right), \tag{A.3g}$$

$$\Sigma_{\phi\theta} = \mu\left(f^e\left[(\nabla u)_{\theta\phi} + (\nabla u)_{\phi\theta}\right] + \omega_0 f^p\left[(\nabla h)_{\theta\phi} + (\nabla h)_{\phi\theta}\right]\right), \tag{A.3h}$$

$$\begin{aligned}\Sigma_{\phi\phi} = {} & \lambda\left(\frac{F^e}{r}(\nabla u)_{rr} + \frac{\omega_0 F^p}{r}(\nabla h)_{rr}\right) + \lambda\left(f^e(\nabla u)_{\theta\theta} + \omega_0 f^p(\nabla h)_{\theta\theta}\right) \\ & + (\lambda + 2\mu)\left(f^e(\nabla u)_{\phi\phi} + \omega_0 f^p(\nabla h)_{\phi\phi}\right),\end{aligned} \tag{A.3i}$$

where g_1, g_2, and g_3 are auxiliary functions defined in Eqns. (3.18d)-(3.18f).

Also, the temporal constants C_4, C_5, C_6, and C_7 introduced in Eqns. (3.18) are defined as:

$$C_4 = (1 + f^e)\left(1 + \frac{F^e}{r}\right)^2, \tag{A.4a}$$

$$C_5 = \left(1 + \frac{F^e}{r}\right)\left(\omega_0 f^p\left(1 + \frac{F^e}{r}\right) + 2\left(1 + f^e\right)\frac{\omega_0 F^p}{r}\right), \tag{A.4b}$$

$$C_6 = \frac{\omega_0 F^p}{r}\left(2\omega_0 f^p\left(1 + \frac{F^e}{r}\right) + \left(1 + f^e\right)\frac{\omega_0 F^p}{r}\right), \tag{A.4c}$$

$$C_7 = \omega_0 f^p\left(\frac{\omega_0 F^p}{r}\right)^2. \tag{A.4d}$$

A.2 Axisymmetric PML

In this section, we present an axisymmetric formulation of our spherical PML. If the problem is axisymmetric, one can set $u_\phi = 0$ and $\frac{\partial}{\partial\phi}(\cdot) = 0$ in the spherical problem (3.18). Then all $r\phi$-, ϕr-, $\theta\phi$-, and $\phi\theta$-components of $\boldsymbol{\nabla u}$, $\boldsymbol{\nabla v}$, and $\boldsymbol{\nabla h}$, and thus $\boldsymbol{\sigma}$ and $\boldsymbol{\Sigma}$, vanish. It is convenient to resolve problem (3.18) in the standard cylindrical coordinate system (R, z, ϕ) with the orthonormal cylindrical basis $\{\boldsymbol{e}_R, \boldsymbol{e}_z, \boldsymbol{e}_\phi\}$, where one finds that $R\phi$-, ϕR-, $z\phi$-, and ϕz- components of $\boldsymbol{\nabla u}$ etc. are zero. Then, one is left with 8 non-trivial equations for 8 unknowns:

Find u_R, u_z, v_R, v_z, h_R, h_z, g_1, g_2 on Ω such that:

$$\dot{u}_R = v_R, \tag{A.5a}$$

$$\dot{u}_z = v_z, \tag{A.5b}$$

$$\begin{aligned}&\rho R C_4 \dot{v}_R - \left[\left\{R\left(\sigma_{RR} + \Sigma_{RR}\right)\right\}_{,R} + \left\{R\left(\sigma_{zR} + \Sigma_{zR}\right)\right\}_{,z}\right] \\ &= -\rho R\left(C_5 v_R + C_6 u_R + C_7 h_R\right) - \left(\sigma_{\phi\phi} + \Sigma_{\phi\phi}\right),\end{aligned} \tag{A.5c}$$

$$\rho R C_4 \dot{v}_z - \left[\left\{R\left(\sigma_{Rz} + \Sigma_{Rz}\right)\right\}_{,R} + \left\{R\left(\sigma_{zz} + \Sigma_{zz}\right)\right\}_{,z}\right]$$

$$= -\rho R\left(C_5 v_z + C_6 u_z + C_7 h_z\right), \tag{A.5d}$$
$$\dot{h}_R = u_R, \tag{A.5e}$$
$$\dot{h}_z = u_z, \tag{A.5f}$$
$$\dot{g}_1 = -C_0 g_1 + C_2(\nabla u)_{rr} + (C_0 C_2 + C_3)(\nabla h)_{rr}, \tag{A.5g}$$
$$\dot{g}_2 = -C_0 g_2 + C_2(\nabla u)_{\theta r} + (C_0 C_2 + C_3)(\nabla h)_{\theta r}, \tag{A.5h}$$

and

$$u_R = \bar{u}_R, \quad u_z = \bar{u}_z \quad \text{on } \partial\Omega_u,$$
$$\begin{bmatrix} R\left(\sigma_{RR} + \Sigma_{RR}\right) & R\left(\sigma_{Rz} + \Sigma_{Rz}\right) \\ R\left(\sigma_{zR} + \Sigma_{zR}\right) & R\left(\sigma_{zz} + \Sigma_{zz}\right) \end{bmatrix}^T \begin{Bmatrix} n_R \\ n_z \end{Bmatrix} = \begin{Bmatrix} t_R \\ t_z \end{Bmatrix} \quad \text{on } \partial\Omega_t,$$

where

$$\sigma_{RR} = 2\mu\left(\nabla u\right)_{RR} + \lambda\left(\left(\nabla u\right)_{RR} + \left(\nabla u\right)_{zz} + \left(\nabla u\right)_{\phi\phi}\right),$$
$$\sigma_{zz} = 2\mu\left(\nabla u\right)_{zz} + \lambda\left(\left(\nabla u\right)_{RR} + \left(\nabla u\right)_{zz} + \left(\nabla u\right)_{\phi\phi}\right),$$
$$\sigma_{\phi\phi} = 2\mu\left(\nabla u\right)_{\phi\phi} + \lambda\left(\left(\nabla u\right)_{RR} + \left(\nabla u\right)_{zz} + \left(\nabla u\right)_{\phi\phi}\right),$$
$$\sigma_{Rz} = \sigma_{zR} = \mu\left(\left(\nabla u\right)_{Rz} + \left(\nabla u\right)_{zR}\right),$$
$$\left(\nabla w\right)_{RR} = w_{R,R}, \quad \left(\nabla w\right)_{zz} = w_{z,z}, \quad \left(\nabla w\right)_{\phi\phi} = \frac{w_R}{R},$$
$$\left(\nabla w\right)_{Rz} = w_{R,z}, \quad \left(\nabla w\right)_{zR} = w_{z,R}, \qquad (w{=}u,\, v,\, h),$$

where n_R and n_z are the R- and z-components of the outward normal vector to $\partial\Omega_t$. The coefficients C_0, ..., and C_7 are given in Eqns. (A.2) and (A.4) and components of $\mathbf{\Sigma}$ are defined in Eqn. (A.3); see Appendix A.1.2. Problem (A.5) inherits the boundary conditions from the original problem (3.18).

A.3 Parameter choice

A.3.1 One-dimensional PML parameter optimization on frequency-domain

A procedure of finding an optimum set of PML parameters was studied in detail for a one-dimensional wave equation in the frequency-domain in [36]. We first summarize the procedure in [36] and introduce heuristics of choosing time-domain parameters using our example in Sec.3.4.

We discretize Eqn. (3.5a) with $k = 0$ on $x \in [0, x_{pml}]$, in which the PML domain corresponds to $x_0 \leq x$, using finite element methods of order q with

$$s(x) = 1 + \frac{1}{\mathrm{i}} f^p(x), \tag{A.6}$$

which is obtained by setting $f^e(x) = 0$ and $\frac{\omega_0}{\omega} = 1$ in Eqn. (3.6). We further restrict the profile of $f^p(x)$ to polynomials as:

$$f^p(x) = \beta^p \left(\frac{x - x_0}{x_{pml} - x_0} \right)^m \tag{A.7}$$

where m and β^p are the order and end-value of the polynomials, respectively. On solving problem (3.5a), one can nondimensionalize the problem to obtain a set of five independent parameters: n_{wpml}, m, β^p, n_{npw}, and q, where n_{wpml} and n_{npw} are number of wavelengths in the PML and number of nodes per wavelength, respectively. Then, for select sets of m, n_{npw}, and q, we vary β^p and n_{wpml} from 0 to 10 and plot contours of reflection coefficients. The procedure to compute the reflection coefficients are briefly explained in the following. On each element in the elastic region, $0 \leq x \leq x_0$, Eqn. (3.5a) produces an element-wise discrete wave operator upon finite element discretization:

$$\square := -k^2 \boldsymbol{m}_e + \boldsymbol{k}_e, \tag{A.8}$$

where $\boldsymbol{m}_e$ and $\boldsymbol{k}_e$ are element mass and element stiffness matrices and $k = \omega/c$ is the wave number. The wave operators (A.8) are then assembled to form a global stiffness matrix $\boldsymbol{K}$. Further, a set of q nodal displacements in the jth element is denoted by $\boldsymbol{u}_j$ and concatenated to form a global solution vector $\boldsymbol{U}$. The structures of $\boldsymbol{K}$ and $\boldsymbol{U}$ are schematically shown for $q = 3$ in Fig. A.1. Three successive nodal displacement vectors in the elastic domain, $\boldsymbol{u}_{j-1}$, $\boldsymbol{u}_j$, and $\boldsymbol{u}_{j+1}$, satisfy a homogeneous system of discrete wave equations which characterizes wave propagation on an unbounded domain:

$$\boldsymbol{B}^T \boldsymbol{u}_{j-1} + \boldsymbol{A}\boldsymbol{u}_j + \boldsymbol{B}\boldsymbol{u}_{j+1} = \boldsymbol{0}, \tag{A.9}$$

as shown in Fig. A.1. Substituting $\boldsymbol{u}_j = \xi^j \boldsymbol{v}$ in the homogeneous system (A.9), one obtains a quadratic eigenvalue problem:

$$\left[\boldsymbol{B}^T + \xi \boldsymbol{A} + \xi^2 \boldsymbol{B} \right] \boldsymbol{v} = \boldsymbol{0},$$

for which there exist two nonzero eigenvalues ξ^+ and ξ^- and corresponding eigenvectors $\boldsymbol{v}^+$ and $\boldsymbol{v}^-$, the former representing outgoing and the latter representing incoming waves so that the total solution on the homogeneous elastic media should be represented as:

$$\boldsymbol{u}_j = \begin{bmatrix} \left(\xi^+\right)^j \boldsymbol{v}^+ & \left(\xi^-\right)^j \boldsymbol{v}^- \end{bmatrix} \begin{pmatrix} c^+ \\ c^- \end{pmatrix}. \tag{A.10}$$

We now compute the solution to the PML problem (3.5) by a standard linear solver and extract two solution vectors $\boldsymbol{u}_j$ and $\boldsymbol{u}_{j+1}$ to solve for c^+ and c^- in Eqn. (A.10). Specifically, we have

$$\begin{pmatrix} c^+ \\ c^- \end{pmatrix} = \begin{bmatrix} \left(\xi^+\right)^j \boldsymbol{v}^+ & \left(\xi^-\right)^j \boldsymbol{v}^- \\ \left(\xi^+\right)^{j+1} \boldsymbol{v}^+ & \left(\xi^-\right)^{j+1} \boldsymbol{v}^- \end{bmatrix}^\dagger \begin{pmatrix} \boldsymbol{u}_j \\ \boldsymbol{u}_{j+1} \end{pmatrix} \tag{A.11}$$

