A new high-order Fourier continuation-based elasticity solver for complex three-dimensional geometries

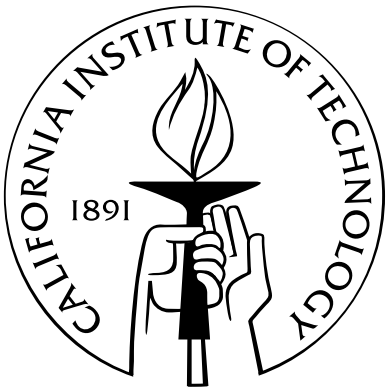
Thesis by

Faisal Amlani

In Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy



California Institute of Technology

Pasadena, California

2014

(Defended September 27, 2013)

In memory of my late father, who
would have read this and pretended he understood it.

Acknowledgements

First and foremost, I would like to express my deepest respect and gratitude to my advisor and mentor, Professor Oscar Bruno, whose passion and mastery of mathematics inspired me to push myself farther than I ever have before; my many years of training could not have been complete without his wisdom, encouragement and remarkable patience. I'd also like to extend my appreciation for being given the honor and privilege to help teach ACM 95/100 under Professor Bruno's tutelage as well as the tutelages of Professors Niles Pierce, Dan Meiron, Houman Owhadi and Tom Hou—it was an experience I will remember most fondly.

Sincerest thanks to Professor José Carlos López-Vázquez for his kindness and accommodations as well as for inspiring the subject of this thesis; and sincerest thanks to my thesis committee, Professors Dan Meiron, Houman Owhadi and Guillaume Blanquart, for their most helpful comments and suggestions. We are the sum of our past experiences and, to that end, I'd also like to thank all my former professors at Rice University for encouraging me to pursue graduate studies outside of the hedges: Professors Mark Embree, Liliana Borcea and Matthias Heinkenschloss for exposing me to academic life through teaching and research; and Professor Steve Cox, who played a most integral part of my time in undergraduate school and whom I will always consider, truly, as a friend.

The work contained in these pages would not have been possible without the help of the many staff, students, post-docs, and other scientists from which I learned a great deal: Daniel Appelö, Nathan Albin, Edwin Jimenez and Aditya Viswanathan for their sage advice, fun discussions and for allowing me to bounce ideas off their whiteboard;

Timothy Elling for fielding my many questions about C++; my dear friend Professor Harsha Bhat for making me appreciate the intricate physics of elastodynamics; Manuel Lombardini for suggestions concerning the thesis text and presentation; Peyman Tavallali and Stephen Becker for the wonderful digressions into philosophy that made my office-life pleasurable; and our department administrators—Sheila Shull, Sydney Garstang and Carmen Sirois—for their maternal friendship and limitless kindness.

And thank you to my friends who provided more than just hot meals: Andrew English for joining me in Southern California and being undoubtedly the nicest friend I've ever had; Paul Minor for his close friendship and for instilling in me a lifelong love of baseball and the piano; Matt Chevlen for always being there; Chris Rogan for teaching me his way of life; Niema Pahlevan for inspiring me as a scientist through his endless brilliance; Manuel Lombardini and Jon Miller, two friends who will be joining me on my next adventure (I couldn't be more excited); and my dearest friend Aldis Seaton, whom I haven't seen in years and hope to again soon.

I'd also like to thank my mother, who is smarter than she will ever acknowledge, and my brother Najeeb, whose wisdom belies his age—I owe everything I ever accomplished to their steadfast support and deep friendship. And, finally, I'd like to thank my wonderful girlfriend, Laurence Bodelot: although we have been apart for the last few months, you have been ever-present through all my work and your endless support and love during the final years of my graduate education I will never forget.

Abstract

This thesis presents a new approach for the numerical solution of three-dimensional problems in elastodynamics. The new methodology, which is based on a recently introduced Fourier continuation (FC) algorithm for the solution of Partial Differential Equations on the basis of accurate Fourier expansions of possibly non-periodic functions, enables fast, high-order solutions of the time-dependent elastic wave equation in a nearly dispersionless manner, and it requires use of CFL constraints that scale only linearly with spatial discretizations. A new FC operator is introduced to treat Neumann and traction boundary conditions, and a block-decomposed (sub-patch) overset strategy is presented for implementation of general, complex geometries in distributed-memory parallel computing environments. Our treatment of the elastic wave equation, which is formulated as a complex system of variable-coefficient PDEs that includes possibly heterogeneous and spatially varying material constants, represents the first fully-realized three-dimensional extension of FC-based solvers to date. Challenges for three-dimensional elastodynamics simulations such as treatment of corners and edges in three-dimensional geometries, the existence of variable coefficients arising from physical configurations and/or use of curvilinear coordinate systems and treatment of boundary conditions, are all addressed. The broad applicability of our new FC elasticity solver is demonstrated through application to realistic problems concerning seismic wave motion on three-dimensional topographies as well as applications to non-destructive evaluation where, for the first time, we present three-dimensional simulations for comparison to experimental studies of guided-wave scattering by through-thickness holes in thin plates.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

> I'm pickin' up good vibrations... she's giving
> me excitations...
>
> - Brian Wilson and
> the Beach Boys

The energy introduced by a disturbance in a medium propagates, as traveling waves, in a phenomenon that is familiar from everyday life, from seismic tremors in the earth to ripples in a pond, and even the stylus that follows the spiral groove in a vinyl music record—picking up these waves in the form of vibrations that travel along a metal band at the end of the tone arm and that eventually convert to electrical signals that are carried to an amplifier. The particular characteristics and information carried by wave propagation—which may vary from medium to medium—can be quite complex, having long been part of a great area of interest to theoreticians and experimentalists alike, and having led to numerous mathematical models for propagation and scattering of waves through gases, liquids, solids and free space [81, 26, 36, 39, 42, 61, 76].

Mechanical waves through a solid medium and, in particular, an elastic medium, are the specific concern of this thesis and cover many interesting propagation phenomena ranging in application from railroad rails to the seismic waves induced by earthquakes which, for example, can travel thousands of miles all the while reflecting and refracting through variations in the Earth's surface and its underground discontinuities. Consequently, the subsequent tools that have been developed from extensive

studies of elastic wave motion vary widely in applicability to many academic and industrial problems, including the field of ultrasonics, which involves the introduction of low energy, high-frequency wave packets into a material to determine fundamental properties (such as elastic constants) or to detect particular defects (such as cracks or holes)—all by measuring and analyzing the propagation, reflection and attenuation of these pulses [53, 67, 11, 12, 70, 34]. (Further details will be provided in the context of the specific material science problems considered in Chapter 4.) An excellent pair of historical sketches and surveys of a great many other applications of elastic wave propagation can be found in [3] and the classic text of [39].

1.1 Numerical PDE solvers for elastodynamics

The aforementioned elastic problems and associated wave motion, which are governed by the equations reviewed in Chapter 2, present a host of challenges and difficulties from a computational standpoint: the accurate, stable and dispersionless modeling and computation of numerically stiff three-dimensional physical elastic systems—including the treatment of curved stress-free surfaces, geometric singularities from corners and edges, complex geometries and manifold other intricacies—is certainly considered to be a highly challenging problem in computational science. These complexities have motivated the development of a plethora of numerical models ranging widely in accuracy and stability. Some well-established methods, such as discontinuous Galerkin methods [46, 29, 47, 27] (developed for homogeneous materials where Lamé material parameters $\lambda(x, y, z) = \lambda_0$ and $\mu(x, y, z) = \mu_0$ are constant) and the (pseudo)spectral-element methods [32, 49], have been applied to the equations of elasticity with arbitrary accuracy and with a sufficient treatment of complex geometries through the use of unstructured grids. These methods, however, can have very restrictive spectral radii—the timestep can scale quadratically or cubically with the spatial mesh sizes. Additionally, to retain high-order approximations, the construction of computational domains must be treated with great care in possibly labor intensive

and un-automated manners. Alternative methods, such as extant finite difference or finite element methods, both of which have been invoked to solve problems in elasticity [7, 50, 72, 59] while still carrying CFL conditions that scale linearly, have been well-known for a long time [60] to suffer from high numerical dispersion: errors in the phase of a solution can accumulate over subsequent periods of the propagating wave-train, demanding ever increasing numbers of points per wavelength to resolve, within a given accuracy, increasingly larger problem sizes.

The Fourier-continuation based solvers considered in this work, which form the foundation of the new numerical methodology we introduce in Chapter 3 for elastodynamics problems, address all of these issues by extending Fourier series-based methods to general, non-periodic domains—allowing for a less restrictive CFL condition as well as a computation of derivatives in domain interiors that is spectrally accurate and enables the method to be nearly dispersionless. As will also be demonstrated, this class of solvers can be applied to arbitrary geometries and spatially varying (heterogeneous) media while retaining high-order accuracy. Additionally, the advent and accessibility of computing clusters makes possible a parallel implementation through use of Message Passing Interfaces (MPI) by sharing the work and memory of solving problems in large domains across many processors in a distributed computing environment—the Fourier continuation framework herein lends itself naturally to implementation in such computer hardware.

1.2 The elasticity solver developed in this thesis

This thesis represents a significant extension of the work by [4, 20, 54, 30, 21, 15, 17] which, in the brief history of Fourier-continuation (FC) differentiation algorithms for solution of Partial Differential Equations, introduces for the first time a full-fledged three-dimensional solver that can be applied to arbitrary geometries for a complex system of PDEs. In an effort to realize the potential of the Fourier continuation method promised by prior successful tests for both linear and non-linear problems, this work

details our development and parallel computing implementation for a three-dimensional model of a variable coefficient elastic wave equation including possibly heterogeneous materials (that is, arbitrarily varying material parameters). The challenges of the elastic Cauchy-Navier equation formulation we consider in this work—notably among them the treatment of complex geometries including corners and edges, the enforcement of physical surface boundary conditions by means of a modified FC operator introduced in Section 3.1.2, and a number of stability concerns—are all addressed as part of this work and are summarized within the general overview in what follows:

Chapter 2 reviews the physics and governing mathematics of elasticity theory, the displacement formulation of the elastic wave equation, and relevant physical boundary conditions. Guided wave theory, where wave motion is directed by what is known as a traction-free boundary condition, will also be discussed briefly to provide some context for the material science and seismology applications considered in Chapter 4. A scalar model for the out-of-plane displacement in some of the non-destructive evaluation problems considered is also presented. This work, which is based on use of the high-order integral equation solver described in Appendix B, was published in [53] and is summarized briefly in Section 4.2.

Chapter 3 extends the Fourier continuation methodology to general, arbitrary overlapping curvilinear geometries in order to accommodate the geometric complexities that include engineering detail (e.g. the real experimental setups of defects in 3D plates modeled in Section 4.2.3) while retaining the spatial high-order accuracy and stability of FC, and to enable implementation in a fully parallel computational infrastructure. A description of the accelerated, FC(Gram) method is presented in full detail, and it is extended to introduce, in Section 3.1.2, a modified FC(Gram) operator suitable for Neumann and traction boundary conditions as part of the general treatment prescribed in Section 3.3.3 for traction boundary conditions. In Section 3.2, a variable coefficient, general

coordinate version of the elasticity equations and traction boundary conditions, which contain 189 and 21 spatial derivative terms, respectively, is formulated in such a way as to enable treatment of general geometries by means of overset patches—allowing for explicit interpolation of high-order of accuracy among possibly hundreds of curvilinear sub-patches. These sub-patches are subsequently assigned to computing processors by a simple load balancing algorithm, presented in Section 3.2.3.3, for efficient implementation in distributed-memory parallel environments. Stability issues and treatments of absorbing boundary conditions (to enable approximation of unbounded domains) are additionally addressed in this chapter.

Chapter 4 demonstrates the new solver in a variety of scientific and engineering contexts relevant to problems in materials science and seismology. In Section 4.2, we will present, for the first time, a thorough comparison between experimental scattering patterns of guided ultrasonic quasi-Rayleigh waves in plates with through-thickness defects and fully three-dimensional elastodynamics numerical simulations. Special consideration will be given to the geometry construction of the respective experimental setups and, in particular, corners and edges will be treated by means of a transfinite interpolation (Appendix A) and the overset strategy (Section 3.2.3).