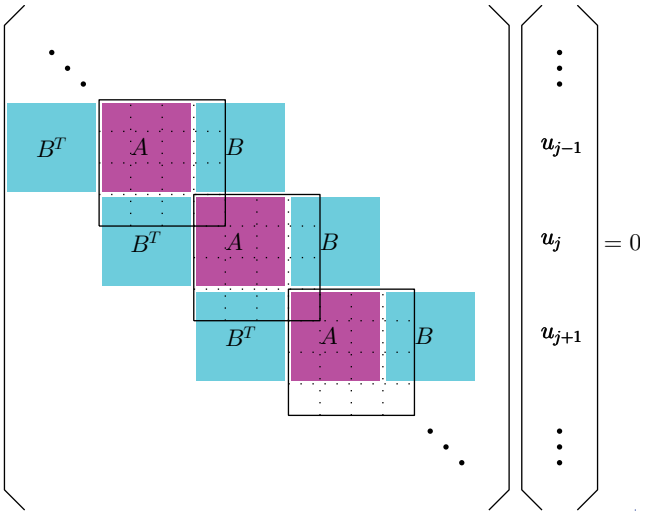


Figure A.1: Schematic of one-dimensional discrete wave equation in frequency-domain on elastic medium with cubic ($q = 3$) interpolation polynomials. Each square with solid sides represents an element-wise stiffness matrix.

where † represents pseudo-inverse. The reflection coefficient r is given by the ratio $|c^-/c^+|$. Note that since we project the solution onto discrete modes and consider the ratio of the discrete incoming wave to the discrete outgoing wave, r represents the reflection due to discretization of the PML as well as its termination.

Figs. A.2(a), (b), (c) and (d) show contour plots of $\log_{10} r$ for $m = 1, 2, 3, 4$, respectively, with fixed $q = 4$ and $n_{npw} = 12$ on a grid of $(n_{wpml}, \beta^p) = [0, 10] \times [0, 10]$. With knowledge of q and n_{npw} and with an allowed level of total reflection in mind, one can readily read off an optimum set of PML parameters, n_{wpml}, β^p, and m, which can achieve the desired level of accuracy with smallest n_{wpml}.

These contour plots characterize the two kinds of reflections $r_{\text{termination}}$ and $r_{\text{discretization}}$ mentioned in Sec. 3.2.1. As a specific example, focus on Fig. A.2(c) with n_{wpml} fixed at 4. Increasing β^p from zero, one observes a rapid decrease of $\log_{10} r$ up to $\beta^p \approx 1$ to achieve $\log_{10} r = -6$. If one further increase β^p, $\log_{10} r$ gradually increases. This suggests that in the first phase $r_{\text{termination}}$ given in Eqn. (3.4) surpasses $r_{\text{discretization}}$, while in the second phase $r_{\text{discretization}}$ surpasses $r_{\text{termination}}$. Indeed, on the region of small β^p and large n_{wpml}, a curve of a constant r, say $\bar{r}$, almost coincides with a curve produced by Eqn. (3.4) with $r_{\text{termination}}$ fixed at $\bar{r}$, which verifies that $r \approx r_{\text{termination}}$ in this region of parameter space. See [36] for further elaboration.

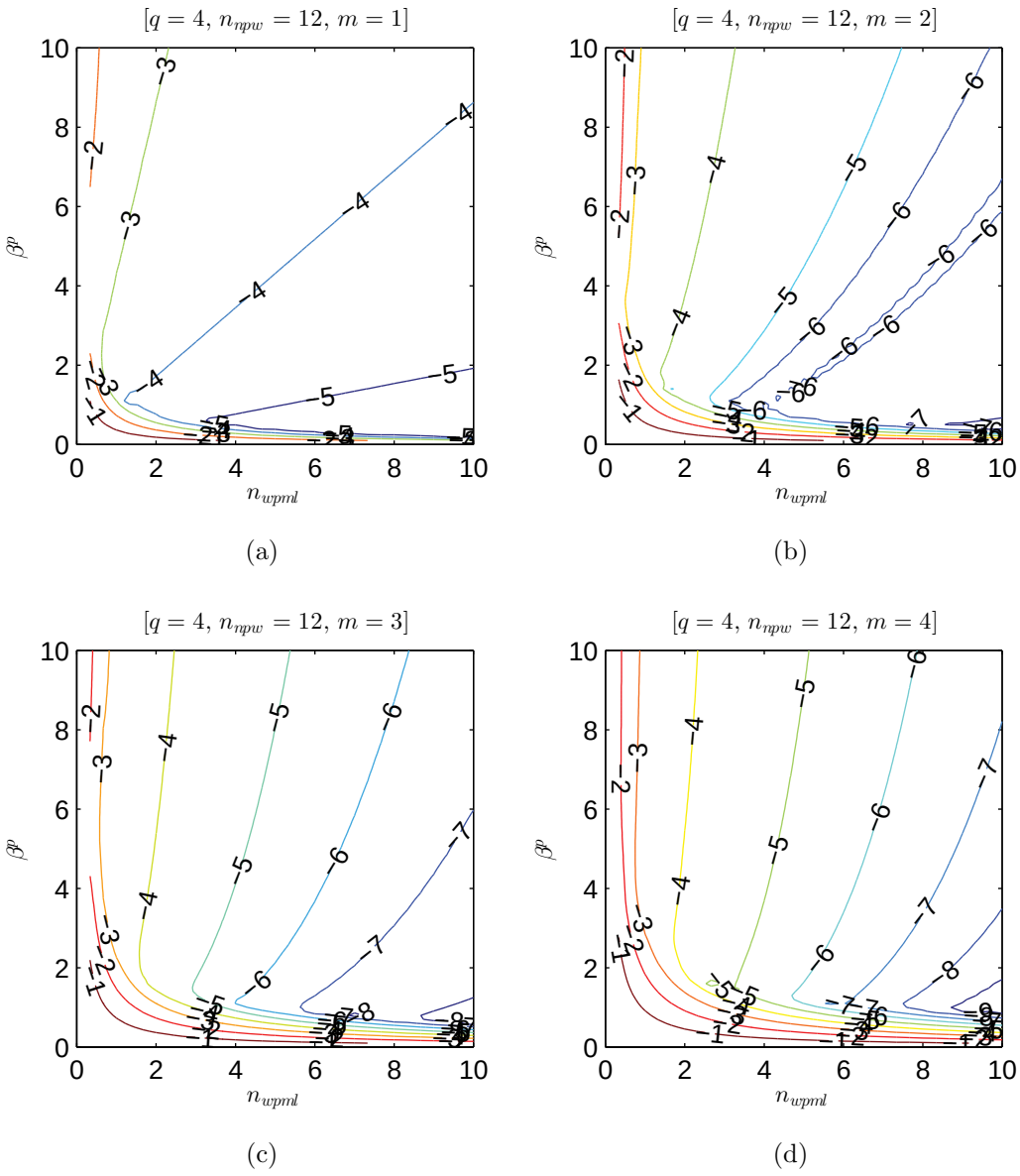


Figure A.2: Contour plots of $\log_{10} r$ for (a) $m = 1$ (b) $m = 2$ (c) $m = 3$ (d) $m = 4$, fixing $q = 4$ and $n_{npw} = 12$.

A.3.2 Parameter choice heuristics on time-domain

Heuristics for choosing PML parameters for elastodynamics in the time-domain for a given level of discretization and an order of element q are summarized below:

- Choose an characteristic frequency (ω_0 in Eqn. (3.6)) and wave speed c_0 and compute a characteristic wavelength λ_0.
- Compute n_{npw}, where n_{npw} denotes the number of nodes per characteristic wavelength.
- Given allowed total reflection coefficients r_{allowed}, read off an optimum pair of parameters n_{wpml} and β^p for an optimum m from contour plots for the desired q and n_{npw}, where n_{wpml} denotes the number of characteristic wavelengths in the PML.

A time-domain PML thus obtained ensures that the reflection coefficient should be r_{allowed} for a mode of frequency ω_0 since Eqn. (3.6) reduces to Eqn. (A.6) when $\omega = \omega_0$.

For our problems in Sec. 3.4, we choose $\omega_0 = 4\pi$ considering the pattern of the waves generated by the excitation (3.24). Setting $c_0 = c_s$, where c_s is the shear wave velocity, we obtain $n_{npw} \approx 12$. r_{allowed} is set to 10^{-3} as is often done in practice. We then look at Figs. A.2(a)-(d) for $m = 1, 2, 3, 4$ with fixed $n_{npw} = 12$ and $q = 4$. According to these figures, the minimum possible n_{wpml} required to achieve $r = 10^{-3}$ for $m = 1$, 2, 3, and 4 are about 1.0, 0.5, 0.6, and 0.7, respectively, so we choose to use $m = 2$. The corresponding parameters are $n_{wpml} \approx 0.5$ and $\beta^p \approx 4$. Also we make a conservative choice of $r_{pml} - r_0 = 1.5[\mu\text{m}]$ which gives $n_{wpml} \approx 0.8$. In summary, we use $\omega_0 = 4\pi$, $m = 2$, $n_{wpml} \approx 0.8$, and $\beta^p = 4$.

Appendix B

High-order absorbing boundary conditions

B.1 Basics and definitions

For convenience, we denote by L_n^ν and $\bar{L}_n^\nu$ the basic operators appearing in the complete radiation boundary conditions (CRBCs) (5.28) as:

$$L_n^\nu = \frac{\cos\phi_n}{c}\partial_t + \partial_\nu + \frac{1}{cT}\frac{\sin^2\phi_n}{\cos\phi_n}, \tag{B.1a}$$

$$\bar{L}_n^\nu = \frac{\cos\bar{\phi}_n}{c}\partial_t - \partial_\nu + \frac{1}{cT}\frac{\sin^2\bar{\phi}_n}{\cos\bar{\phi}_n}, \tag{B.1b}$$

where ν represents normal directions, i.e. $\nu = x$, y, or z.

We now prove the following statement on $\nu \in [0, \infty)$:

$$\begin{cases} \bar{L}_n^\nu g(\nu, t) = 0, \\ g(\nu, 0) = 0. \end{cases} \quad \Leftrightarrow \quad g(\nu, t) = 0. \tag{B.2}$$

Proof. The general solution is given by $g(\nu,t) = e^{\frac{1}{cT}\frac{\sin^2\bar{\phi}_j}{\cos\bar{\phi}_j}\nu} f\left(t + \frac{\cos\bar{\phi}}{c}\nu\right)$ where f is an undetermined function. Initial conditions readily give $f = 0$ and thus $g(\nu, t) = 0$. (*q.e.d.*)

As a consequence of (B.2), we can prove the following statement on $\nu \in [0, \infty)$ on the functions $u^{(n)}$ $(n = 0, 1, \cdots, N+1)$ appearing in (5.28):

$$\begin{cases} \Box u^{(n)} = 0, \\ L_n^\nu u^{(n)} = \bar{L}_n^\nu u^{(n+1)}, \\ \Box u^{(n+1)}(\nu, 0) = 0. \end{cases} \quad \Leftrightarrow \quad \Box u^{(n+1)} = 0. \tag{B.3}$$

where $\Box$ represents the wave operator.

Proof. By commutativity of the linear operators in (B.1), and using (B.2):

$$\Box L_n^\nu u^{(n)} = \Box \bar{L}_n^\nu u^{(n+1)} \quad \Leftrightarrow \quad L_n^\nu \Box u^{(n)} = \bar{L}_n^\nu \Box u^{(n+1)} \quad \Leftrightarrow$$
$$0 = \bar{L}_n^\nu \Box u^{(n+1)} \quad \Leftrightarrow \quad \Box u^{(n+1)} = 0.$$

(*q.e.d.*)

Thus, knowing $\Box u^{(0)} = 0$ on $\nu \in [0, \infty)$, we can inductively prove that $\Box u^{(n)} = 0$ on $\nu \in [0, \infty)$ assuming up to second temporal and spatial derivatives of $u^{(n+1)} = 0$ are zero at $t = 0$.

Proofs presented in this section is found in [11].

B.2 Derivation of edge and corner auxiliary functions

In Sec.5.3 we rederived complete radiation boundary conditions proposed in [27] based on a half-space problem on $x < 0$ in three dimensions ($d = 3$). In the derivation, auxiliary functions $u^{(n)}$ were defined on $x \geq 0$. For computational efficiency, however, these auxiliary functions are defined only on the boundary $x = 0$ in practical implementations and this boundary always intersects with other boundaries, e.g. Dirichlet boundaries, Neumann boundaries, and/or other absorbing boundaries, on, say $y = 0$. On these intersections, one needs a set of boundary conditions for $u^{(n)}$ themselves. These additional conditions were derived in two dimensions in [29] for the second-order formulation of (5.14) and in [27] for the first-order formulation (5.28). In this section we derive these conditions based on the same idea employed in [29, 27], but in three dimensions. See also Appendix B.3 for their second-order counterpart to be used with the second-order formulation proposed in Sec.5.4.

Suppose again that a set of face functions $u^{(n)}$ ($n = 0, 1, \cdots, N$) is defined on $x \geq 0$ and suppose for a moment that we should apply Dirichlet boundary condition $u^{(0)} = 0$ on $y = 0$. Then on $\{x \geq 0\} \cap \{y = 0\}$ one has:

$$L_0^x u^{(0)} = \bar{L}_0^x u^{(1)} \quad \Leftrightarrow \quad 0 = \bar{L}_0^x u^{(1)} \quad \Leftrightarrow \quad u^{(1)} = 0,$$

where the last identity was aided by (B.2) provided $u^{(1)} = 0$ at $t = 0$. Inductively, one can prove $u^{(n)} = 0$ ($n = 1, \cdots, N$) on $y = 0$, which provides a set of boundary condition for auxiliary functions on $y = 0$.

Suppose next that we should apply Neumann boundary condition $\partial_y u^{(0)} = 0$ on $y = 0$. Then on $x \geq 0$:

$$\partial_y L_0^x u^{(0)} = \partial_y \bar{L}_0^x u^{(1)} \quad \Leftrightarrow \quad L_0^x \partial_y u^{(0)} = \bar{L}_0^x \partial_y u^{(1)},$$

and evaluating this on $y = 0$, one has on $\{x \geq 0\} \cap \{y = 0\}$:

$$0 = \bar{L}_0^x \partial_y u^{(1)} \quad \Leftrightarrow \quad \partial_y u^{(1)} = 0,$$

where the identity was again aided by (B.2) provided $\partial_y u^{(1)} = 0$ at $t = 0$. Inductively, one can prove $\partial_t u^{(n)} = 0$ $(n = 1, \cdots, N)$ on $y = 0$, which provides a set of boundary condition for auxiliary functions on $y = 0$.

Things become more involved when one should apply another set of absorbing boundary conditions on $y = 0$, which is the focus of Sec.B.2.1. In Sec.B.2.1 we end up introducing another set of auxiliary functions $u^{(i,j)}$ $(i, j = 0, 1, \cdots, N + 1)$ which would, in practical implementations, only be introduced on the edge $\{x = 0, y = 0\}$. In practice, however, domains are also bounded in the z-direction and the edge $\{x = 0, y = 0\}$ also intersects with another boundary, say $z = 0$, which requires one to apply appropriate boundary conditions to $u^{(i,j)}$ on $z = 0$. If one imposes Dirichlet $u^{(0)} = 0$ or Neumann $\partial_z u^{(0)} = 0$ boundary conditions on $z = 0$, we can readily obtain the consistent boundary conditions for $u^{(i,j)}$ on $z = 0$ just as we have done above. If one imposes yet another set of absorbing boundary conditions on $z = 0$, one would need to define yet another set of auxiliary functions $u^{(i,j,k)}$ $(i, j, k = 0, 1, \cdots, N + 1)$, which would, in practical implementations, only be introduced at the corner $\{x = 0, y = 0, z = 0\}$. This condition is discussed in Sec.B.2.2.

For simplicity we will only discuss cases in which absorbing boundaries intersect with each other at right angles. This condition is relaxed in the context of PMDLs in [25]; see Sec.4 and Appendix B.4 for the equivalence between CRBCs and PMDLs. Without loss of generality, we derive edge conditions for the edge $\{x = 0, y = 0\}$ and corner conditions for the corner $\{x = 0, y = 0, z = 0\}$. We will also assume that all absorbing boundaries have the same order N. Extension of our derivation to a more general cases where each of them have a different order is straightforward.

We consider as an example the domain $\Omega = (-\infty, 0)^3$ with CRBCs on three faces $x = 0$, $y = 0$, and $z = 0$. We assume that all sources and initial data are supported on $(-\infty, 0)^3$.

Fig.B.1 depicts auxiliary functions to be introduced for this example problem for CRBC order of $N = 3$. Planes (shown in green) represent *face* auxiliary functions $u^{(i)}$ $(i = 0, a, 1, \cdots, N)$ introduced in (5.28) with $u^{(0)} := u$ being the primary function defined on Ω. $u^{(a)}$ is an auxiliary function introduced in Sec.5.4 for our second-order formulation. Lines (shown in red) represent *edge* auxiliary functions $u^{(i,j)}$ $(i, j = 0, a, 1, \cdots, N)$ with $u^{(i,0)}$ and $u^{(0,j)}$ identified as face auxiliary functions introduced on two absorbing faces $x = x_0$ and $y = y_0$ (The face auxiliary functions on $y = y_0$ are not shown). $u^{(i,a)}$ and $u^{(a,j)}$ are only introduced when we derive corresponding second-order formulation on edges in Appendix B.3.1. Finally, points (shown in black) represent *corner* auxiliary functions $u^{(i,j,k)}$ $(i, j, k = 0, a, 1, \cdots, N)$ with $u^{(i,j,0)}$, $u^{(i,0,k)}$, and $u^{(0,j,k)}$ identified as edge auxiliary functions introduced on three absorbing edges $\{x = 0, y = 0\}$, $\{x = 0, z = 0\}$, and $\{y = 0, z = 0\}$ (The edge auxiliary functions on $\{x = 0, z = 0\}$ and $\{y = 0, z = 0\}$ are not shown). $u^{(i,j,a)}$, $u^{(i,a,k)}$, and $u^{(a,j,k)}$ are only introduced when we derive the corresponding second-order formulation on corners in Appendix B.3.2. The use of these sets of auxiliary functions on edges and at corners is justified in subsequent sections.

It is emphasized that, though otherwise depicted in Fig.B.1, auxiliary functions $u^{(i)}$, $u^{(i,j)}$, and $u^{(i,j,k)}$ are not restricted on faces, edges, or corners in the derivation phase. These restrictions are added in practical implementations, in which case Fig.B.1 is greatly appreciated.

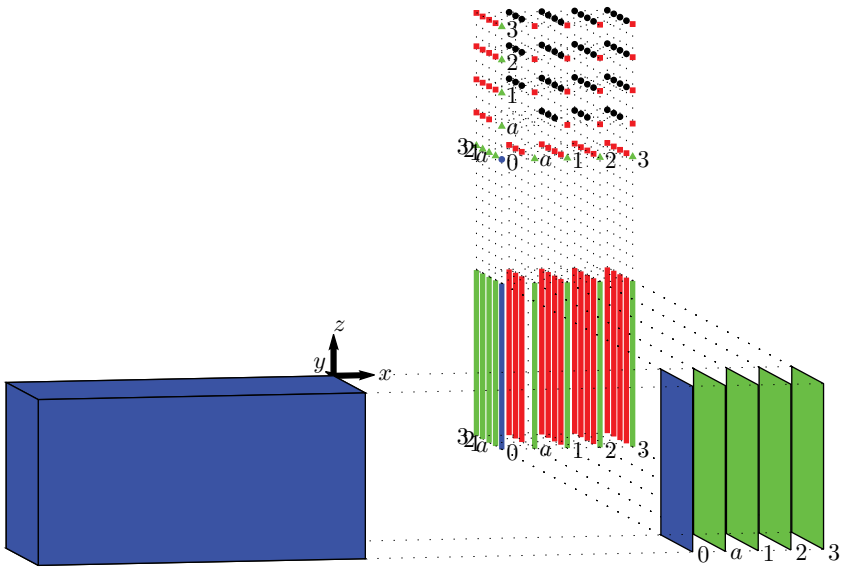


Figure B.1: Schematic of auxiliary functions for face $\{x = 0\}$, edge $\{x = 0, y = 0\}$, and corner $\{x = 0, y = 0, z = 0\}$ for order $N = 3$.

In the sequel, we use the following notation:

$$\mathbb{R}^d_\nu = \{(x, y, z) \in \mathbb{R}^d : \nu \geq 0\},$$

for $\nu = x, y, z$.

B.2.1 Edges

In this section we derive a set of auxiliary functions to be introduced on the edge $\{x = 0, y = 0\}$ when we have two sets of absorbing boundaries on $x = 0$ and $y = 0$ both of order N.

Denote by $u^{(i)}$ $(i = 0, 1, \cdots, N+1)$ and $u^{(j)}$ $(j = 0, 1, \cdots, N+1)$ the set of face auxiliary functions for the faces $\{x = 0\}$ and $\{y = 0\}$, respectively. $u^{(i)}$ and $u^{(j)}$ represent two distinct sets of functions, but we merely distinguish them by using different indices i and j to simplify the notation.

Let us focus on $u^{(i)}$ for a moment. We seek an appropriate representation of $\partial_y u^{(i)}\big|_{y=0}$ $(i = 1, \cdots, N + 1)$. We know from Appendix B.1 that $u^{(i)}$ also satisfies the wave equation on $\mathbb{R}^d_x$, which means that $u^{(i)}$ itself represents a wave propagating on an unbounded domain $\mathbb{R}^d_x$. Therefore it is natural to introduce similar recursion to (5.28) for each $u^{(i)}$ on $\mathbb{R}^d_x \cap \mathbb{R}^d_y$ to represent the unboundedness on $y \geq 0$. Specifically, we introduce, for each $i \in \{0, 1, \cdots, N+1\}$ and for $j = 0, 1, \cdots, N$:

$$L^y_j u^{(i,j)} = \bar{L}^y_j u^{(i,j+1)}, \quad \text{on } \mathbb{R}^d_x \cap \mathbb{R}^d_y, \tag{B.4}$$

where $u^{(i,j)}$ $(i,j = 0,1,\cdots,N,N+1)$ is a set of *edge* auxiliary functions with $u^{(i,0)} := u^{(i)}$. One can introduce a similar recursion for $u^{(j)}$ in the same manner for each $j \in \{0,1,\cdots,N,N+1\}$ and for $i = 0,1,\cdots,N$ as:

$$L_j^x u^{(i,j)'} = \bar{L}_j^x u^{(i+1,j)'}, \quad \text{on } \mathbb{R}_x^d \cap \mathbb{R}_y^d, \tag{B.5}$$

where $u^{(i,j)'}$ $(i,j = 0,1,\cdots,N+1)$ is another set of edge auxiliary functions with $u^{(0,j)'} := u^{(j)}$.