Chapter 2

The mathematical and physical framework

The mathematics of continuum mechanics concerns the description of the mechanical behavior of materials (e.g. solids, fluids, gases) under the action of forces and a fundamental assumption that the respective material is modeled as a continuum. For a solid material, external forces result in an internal restoring force (stress) given by the (symmetric) stress tensor

$$\boldsymbol{\sigma} = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{12} & \sigma_{22} & \sigma_{23} \\ \sigma_{13} & \sigma_{23} & \sigma_{33} \end{pmatrix}. \tag{2.1}$$

The motion of a point in a material sitting at $(x, y, z)^T$ for a certain initial time $t = t_0$ can be described by its displacement to a new position $(\tilde{x}(x, y, z, t_0 + \Delta t), \tilde{y}(x, y, z, t_0 + \Delta t, \tilde{z}(x, y, z, t_0 + \Delta t))^T$ by means of the displacement vector $(u, v, w)^T$:

$$\begin{pmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \end{pmatrix} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} u(x, y, z, t) \\ v(x, y, z, t) \\ w(x, y, z, t) \end{pmatrix}. \tag{2.2}$$

Using the notation $f_x = \partial f/\partial x, f_{xx} = \partial^2 f/\partial x^2$ for derivatives, the gradient tensor $\nabla \mathbf{u}$ of the displacement vector $\mathbf{u} = (u, v, w)^T$ is given by

$$\nabla \mathbf{u} = \begin{pmatrix} u_x & u_y & u_z \\ v_x & v_y & v_z \\ w_x & w_y & w_x \end{pmatrix}. \tag{2.3}$$

The relative deformation induced by stresses given by the tensor (2.1) can be described using the Green-Lagrange strain tensor $\mathbf{E}$, which is related to the displacement gradient (2.3) by the expression

$$\mathbf{E} = \frac{1}{2}[\nabla \mathbf{u} + (\nabla \mathbf{u})^T + \nabla \mathbf{u} \cdot (\nabla \mathbf{u})^T]. \tag{2.4}$$

Materials elastic in nature, that is, that return to their original shape after deformation, are generally governed by a non-linear constitutive law describing the relationship between the stress $\boldsymbol{\sigma}$ given in (2.1) and the strain tensor $\mathbf{E}$ given in (2.4). (Other formulations, such as for hyperelastic materials, derive the stress-strain relationship from an energy density function.)

2.1 Linear elasticity

This thesis is concerned with linear elasticity: small elastic deformations that can be accurately described under linearization of the general non-linear laws that govern elastic bodies. Under such assumptions the third term in the strain tensor (2.4) can be neglected and an approximate deformation can be defined by the infinitesimal strain tensor $\boldsymbol{\varepsilon}$ given by

$$\boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_{11} & \varepsilon_{12} & \varepsilon_{13} \\ \varepsilon_{21} & \varepsilon_{22} & \varepsilon_{23} \\ \varepsilon_{31} & \varepsilon_{32} & \varepsilon_{33} \end{pmatrix} = \begin{pmatrix} u_x & \frac{1}{2}\left(u_y + v_x\right) & \frac{1}{2}\left(u_z + w_x\right) \\ \frac{1}{2}\left(v_x + u_y\right) & v_y & \frac{1}{2}\left(v_z + w_y\right) \\ \frac{1}{2}\left(w_x + u_z\right) & \frac{1}{2}\left(w_y + v_z\right) & w_z \end{pmatrix}. \tag{2.5}$$

Linear elasticity further provides a linear relationship, known as Hooke's law, between the stress $\boldsymbol{\sigma}$ and the infinitesimal strain tensor $\boldsymbol{\varepsilon}$. For a linear isotropic material in the framework of this theory, there is no preferred direction in Hooke's law and the components of $\boldsymbol{\sigma}$ are linearly related to components of $\boldsymbol{\varepsilon}$ through certain material constants λ and μ (known as Lamé parameters); in such cases Hooke's law takes the form

$$\sigma_{ij} = \lambda \delta_{ij} \varepsilon_{kk} + 2\mu \varepsilon_{ij}, \tag{2.6}$$

where δ_{ij} is the Kronecker delta given by a value of $\delta_{ij} = 1$ when $i = j$ and $\delta_{ij} = 0$ when $i \neq j$.

2.1.1 The Cauchy-Navier equations for displacement

For an isotropic linear elastic solid contained in a domain Ω, the momentum-balance equation in Cartesian coordinates, together with (2.6), becomes the system of Partial Differential Equations given by [79]

$$\begin{aligned} \rho u_{tt} &= \Big[(\lambda + 2\mu)u_x + \lambda(v_y + w_z)\Big]_x + \Big[\mu(v_x + u_y)\Big]_y + \Big[\mu(w_x + u_y)\Big]_z + f_1, \\ \rho v_{tt} &= \Big[(\lambda + 2\mu)v_y + \lambda(u_x + w_z)\Big]_y + \Big[\mu(v_x + u_y)\Big]_x + \Big[\mu(v_z + w_y)\Big]_z + f_2, \\ \rho w_{tt} &= \Big[(\lambda + 2\mu)w_z + \lambda(u_x + v_y)\Big]_z + \Big[\mu(w_x + u_z)\Big]_x + \Big[\mu(w_y + v_z)\Big]_y + f_3, \end{aligned} \tag{2.7}$$

where $\mathbf{f} = \mathbf{f}(x, y, z, t)$ is a vector of body forces. The material properties are given by the Lamé parameters $\mu = \mu(x, y, z), \lambda = \lambda(x, y, z)$ and the density $\rho = \rho(x, y, z)$, which are real-valued positive functions of space. For a homogeneous solid, where

$\lambda(x,y,z) = \lambda$ and $\mu(x,y,z) = \mu$ are constant, Equation (2.7) can be written as

$$\rho\ddot{\mathbf{u}} = \mu\nabla^2\mathbf{u} + (\lambda+\mu)\nabla\nabla\cdot\mathbf{u} + \mathbf{f}, \quad (x,y,z) \in \Omega, \tag{2.8}$$

where $\mathbf{u} = (u,v,w)^T$.

The initial displacement and velocities at an initial time $t = t_0$ must be prescribed:

$$\begin{aligned} u(x,y,z,t_0) &= a_1(x,y,z), & u_t(x,y,z,t_0) &= b_1(x,y,z), \\ v(x,y,z,t_0) &= a_2(x,y,z), & v_t(x,y,z,t_0) &= b_2(x,y,z), \\ w(x,y,z,t_0) &= a_3(x,y,z), & w_t(x,y,z,t_0) &= b_3(x,y,z). \end{aligned} \tag{2.9}$$

Boundary conditions must also be given, as discussed in the following section.

2.1.2 Boundary conditions

On the boundary $\partial\Omega$ of a solid geometry, different forms of data can be imposed for the elastic wave equation to model physical configurations. The traction boundary condition, which prescribes the normal stresses on the solid boundary, is given by

$$\begin{pmatrix} \sigma_{11} & \sigma_{21} & \sigma_{31} \\ \sigma_{12} & \sigma_{22} & \sigma_{32} \\ \sigma_{13} & \sigma_{23} & \sigma_{33} \end{pmatrix} \begin{pmatrix} n_1 \\ n_2 \\ n_3 \end{pmatrix} = \begin{pmatrix} d_1(x,y,z,t) \\ d_2(x,y,z,t) \\ d_3(x,y,z,t) \end{pmatrix} \quad \text{on } \partial\Omega, \tag{2.10}$$

where $[n_1, n_2, n_3]^T$ is the inward unit normal on the surface and d_1, d_2, d_3 are functions of the spatial coordinates and time. In terms of displacements, the traction boundary conditions can be made to read

$$\begin{aligned} \Big((2\mu+\lambda)u_x + \lambda(v_y+w_z)\Big)n_1 + \mu\Big(v_x+u_y\Big)n_2 + \mu\Big(w_x+u_z\Big)n_3 &= d_1(x,y,z,t), \\ \mu\Big(v_x+u_y\Big)n_1 + \Big((2\mu+\lambda)v_y + \lambda(u_x+w_z)\Big)n_2 + \mu\Big(w_y+v_z\Big)n_3 &= d_2(x,y,z,t), \\ \mu\Big(w_x+u_z\Big)n_1 + \mu\Big(w_y+v_z\Big)n_2 + \Big((2\mu+\lambda)w_z + \lambda(u_x+v_y)\Big)n_3 &= d_3(x,y,z,t). \end{aligned} \tag{2.11}$$

Many applications—and in particular, those in this work—involve traction-free (or free-surface) boundary conditions. These are characterized by vanishing normal stresses on the surface of the domain, i.e. $d_1(x,y,z,t), d_2(x,y,z,t), d_3(x,y,z,t) \equiv 0$.

A Dirichlet boundary condition can also be imposed as data on the displacements u, v, w at a boundary as

$$u = c_1(x,y,z,t), \quad v = c_2(x,y,z,t), \quad w = c_3(x,y,z,t) \quad \text{on } \partial\Omega. \tag{2.12}$$

It is often necessary to additionally utilize techniques that enable simulation of an actually infinite medium by means of a bounded computational domain. In this thesis this is accomplished by resorting to use of certain sponge regions (absorbing layers) surrounding the actual physical domain. In the sponge layers we employ, both the computational domain is stretched outside the region of interest and a corresponding numerical dissipation term is added to the system of PDEs to dampen the amplitude of the displacements. Details regarding these are provided in Section 3.3.4.

2.2 Longitudinal and transverse waves

A Helmholtz decomposition of the displacement vector $(u,v,w)^T$ into a scalar potential $\phi(x,y,z,t)$ and a vector potential $\boldsymbol{\psi}(x,y,z,t)$ in the form

$$(u,v,w)^T = \nabla\phi + \nabla \times \boldsymbol{\psi} \tag{2.13}$$

yields from Equation (2.8), for constant material parameters $\lambda(x,y,z) = \lambda_0$ and $\mu(x,y,z) = \mu_0$, the expression given by

$$\rho\frac{\partial^2}{\partial t^2}[\nabla\phi + \nabla\times\boldsymbol{\psi}] = [\mu\nabla^2[\nabla\phi + \nabla\times\boldsymbol{\psi}] + (\lambda+\mu)\nabla\nabla\cdot[\nabla\phi + \nabla\times\boldsymbol{\psi}] + \mathbf{f}. \tag{2.14}$$

Noting the fact that $\nabla \cdot \nabla\phi = \nabla^2\phi$ and $\nabla \cdot \nabla \times \boldsymbol{\psi} = 0$ and in absence of the body forces, a rearrangement of this equation yields

$$\nabla[(\lambda + 2\mu)\nabla^2\phi - \rho\phi_{tt}] + \nabla \times [\mu\nabla^2\boldsymbol{\psi} - \rho\boldsymbol{\psi}_{tt}] = 0. \tag{2.15}$$

Clearly, this equation is satisfied provided a corresponding pair of wave equations, namely, the scalar wave equation

$$\phi_{tt} = c_L^2\nabla^2\phi, \tag{2.16}$$

with the *longitudinal* wave speed

$$c_L = \sqrt{\frac{\lambda + 2\mu}{\rho}}, \tag{2.17}$$

and the vector wave equation

$$\boldsymbol{\psi}_{tt} = c_T^2\nabla^2\boldsymbol{\psi}, \tag{2.18}$$

with the *transverse (shear)* wave speed

$$c_T = \sqrt{\frac{\mu}{\rho}} \tag{2.19}$$

are satisfied. The wave speeds (2.17) and (2.19) (which only depend on the material parameters ρ, λ and μ) satisfy the relation

$$\frac{c_L}{c_T} = \sqrt{\frac{2(1-\nu)}{1-2\nu}}, \tag{2.20}$$

where $\nu = \dfrac{\lambda}{2(\lambda+\mu)}$ denotes the material's Poisson's ratio.

The solutions ϕ to the scalar wave equation (2.16) are known as P-waves (or *pressure* waves, or *longitudinal* waves): they involve wave motion of particles in the direction of wave propagation, perpendicular to the wavefront. Solutions $\boldsymbol{\psi}$ to the

vector wave equation (2.18) are commonly known as S-waves (or *shear* waves, or *transverse* waves) whose motion is contained in any direction of a plane normal to the direction of propagation. Motion of S-waves can be horizontally (SH-waves) or vertically (SV-waves) polarized; for example, in the aluminum plates considered in this thesis, an SH-wave travels within the plane of the plate and an SV-wave travels through its thickness. For a more extensive review of the many sorts of waves, we refer to the textbooks [3, 42, 39, 36, 76].

2.3 Guided-wave theory

The bulk waves that travel through the interior of a solid are characteristically non-dispersive, that is, c_L and c_T are functions of ρ, λ and μ, only: they do not depend on frequency. However, mechanical waves can also propagate along an interface between media of different acoustic impedances: these are known as *surface waves*. When one of the two media is a vacuum, surface waves are simply guided by vanishing stress (traction-free) boundary conditions. Unlike bulk waves, these waves are dispersive in nature (wave speed is a function of frequency) and exhibit an infinite number of wave modes.

In seismology, surface waves are commonly categorized as either *Love* waves or *Rayleigh* waves. Love waves, which were predicted mathematically by A. E. H. Love, are SH-waves guided by an elastic layer (an elastic half space bordered by a vacuum) and decay within the depth of the layer. These waves cause the horizontal shifting observed during an earthquake and travel, for typical homogeneous elastic media, at a speed that is around 90% of the transverse wave speed c_T.