Note that $u^{(0,0)} = u^{(0,0)'} = u^{(0)}$ is the primary unknown function. We now show that $u^{(i,j)'}$ is identified as $u^{(i,j)}$ $(i,j = 1,\cdots,N+1)$.

Provided that $u^{(1,1)}$ is smooth enough, one has:

$$\begin{aligned}
&\bar{L}_0^x \bar{L}_0^y u^{(1,1)} = \bar{L}_0^x L_0^y u^{(1,0)} = L_0^y \bar{L}_0^x u^{(1,0)} = L_0^y \bar{L}_0^x u^{(i=1)} = L_0^y L_0^x u^{(0)}, \\
&\bar{L}_0^y \bar{L}_0^x u^{(1,1)'} = \bar{L}_0^y L_0^x u^{(0,1)'} = L_0^x \bar{L}_0^y u^{(0,1)'} = L_0^x \bar{L}_0^y u^{(j=1)} = L_0^x L_0^y u^{(0)},
\end{aligned}$$

where (B.4) and (B.5) were used. Taking a difference, one has:

$$\bar{L}_0^x \bar{L}_0^y \left(u^{(1,1)} - u^{(1,1)'} \right) = 0,$$

Provided that up to first temporal derivatives of $u^{(1,1)}$ and $u^{(1,1)'}$ are zero on $\mathbb{R}_x^d \cap \mathbb{R}_y^d$ at $t = 0$, $\bar{L}_0^y \left(u^{(1,1)} - u^{(1,1)'} \right) = 0$ on $\mathbb{R}_x^d \cap \mathbb{R}_y^d$ at $t = 0$, so (B.2) successively implies:

$$\bar{L}_0^y \left(u^{(1,1)} - u^{(1,1)'} \right) = 0 \quad \Rightarrow \quad u^{(1,1)} - u^{(1,1)'} = 0.$$

Likewise, one can inductively show $u^{(i+1,j+1)} = u^{(i+1,j+1)'}$ knowing $u^{(i,j)} = u^{(i,j)'}$, $u^{(i+1,j)} = u^{(i+1,j)'}$, $u^{(i,j+1)} = u^{(i,j+1)'}$, provided up to first temporal derivatives of $u^{(i+1,j+1)}$ and $u^{(i+1,j+1)'}$ are zero on $\mathbb{R}_x^d \cap \mathbb{R}_y^d$ at $t = 0$, concluding that $u^{(i,j)} = u^{(i,j)'}$ $(i,j = 0,1,\cdots,N,N+1)$.

Note that we can also show that $u^{(i,j)}$ $(i,j = 1,\cdots,N+1)$ also satisfies the wave equation on $\mathbb{R}_x^d \cap \mathbb{R}_y^d$ just in the same way as we have shown in Appendix B.1 for $u^{(i)}$.

Finally, we derive the termination condition similar to (5.28b). Noting that $u^{(N+1,0)}\big|_{x=0} = u^{(i=N+1)}\big|_{x=0} = 0$ from (5.28b), one has:

$$\bar{L}_0^y u^{(N+1,1)}\big|_{x=0} = L_0^y u^{(N+1,0)}\big|_{x=0} = 0.$$

(B.2) then implies $u^{(N+1,1)}\big|_{x=0} = 0$. One can inductively show:

$$u^{(N+1,j)}\big|_{x=0} = 0, \quad j = 0,1,\cdots,N+1. \tag{B.6}$$

In the same manner, one can also show:

$$u^{(i,N+1)}\big|_{y=0} = 0, \quad i = 0,1,\cdots,N+1. \tag{B.7}$$

B.2.2 Corners

In this section we derive a set of auxiliary functions to be introduced at the corner $\{x = 0, y = 0, z = 0\}$ when we have three sets of absorbing boundaries on $x = 0$, $y = 0$, and $z = 0$ all of order N.

Denote by $u^{(i,j)}$ $(i, j = 0, 1, \cdots, N+1)$, $u^{(i,k)}$ $(i, k = 0, 1, \cdots, N+1)$, and $u^{(j,k)}$ $(j, k = 0, 1, \cdots, N+1)$ the set of edge auxiliary functions for the edges $\{x = 0, y = 0\}$, $\{x = 0, z = 0\}$, and $\{y = 0, z = 0\}$, respectively. $u^{(i,j)}$, $u^{(i,k)}$, and $u^{(j,k)}$ represent three distinct sets of functions, but we merely distinguish them by using different sets of indices (i, j), (i, k), and (j, k) to simplify the notation.

Let us focus on $u^{(i,j)}$ for a moment. We seek an appropriate representation of $\partial_z u^{(i,j)}\big|_{z=0}$ $(i = 1, \cdots, N+1)$. As mentioned in Appendix B.2.1, $u^{(i,j)}$ also satisfies the wave equation on $\mathbb{R}^d_x \cap \mathbb{R}^d_y$, which means that $u^{(i,j)}$ itself represents a wave propagating on an unbounded domain $\mathbb{R}^d_x \cap \mathbb{R}^d_y$. Therefore it is natural to introduce similar recursion to (5.28) for each $u^{(i,j)}$ on $\mathbb{R}^d_x \cap \mathbb{R}^d_y \cap \mathbb{R}^d_z$ to represent the unboundedness on $z \geq 0$. Specifically, we introduce, for each pair of (i, j) $(i, j = 0, 1, \cdots, N+1)$ and for $k = 0, 1, \cdots, N$:

$$L^z_k u^{(i,j,k)} = \bar{L}^z_k u^{(i,j,k+1)}, \quad \text{on } \mathbb{R}^d_x \cap \mathbb{R}^d_y \cap \mathbb{R}^d_z, \tag{B.8}$$

where $u^{(i,j,k)}$ $(i, j, k = 0, 1, \cdots, N+1)$ is a set of *corner* auxiliary functions with $u^{(i,j,0)} := u^{(i,j)}$. One can introduce a similar recursion for $u^{(i,k)}$ in the same manner for each pair of (i, k) $(i, k = 0, 1, \cdots, N+1)$ and for $j = 0, 1, \cdots, N$ as:

$$L^y_j u^{(i,j,k)'} = \bar{L}^y_j u^{(i,j+1,k)'}, \quad \text{on } \mathbb{R}^d_x \cap \mathbb{R}^d_y \cap \mathbb{R}^d_z, \tag{B.9}$$

where $u^{(i,j,k)'}$ $(i, j, k = 0, 1, \cdots, N+1)$ is another set of corner auxiliary functions with $u^{(i,0,k)} := u^{(i,k)}$, and for $u^{(j,k)}$ for each pair of (j, k) $(j, k = 0, 1, \cdots, N+1)$ and for $i = 0, 1, \cdots, N$ as:

$$L^x_i u^{(i,j,k)''} = \bar{L}^x_i u^{(i+1,j,k)''}, \quad \text{on } \mathbb{R}^d_x \cap \mathbb{R}^d_y \cap \mathbb{R}^d_z, \tag{B.10}$$

where $u^{(i,j,k)''}$ $(i, j, k = 0, 1, \cdots, N+1)$ is yet another set of corner auxiliary functions with $u^{(0,j,k)} := u^{(j,k)}$.

Note that $u^{(0,0,0)} = u^{(0,0,0)'} = u^{(0,0,0)''} = u^{(0)}$ is the primary unknown function. We now show that $u^{(i,j,k)'}$ and $u^{(i,j,k)''}$ are identified as $u^{(i,j,k)}$ $(i, j, k = 1, \cdots, N+1)$.

Provided that $u^{(1,1,1)}$ is smooth enough, one has:

$$\begin{aligned}
\bar{L}^x_0 \bar{L}^y_0 \bar{L}^z_0 u^{(1,1,1)} &= \bar{L}^x_0 \bar{L}^y_0 L^z_0 u^{(1,1,0)} = L^z_0 \bar{L}^x_0 \bar{L}^y_0 u^{(i=1,j=1)} = L^z_0 L^x_0 L^y_0 u^{(0)}, \\
\bar{L}^x_0 \bar{L}^z_0 \bar{L}^y_0 u^{(1,1,1)'} &= \bar{L}^x_0 \bar{L}^z_0 L^y_0 u^{(1,0,1)'} = L^y_0 \bar{L}^x_0 \bar{L}^z_0 u^{(i=1,k=1)'} = L^y_0 L^x_0 L^z_0 u^{(0)}, \\
\bar{L}^y_0 \bar{L}^z_0 \bar{L}^x_0 u^{(1,1,1)''} &= \bar{L}^y_0 \bar{L}^z_0 L^x_0 u^{(0,1,1)''} = L^x_0 \bar{L}^y_0 \bar{L}^z_0 u^{(j=1,k=1)} = L^x_0 L^y_0 L^z_0 u^{(0)},
\end{aligned}$$

where (B.8), (B.9), and (B.10) and results from Sec.B.2.1 were used. Taking differences, one has:

$$\bar{L}^x_0 \bar{L}^y_0 \bar{L}^z_0 \left(u^{(1,1,1)} - u^{(1,1,1)'} \right) = 0,$$

$$\bar{L}_0^x \bar{L}_0^y \bar{L}_0^z \left(u^{(1,1,1)} - u^{(1,1,1)''} \right) = 0,$$

Provided that up to second temporal derivatives of $u^{(1,1,1)}$, $u^{(1,1,1)'}$, and $u^{(1,1,1)''} = 0$ are zero on $\mathbb{R}_x^d \cap \mathbb{R}_y^d \cap \mathbb{R}_z^d$ at $t = 0$, (B.2) successively implies:

$$\bar{L}_0^y \bar{L}_0^z \left(u^{(1,1,1)} - u^{(1,1,1)'} \right) = 0 \quad \Rightarrow \quad \bar{L}_0^z \left(u^{(1,1,1)} - u^{(1,1,1)'} \right) = 0 \quad \Rightarrow \quad u^{(1,1,1)} - u^{(1,1,1)'} = 0,$$
$$\bar{L}_0^y \bar{L}_0^z \left(u^{(1,1,1)} - u^{(1,1,1)''} \right) = 0 \quad \Rightarrow \quad \bar{L}_0^z \left(u^{(1,1,1)} - u^{(1,1,1)''} \right) = 0 \quad \Rightarrow \quad u^{(1,1,1)} - u^{(1,1,1)''} = 0.$$

Likewise, one can inductively show $u^{(i+1,j+1,k+1)} = u^{(i+1,j+1,k+1)'} = u^{(i+1,j+1,k+1)''}$ knowing $u^{(\bar{i},\bar{j},\bar{k})} = u^{(\bar{i},\bar{j},\bar{k})'} = u^{(\bar{i},\bar{j},\bar{k})''}$ for $(\bar{i},\bar{j},\bar{k}) = (i,j,k),(i,j,k+1),(i,j+1,k),(i,j+1,k+1),(i+1,k,j),(i+1,k,j+1),(i+1,k+1,j)$, provided up to second temporal derivatives of $u^{(i+1,j+1,k+1)}$, $u^{(i+1,j+1,k+1)'}$, and $u^{(i+1,j+1,k+1)''}$ are zero on $\mathbb{R}_x^d \cap \mathbb{R}_y^d \cap \mathbb{R}_z^d$ at $t = 0$, concluding that $u^{(i,j,k)} = u^{(i,j,k)'} = u^{(i,j,k)''}$ $(i,j,k = 0,1,\cdots,N,N+1)$.

Note that we can also show that $u^{(i,j,k)}$ $(i,j,k = 1,\cdots,N+1)$ also satisfies the wave equation on $\mathbb{R}_x^d \cap \mathbb{R}_y^d \cap \mathbb{R}_z^d$ just in the same way as we have shown in Appendix B.1 for $u^{(i)}$.

Finally, we derive the termination condition similar to (5.28b). Noting from Sec.B.2.1 that $u^{(N+1,1,0)}\big|_{x=0} = u^{(i=N+1,j=1)}\big|_{x=0} = 0$, one has:

$$\bar{L}_0^z u^{(N+1,1,1)}\big|_{x=0} = L_0^z u^{(N+1,1,0)}\big|_{x=0} = 0,$$

(B.2) then implies $u^{(N+1,1,1)}\big|_{x=0} = 0$. One can inductively show:

$$u^{(N+1,j,k)}\big|_{x=0} = 0, \quad j,k = 0,1,\cdots,N+1, \tag{B.11}$$

In the same manner, one can also show:

$$u^{(i,N+1,k)}\big|_{y=0} = 0, \quad i,k = 0,1,\cdots,N+1, \tag{B.12}$$
$$u^{(i,j,N+1)}\big|_{z=0} = 0, \quad j,k = 0,1,\cdots,N+1. \tag{B.13}$$

B.3 Second-order formulations on edges and corners

In this section we derive second-order formulations of edge and corner CRBCs derived in Appendices B.2.1 and B.2.2 which are to be used in conjunction with the second-order formulation for face CRBCs (5.38) derived in Sec.5.4.1. In the sequel, we will not present explicit matrix formulas, but systematic procedures to construct them.

B.3.1 Edges

In this section we present a systematic procedures to construct a second-order formulation of the edge CRBCs derived for the edge $\{x = 0, y = 0\}$ in Appendix B.3.1.

Recall that edge auxiliary functions $u^{(i,j)}$ $(i,j=0,1,\cdots,N+1)$ satisfy recursions (B.4) and (B.5) on $\mathbb{R}^d_x \cap \mathbb{R}^d_y$ along with termination conditions (B.6) and (B.7). For each such recursion, one can proceed as in Sec.5.4.1 to obtain a second-order formulation similar to (5.36) defined *on* $\{x=0, y=0\}$. Specifically, from recursion (B.4) and termination condition (B.7) we obtain $N+1$ equations for each $i \in \{1,\cdots,N\}$:

$$\boldsymbol{M}'_{tt}\begin{pmatrix} u^{(i,0)} \\ \boldsymbol{U} \end{pmatrix}_{,tt} = -\frac{1}{T}\boldsymbol{M}'_t\begin{pmatrix} u^{(i,0)} \\ \boldsymbol{U} \end{pmatrix}_{,t} - \frac{1}{T^2}\boldsymbol{M}_0\begin{pmatrix} u^{(i,0)} \\ \boldsymbol{U} \end{pmatrix} - c^2\boldsymbol{M}_{\tan}\nabla^2_{\tan\perp\{y=0\}}\begin{pmatrix} u^{(i,0)} \\ \boldsymbol{U} \end{pmatrix} - \frac{1}{T}\boldsymbol{m}'_t u^{(i,a)}_{,t} - \frac{1}{T^2}\boldsymbol{m}'_0 u^{(i,a)}, \qquad \text{(B.14)}$$

where, $\boldsymbol{U} = \left[u^{(i,1)},...,u^{(i,N)}\right]^{\mathrm{T}}$ and $(N+1)\times(N+1)$ matrices $\boldsymbol{M}'_{tt}$, $\boldsymbol{M}'_t$, $\boldsymbol{M}_0$, and $\boldsymbol{M}_F$ and $(N+1)\times 1$ vectors $\boldsymbol{m}'_t$ and $\boldsymbol{m}'_0$ are the same as those in (5.36). $\nabla^2_{\tan\perp\{y=0\}}$ represents Laplacian on plane $y=0$. As in Sec.5.4.1, we introduced another set of functions $u^{(i,a)} := (cT)\, u^{(i,0)}_{,y}$ for a direct representation of a normal derivative.

Similarly, from recursion (B.5) and termination condition (B.6) we obtain $N+1$ equations for each $j \in \{1,\cdots,N\}$:

$$\boldsymbol{M}'_{tt}\begin{pmatrix} u^{(0,j)} \\ \boldsymbol{U} \end{pmatrix}_{,tt} = -\frac{1}{T}\boldsymbol{M}'_t\begin{pmatrix} u^{(0,j)} \\ \boldsymbol{U} \end{pmatrix}_{,t} - \frac{1}{T^2}\boldsymbol{M}_0\begin{pmatrix} u^{(j,0)} \\ \boldsymbol{U} \end{pmatrix} - c^2\boldsymbol{M}_{\tan}\nabla^2_{\tan\perp\{x=0\}}\begin{pmatrix} u^{(0,j)} \\ \boldsymbol{U} \end{pmatrix} - \frac{1}{T}\boldsymbol{m}'_t u^{(a,j)}_{,t} - \frac{1}{T^2}\boldsymbol{m}'_0 u^{(a,j)}, \qquad \text{(B.15)}$$

where, $\boldsymbol{U} = \left[u^{(1,j)},...,u^{(N,j)}\right]^{\mathrm{T}}$, $\nabla^2_{\tan\perp\{x=0\}}$ represents Laplacian on plane $x=0$, and $u^{(a,j)} := (cT)\, u^{(0,j)}_{,x}$. Note again that these matrix equations as well as all functions and their derivatives are already evaluated on the edge $\{x=0, y=0\}$ and they are merely functions of z and t.

In addition to these matrix equations, we know from Appendix B.2.1 that each $u^{(i,j)}$ $(i,j=0,1,\cdots,N)$ satisfies the wave equation on $\mathbb{R}^d_x \cap \mathbb{R}^d_y$. We will evaluate these equations on the edge $\{x=0, y=0\}$ and use them in the sequel.

We have N^2+2N unknowns;

- $u^{(i,a)}$ $(i=1,\cdots,N)$,
- $u^{(a,j)}$ $(j=1,\cdots,N)$,
- $u^{(i,j)}$ $(i,j=1,\cdots,N)$,

and $3N^2+2N$ equations:

- the wave equation for $u^{(i,j)}$ $(i,j=1,\cdots,N)$,
- $(N+1)$ equations in (B.14) for each $i \in \{1,\cdots,N\}$,

- $(N+1)$ equations in (B.15) for each $j \in \{1, \cdots, N\}$.

This system involves first-order temporal derivatives of $u^{(i,a)}$ and $u^{(a,j)}$ and second-order temporal derivatives of $u^{(i,j)}$ $(i, j = 0, 1, \cdots, N)$. We solve this system of $3N^2+2N$ equations algebraically for:

- $u_{,t}^{(i,a)}$ $(i = 1, \cdots, N)$,
- $u_{,t}^{(a,j)}$ $(j = 1, \cdots, N)$,
- $u_{,tt}^{(i,j)}, u_{,xx}^{(i,j)}, u_{,yy}^{(i,j)}$ $(i, j = 1, \cdots, N)$,

in terms of:

- $u_{,tt}^{(i,0)}, u_{,t}^{(i,0)}, u^{(i,0)}, u_{,xx}^{(i,0)}, u_{,zz}^{(i,0)}, u^{(i,a)}$ $(i = 1, \cdots, N)$,
- $u_{,tt}^{(0,j)}, u_{,t}^{(0,j)}, u^{(0,j)}, u_{,yy}^{(0,j)}, u_{,zz}^{(0,j)}, u^{(a,j)}$ $(j = 1, \cdots, N)$,
- $u_{,t}^{(i,j)}, u^{(i,j)}, u_{,zz}^{(i,j)}$ $(i, j = 1, \cdots, N)$,

among which $u_{,xx}^{(i,0)}$ and $u_{,yy}^{(0,j)}$ can not be represented directly on the edge $\{x = 0, y = 0\}$. We use the wave equation to convert $u_{,xx}^{(i,0)}$ into $u_{,tt}^{(i,0)}$ and $\nabla^2_{\tan\perp\{x=0\}} u^{(i,0)}$ and then use (5.36) to convert $\nabla^2_{\tan\perp\{x=0\}} u^{(i,0)}$ into $u_{,tt}^{(i',0)}, u_{,t}^{(i',0)}, u^{(i',0)}$ $(i' = 0, \cdots, N)$ and $u_{,t}^{(a,0)}$ and $u^{(a,0)}$, which do not involve neither x- nor y-derivatives. $u_{,yy}^{(0,j)}$ $(j = 1, \cdots, N)$ is treated in the same way. Then we have $N^2 + 2N$ equations for $N^2 + 2N$ unknowns:

- $u_{,t}^{(i,a)}$ $(i = 1, \cdots, N)$,
- $u_{,t}^{(a,j)}$ $(j = 1, \cdots, N)$,
- $u_{,tt}^{(i,j)}$ $(i, j = 1, \cdots, N)$,

merely in terms of:

- $u_{,tt}^{(0,0)}, u_{,t}^{(0,0)}, u^{(0,0)}$,
- $u_{,t}^{(a,0)}, u^{(a,0)}, u_{,t}^{(0,a)}, u^{(0,a)}$,
- $u_{,tt}^{(i,0)}, u_{,t}^{(i,0)}, u^{(i,0)}, u_{,zz}^{(i,0)}, u^{(i,a)}$ $(i = 1, \cdots, N)$,
- $u_{,tt}^{(0,j)}, u_{,t}^{(0,j)}, u^{(0,j)}, u_{,zz}^{(0,j)}, u^{(a,j)}$ $(j = 1, \cdots, N)$,
- $u_{,t}^{(i,j)}, u^{(i,j)}, u_{,zz}^{(i,j)}$ $(i, j = 1, \cdots, N)$.

Introducing $v^{(i,j)} = u_{,t}^{(i,j)}$ $(i, j = 0, \cdots, N)$ to write the system in first order in time, we have $N^2 + 2N$ equations for $2N^2 + 2N$ unknowns:

- $u_{,t}^{(i,a)}$ $(i = 1, \cdots, N)$,
- $u_{,t}^{(a,j)}$ $(j = 1, \cdots, N)$,
- $u_{,t}^{(i,j)}$ $(i, j = 1, \cdots, N)$,
- $v_{,t}^{(i,j)}$ $(i, j = 1, \cdots, N)$,

in terms of:

- $v_{,t}^{(0,0)}, v^{(0,0)}, u^{(0,0)}$,
- $u_{,t}^{(a,0)}, u^{(a,0)}, u_{,t}^{(0,a)}, u^{(0,a)}$,
- $v_{,t}^{(i,0)}, v^{(i,0)}, u^{(i,0)}, u_{,zz}^{(i,0)}, u^{(i,a)}$ $(i = 1, \cdots, N)$,
- $v_{,t}^{(0,j)}, v^{(0,j)}, u^{(0,j)}, u_{,zz}^{(0,j)}, u^{(a,j)}$ $(j = 1, \cdots, N)$,
- $v^{(i,j)}, u^{(i,j)}, u_{,zz}^{(i,j)}$ $(i, j = 1, \cdots, N)$,

which can all be discretized on the edge $\{x = 0, y = 0\}$. This set of equations defines our second-order formulation of edge CRBCs.