Rayleigh waves, which were predicted by Lord Rayleigh in 1885, carry both a longitudinal and transverse motion along the surface of a semi-infinite elastic solid and are slightly slower than Love waves. Ever-present in earthquakes, these waves can also be produced by piezo-electric transduction or by a localized impact, and they are therefore used to detect material defects in a non-destructive manner. When

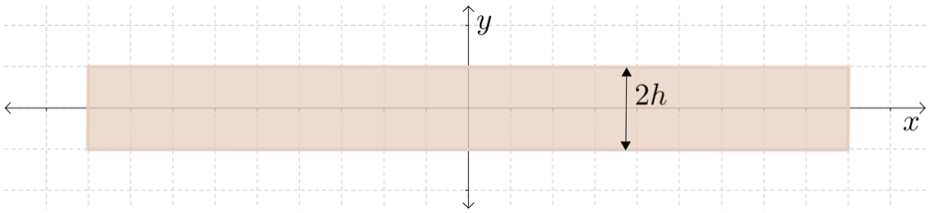


Figure 2.1: *A thin plate.*

neither material on each side of an interface is a vacuum, Rayleigh waves are also called *Stoneley* waves (if the two materials are solids) or *Scholte* waves (if one is a liquid and the other a solid). When guided in layers, Rayleigh waves are called *generalized Rayleigh* waves or *Lamb* waves—they travel through the body and along both traction-free surfaces of a thin plate or shell.

Lamb wave theory [42, 52, 64, 55, 56, 57]—also known as "plate theory"—was pioneered by its namesake Sir Horace Lamb at the turn of the 20th century [52] and is of great practical interest to the non-destructive techniques introduced in the next section and modeled in Chapter 4. It considers a plate, illustrated in Figure 2.1, of infinite extent in the x and z coordinate directions. The plate has a finite thickness of $2h$ in the y coordinate direction and is under conditions of plain-strain: traction-free surfaces at $y = \pm h$. In light of the rectangular geometry, the corresponding boundary condition (2.11) reduces to the system given by

$$\begin{aligned} \sigma_{21} &= 0 = \mu(v_x + u_y), \\ \sigma_{22} &= 0 = (2\mu + \lambda)v_y + \lambda(u_x + w_z), \\ \sigma_{23} &= 0 = \mu(w_y + v_z). \end{aligned} \tag{2.21}$$

Solutions to this problem can be expressed in terms of two sets of modes: 1) symmetric modes, where displacement in the x-direction is symmetric to the midplane of the plate; and, 2) antisymmetric modes, where displacement in the x-direction is anti-symmetric to the midplane of the plate (and vice versa for the y-direction). By considering the

general solutions

$$\begin{aligned} \phi &= \Phi(y)\exp(ikx - \omega t), \\ \boldsymbol{\psi} &= \boldsymbol{\Psi}(y)\exp(ikx - \omega t), \end{aligned} \tag{2.22}$$

of the uncoupled wave equations (2.16) and (2.18), where ω is the angular frequency and k its corresponding wave number, the displacement solutions for both the symmetric and anti-symmetric cases are of the form [2, 39]

$$\begin{aligned} u &= i(B\xi\cos(\alpha y) + C\beta\cos(\beta y))e^{i(x-\omega t)}, \\ v &= (-B\alpha\sin(\alpha y) + C\xi\sin(\beta y))e^{i(x-\omega t)}, \end{aligned} \tag{2.23}$$

and, respectively,

$$\begin{aligned} u &= i(\xi A\sin(\alpha y) - D\beta\sin(\beta y))e^{i(x-\omega t)}, \\ v &= (A\alpha\cos(\alpha y) + D\xi\cos(\beta y))e^{i(x-\omega t)}, \end{aligned} \tag{2.24}$$

where $\alpha^2 = \omega^2/c_L^2 - k^2$ and $\beta^2 = \omega^2/c_T^2 - k^2$. An application of the boundary conditions given by (2.21) yields the well-known dispersion relation between the angular frequency ω and the corresponding wavenumber k given by

$$\frac{\tan(\beta h)}{\tan(\alpha h)} = -\left(\frac{4\alpha\beta k^2}{(k^2-\beta^2)^2}\right)^{\pm 1}, \quad \begin{cases} +1 & \text{symmetric} \\ -1 & \text{antisymmetric} \end{cases}, \tag{2.25}$$

and known as the Rayleigh-Lamb frequency equations for a thin plate. Solutions to the highly non-linear equations of (2.25) were once considered intractable, but computers have enabled calculations of the full frequency spectrum in a reasonable amount of computational time, as we have done in Figure (2.2) via a simple application of Newton's Method [62]. For the non-destructive applications considered in later chapters, the region of excitation for the incident fields induced in the experimental samples of sufficiently thick plates is located at the upper end of the zeroth symmetric and anti-symmetric modes of the spectrum displayed in Figure 2.2; in this region—where the corresponding wave numbers are nearly identical in both modes—the waves

can also be described as *quasi-Rayleigh* waves [67, 78]: the phase and group velocities are close to the phase velocity of the Rayleigh wave and the distribution through the upper and lower halves of the plate mimic that of a Rayleigh wave as well.

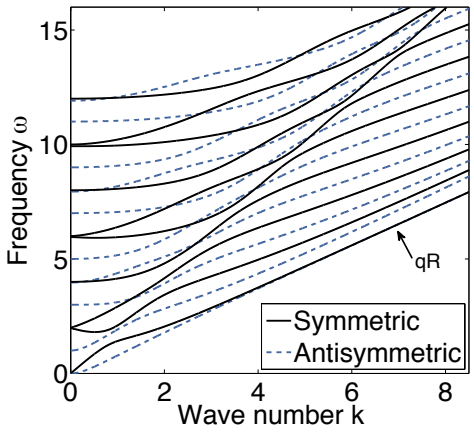


Figure 2.2: *Non-dimensionalized frequency spectrum, computed by Newton's Method, showing the first 10 symmetric (solid lines) and antisymmetric (dashed lines) Lamb modes for a stress-free plate with Poisson's ratio of $\nu = .33$ corresponding to aluminum. This spectrum identifies the corresponding Lamb modes that could be excited by the temporal frequency ω. qR identifies the region of the spectrum corresponding to the quasi-Rayleigh waves that are considered in the numerical applications of Section 4.2.3.*

2.3.1 Scalar model

As discussed above, Lamb modes are described using scalar and vector potentials in (2.22), but an alternative decomposition introduced by [2, 3] can be used to describe the modes with one scalar potential ϕ superposed with carrier waves (one-dimensional functions $V(y)$ and $W(y)$ in the thickness of the plate) that act as "membrane-like waves over the guiding plane" [3]. The decomposition results in displacement solutions

to (2.8) for each mode of the form

$$\begin{aligned} u(x,y,z,t) &= \frac{1}{k}V(y)e^{i\omega t}\frac{\partial}{\partial x}\phi(x,z), \\ v(x,y,z,t) &= W(y)e^{i\omega t}\phi(x,z), \\ w(x,y,z,t) &= \frac{1}{k}V(y)e^{i\omega t}\frac{\partial}{\partial z}\phi(x,z), \end{aligned} \tag{2.26}$$

where $V(y)$ and $W(y)$ are smooth functions in the thickness dimension of the plate that characterize transversal shapes for a given mode. The function $\phi(x,z)$ satisfies the two-dimensional Helmholtz equation

$$\nabla^2\phi + k^2\phi = 0 \tag{2.27}$$

in the plane given by x and z. For a given mode n, then, it follows that at $y = constant$, the out-of-plane displacement $v(x,y,z,t)$ satisfies the equation

$$\nabla^2 v + k^2 v = 0 \tag{2.28}$$

for the wave number k which, for plates, is given by the Rayleigh-Lamb frequency equation (2.25). The decomposition given by (2.26) prompted the work [53], in which a two-dimensional scalar potential was used to approximately model through-thickness defects using the time-harmonic acoustic equation—the geometry depends only on the (x,z) coordinates and in a sense is two-dimensional. Scattering that was produced under conditions where only one propagating mode contributed to v enabled the use of (2.28), supplemented by a Neumann boundary condition, as a first approximation (see Appendix B). The validity of this model was justified in [53] "a posteriori on the basis of comparison of model results with experimental data," but, as demonstrated in Chapter 4, this approximation, while in fairly good agreement, does not fully capture important physical quantities and behavior.

Chapter 3

A new methodology for solving the elastic wave equation

This chapter presents a high-performance numerical methodology for the solution of elastodynamics problems. Based on a certain "Fourier continuation" (FC) approach for accurate Fourier expansion of possibly non-periodic functions [20], the proposed methodology possesses a number of appealing properties: it yields essentially dispersionless results, it gives rise to CFL constraints that scale only linearly with the spatial mesh size h, it runs at a cost that scales linearly with the discretization sizes, and it lends itself easily to parallelization in distributed-memory computing clusters. The new FC algorithm, which applies to a great many problems in elastodynamics, including general geometries and possibly heterogeneous and spatially varying elastic constants, is demonstrated in this thesis via a number of applications to problems concerning materials science and, more specifically, non-destructive evaluation. Our FC treatment of the elastic (Cauchy-Navier) equations is based on the high-order-accurate FC method; other extant FC-based PDE solvers include solvers for the classical wave and diffusion equations [20, 54, 21, 30] as well as the compressible Navier-Stokes equations [4]. The development described in this thesis of FC solvers for elastic wave-propagation problems represents an important extension of the FC methodology: the equations pose a number of significant challenges concerning stability, enforcement of boundary conditions, and geometry treatment, all of which are treated as part of this work.

3.1 The treatment of spatial derivatives

In view of their simplicity, classical differentiation methods such as those based on use of finite differences or finite elements are often used as part of PDE discretization schemes. As is well known, however, Finite Difference (FD) and Finite Element Methods (FEM) for PDEs governing wave-like phenomena do give rise to high numerical dispersion: phase errors accumulate over subsequent periods of a wave field, and thus FD and FEM solvers require increasing numbers of discretization points per wavelength to resolve the solution within a given accuracy tolerance as the size of the problem grows. As is well known, methods based on use of Fourier series do not suffer from this problem: a fixed number of discretization points per wavelength suffices to produce wave-like solutions with a fixed accuracy in domains containing arbitrarily large numbers of waves (up to the wavenumber associated with the finest resolution).

Unfortunately, however, classical Fourier methods are only applicable to periodic functions—since Fourier expansion of a non-periodic functions gives rise to the well-known "ringing effect" (the Gibb's phenomenon [38]): the jump that results in the periodic extension of an inherently non-periodic function leads to an "overshoot" in the Fourier representation at points of discontinuity of the periodic continuation of the given function. This inaccuracy does not subside as more terms are added and, in fact, it results in extremely slow pointwise convergence throughout the interior of the computational domain—in addition to the unmitigated error at domain boundaries.

A goal to extend the applicability of Fourier methods (together with its inherent excellent qualities, most notably dispersionless-ness and high-order accuracy) to general non-periodic configurations has lead to the development of the FC methods that play a central role in this thesis. The Fourier continuation (FC) method produces a rapidly-convergent interpolating Fourier series representation of a given function on a region larger than the given physical domain. This is accomplished by relying on a "periodic extension" of the given function, that closely approximates the given function values in the original domain, but which is periodic (albeit in a slightly enlarged domain)

and thus suitable for Fourier approximation. In detail, given a function f which is defined, without loss of generality, on the unit interval as

$$f(x) : [0,1] \subset \mathbb{R} \to \mathbb{R},$$

the FC method produces a periodic function f^c defined on an extended interval,

$$f^c(x) : [0,b] \subset \mathbb{R} \to \mathbb{R}, \quad b > 1$$

which closely approximates $f(x)$ on the original interval $[0,1]$.

Fully discrete Fourier continuation algorithms generally proceed as follows: letting N be the number of discretization points over the unit interval (yielding a uniform grid $x_i = ih, i = 0, \ldots, N-1,\ h = 1/(N-1)$) together with point values $f(x_i)$ of the function of interest, the Fourier continuation method produces a b-periodic trigonometric polynomial f^c of the form

$$f^c(x) = \sum_{k=-M}^{M} a_k e^{\frac{2\pi i k x}{b}}, \tag{3.1}$$

that matches the given discrete values of f: $f^c(x_i) = f(x_i), i = 0, ..., N-1$. Derivatives of the function can then be easily computed through term-by-term differentiation, e.g.

$$\begin{aligned} f_x(x) = \quad & f_x^c(x) = \quad \sum_{k=-M}^{M} \left(\frac{2\pi i k}{b}\right) a_k e^{\frac{2\pi i k x}{b}}, \\ f_{xx}(x) = \quad & f_{xx}^c(x) = \quad -\sum_{k=-M}^{M} \left(\frac{2\pi k^2}{b^2}\right) a_k e^{\frac{2\pi i k x}{b}}, \\ & \vdots \end{aligned} \tag{3.2}$$

In the simplest treatment [18, 15, 17], the coefficients a_k of (3.1) are found by invoking the Singular Value Decomposition to solve the least squares system

$$\min_{a_k} \sum_{i=0}^{N-1} \left| f^c(x_i) - f(x_i) \right|. \tag{3.3}$$

Although straightforward, repeated applications of the SVD for time-dependent function values on each line of a three-dimensional mesh in a PDE is, in general, unacceptably costly. To circumvent this difficulty, an accelerated method was proposed in [20, 54] which allows for Fourier continuation of functions on the basis of a number of $d = d_\ell, d_r$ end function values at the left and right intervals and a projection onto a Gram polynomial basis whose FC extensions are precomputed via a high-precision SVD. In effect, this procedure produces a "basis" of continuation functions that can be utilized by a PDE solver. The technique was modified in [4] into a "biased-order" algorithm appropriate for high-performance computing including domain decomposition. The latter "FC(Gram)" method is described in Section 3.1.1 and extended in Section 3.1.2 to a form suitable for our elasticity solver.