B.3.2 Corners

In this section we present a systematic procedures to construct a second-order formulation of the corner CRBCs derived for the corner $\{x = 0, y = 0, z = 0\}$ in Appendix B.3.2.

Recall that corner auxiliary functions $u^{(i,j,k)}$ $(i, j, k = 0, 1, \cdots, N + 1)$ satisfy recursions (B.10), (B.9), and (B.8) on $\mathbb{R}_x^d \cap \mathbb{R}_y^d \cap \mathbb{R}_z^d$ along with termination conditions (B.11), (B.12), and (B.13). For each such recursion, one can proceed as in Sec.5.4.1 to obtain a second-order formulation similar to (5.36) defined *on* $\{x = 0, y = 0, z = 0\}$. Specifically, from recursion (B.8) and termination condition (B.13) we obtain $N + 1$ equations for each pair of (i, j), $i, j \in \{1, \cdots, N\}$:

$$\boldsymbol{M}'_{tt} \begin{pmatrix} u^{(i,j,0)} \\ \boldsymbol{U} \end{pmatrix}_{,tt} = -\frac{1}{T}\boldsymbol{M}'_t \begin{pmatrix} u^{(i,j,0)} \\ \boldsymbol{U} \end{pmatrix}_{,t} - \frac{1}{T^2}\boldsymbol{M}_0 \begin{pmatrix} u^{(i,j,0)} \\ \boldsymbol{U} \end{pmatrix} - c^2 \boldsymbol{M}_{\tan} \nabla^2_{\tan\perp\{z=0\}} \begin{pmatrix} u^{(i,j,0)} \\ \boldsymbol{U} \end{pmatrix} - \frac{1}{T}\boldsymbol{m}'_t u_{,t}^{(i,j,a)} - \frac{1}{T^2}\boldsymbol{m}'_0 u^{(i,j,a)}, \quad \text{(B.16)}$$

where, $\boldsymbol{U} = \left[u^{(i,j,1)}, ..., u^{(i,j,N)}\right]^{\mathrm{T}}$ and $(N+1)\times(N+1)$ matrices $\boldsymbol{M}'_{tt}$, $\boldsymbol{M}'_t$, $\boldsymbol{M}_0$, and $\boldsymbol{M}_F$ and $(N+1)\times 1$ vectors $\boldsymbol{m}'_t$ and $\boldsymbol{m}'_0$ are the same as those in (5.36). $\nabla^2_{\tan\perp\{z=0\}}$ represents Laplacian on plane $z = 0$. As in Sec.5.4.1, we introduced another set of functions $u^{(i,j,a)} := (cT)\, u_{,z}^{(i,j,0)}$ for a direct representation of a normal derivative.

Similarly, from recursion (B.9) and termination condition (B.12) we obtain $N+1$ equations for each pair of (i,k), $i,k \in \{1,\cdots,N\}$:

$$\boldsymbol{M}'_{tt}\begin{pmatrix} u^{(i,0,k)} \\ \boldsymbol{U} \end{pmatrix}_{,tt} = -\frac{1}{T}\boldsymbol{M}'_{t}\begin{pmatrix} u^{(i,0,k)} \\ \boldsymbol{U} \end{pmatrix}_{,t} - \frac{1}{T^2}\boldsymbol{M}_0\begin{pmatrix} u^{(i,0,k)} \\ \boldsymbol{U} \end{pmatrix} - c^2\boldsymbol{M}_{\tan}\nabla^2_{\tan\perp\{y=0\}}\begin{pmatrix} u^{(i,0,k)} \\ \boldsymbol{U} \end{pmatrix} - \frac{1}{T}\boldsymbol{m}'_t u^{(i,a,k)}_{,t} - \frac{1}{T^2}\boldsymbol{m}'_0 u^{(i,a,k)}, \quad \text{(B.17)}$$

where, $\boldsymbol{U} = \left[u^{(i,1,k)},...,u^{(i,N,k)}\right]^{\mathrm{T}}$ and $u^{(i,a,k)} := (cT)\,u^{(i,0,k)}_{,y}$, and from recursion (B.10) and termination condition (B.11) we obtain $N+1$ equations for each pair of (j,k), $j,k \in \{1,\cdots,N\}$:

$$\boldsymbol{M}'_{tt}\begin{pmatrix} u^{(0,j,k)} \\ \boldsymbol{U} \end{pmatrix}_{,tt} = -\frac{1}{T}\boldsymbol{M}'_{t}\begin{pmatrix} u^{(0,j,k)} \\ \boldsymbol{U} \end{pmatrix}_{,t} - \frac{1}{T^2}\boldsymbol{M}_0\begin{pmatrix} u^{(0,j,k)} \\ \boldsymbol{U} \end{pmatrix} - c^2\boldsymbol{M}_{\tan}\nabla^2_{\tan\perp\{x=0\}}\begin{pmatrix} u^{(0,j,k)} \\ \boldsymbol{U} \end{pmatrix} - \frac{1}{T}\boldsymbol{m}'_t u^{(a,j,k)}_{,t} - \frac{1}{T^2}\boldsymbol{m}'_0 u^{(a,j,k)}, \quad \text{(B.18)}$$

where, $\boldsymbol{U} = \left[u^{(1,j,k)},...,u^{(N,j,k)}\right]^{\mathrm{T}}$ and $u^{(a,j,k)} := (cT)\,u^{(0,j,k)}_{,x}$. Note again that these matrix equations as well as all functions and their derivatives are already evaluated at the corner $\{x=0,y=0,z=0\}$ and they are merely functions of t.

In addition to these matrix equations, we know from Appendix B.2.2 that each $u^{(i,j,k)}$ $(i,j,k=0,1,\cdots,N)$ satisfies the wave equation on $\mathbb{R}^d_x \cap \mathbb{R}^d_y \cap \mathbb{R}^d_z$. We will evaluate these equations at the corner $\{x=0,y=0,z=0\}$ and use them in the sequel.

We have N^3+3N^2 unknowns;

- $u^{(i,j,a)}$ $(i,j=1,\cdots,N)$,
- $u^{(i,a,k)}$ $(i,k=1,\cdots,N)$,
- $u^{(a,j,k)}$ $(j,k=1,\cdots,N)$,
- $u^{(i,j,k)}$ $(i,j,k=1,\cdots,N)$,

and $4N^3+3N^2$ equations:

- the wave equation for $u^{(i,j,k)}$ $(i,j,k=1,\cdots,N)$,
- $(N+1)$ equations in (B.16) for each pair of (i,j), $i,j \in \{1,\cdots,N\}$,
- $(N+1)$ equations in (B.17) for each pair of (i,k), $i,k \in \{1,\cdots,N\}$,
- $(N+1)$ equations in (B.18) for each pair of (j,k), $j,k \in \{1,\cdots,N\}$,

The system involves first-order temporal derivatives of $u^{(i,j,a)}$, $u^{(i,a,k)}$, and $u^{(a,j,k)}$ and second-order temporal derivatives of $u^{(i,j,k)}$ $(i,j,k=0,1,\cdots,N)$. We solve this system of $4N^3+3N^2$ equations algebraically for:

- $u_{,t}^{(i,j,a)}$ $(i,j=1,\cdots,N)$,
- $u_{,t}^{(i,a,k)}$ $(i,k=1,\cdots,N)$,
- $u_{,t}^{(a,j,k)}$ $(j,k=1,\cdots,N)$,
- $u_{,tt}^{(i,j,k)}, u_{,xx}^{(i,j,k)}, u_{,yy}^{(i,j,k)}, u_{,zz}^{(i,j,k)}$ $(i,j,k=1,\cdots,N)$,

in terms of:

- $u_{,tt}^{(i,j,0)}, u_{,t}^{(i,j,0)}, u^{(i,j,0)}, u_{,xx}^{(i,j,0)}, u_{,yy}^{(i,j,0)}, u^{(i,j,a)}$ $(i,j=1,\cdots,N)$,
- $u_{,tt}^{(i,0,k)}, u_{,t}^{(i,0,k)}, u^{(i,0,k)}, u_{,xx}^{(i,0,k)}, u_{,zz}^{(i,0,k)}, u^{(i,a,k)}$ $(i,k=1,\cdots,N)$,
- $u_{,tt}^{(0,j,k)}, u_{,t}^{(0,j,k)}, u^{(0,j,k)}, u_{,yy}^{(0,j,k)}, u_{,zz}^{(0,j,k)}, u^{(a,j,k)}$ $(j,k=1,\cdots,N)$,
- $u_{,t}^{(i,j,k)}, u^{(i,j,k)}$ $(i,j,k=1,\cdots,N)$,

among which terms with second-order spatial derivatives can not be represented directly at the corner $\{x=0,y=0,z=0\}$. We use the wave equation to convert $u_{,xx}^{(i,j,0)}$ into $u_{,tt}^{(i,j,0)}$ and $\nabla^2_{\tan\perp\{x=0\}}u^{(i,j,0)}$ and then use (5.36) to convert $\nabla^2_{\tan\perp\{x=0\}}u^{(i,j,0)}$ into $u_{,tt}^{(i',j,0)}, u_{,t}^{(i',j,0)}, u^{(i',j,0)}$ $(i'=0,\cdots,N)$ and $u_{,t}^{(a,j,0)}$ and $u^{(a,j,0)}$, which do not involve any spatial derivatives. Other terms with second-order spatial derivatives are treated in the same way. Then we have N^3+3N^2 equations for N^3+3N^2 unknowns:

- $u_{,t}^{(i,j,a)}$ $(i,j=1,\cdots,N)$,
- $u_{,t}^{(i,a,k)}$ $(i,k=1,\cdots,N)$,
- $u_{,t}^{(a,j,k)}$ $(j,k=1,\cdots,N)$,
- $u_{,tt}^{(i,j,k)}$ $(i,j,k=1,\cdots,N)$,

merely in terms of:

- $u_{,tt}^{(i,0,0)}, u_{,t}^{(i,0,0)}, u^{(i,0,0)}$ $(i=1,\cdots,N)$,
- $u_{,tt}^{(0,j,0)}, u_{,t}^{(0,j,0)}, u^{(0,j,0)}$ $(j=1,\cdots,N)$,
- $u_{,tt}^{(0,0,k)}, u_{,t}^{(0,0,k)}, u^{(0,0,k)}$ $(k=1,\cdots,N)$,
- $u_{,t}^{(i,a,0)}, u^{(i,a,0)}, u_{,t}^{(i,0,a)}, u^{(i,0,a)}$ $(i=1,\cdots,N)$,
- $u_{,t}^{(a,j,0)}, u^{(a,j,0)}, u_{,t}^{(0,j,a)}, u^{(0,j,a)}$ $(j=1,\cdots,N)$,
- $u_{,t}^{(a,0,k)}, u^{(a,0,k)}, u_{,t}^{(0,a,k)}, u^{(0,a,k)}$ $(k=1,\cdots,N)$,

- $u_{,tt}^{(i,j,0)}, u_{,t}^{(i,j,0)}, u^{(i,j,0)}, u^{(i,j,a)}$ $(i,j=1,\cdots,N)$,
- $u_{,tt}^{(i,0,k)}, u_{,t}^{(i,0,k)}, u^{(i,0,k)}, u^{(i,a,k)}$ $(i,k=1,\cdots,N)$,
- $u_{,tt}^{(0,j,k)}, u_{,t}^{(0,j,k)}, u^{(0,j,k)}, u^{(a,j,k)}$ $(j,k=1,\cdots,N)$,
- $u_{,t}^{(i,j,k)}, u^{(i,j,k)}$ $(i,j,k=1,\cdots,N)$,

Introducing $v^{(i,j,k)} = u_{,t}^{(i,j,k)}$ $(i,j,k=0,\cdots,N)$ to write the system in first order in time, we have $2*N^3+3N^2$ equations for $2*N^3+3N^2$ unknowns:

- $u_{,t}^{(i,j,a)}$ $(i,j=1,\cdots,N)$,
- $u_{,t}^{(i,a,k)}$ $(i,k=1,\cdots,N)$,
- $u_{,t}^{(a,j,k)}$ $(j,k=1,\cdots,N)$,
- $u_{,t}^{(i,j,k)}$ $(i,j,k=1,\cdots,N)$,
- $v_{,t}^{(i,j,k)}$ $(i,j,k=1,\cdots,N)$,

in terms of:

- $v_{,t}^{(i,0,0)}, v^{(i,0,0)}, u^{(i,0,0)}$ $(i=1,\cdots,N)$,
- $v_{,t}^{(0,j,0)}, v^{(0,j,0)}, u^{(0,j,0)}$ $(j=1,\cdots,N)$,
- $v_{,t}^{(0,0,k)}, v^{(0,0,k)}, u^{(0,0,k)}$ $(k=1,\cdots,N)$,
- $u_{,t}^{(i,a,0)}, u^{(i,a,0)}, u_{,t}^{(i,0,a)}, u^{(i,0,a)}$ $(i=1,\cdots,N)$,
- $u_{,t}^{(a,j,0)}, u^{(a,j,0)}, u_{,t}^{(0,j,a)}, u^{(0,j,a)}$ $(j=1,\cdots,N)$,
- $u_{,t}^{(a,0,k)}, u^{(a,0,k)}, u_{,t}^{(0,a,k)}, u^{(0,a,k)}$ $(k=1,\cdots,N)$,
- $v_{,t}^{(i,j,0)}, v^{(i,j,0)}, u^{(i,j,0)}, u^{(i,j,a)}$ $(i,j=1,\cdots,N)$,
- $v_{,t}^{(i,0,k)}, v^{(i,0,k)}, u^{(i,0,k)}, u^{(i,a,k)}$ $(i,k=1,\cdots,N)$,
- $v_{,t}^{(0,j,k)}, v^{(0,j,k)}, u^{(0,j,k)}, u^{(a,j,k)}$ $(j,k=1,\cdots,N)$,
- $v^{(i,j,k)}, u^{(i,j,k)}$ $(i,j,k=1,\cdots,N)$,

which does not require further spatial discretization. This set of equations defines our second-order formulation of corner CRBCs.

B.4 An alternative second-order formulation

In this section, we derive an alternative second-order formulation of the complete radiation boundary conditions (5.28) with (5.29) for a more general two-dimensional ($d = 2$) vector equation (4.1) in the same setting as considered in the context of the PMDLs in Sec.4.2. Extension of the following discussion to the three-dimensional counterpart of (4.1) should be straightforward. To employ complete radiation boundary conditions which is developed for the scalar wave equation for the more general vector equation (4.1), we merely apply (5.28) with (5.29) to each component u_i $(i = 1, 2)$ of the solution vector $\boldsymbol{u}$, that is:

$$\left(\frac{\cos\phi_n}{c}\partial_t + \partial_x + \frac{1}{cT}\frac{\sin^2\phi_n}{\cos\phi_n}\right) u_i^{(n)} = \left(\frac{\cos\bar{\phi}_n}{c}\partial_t - \partial_x + \frac{1}{cT}\frac{\sin^2\bar{\phi}_n}{\cos\bar{\phi}_n}\right) u_i^{(n+1)}, \quad \text{on } \mathbb{R}_x^d, \tag{B.19a}$$

$$u_i^{(N+1)} = 0, \quad \text{on } x = 0, \tag{B.19b}$$

with initial conditions:

$$u_i^{(n)}(x, y, 0) = 0, \quad \text{on } \mathbb{R}_x^d, \tag{B.20a}$$

$$\partial_t u_i^{(n)}(x, y, 0) = 0, \quad \text{on } \mathbb{R}_x^d, \tag{B.20b}$$

where $\mathbb{R}_x^d = \{(x, y) : x \geq 0\}$, $u_i^{(0)} := u_i$, $u_i^{(n)}$ $(n = 1, \cdots, N+1)$ is a set of auxiliary functions defined on $\mathbb{R}_x^d$, ϕ_n and $\bar{\phi}_n$ $(n = 0, 1, \cdots, N)$ are two sets of parameters. In this section, we impose a minor restriction to ϕ_n and $\bar{\phi}_n$ and set $\bar{\phi}_n = \phi_n$. These boundary conditions should give a good approximation to the boundary traction (4.2) at $x = 0$.

For convenience, we denote $\boldsymbol{u}_i = \left[u_i^{(1)}, \cdots, u_i^{(N)}, u_i^{(N+1)}\right]^{\mathrm{T}}$. It can be shown, in the same manner as we have done for the scalar wave equation in Appendix B.1, that each $u_i^{(n)}$ $(n = 1, \cdots, N+1)$ also satisfies (4.1) on $\mathbb{R}_x^d$. Therefore, using indicial notation, we have:

$$\boldsymbol{u}_{i,tt} - G_{xx}(i,j)\boldsymbol{u}_{j,xx} - \left(G_{xy}(i,j) + G_{xy}^{\mathrm{T}}(i,j)\right)\boldsymbol{u}_{j,xy} - G_{yy}(i,j)\boldsymbol{u}_{j,yy} = 0, \quad \text{on } \mathbb{R}_x^d, \quad i = 1, 2. \tag{B.21}$$

The alternative formulation to be derived turns out to be identical to the second-order formulation of PMDLs. Though in Sec.4.3, equivalence of the first-order CRBCs and the first-order PMDLs is discussed, second-order counterpart is not trivial. To our knowledge, this is the first work in which a second-order CRBCs are identified as the second-order PMDLs.

We employ exactly the same procedure as in Sec.5.4.1, but with different manipulation strategy and in transformed space. We first take Laplace- and Fourier transforms in t- and y-directions in (4.1) defined on $\mathbb{R}_x^d$ and work with the transformed equations (4.3). Note since all sources and initial data are supported in $x < 0$, we set $\boldsymbol{f} = \boldsymbol{0}$ in (4.3).

Boundary conditions (B.19) are transformed accordingly incorporating the initial conditions (B.20) to yield:

$$(\gamma_n + \partial_x)\, u_i^{(n)} = (\gamma_n - \partial_x)\, u_i^{(n+1)}, \quad \text{on } x \geq 0 \tag{B.22a}$$

$$u_i^{(N+1)} = 0, \quad \text{on } x = 0, \tag{B.22b}$$

where

$$\gamma_n = \frac{\cos\phi_n}{c}s + \frac{\sin^2\phi_n}{\cos\phi_n}. \tag{B.23}$$

This set of boundary conditions are to represent the transformed boundary traction (4.4) on $x = 0$. We also transform (B.21) incorporating (B.20) to obtain:

$$s^2\boldsymbol{u}_i - G_{xx}(i,j)\boldsymbol{u}_{j,xx} - \mathrm{i}k_y\left(G_{xy}(i,j) + G_{xy}^{\mathrm{T}}(i,j)\right)\boldsymbol{u}_{j,x} - (\mathrm{i}k_y)^2 G_{yy}(i,j)\boldsymbol{u}_j = 0, \quad \text{on } \mathbb{R}_x^d, \quad i = 1,2. \tag{B.24}$$

One can write (B.22a) in a matrix form as:

$$\boldsymbol{M}\boldsymbol{u}_{i,x} = -\frac{1}{2}\boldsymbol{\Gamma}\boldsymbol{u}_i - \frac{1}{2\gamma_0}\boldsymbol{e}u_{i,x}^{(0)} - \frac{1}{2}\boldsymbol{e}u_i^{(0)}, \tag{B.25}$$

where

$$\boldsymbol{M} = \frac{1}{4}\begin{pmatrix} \frac{2}{\gamma_0} & & & & & \\ \frac{2}{\gamma_1} & \frac{2}{\gamma_1} & & & & \\ & \frac{2}{\gamma_2} & \frac{2}{\gamma_2} & & & \\ & & \ddots & & & \\ & & & \frac{2}{\gamma_{N-1}} & & \\ & & & \frac{2}{\gamma_N} & \frac{2}{\gamma_N} \end{pmatrix}, \quad \boldsymbol{\Gamma} = \begin{pmatrix} -1 & & & & & \\ 1 & -1 & & & & \\ & 1 & -1 & & & \\ & & & \ddots & & \\ & & & & -1 & \\ & & & & 1 & -1 \end{pmatrix}, \quad \boldsymbol{e} = \begin{pmatrix} 1 \\ 0 \\ \\ \vdots \\ \\ 0 \end{pmatrix}.$$

Taking x-derivative of (B.25) and eliminating resulting $\boldsymbol{u}_{i,x}$ again using (B.25), one obtains:

$$\boldsymbol{M}\boldsymbol{u}_{i,xx} = \frac{1}{4}\boldsymbol{\Gamma}\boldsymbol{M}^{-1}\boldsymbol{\Gamma}\boldsymbol{u}_i + \frac{1}{4\gamma_0}\boldsymbol{\Gamma}\boldsymbol{M}^{-1}\boldsymbol{e}u_{i,x}^{(0)} + \frac{1}{4}\boldsymbol{\Gamma}\boldsymbol{M}^{-1}\boldsymbol{e}u_i^{(0)} - \frac{1}{2\gamma_0}\boldsymbol{e}u_{i,xx}^{(0)} - \frac{1}{2}\boldsymbol{e}u_{i,x}^{(0)}. \tag{B.26}$$

We here premultiply (B.24) by $\boldsymbol{M}$ to obtain:

$$s^2\boldsymbol{M}\boldsymbol{u}_i - G_{xx}(i,j)\boldsymbol{M}\boldsymbol{u}_{j,xx} - \mathrm{i}k_y\left(G_{xy}(i,j) + G_{xy}^{\mathrm{T}}(i,j)\right)\boldsymbol{M}\boldsymbol{u}_{j,x} - (\mathrm{i}k_y)^2 G_{yy}(i,j)\boldsymbol{M}\boldsymbol{u}_j = 0. \tag{B.27}$$

We then substitute (B.25) and (B.26) for $\boldsymbol{M}\boldsymbol{u}_{j,x}$ and $\boldsymbol{M}\boldsymbol{u}_{j,xx}$ in (B.27), to obtain:

$$\begin{aligned} &G_{xx}(i,j)\left(-\frac{1}{4\gamma_0}\boldsymbol{\Gamma}\boldsymbol{M}^{-1}\boldsymbol{e} + \frac{1}{2}\boldsymbol{e}\right)u_{j,x}^{(0)} \\ &+\frac{1}{2\gamma_0}\left(G_{xx}(i,j)u_{j,xx}^{(0)} + (\mathrm{i}k_y)\left(G_{xy}(i,j) + G_{xy}^{\mathrm{T}}(i,j)\right)u_{j,x}^{(0)}\right)\boldsymbol{e} + s^2\boldsymbol{M}\boldsymbol{u}_i - (\mathrm{i}k_y)^2 G_{yy}(i,j)\boldsymbol{M}\boldsymbol{u}_j \\ &+\frac{\mathrm{i}k_y}{2}\left(G_{xy}(i,j) + G_{xy}^{\mathrm{T}}(i,j)\right)\left(\boldsymbol{e}u_j^{(0)} + \boldsymbol{\Gamma}\boldsymbol{u}_j\right) \end{aligned}$$

$$+G_{xx}(i,j)\left(-\frac{1}{4}\boldsymbol{\Gamma}\boldsymbol{M}^{-1}\boldsymbol{e}u_j^{(0)}-\frac{1}{4}\boldsymbol{\Gamma}\boldsymbol{M}^{-1}\boldsymbol{\Gamma}\boldsymbol{u}_j\right)=\mathbf{0}. \tag{B.28}$$

Note that (B.28) does not involve x-derivatives of $\boldsymbol{u}_j$ ($j=1,\cdots,N+1$) anymore. We now restrict (B.28) to $x=0$ and apply the termination conditions (B.22b). All functions and derivatives of functions hereafter should be understood as their restrictions to $x=0$.