3.1.1 Accelerated Fourier continuation: FC(Gram)

Given a column vector $\mathbf{f} = (f_0, \ldots, f_{N-1})^T$ containing N values of a given function f in its domain $[0, 1]$, $f_i = f(x_i)$ $(0 = x_0 < x_1 < \cdots < x_{N-1} = 1)$, the accelerated FC(Gram) method [20, 54, 4] uses a subset of the given function values on small numbers d_ℓ and d_r of matching points $\{x_0, .., x_{d_\ell - 1}\}$ and $\{x_{N-d}, ..., x_{N-1}\}$ contained in small subintervals on the left and right ends of the interval $[0, 1]$ (of lengths $\delta_\ell = (d_\ell - 1)h$ and $\delta_r = (d_r - 1)h$) to produce, at first, a discrete periodic extension. Using such data, the FC(Gram) algorithm thus appends a number C of continuation function values for the periodic extension in $[1, b]$ to transition smoothly from f_{N-1} back to f_0; see Figure 3.1. The resulting vector $\mathbf{f}^c$ can be viewed as a discrete set of values of a smooth and periodic function which is suitable for high-order approximation by means of the FFT algorithm in an interval of length $(N + C)\Delta x$.

In order to evaluate the C necessary extension values mentioned above, the accelerated FC(Gram) method uses the given function f defined in the interval $[0, 1]$ together with a translation of it by a distance b. Defining the sets $\mathcal{D}_\ell = \{b + x_0, b + x_1, ..., b + x_{d_\ell - 1}\}$ and $\mathcal{D}_r = \{x_{N-d_\ell}, x_{N-d+1}, ..., x_{N-1}\}$, the additional C needed values in

the interval $[1, b]$ are obtained as point values of an auxiliary trigonometric polynomial of periodicity interval $[1 - \delta_r, 2b - (1 - \delta_r)]$ (with appropriately selected bandwidth) which closely approximate the function values on $\mathcal{D}_r \cup \mathcal{D}_\ell$. This approximating trigonometric polynomial is obtained as the result of a two-step process, namely 1) projection onto bases of orthogonal polynomials, and 2) continuation through use of a precomputed set of continuations-to-zero of each Gram polynomial, as explained in what follows.

The polynomial projection mentioned in step 1) above for the function values on $\mathcal{D}_r \cup \mathcal{D}_\ell$ relies on use of two Gram (orthonormal) bases, $\mathcal{B}_r$ and $\mathcal{B}_\ell$, of the respective spaces of polynomials of degree $< d_\ell, d_r$ on the intervals $[1 - \delta_r, 1]$ and $[b, b + \delta_\ell]$ with respect to the natural discrete scalar product defined by the discretization points $\mathcal{D}_r$ and $\mathcal{D}_\ell$ [20, 54] (see Equation (3.9) below). The algorithm also utilizes precomputed extensions, one for each polynomial $p_r \in \mathcal{B}_r$, into a smooth function defined for $x \geq 1 - \delta_r$, which approximates p_r closely in the matching interval $[1 - \delta_r, 1]$, and which blends smoothly to zero for $x \geq b$. Such rightward extension is obtained by means of appropriately oversampled least squares approximations by Fourier series of periodicity interval $[1 - \delta_r, b - (1 - \delta_r)]$, as described in [20, 54]. Similarly, the scheme obtains, for each polynomial $p_\ell \in \mathcal{B}_\ell$, a smooth blending function that agrees with p_ℓ in the matching interval $[b, b + \delta_\ell]$ and which vanishes for $x \leq 1$.

In presence of such smooth blending, step 2) proceeds: an FC extension is produced, for the given function, from its values at the set of points $\mathcal{D}_r \cup \mathcal{D}_\ell$—since the interpolating polynomials of degree $d_r - 1$ on $\mathcal{D}_r$ and $d_\ell - 1$ on $\mathcal{D}_\ell$ can be expressed as linear combinations of the polynomials in the bases $\mathcal{B}_\ell$ and $\mathcal{B}_r$, with coefficients that can be obtained rapidly by means of scalar products. With the extension in hand, an application of the discrete Fourier transform on the interval $[0, b]$ to the vector of function values $\mathbf{f}$ augmented by the C "continuation" values yields the desired trigonometric polynomial (3.1). An example of the blending procedure is depicted in Figure 3.1.

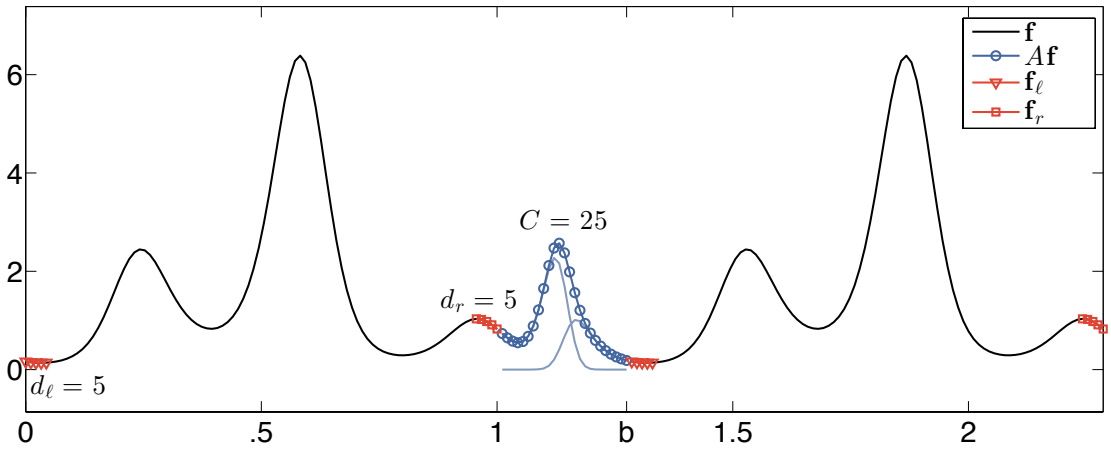


Figure 3.1: *Fourier continuation of the non-periodic function given by* $f(x) = e^{\sin(5.4\pi x - 2.7\pi) - \cos(2\pi x)}$. *Red triangles/squares and blue circles represent* $d_\ell = d_r = 5$ *matching points and* $C = 25$ *continuation points, respectively.*

The resulting continuation operation can be expressed in a block matrix form as

$$\mathbf{f}^c = \begin{bmatrix} I \\ A \end{bmatrix} \mathbf{f} = \begin{bmatrix} \mathbf{f} \\ A\mathbf{f} \end{bmatrix}, \tag{3.4}$$

where $\mathbf{f}^c$ is a vector of the $N + C$ continued function values, I is the $N \times N$ identity matrix and A is the matrix containing the blend-to-zero continuation information. Defining the vector of matching points for the left and right as

$$\mathbf{f}_\ell = \begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_{d_\ell - 1} \end{pmatrix}, \quad \mathbf{f}_r = \begin{pmatrix} f_{N-d_r} \\ f_{N-d_r+1} \\ \vdots \\ f_{N-1} \end{pmatrix}, \tag{3.5}$$

gives an $A\mathbf{f}$ of the form

$$A\mathbf{f} = A_\ell Q_\ell^T \mathbf{f}_\ell + A_r Q_r^T \mathbf{f}_r, \tag{3.6}$$

where the columns of Q_ℓ and Q_r contain the d_ℓ, d_r point values of each element of the

corresponding Gram polynomial basis, and where the columns of A_ℓ and A_r contain the corresponding C values that blend the left and the right bases to zero. For $d_\ell = d_r$ the matrices A_ℓ and A_r (resp. Q_ℓ and Q_r) differ only by row-ordering (resp. only by column ordering). The matrices $Q = Q_\ell$ and $Q = Q_r$ for given numbers $d = d_\ell$ and $d = d_r$ of matching points may be obtained by orthogonalizing the $d \times d$ Vandermonde matrix

$$P = \begin{pmatrix} 1 & x_0 & (x_0)^2 & ... & (x_0)^{d-1} \\ 1 & x_1 & (x_1)^2 & ... & (x_1)^{d-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{d-1} & (x_{d-1})^2 & ... & (x_{d-1})^{d-1} \end{pmatrix}, \tag{3.7}$$

of point values of the monomials x^j $(j = 1, \ldots, d-1)$ at the discrete points $x_0, x_1, ..., x_{d-1}$. (A substitution of x by $(x-1)$ must be used for the right matching problem.) This is performed by applying a Gram-Schmidt orthogonalization process

$$P = QR \tag{3.8}$$

with respect to the inner product

$$(g, h)_r = \sum_{x_i \in \mathcal{D}_r} g(x_i) h(x_i), \tag{3.9}$$

and a similar inner product for $(g, h)_\ell$, where the jth column of Q in (3.8) contains the d point values of the jth Gram basis polynomial. To ensure a good agreement with the constraints described by the least squares problem introduced in what follows, we additionally oversample the monomial basis by a factor of $n_{over} = 20$, which leads to a Vandermonde matrix P^{over} similar to (3.7) but of size $(n_{over}(d-1)+1) \times d$. The corresponding matrix whose columns are the Gram basis polynomials evaluated on a grid n_{over} times finer can be reconstructed by

$$Q^{over} = P^{over} R^{-1}, \tag{3.10}$$

where R is the upper triangular matrix in the orthogonalization of the coarse P in (3.8).

Remark 1. *The ill-conditioning of the full continuation problem is concentrated here at the orthogonalization of the poorly conditioned Vandermonde matrix (3.7); such a decomposition is well known to possibly suffer from a numerical loss of orthogonality. This difficulty is addressed by relying on high-precision (using* 256 *digits) and symbolic arithmetic throughout the process of evaluation of precomputed continuations for the polynomial basis elements. This is an operation that needs to be performed only once and stored in file for use each time the Fourier continuation is invoked. Technical details concerning the implementation of this high-precision precomputation procedure can be found in Appendix D.*

Letting Z be the number of zero points used for the blending and letting E be a certain number of additional points that are used to allow for continuation regions of user-prescribed lengths, we define the intervals

$$\begin{aligned} I_{match} &= [0, (d-1)\Delta x], \\ I_{blend} &= [(d-1)\Delta x, (d+C-1)\Delta x], \\ I_{zero} &= [(d+C-1)\Delta x, (d+C+Z-1)\Delta x], \\ I_{extra} &= [(d+C+Z-1)\Delta x, (d+C+Z+E-1)\Delta x] \end{aligned} \tag{3.11}$$

which contain d, C, Z and E discretization points, respectively. The function that blends the discrete values of a given Gram polynomial on I_{match} to the zero function values in I_{zero} is obtained as a band-limited trigonometric polynomial

$$f(x) = \sum_{k=-M}^{M} a_k e^{\frac{2\pi i k x}{(d+C+Z+E-1)\Delta x}} \tag{3.12}$$

that matches closely (in a least-squares sense) the Gram polynomials and an interval of vanishing function values, and where $M = (d+C+Z+E)/2$. The coefficients

$\mathbf{a} = (a_{-M}, ..., a_M)^T$ are found by solving the minimization problem (posed on the oversampled Gram basis Q^{over} with columns q_k^{over} in (3.10)) given by

$$\min_{\mathbf{a}=(a_{-M},...,a_M)^T} \left\| B^{over}\mathbf{a} - \begin{pmatrix} q_j^{over} \\ \mathbf{0} \end{pmatrix} \right\|_2, \tag{3.13}$$

where B^{over} is a matrix formed from values of the function (3.12) at the $\Delta x/n_{over}$ discretizations of I_{match} and I_{zero}, and where $\mathbf{0}$ is the zero vector of length $(Z-1)n_{over}+1$. The minimizing Fourier coefficients $\mathbf{a} = (a_{-M}, ..., a_M)^T$ in (3.13) are then found by a Singular Value Decomposition and, once determined, the corresponding columns of A_ℓ, A_r are constructed by evaluating (3.12) on the coarse discretization Δx of the full continuation interval $[0, (d + C + Z + E - 1)\Delta x]$.