(B.28) now represents a system of $N+1$ equations on $\boldsymbol{u}_i=\left[u_i^{(0)},u_i^{(1)},\cdots,u_i^{(N)}\right]^{\mathrm{T}}$ with an abuse of notation $\boldsymbol{u}_i$. Multiplying (B.28) by a $(N+1)\times(N+1)$ permutation matrix:

$$\begin{pmatrix} 1 & & & & & \\ 1 & 1 & & & & \\ & 1 & 1 & & & \\ & & & \ddots & & \\ & & & & 1 & \\ & & & & 1 & 1 \end{pmatrix},$$

one obtains:

$$\begin{aligned} &G_{xx}(i,j)\boldsymbol{e}u_{j,x}^{(0)} \\ &+\left[\left(s^2\delta_{ij}-(\mathfrak{i}k_y)^2\,G_{yy}(i,j)\right)\boldsymbol{M}_0+\frac{\mathfrak{i}k_y}{2}\left(G_{xy}(i,j)+G_{xy}^{\mathrm{T}}(i,j)\right)\boldsymbol{M}_1+G_{xx}(i,j)\boldsymbol{M}_2\right]\boldsymbol{u}_j=\mathbf{0}, \end{aligned} \tag{B.29}$$

where

$$\boldsymbol{M}_0=\frac{1}{4}\left(\begin{array}{c:cccc} \frac{2}{\gamma_0} & \frac{2}{\gamma_0} & & & \\ \hdashline \frac{2}{\gamma_0} & \frac{2}{\gamma_0}+\frac{2}{\gamma_1} & \frac{2}{\gamma_1} & & \\ & \frac{2}{\gamma_1} & \frac{2}{\gamma_1}+\frac{2}{\gamma_2} & & \\ & & & \ddots & \\ & & & \frac{2}{\gamma_{N-2}}+\frac{2}{\gamma_{N-1}} & \frac{2}{\gamma_{N-1}} \\ & & & \frac{2}{\gamma_{N-1}} & \frac{2}{\gamma_{N-1}}+\frac{2}{\gamma_N} \end{array}\right),$$

$$\boldsymbol{M}_1=\left(\begin{array}{c:cccc} 1 & -1 & & & \\ \hdashline 1 & 0 & -1 & & \\ & 1 & 0 & -1 & \\ & & \ddots & & \\ & & 1 & 0 & -1 \\ & & & 1 & 0 \end{array}\right),$$

$$\boldsymbol{M}_2 = \left(\begin{array}{c:cccccc} \frac{\gamma_0}{2} & -\frac{\gamma_0}{2} & & & & & \\ -\frac{\gamma_0}{2} & \frac{\gamma_0}{2}+\frac{\gamma_1}{2} & -\frac{\gamma_1}{2} & & & & \\ & -\frac{\gamma_1}{2} & \frac{\gamma_1}{2}+\frac{\gamma_2}{2} & & & & \\ & & & \ddots & & & \\ & & & & \frac{\gamma_{N-2}}{2}+\frac{\gamma_{N-1}}{2} & -\frac{\gamma_{N-1}}{2} & \\ & & & & -\frac{\gamma_{N-1}}{2} & \frac{\gamma_{N-1}}{2}+\frac{\gamma_N}{2} \end{array}\right).$$

Due to its special structure, (B.29) can be deassembled as:

$$\begin{aligned} \begin{pmatrix} t_i^{(n)} \\ 0 \end{pmatrix} + \Bigg[& \frac{\frac{2}{\gamma_n}}{4}\left(s^2\delta_{ij} - (\mathrm{i}k_y)^2 G_{yy}(i,j)\right)\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \\ & + \frac{\mathrm{i}k_y}{2} G_{xy}(i,j)\begin{pmatrix} -1 & -1 \\ 1 & 1 \end{pmatrix} \\ & + \frac{\mathrm{i}k_y}{2} G_{xy}^{\mathrm{T}}(i,j)\begin{pmatrix} 1 & -1 \\ 1 & -1 \end{pmatrix} \\ & + \frac{\gamma_n}{2} G_{xx}(i,j)\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \qquad\qquad \Bigg] \begin{pmatrix} u_j^{(n)} \\ u_j^{(n+1)} \end{pmatrix} - \begin{pmatrix} 0 \\ t_i^{(n+1)} \end{pmatrix} = \mathbf{0}, \end{aligned} \tag{B.30}$$

where

$$t_i^{(n)} = G_{xx}(i,j)u_{j,x}^{(n)} + \mathrm{i}k_y G_{xy}(i,j)u_j^{(n)},$$

for $n = 0, ..., N$. Note that one can reassemble (B.30) to obtain (B.28) by *gluing* together the $t^{(n)}$ appearing in $(n-1)$th and nth sets of equations.

Finally we write (B.30) in a matrix/vector form as:

$$\begin{aligned} \begin{pmatrix} \boldsymbol{t}^{(n)} \\ \mathbf{0} \end{pmatrix} + \Bigg[& \frac{\frac{2}{\gamma_n}}{4}\begin{pmatrix} s^2\boldsymbol{I} - (\mathrm{i}k_y)^2\boldsymbol{G}_{yy} & s^2\boldsymbol{I} - (\mathrm{i}k_y)^2\boldsymbol{G}_{yy} \\ s^2\boldsymbol{I} - (\mathrm{i}k_y)^2\boldsymbol{G}_{yy} & s^2\boldsymbol{I} - (\mathrm{i}k_y)^2\boldsymbol{G}_{yy} \end{pmatrix} \\ & + \frac{\mathrm{i}k_y}{2}\begin{pmatrix} -\boldsymbol{G}_{xy} & -\boldsymbol{G}_{xy} \\ \boldsymbol{G}_{xy} & \boldsymbol{G}_{xy} \end{pmatrix} \\ & + \frac{\mathrm{i}k_y}{2}\begin{pmatrix} \boldsymbol{G}_{xy}^{\mathrm{T}} & -\boldsymbol{G}_{xy}^{\mathrm{T}} \\ \boldsymbol{G}_{xy}^{\mathrm{T}} & -\boldsymbol{G}_{xy}^{\mathrm{T}} \end{pmatrix} \\ & + \frac{\gamma_n}{2}\begin{pmatrix} \boldsymbol{G}_{xx} & -\boldsymbol{G}_{xx} \\ -\boldsymbol{G}_{xx} & \boldsymbol{G}_{xx} \end{pmatrix} \qquad\qquad \Bigg] \begin{pmatrix} \boldsymbol{u}^{(n)} \\ \boldsymbol{u}^{(n+1)} \end{pmatrix} - \begin{pmatrix} \mathbf{0} \\ \boldsymbol{t}^{(n+1)} \end{pmatrix} = \mathbf{0}, \end{aligned}$$

where

$$\boldsymbol{t}^{(n)} = \boldsymbol{G}_{xx}\boldsymbol{u}_{,x}^{(n)} + \mathrm{i}k_y\boldsymbol{G}_{xy}\boldsymbol{u}^{(n)},$$

which is nothing but PMDL formulation (4.7) with $L_n = \frac{2}{\gamma_n}$ recognizing that $\boldsymbol{t}^{(n)}$ given here is identical to the one given in (4.8).

Appendix C

Plancherel's and Parseval's identities

C.1 Plancherel's identity

Let $f(X) \in L^1(\mathbb{R})$. The Fourier transform of $f(X) \in L^1(\mathbb{R})$ is defined as:

$$\hat{f}(x) = \int_{-\infty}^{+\infty} e^{-\mathrm{i}xX} f(X) \mathrm{d}X.$$

Assuming that $\hat{f}(x) \in L^1(\mathbb{R})$, the inverse Fourier transform is defined as:

$$f(X) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} e^{\mathrm{i}xX} \hat{f}(x) \mathrm{d}x.$$

If $f(X), g(X) \in L^1(\mathbb{R}) \cap L^2(\mathbb{R})$, one can prove the following Plancherel's identity:

$$\begin{aligned} \int_{-\infty}^{+\infty} f(X)\overline{g(X)}\mathrm{d}X &= \int_{-\infty}^{+\infty} f(X) \cdot \frac{1}{2\pi} \int_{-\infty}^{+\infty} \overline{\hat{g}(x)} e^{-\mathrm{i}xX} \mathrm{d}x\mathrm{d}X \\ &= \frac{1}{2\pi} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(X) e^{-\mathrm{i}xX} \mathrm{d}X \cdot \overline{\hat{g}(x)} \mathrm{d}x = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \hat{f}(x)\overline{\hat{g}(x)} \mathrm{d}x, \end{aligned} \tag{C.1}$$

which in turn states that $\hat{f}(x), \hat{g}(x) \in L^2(\mathbb{R})$. In the above proof, we used the fact that $\hat{g}(x) \in L^1(\mathbb{R})$ without proof. It should also be noted that, by using Plancherel's identity (C.1), one can extend the definition of Fourier transform to functions in $L_2(\mathbb{R})$. Rigorous proofs are found in [33].

The Plancherel's identity (C.1) can be extended to d-dimensional spaces as:

$$\int_{\mathbb{R}^d} f(\boldsymbol{X})\overline{g(\boldsymbol{X})}\mathrm{d}\boldsymbol{X} = \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} \hat{f}(\boldsymbol{x})\overline{\hat{g}(\boldsymbol{x})}\mathrm{d}\boldsymbol{x}. \tag{C.2}$$

C.2 Parseval's identity

Setting $g = f$ in the Plancherel's identity (C.1), one obtains the Parseval's identity:

$$\int_{-\infty}^{+\infty} |f(X)|^2 \, \mathrm{d}X = \frac{1}{2\pi} \int_{-\infty}^{+\infty} |\hat{f}(x)|^2 \mathrm{d}x. \tag{C.3}$$

Setting $g = f$ in (C.2), one obtains in d dimensions:

$$\int_{\mathbb{R}^d} |f(\boldsymbol{X})|^2 \, \mathrm{d}\boldsymbol{X} = \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} |\hat{f}(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x}. \tag{C.4}$$