Remark 2. *For all computations in this thesis, the parameters $C = 25, Z = 12, E = 25$ and $n_{over} = 20$ were used to construct continuations for various integer values of $d = d_\ell, d_r$. These choices were made in accordance with previous FC-based algorithms [4, 30], and were determined empirically to provide an appropriate trade-off between computational cost in a PDE solver and stability of the operator constructed by the above procedure.*

3.1.2 A modified accelerated FC operator for Neumann boundary conditions

In order to treat Neumann boundary conditions (and, ultimately, traction boundary conditions), a modified version of the accelerated (biased-order) algorithm described in Section 3.1.1 is needed to construct an operator appropriate to match the values of the derivative of a function f at the end point x_{d-1} in addition to values of f at $x_0, ..., x_{d-2}$. A simple extension of the algorithm presented above can be made to achieve this goal. Indeed it suffices to use a polynomial interpolant that, in particular, matches the derivative value $f'(x_{d-1})$ at the endpoint x_{d-1}. This is achieved by orthogonalizing

the modified Vandermonde matrix

$$P_{der} = \begin{pmatrix} 1 & x_0 & (x_0)^2 & ... & (x_0)^{d-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{d-2} & (x_{d-1})^2 & ... & (x_{d-2})^{d-1} \\ 0 & 1 & 2x_{d-1} & ... & (d-1)(x_{d-1})^{d-2} \end{pmatrix} \tag{3.14}$$

instead of (3.7) by means of the (high-precision) QR-decomposition

$$P_{der} = Q_{der} R_{der}. \tag{3.15}$$

The modified Fourier continuation blend-to-zero information can be obtained in two ways, both of which in practice have been found to give the same results: 1) by proceeding to form a new continuation basis in a manner analogous to Section 3.1.1, replacing Q with Q_{der} and the appropriate B^{over} with a corresponding B^{over}_{der} in the least squares formulation; or, 2) by reconstructing the coefficients in the original Gram polynomial basis and simply replacing the operator Q $(= Q_\ell, Q_r)$ in (3.6) with a new operator $\tilde{Q}$ $(= \tilde{Q}_\ell, \tilde{Q}_r)$ for which the same pre-constructed blend-to-zero Dirichlet operators A_ℓ, A_r obtained in the previous section can be employed. The latter method—which is used in this work and introduced in what follows—carries the advantage that the Gram polynomials blending-to-zero data computed in the previous section, which was produced in the context of Dirichlet boundary conditions, may be re-used for treatment of Neumann and traction boundary conditions as well.

The construction of $\tilde{Q}$ is obtained by solving the system

$$P_{der}\begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_{d-1} \\ c_{d-1} \end{pmatrix} = \begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_{d-2} \\ f'(x_{d-1}) \end{pmatrix}. \tag{3.16}$$

for the coefficients in (3.16) via the decomposition (3.15) as

$$\begin{pmatrix} c_0 \\ c_2 \\ \vdots \\ c_{d-2} \\ c_{d-1} \end{pmatrix} = R_{der}^{-1} Q_{der}^T \begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_{d-2} \\ f'(x_{d-1}) \end{pmatrix}. \tag{3.17}$$

Recalling the Vandermonde matrix P and its QR-decomposition in (3.8), substitution of these coefficients into

$$P\begin{pmatrix} c_0 \\ c_2 \\ \vdots \\ c_{d-2} \\ c_{d-1} \end{pmatrix} = QR\begin{pmatrix} c_0 \\ c_2 \\ \vdots \\ c_{d-2} \\ c_{d-1} \end{pmatrix} = \begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_{d-2} \\ f_{d-1} \end{pmatrix} \tag{3.18}$$

yields the expressions

$$QRR_{der}^{-1}Q_{der}^{T}\begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_{d-2} \\ f'(x_{d-1}) \end{pmatrix} = \begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_{d-2} \\ f_{d-1} \end{pmatrix} \iff \tilde{Q}^T \begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_{d-2} \\ f'(x_{d-1}) \end{pmatrix} = Q^T \begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_{d-2} \\ f_{d-1} \end{pmatrix} \tag{3.19}$$

for $\tilde{Q} = (RR_{der}^{-1}Q_{der}^{T})^T$. Hence the continuation procedure constructed in the previous section and embodied in the formula

$$A\mathbf{f} = A_\ell Q_\ell^T \mathbf{f}_\ell + A_r Q_r^T \mathbf{f}_r \tag{3.20}$$

is modified by substitution of (3.19) into (3.20) to yield

$$A\mathbf{f} = A_\ell \tilde{Q}_\ell^T \tilde{\mathbf{f}}_\ell + A_r \tilde{Q}_r^T \tilde{\mathbf{f}}_r, \tag{3.21}$$

where $\tilde{\mathbf{f}}_\ell = (f_0, ..., f_{d_\ell - 2}, f'(x_{d_\ell - 1}))^T, \tilde{\mathbf{f}}_r = (f_{N-d_r}, ..., f_{N-2}, f'(x_{N-1}))^T$ and where A_ℓ, A_r are the same blend-to-zero operators. Note that if, for example, one requires the continuation to approximate the value $f(x_0)$ and the derivative $f'(x_{N-1})$ (as is often necessary as a result of the domain decomposition strategy that we introduce later in this thesis), one needs only to form the corresponding biased continuation using the appropriate projections on the left and the right as

$$A\mathbf{f} = A_\ell Q_\ell^T \mathbf{f}_\ell + A_r \tilde{Q}_r^T \tilde{\mathbf{f}}_r, \tag{3.22}$$

and similarly to approximate $f'(x_0)$ and $f(x_{N-1})$.

3.1.3 A simple preliminary 1D example with Dirichlet and Neumann boundaries

As a simple example to verify the character of the FC methodology for both Dirichlet and Neumann problems, consider a 1D wave propagation problem with initial values given by a Gaussian profile:

$$\begin{cases} u_{tt}(x,t) = u_{xx}(x,t), \\ u(x,0) = e^{-300(x+.5)^2}. \end{cases} \tag{3.23}$$

In order to facilitate evaluation of errors in our Dirichlet and Neumann test cases, we set up Dirichlet and Neumann boundary conditions at $x = 0$ and $x = 1$ in such a way that the exact solution is given by the traveling wave function

$$u(x) = e^{-300(x-ct+.5)^2}, \quad 0 \le x \le 1, \tag{3.24}$$

which is depicted in Figure 3.2.

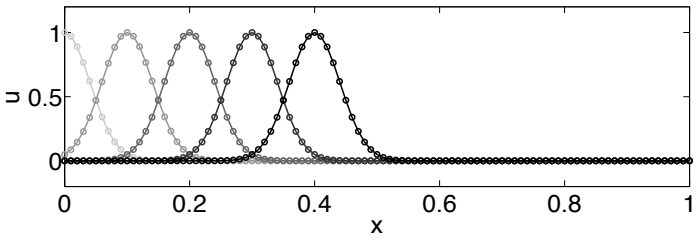


Figure 3.2: *Numerical solution to the one-dimensional wave equation* (3.23) *at times* $t = 0.5, 0.6, 0.7, 0.8, 0.9s$ *based on use of Fourier continuation* ($d_\ell = d_r = 5, C = 25, \Delta x = 1/100$) *and explicit time marching* ($\Delta t = \Delta x/32$).

Using the FC method for evaluation of spatial derivatives, and utilizing the explicit fourth-order Adams-Bashforth method to evolve the resulting set of ODEs, we obtain a PDE solver for the wave equation (3.23). Using a small time-step enables us to demonstrate the accuracy of the FC based spatial discretization method for various

mesh sizes. Table 3.1 displays, for both the Dirichlet and Neumann problems, the maximum error over all spatial points $x \in [0, 1]$ and for all times between 0 and T, where T is taken to be large enough that the wave has passed through both interval endpoints. As expected, a 4th-order Gram polynomial basis (using 5 matching points) results in essentially 5th-order convergence; since the derivatives of the continued function are spectrally accurate, the error should be dominated by the polynomial approximation used to project the end function values onto a Fourier continuation basis.

$1/\Delta x$	L^∞ error (Dirichlet)	$O(L^\infty)$	L^∞ error (Neumann)	$O(L^\infty)$
50	3.60e-2	—	6.15e-2	—
100	1.05e-3	5.10	1.44e-3	5.41
200	3.69e-5	4.83	3.21e-5	5.49
300	4.93e-6	4.96	4.20e-6	5.02
400	1.16e-6	5.03	1.00e-6	4.99

Table 3.1: *Convergence results for the maximum error over all space and time of the solution to the 1D wave equation* (3.23).

Even though the order of the error is the same as that given by a finite-difference solver of the same order, the FC methodology provides a significant advantage: it is essentially dispersionless. As mentioned earlier, wave propagation by means of finite difference schemes leads to an accumulation of errors that compound over the length of the domain. The errors that accumulate over each wave demand a significant increase in the number of points per wavelength (PPW) to resolve the solution for a given accuracy. This behavior is also found in Finite Element Methods and usually discussed in the context of "pollution errors" [8]. Figure 3.3 and Figure 3.4 demonstrate these characteristics for Dirichlet and Neumann boundary conditions using both a second-order-in-space FD scheme and the FC scheme to advance a sinusoidal solution given by

$$u = \sin(2\pi n(x - t)) \tag{3.25}$$

over one period ($t = 0$ to $t = 1$) for varying wavenumbers n and discretized by fixed

numbers $10, 15, 20$ PPW for FC and $75, 150$ PPW for FD. All errors are taken over all time and space, with a time-step again chosen small enough in all cases so that the errors are dominated by the spatial discretization. It is pertinent to note that even though the given solution is periodic, the FC methodology does not exploit this property—since it still extends the domain and creates a new periodic extension. Clearly the PPW required to support the demand of a longer wave-train remains fixed for FC but certainly not for FD, placing a prohibitive requirement on the number of points required to resolve large-scale problems—especially in a full three-dimensional volume—in the latter case. Additionally, Figures 3.3 and 3.4 also display errors over many temporal cycles of a ten-wavelength solution to demonstrate the long-term stability of the FC method as has been described in [20].

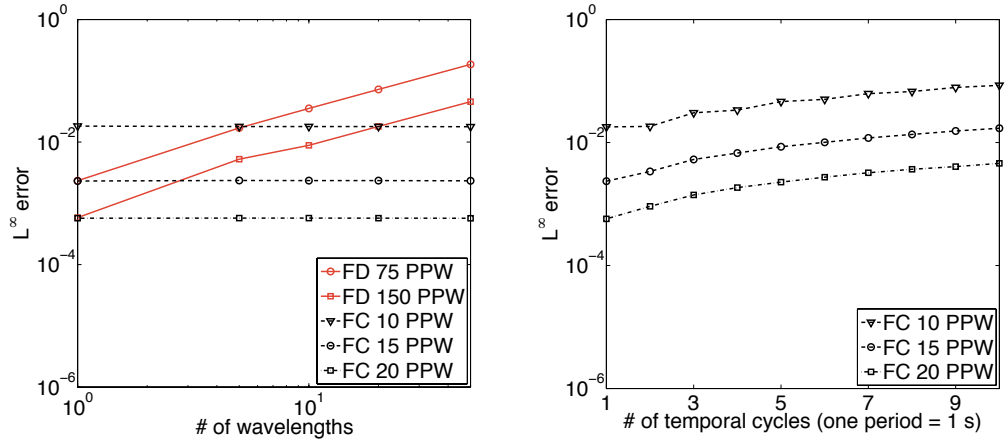


Figure 3.3: *Left: Maximum numerical errors over space and time for solutions of the wave equation Dirichlet problem resulting from use of FC and FD solvers over one full temporal cycle of a sinusoidal solution with increasing number of wavelengths. Right: Maximum numerical errors resulting from use of the FC solver over many temporal cycles for a five-wavelength traveling solution.*

Similar comparisons of FC algorithms and FD methods have been obtained for fourth- and eighth-order finite-differences, as well as the "spectral-like" high-order Padé schemes [4, 30, 54]. In each of these respective studies, the behavior of the FC algorithms showed significant advantages over the previous—either, in view of their favorable dispersion characteristics over those arising from low-order methods, or as

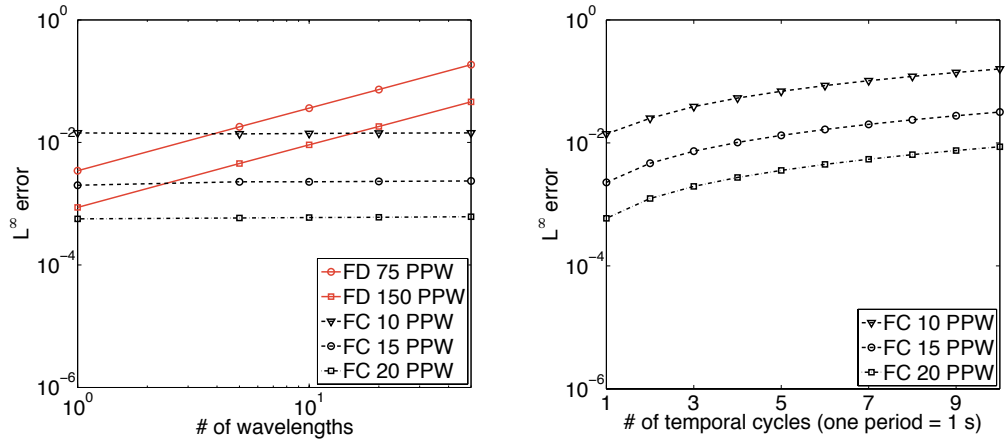


Figure 3.4: *Same as Figure 3.3 but for the Neumann problem instead of the Dirichlet problem.*

a result of the reduced time-steps required compared to other high-order methods, or both. The ability of FC to maintain accuracy over long distances by accurately approximating the dispersion characteristics of the corresponding continuous problem at an $O(N \log N)$ computational cost makes it a competitive approach for solution of general Partial Differential Equations in the time domain.

3.2 Complex geometries

3.2.1 Curvilinear coordinate systems

One of the main advantages of the FC method lies in its use of uniform meshes—for which, as demonstrated in [4], the FC differentiation operator has optimal spectral radius, and for which, therefore, the CFL constraints scale linearly with the size of spatial discretizations (and not quadratically, as do certain other spectral algorithms such as those based on the use of Chebyshev polynomials). Cartesian meshes in physical space cannot capture the geometry of an object of interest with high-order accuracy, however, unless the object has particularly simple shapes: any curvature on an object boundary precludes the use of Cartesian meshes in physical space if high-order accuracy is required. In order to enable applicability to general domains

the FC methods rely on use of local curvilinear grids, which are themselves mapped via a domain mapping

$$\mathcal{M} : [0,1]^3 \to \Omega_i \tag{3.26}$$

from the Cartesian domain $[0,1]^3$ into a portion Ω_i of the physical domain (which are called "patches" in what follows). In explicit coordinates the mapping $\mathcal{M}$ is given by three scalar functions (x, y, z) of three independent variables (q, r, s):

$$\mathcal{M}(q,r,s) = (x(q,r,s), y(q,r,s), z(q,r,s)) : [0,1]^3 \to \Omega. \tag{3.27}$$

A two dimensional example of such a mapping is provided in Figure 3.5.

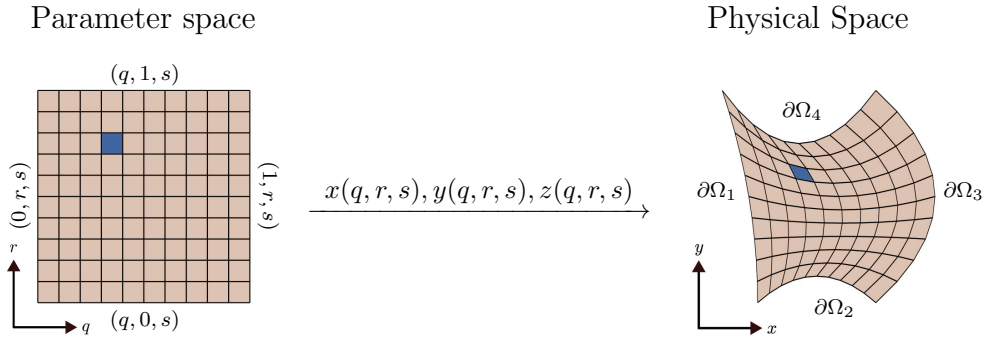


Figure 3.5: *An example of a curvilinear mapping.*

These boundary-conforming patches can be constructed either analytically (for simple geometries), algebraically (on the basis of transfinite interpolation [31]), on the basis of PDE solves (as is the case in elliptic mesh generation [33]), or via combinations thereof [73]. Such patches are endowed with Cartesian-like discretizations: the coordinate lines in a given patch are images of Cartesian coordinate lines in parameter space. Each map is locally invertible with a positive, non-singular Jacobian of transformation that preserves the mathematical type of the PDE [48]. The full domain of interest can then be viewed as a union $\Omega = \cup\Omega_i$ of separate but overset patches Ω_i. (In order to maintain consistency—continuity and differentiability—of the solution

across patches, an interpolation strategy is used, as detailed in Section 3.2.3.)

3.2.2 The governing equations in curvilinear coordinates

Adopting again the shorthand $u_x = \partial u/\partial x$ for derivatives, the chain rule gives on each patch the expressions

$$\begin{aligned} u_q &= x_q u_x + y_q u_y + z_q u_z, \qquad & u_x &= q_x u_q + r_x u_r + s_x u_s, \\ u_r &= x_r u_x + y_r u_y + z_r u_z, \qquad & u_y &= q_y u_q + r_y u_r + s_y u_s, \\ u_s &= x_s u_x + y_s u_y + z_s u_z, \qquad & u_z &= q_z u_q + r_z u_r + s_z u_s. \end{aligned} \tag{3.28}$$

To express the elasticity equations in a curvilinear patch, one seeks expressions for u_x, u_y, u_z (and similarly for v and w) in terms of the derivatives in parameter space $u_q, u_r, u_s, \ldots$ via the chain rules (3.28). The functions $q_x, q_y, q_z, ...$ (also known as *metrics*) can be found provided the corresponding Jacobian

$$J = \begin{vmatrix} x_q & x_r & x_s \\ y_q & y_r & y_s \\ z_q & z_r & z_s \end{vmatrix} = \begin{array}{l} x_q y_r z_s + y_q z_r x_s + z_q x_r y_s \\ \quad - x_q z_r y_s - y_q x_r z_s - z_q y_r x_s \end{array} \tag{3.29}$$

does not vanish. In this case, taking into account the right set of equations in (3.28), we obtain

$$\begin{aligned} \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix} &= \frac{1}{J} \begin{pmatrix} -y_s z_r + y_r z_s & y_s z_q - y_q z_s & -y_r z_q + y_q z_r \\ x_s z_r - x_r z_s & -x_s z_q + x_q z_s & x_r z_q - x_q z_r \\ -x_s y_r + x_r y_s & x_s y_q - x_q y_s & -x_r y_q + x_q y_r \end{pmatrix} \begin{pmatrix} u_q \\ u_r \\ u_s \end{pmatrix} \\ &= \begin{pmatrix} q_x & r_x & s_x \\ q_y & r_y & s_y \\ q_z & r_z & s_z \end{pmatrix} \begin{pmatrix} u_q \\ u_r \\ u_s \end{pmatrix}. \end{aligned} \tag{3.30}$$

The derivatives $x_q, x_r, x_s, ...$ and the corresponding metrics can be computed from the mapping (3.27) either numerically (e.g. via an application of the Fourier continuation method) or analytically.

The curvilinear form of the Cauchy-Navier equations follows from (3.30): denoting by

$$\begin{aligned}
\widetilde{u}(q,r,s) &= u\left(x(q,r,s), y(q,r,s), z(q,r,s)\right), \\
\widetilde{v}(q,r,s) &= v\left(x(q,r,s), y(q,r,s), z(q,r,s)\right), \\
\widetilde{w}(q,r,s) &= w\left(x(q,r,s), y(q,r,s), z(q,r,s)\right),
\end{aligned} \tag{3.31}$$

the displacements as functions of the parameters (q,r,s), and defining

$$\begin{aligned}
\widetilde{D}_x[\widetilde{u}] &= q_x\widetilde{u}_q + r_x\widetilde{u}_r + s_x\widetilde{u}_s, \\
\widetilde{D}_y[\widetilde{u}] &= q_y\widetilde{u}_q + r_y\widetilde{u}_r + s_y\widetilde{u}_s, \\
\widetilde{D}_z[\widetilde{u}] &= q_z\widetilde{u}_q + r_z\widetilde{u}_r + s_z\widetilde{u}_s,
\end{aligned} \tag{3.32}$$

the elastic wave equation (2.7) becomes [48]

$$\begin{aligned}
J\rho\widetilde{u}_{tt} = &\left[Jq_x\Big((\lambda+2\mu)\widetilde{D}_x[\widetilde{u}] + \lambda\widetilde{D}_y[\widetilde{v}] + \lambda\widetilde{D}_z[\widetilde{w}]\Big) + Jq_y\Big(\mu\widetilde{D}_x[\widetilde{v}] + \mu\widetilde{D}_y[\widetilde{u}]\Big) + Jq_z\Big(\mu\widetilde{D}_x[\widetilde{w}] + \mu\widetilde{D}_z[\widetilde{u}]\Big)\right]_q \\
&+ \left[Jr_x\Big((\lambda+2\mu)\widetilde{D}_x[\widetilde{u}] + \lambda\widetilde{D}_y[\widetilde{v}] + \lambda\widetilde{D}_z[\widetilde{w}]\Big) + Jr_y\Big(\mu\widetilde{D}_x[\widetilde{v}] + \mu\widetilde{D}_y[\widetilde{u}]\Big) + Jr_z\Big(\mu\widetilde{D}_x[\widetilde{w}] + \mu\widetilde{D}_z[\widetilde{u}]\Big)\right]_r \\
&+ \left[Js_x\Big((\lambda+2\mu)\widetilde{D}_x[\widetilde{u}] + \lambda\widetilde{D}_y[\widetilde{v}] + \lambda\widetilde{D}_z[\widetilde{w}]\Big) + Js_y\Big(\mu\widetilde{D}_x[\widetilde{v}] + \mu\widetilde{D}_y[\widetilde{u}]\Big) + Js_z\Big(\mu\widetilde{D}_x[\widetilde{w}] + \mu\widetilde{D}_z[\widetilde{u}]\Big)\right]_s \\
&+ f_1(x(q,r,s), y(q,r,s), z(q,r,s), t), \\
J\rho\widetilde{v}_{tt} = &\left[Jq_y\Big((\lambda+2\mu)\widetilde{D}_y[\widetilde{v}] + \lambda\widetilde{D}_x[\widetilde{u}] + \lambda\widetilde{D}_z[\widetilde{w}]\Big) + Jq_x\Big(\mu\widetilde{D}_x[\widetilde{v}] + \mu\widetilde{D}_y[\widetilde{u}]\Big) + Jq_z\Big(\mu\widetilde{D}_z[\widetilde{v}] + \mu\widetilde{D}_y[\widetilde{w}]\Big)\right]_q \\
&+ \left[Jr_y\Big((\lambda+2\mu)\widetilde{D}_y[\widetilde{v}] + \lambda\widetilde{D}_x[\widetilde{u}] + \lambda\widetilde{D}_z[\widetilde{w}]\Big) + Jr_x\Big(\mu\widetilde{D}_x[\widetilde{v}] + \mu\widetilde{D}_y[\widetilde{u}]\Big) + Jr_z\Big(\mu\widetilde{D}_z[\widetilde{v}] + \mu\widetilde{D}_y[\widetilde{w}]\Big)\right]_r \\
&+ \left[Js_y\Big((\lambda+2\mu)\widetilde{D}_y[\widetilde{v}] + \lambda\widetilde{D}_x[\widetilde{u}] + \lambda\widetilde{D}_z[\widetilde{w}]\Big) + Js_x\Big(\mu\widetilde{D}_x[\widetilde{v}] + \mu\widetilde{D}_y[\widetilde{u}]\Big) + Js_z\Big(\mu\widetilde{D}_z[\widetilde{v}] + \mu\widetilde{D}_y[\widetilde{w}]\Big)\right]_s \\
&+ f_2(x(q,r,s), y(q,r,s), z(q,r,s), t), \\
J\rho\widetilde{w}_{tt} = &\left[Jq_z\Big((\lambda+2\mu)\widetilde{D}_z[\widetilde{w}] + \lambda\widetilde{D}_x[\widetilde{u}] + \lambda\widetilde{D}_y[\widetilde{v}]\Big) + Jq_x\Big(\mu\widetilde{D}_x[\widetilde{w}] + \mu\widetilde{D}_z[\widetilde{u}]\Big) + Jq_y\Big(\mu\widetilde{D}_y[\widetilde{w}] + \mu\widetilde{D}_z[\widetilde{v}]\Big)\right]_q \\
&+ \left[Jr_z\Big((\lambda+2\mu)\widetilde{D}_z[\widetilde{w}] + \lambda\widetilde{D}_x[\widetilde{u}] + \lambda\widetilde{D}_y[\widetilde{v}]\Big) + Jr_x\Big(\mu\widetilde{D}_x[\widetilde{w}] + \mu\widetilde{D}_z[\widetilde{u}]\Big) + Jr_y\Big(\mu\widetilde{D}_y[\widetilde{w}] + \mu\widetilde{D}_z[\widetilde{v}]\Big)\right]_r
\end{aligned}$$

$$+\left[Js_z\Big((\lambda+2\mu)\widetilde{D}_z[\widetilde{w}]+\lambda\widetilde{D}_x[\widetilde{u}]+\lambda\widetilde{D}_y[\widetilde{v}]\Big)+Js_x\Big(\mu\widetilde{D}_x[\widetilde{w}]+\mu\widetilde{D}_z[\widetilde{u}]\Big)+Js_y\Big(\mu\widetilde{D}_y[\widetilde{w}]+\mu\widetilde{D}_z[\widetilde{v}]\Big)\right]_s$$
$$+f_3(x(q,r,s),y(q,r,s),z(q,r,s),t), \tag{3.33}$$

with the initial data

$$\begin{aligned}
\widetilde{u}(q,r,s,t_0)&=\widetilde{a}_1(q,r,s), \qquad & \widetilde{u}_t(q,r,s,t_0)&=\widetilde{b}_1(q,r,s),\\
\widetilde{v}(q,r,s,t_0)&=\widetilde{a}_2(q,r,s), \qquad & \widetilde{v}_t(q,r,s,t_0)&=\widetilde{b}_2(q,r,s),\\
\widetilde{w}(q,r,s,t_0)&=\widetilde{a}_3(q,r,s), \qquad & \widetilde{w}_t(q,r,s,t_0)&=\widetilde{b}_3(q,r,s).
\end{aligned} \tag{3.34}$$

The q,r,s derivatives in this formulation can be discretized on the basis of the Fourier continuation method (see Section 3.2.2.1).

The boundary conditions of the problem can be applied easily in the (q,r,s) coordinates. For instance, on the plane corresponding to $q=0$ (and similarly for the other faces), the Dirichlet boundary condition (2.12) becomes

$$\begin{aligned}
\widetilde{u}(0,r,s,t)&=\widetilde{c}_1(r,s,t),\\
\widetilde{v}(0,r,s,t)&=\widetilde{c}_2(r,s,t),\\
\widetilde{w}(0,r,s,t)&=\widetilde{c}_3(r,s,t),
\end{aligned} \tag{3.35}$$

where $\widetilde{c}_j(r,s,t)=c_j(x(q,r,s),y(q,r,s),z(q,r,s),t)$. Similarly, the traction boundary

conditions (2.11) are expressed in the form

$$\begin{aligned}
\widetilde{d}_1(0,r,s,t) &= J\left((2\mu+\lambda)\widetilde{D}_x[\widetilde{u}] + \lambda\widetilde{D}_y[\widetilde{v}] + \lambda\widetilde{D}_z[\widetilde{w}]\right) n_1 \\
&\quad + J\left(\mu\widetilde{D}_x[\widetilde{v}] + \mu\widetilde{D}_y[\widetilde{u}]\right) n_2 \\
&\quad + J\left(\mu\widetilde{D}_x[\widetilde{w}] + \mu\widetilde{D}_z[\widetilde{u}]\right) n_3, \\
\widetilde{d}_2(0,r,s,t) &= J\left(\mu\widetilde{D}_x[\widetilde{v}] + \mu\widetilde{D}_y[\widetilde{u}]\right) n_1 \\
&\quad + J\left((2\mu+\lambda)\widetilde{D}_y[\widetilde{v}] + \lambda\widetilde{D}_x[\widetilde{u}] + \lambda\widetilde{D}_z[\widetilde{w}]\right) n_2 \\
&\quad + J\left(\mu\widetilde{D}_y[\widetilde{w}] + \mu\widetilde{D}_z[\widetilde{v}]\right) n_3, \\
\widetilde{d}_3(0,r,s,t) &= J\left(\mu\widetilde{D}_x[\widetilde{w}] + \mu\widetilde{D}_z[\widetilde{u}]\right) n_1 \\
&\quad + J\left(\mu\widetilde{D}_y[\widetilde{w}] + \mu\widetilde{D}_z[\widetilde{v}]\right) n_2 \\
&\quad + J\left((2\mu+\lambda)\widetilde{D}_z[\widetilde{w}] + \lambda\widetilde{D}_x[\widetilde{u}] + \lambda\widetilde{D}_y[\widetilde{v}]\right) n_3.
\end{aligned} \tag{3.36}$$

The (unit) normal vector used in (3.36) is given by the expression

$$(n_1,n_2,n_3)^T = \frac{(q_x,q_y,q_z)^T}{\sqrt{q_x^2+q_y^2+q_z^2}} \tag{3.37}$$

evaluated at $(0,r,s)$ for $(s,r)\in[0,1]^2$. The numerical treatment and application of boundary conditions in the form of (3.36) is detailed in Section 3.3.3.

3.2.2.1 The discrete curvilinear formulation

A discretization of each patch is obtained, simply, by using a Cartesian parameter-space mesh

$$\begin{aligned}
q_i &= i\Delta q, \quad 0 \le i \le N_q - 1, \\
r_j &= j\Delta r, \quad 0 \le j \le N_r - 1, \\
s_k &= k\Delta s, \quad 0 \le k \le N_s - 1,
\end{aligned} \tag{3.38}$$

containing $N_q \cdot N_r \cdot N_s$ discretization points in $[0,1]^3$—so that the resulting uniform parameter-space mesh-sizes equal $\Delta q = 1/(N_q-1), \Delta r = 1/(N_r-1)$ and $\Delta s =$

$1/(N_s-1)$, respectively. The mapping defined by (3.27) carries curvilinear mesh points (q_i, r_j, s_k) to physical space mesh points

$$\begin{aligned} x_{ijk} &= x(q_i, r_j, s_k), \quad 0 \le i \le N_q - 1, \quad 0 \le j \le N_r - 1, \quad 0 \le k \le N_s - 1, \\ y_{ijk} &= y(q_i, r_j, s_k), \quad 0 \le i \le N_q - 1, \quad 0 \le j \le N_r - 1, \quad 0 \le k \le N_s - 1, \\ z_{ijk} &= y(q_i, r_j, s_k), \quad 0 \le i \le N_q - 1, \quad 0 \le j \le N_r - 1, \quad 0 \le k \le N_s - 1, \end{aligned} \tag{3.39}$$

and solutions in curvilinear coordinates $\widetilde{u}_{ijk} = \widetilde{u}(q_i, r_j, s_k), \widetilde{v}_{ijk} = \widetilde{v}(q_i, r_j, s_k), \widetilde{w}_{ijk} = \widetilde{w}(q_i, r_j, s_k)$ to physical space solutions

$$\begin{aligned} u_{ijk} &= u(x(q_i, r_j, s_k), y(q_r, r_j, s_k), z(q_i, r_j, s_k)) \\ &= \widetilde{u}(q_i, r_j, s_k) \\ &= \widetilde{u}_{ijk}, \quad 0 \le i \le N_q - 1, \quad 0 \le j \le N_r - 1, \quad 0 \le k \le N_s - 1, \\ v_{ijk} &= \widetilde{v}_{ijk}, \quad 0 \le i \le N_q - 1, \quad 0 \le j \le N_r - 1, \quad 0 \le k \le N_s - 1, \\ w_{ijk} &= \widetilde{w}_{ijk}, \quad 0 \le i \le N_q - 1, \quad 0 \le j \le N_r - 1, \quad 0 \le k \le N_s - 1. \end{aligned} \tag{3.40}$$

The discrete Fourier continuation detailed in Section 3.1 can be applied to compute spatial first and second derivatives of $\widetilde{u}, \widetilde{v}, \widetilde{w}$ with respect to the curvilinear coordinates q, r and s via a Fast Fourier Transform on the above uniform discretization.

From (3.39) we see that the boundary discretization points at the left, right, bottom, top, backward, and forward faces of the parameter cube (where boundary conditions are enforced) are characterized by the indices

$$\begin{aligned} &i = 0 \quad \text{or} \quad i = N_q - 1, \quad \text{with} \quad 0 \le j \le N_r - 1, \quad 0 \le k \le N_s - 1, \\ &j = 0 \quad \text{or} \quad j = N_r - 1, \quad \text{with} \quad 0 \le i \le N_q - 1, \quad 0 \le k \le N_s - 1, \\ &k = 0 \quad \text{or} \quad k = N_s - 1, \quad \text{with} \quad 0 \le i \le N_q - 1, \quad 0 \le j \le N_r - 1, \end{aligned} \tag{3.41}$$

respectively. As intended, in view of the boundary-conforming curvilinear transformation, boundary conditions are applied—with high-order accuracy and for an arbitrarily complex surface—at either the first or last index in each dimension in the

corresponding data structure.

3.2.3 A block decomposed, overlapping grid strategy

For a full physical domain, the final discretization strategy adopted by our elasticity solver (see Figure 3.6) consists of two key components [4, 5, 16, 43]: 1) a decomposition of the domain Ω into a collection of overlapping curvilinear patches $\Omega_1, \Omega_2, ..\Omega_n$ (introduced in Section 3.2.1) whose solutions are communicated across overlapping patch boundaries by means of a high-order polynomial interpolation (Section 3.2.3.1); and 2) a further decomposition of each curvilinear patch Ω_i into mutually disjoint "sub-patches" that are then extended by a shared region of subsets of neighboring sub-patches to communicate information within a patch for use in distributed parallel computing environments (Section 3.2.3.2). To facilitate an efficient implementation of element 2), in Section 3.2.3.3 we additionally put forth a simple load-balancing algorithm which, as demonstrated in Section 3.4.3, gives rise to excellent parallel scaling.

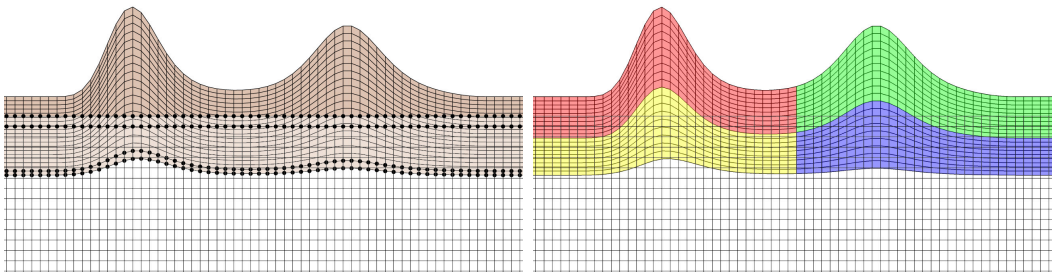


Figure 3.6: *Left: A portion of a physical domain covered by two patches where the bold dots indicate the layers of interpolation points. Right: One such curvilinear patch decomposed into four sub-patches.*

3.2.3.1 Overset meshes and artificial "inter-patch" boundaries

As discussed earlier, a general geometry can be represented by a finite number of overlapping curvilinear patches within each one of which the PDE (3.33) is evolved—

while accounting for smooth matching of solution values between neighboring patches around patch boundaries. In this method, commonly known as the *overset* [16] grid method, a patch $\Omega_i \subset \Omega$ in the PDE domain exchanges information with adjacent patches $\Omega_j \subset \Omega$ by means of a suitable interpolation scheme. Figure 3.6 (left) shows an example of part of a computational domain that is covered by two curvilinear patches, with interpolation points shown as bold dots.

By construction, subsets of a patch boundary $\partial\Omega_i$ that do not coincide with a physical boundary of the computational domain Ω will overlap one or more of the other patches composing a geometry. Along these boundaries, unknown values from neighboring patches are interpolated by means of tensor-product polynomial. In this thesis we employ stencils of size $7 \times 7 \times 7$ points (that is, 6th order polynomial interpolation in each dimension) and interpolate to a layer of width 3 points on the overlapping boundary. The overlapping regions between patches are chosen sufficiently large enough so that interpolation stencils are contained sufficiently far away from the respective donor patch boundary. Patches are constructed to sufficiently overlap each other so that only interpolation from points at a distance from the boundary of the subsequent "donor" patch is performed. That is, given a point $(x_0, y_0, z_0) \in \Omega_i$, an adjacent donor patch Ω_j is found that contains (x_0, y_0, z_0) at its most interior and a corresponding seven-point stencil is located (stored by its lower left-hand corner index) in patch Ω_j to have the boundary point of patch Ω_i as close to its center as possible. Interpolation is then performed in the corresponding (q, r, s) parameter space of the patch Ω_j via a straightforward application of Neville's algorithm [62]. Further details in these regards are provided in Appendix D.

3.2.3.2 Parallel decomposition and artificial "intra-patch" boundaries

A parallel implementation of the FC solver described in this thesis can be produced by means of a decomposition of each curvilinear patch into a union of disjoint sub-patches within each one of which the PDE (3.33) is evolved. At each time-step, solution values

in the sub-patches are used in conjunction with a certain number of fringe discretization points of neighboring sub-patches to produce the spatial derivatives necessary during time marching. This concept is illustrated by a simple one-dimensional example in Figure 3.7: in this simplified example the center sub-domain (sub-patch) has (only) two shared points (which, following [4] we call "fringe points") on either side; the left and right boundary sub-domains contain fringe points on one side only. In our actual algorithm implementation a total of four fringe points in each dimension are used for each interior boundary.

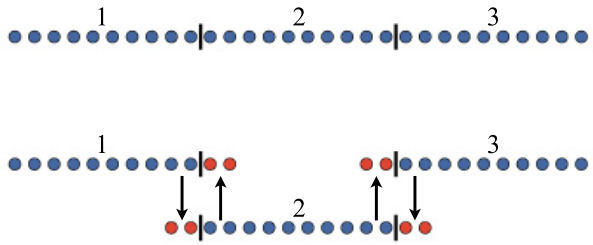


Figure 3.7: *One dimensional line segmentation analogous to the multidimensional parallel decomposition of a patch.*

Remark 3. *To maintain the excellent dispersion characteristics of the FC differentiation scheme, the FC algorithm is set up to use* $d_\ell, d_r = 12$ *discretization points in the Gram polynomial projection interval at interior boundaries; all other (physical) boundaries employ* $d_\ell, d_r = 5$ *Gram projection points.*

The multidimensional version of the shared (fringe) region strategy is demonstrated in Figure 3.8: the continuations needed to apply the differentiation algorithm along a vertical or horizontal line in a sub-patch is obtained by applying FC to the combination of discrete values at the light-gray fringe points and the black sub-patch points. As additionally illustrated in the figure, the fringe discretization region associated with a sub-patch may extend not only into the six neighbors with which the sub-patch shares a face in three-dimensions but also into corner-adjacent neighboring sub-patches, for up to twenty-six neighbors. The potential additional communication required for

all neighbors can be circumvented by communicating the fringe information in one coordinate direction at a time (as demonstrated for a two-dimensional case in the right-hand figure), so that information with corner-adjacent sub-patches is indirectly passed along, thus reducing the overall communication load: this allows the full exchange of information to be carried out by only six send-and-receive calls from a sub-patch with its (up to six) face-sharing neighbors.

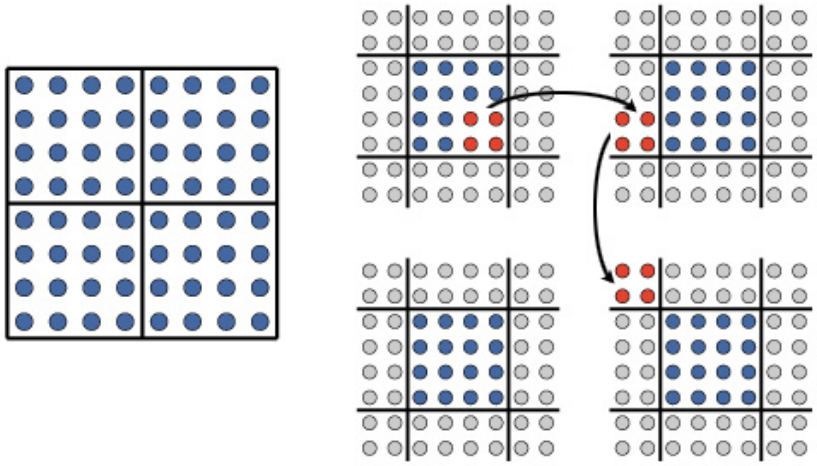


Figure 3.8: *Left: A two-dimensional example of a square patch decomposed into four disjoint sub-patches. Right: The four sub-patches augmented by fringe regions and an example of the transmission of data to corner patches.*

For a parallel implementation in which a processor is responsible for advancing the solution of a single sub-patch, the solution values for the assigned sub-patch are stored together with the values in the fringe discretization regions. The fringe regions are subsequently updated at each time-step to allow for transmission of information across component sub-patches to take place via neighbor-to-neighbor parallel communication. This procedure is akin to the one used in connection with the "artificial" interpolating inter-patch boundaries considered in Section 3.2.3.1, but here no interpolation is necessary—since neighboring sub-patches share the same discretization grid of the underlying patch. For all simulations in this work, we have employed a number of

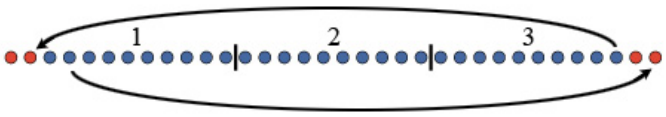


Figure 3.9: *One dimensional line segmentation for periodic boundary conditions.*

four points in the fringe intervals in each dimension.

Remark 4. *Periodic boundary conditions on curvilinear patches such as annuli can be applied by the decomposition described above using fringe points on each end of a periodic interval and a communication strategy analogous to the one described above—as demonstrated in Figure 3.9.*

3.2.3.3 Load balancing

An efficient parallel decomposition strategy should give rise to minimal communication between processors, on one hand, and to a balanced overall workload, on the other. In our approach, one processor is assigned to each sub-patch. In order to determine the number of sub-patches (processors) to be used along each dimension of a given patch the algorithm proceeds in two steps: the number of total processors assigned to a problem are distributed evenly to each of the curvilinear patches; each patch is then subdivided into a number sub-patches that are balanced in an effort to optimize the execution of FFTs.

Given a number of patches $\Omega_1, .., \Omega_M$ that cover the computational domain, and a number p_{total} of processors assigned to the solver, then each patch Ω_i is assigned at least

$$p^i = \left\lfloor \frac{N_q^i N_r^i N_s^i}{\sum_{k=1}^{M} N_q^k N_r^k N_s^k} p_{total} \right\rfloor, \quad i = 1, 2, .., M$$

processors, where N_q^i, N_r^i, N_s^i are the number of discretization points along each side of the computational domain of Ω_i, and where $\lfloor \cdot \rfloor$ is the floor function. The remainder

$p_{total} - \sum_{i=1}^{M} p^i$ remains un-utilized or is redistributed over all sub-domains on the basis of some adequate ad-hoc criterion.

The number of processors now assigned to a curvilinear patch Ω_i can be factored as $p^i = p^i_q p^i_r p^i_s$ for where $p^i_q, p^i_r, p^i_s \geq 1$ denote the number of processors assigned in the q, r, s directions, respectively. The sizes of the blocks used in each dimension should be balanced so as to achieve good parallel scaling; this can be accomplished by minimizing the quantity

$$\min_{p_q p_r p_s = N^i_{procs}} \left| \frac{N^i_q}{p^i_q} - \frac{N^i_r}{p^i_r} \right| + \left| \frac{N^i_r}{p^i_r} - \frac{N^i_s}{p^i_s} \right| + \left| \frac{N^i_q}{p^i_q} - \frac{N^i_s}{p^i_s} \right| ; \tag{3.42}$$

this procedure results in patches that contain approximately the same number of discretization points in each one of the Cartesian directions q, r and s: e.g. $N^i_q/p^i_q \approx N^i_r/p^i_r \approx N^i_s/p^i_s$. In addition to load balancing the parallel code, this strategy results in meshes of approximately equal sizes, which provides the added advantage that FFTs of fixed sizes can be used, as discussed in Appendix D.3—giving rise to additional run-time efficiency.

3.3 Implementation details

This section completes the full description of our FC-based linear elasticity solver for general domains. An explicit treatment of temporal derivatives (a choice made for easy implementation and rapid performance per iteration) will be described in Section 3.3.1 for time marching and, subsequently, a filter will be introduced in Section 3.3.2 to prevent an undesirable growth in time of high-frequency errors that may in some cases result from the FFT approximation in space. Treatment of the traction boundary conditions, which are applied by solving for the relevant out-of-plane directional derivatives in terms of approximate in-plane derivatives, will be discussed in Section 3.3.3 along with a formulation in Section 3.3.4 of "absorbing" boundary

patches used to approximate infinite geometries in an overset strategy for a bounded computational domain.

3.3.1 Explicit treatment of temporal derivatives

A high-order discrete temporal formulation is employed for time integration of the elasticity PDEs; explicit fourth-order Runge-Kutta (RK4) and Adams-Bashforth (AB4) methods are straightforward and provide adequate regions of absolute stability [10, 41]. While the RK4 method requires initialization of only the first step, each subsequent time-step demands four evaluations of the right-hand-side, and the proper enforcement of boundary conditions at intermediate steps may be problematic, particularly for time-dependent boundary conditions [1, 23]. On the other hand, the AB4 method—which is employed everywhere in this thesis—requires only one evaluation of the right-hand-side and a more straightforward application of boundary conditions, but necessitates an initialization of the first three steps. The latter point, however, is moot in most relevant contexts, in which sources ramp up from zero (carrying zero initial displacement), rendering the initial steps identically equal to zero.

Recalling the second-order-in-time elastic wave equation in the form

$$\begin{aligned}
\widetilde{u}_{tt} &= \widetilde{F}_1(\widetilde{u}, \widetilde{u}_q, \widetilde{u}_r, \widetilde{u}_s, \widetilde{v}, \widetilde{v}_q, \widetilde{v}_r, \widetilde{v}_s, \widetilde{w}_q, \widetilde{w}_r, \widetilde{w}_s, t), \\
\widetilde{v}_{tt} &= \widetilde{F}_2(\widetilde{u}, \widetilde{u}_q, \widetilde{u}_r, \widetilde{u}_s, \widetilde{v}, \widetilde{v}_q, \widetilde{v}_r, \widetilde{v}_s, \widetilde{w}_q, \widetilde{w}_r, \widetilde{w}_s, t), \\
\widetilde{w}_{tt} &= \widetilde{F}_3(\widetilde{u}, \widetilde{u}_q, \widetilde{u}_r, \widetilde{u}_s, \widetilde{v}, \widetilde{v}_q, \widetilde{v}_r, \widetilde{v}_s, \widetilde{w}_q, \widetilde{w}_r, \widetilde{w}_s, t),
\end{aligned} \tag{3.43}$$

and the transformation

$$\widetilde{g}_1 = \widetilde{u}_t, \quad \widetilde{g}_2 = \widetilde{v}_t, \quad \widetilde{g}_3 = \widetilde{w}_t,$$

the PDE formulation can be recast to a first-order-in-time system as

$$\begin{aligned}
\widetilde{u}_t &= \widetilde{g}_1, & \widetilde{g}_{1,t} &= \widetilde{F}_1(\widetilde{u}, \widetilde{u}_q, \widetilde{u}_r, \widetilde{u}_s, \widetilde{v}, \widetilde{v}_q, \widetilde{v}_r, \widetilde{v}_s, \widetilde{w}_q, \widetilde{w}_r, \widetilde{w}_s, t), \\
\widetilde{v}_t &= \widetilde{g}_2, & \widetilde{g}_{2,t} &= \widetilde{F}_2(\widetilde{u}, \widetilde{u}_q, \widetilde{u}_r, \widetilde{u}_s, \widetilde{v}, \widetilde{v}_q, \widetilde{v}_r, \widetilde{v}_s, \widetilde{w}_q, \widetilde{w}_r, \widetilde{w}_s, t), \\
\widetilde{w}_t &= \widetilde{g}_3, & \widetilde{g}_{3,t} &= \widetilde{F}_3(\widetilde{u}, \widetilde{u}_q, \widetilde{u}_r, \widetilde{u}_s, \widetilde{v}, \widetilde{v}_q, \widetilde{v}_r, \widetilde{v}_s, \widetilde{w}_q, \widetilde{w}_r, \widetilde{w}_s, t),
\end{aligned} \tag{3.44}$$

for unknowns $\widetilde{u}, \widetilde{v}, \widetilde{w}, \widetilde{g}_1, \widetilde{g}_2, \widetilde{g}_3$. For integration in time using a temporal step size Δt, the fourth-order explicit Adams-Bashforth discretization yields the fully discrete equations as

$$\begin{aligned}
\widetilde{g}_1(t+\Delta t) &= \widetilde{g}_1(t) + \frac{\Delta t}{24}\Big(55\widetilde{F}_1(t) - 59\widetilde{F}_1(t-\Delta t) + 37\widetilde{F}_1(t-2\Delta t) - 9\widetilde{F}_1(t-3\Delta t)\Big), \\
\widetilde{g}_2(t+\Delta t) &= \widetilde{g}_2(t) + \frac{\Delta t}{24}\Big(55\widetilde{F}_2(t) - 59\widetilde{F}_2(t-\Delta t) + 37\widetilde{F}_2(t-2\Delta t) - 9\widetilde{F}_2(t-3\Delta t)\Big), \\
\widetilde{g}_3(t+\Delta t) &= \widetilde{g}_3(t) + \frac{\Delta t}{24}\Big(55\widetilde{F}_3(t) - 59\widetilde{F}_3(t-\Delta t) + 37\widetilde{F}_3(t-2\Delta t) - 9\widetilde{F}_3(t-3\Delta t)\Big), \\
\widetilde{u}(t+\Delta t) &= \widetilde{u}(t) + \frac{\Delta t}{24}\Big(55\widetilde{g}_1(t) - 59\widetilde{g}_1(t-\Delta t) + 37\widetilde{g}_1(t-2\Delta t) - 9\widetilde{g}_1(t-3\Delta t)\Big), \\
\widetilde{v}(t+\Delta t) &= \widetilde{v}(t) + \frac{\Delta t}{24}\Big(55\widetilde{g}_2(t) - 59\widetilde{g}_2(t-\Delta t) + 37\widetilde{g}_2(t-2\Delta t) - 9\widetilde{g}_2(t-3\Delta t)\Big), \\
\widetilde{w}(t+\Delta t) &= \widetilde{w}(t) + \frac{\Delta t}{24}\Big(55\widetilde{g}_3(t) - 59\widetilde{g}_3(t-\Delta t) + 37\widetilde{g}_3(t-2\Delta t) - 9\widetilde{g}_3(t-3\Delta t)\Big).
\end{aligned} \tag{3.45}$$

Dirichlet conditions are imposed by enforcing their values on the given boundary at the end of each time-step of (3.45), whereas time-dependent traction boundary conditions are applied at the beginning of each time-step by the evaluation of the FC-Neumann operator used in the right hand sides $\widetilde{F}_1, \widetilde{F}_2, \widetilde{F}_3$. That is, one time-step of the solver will first compute the requisite traction boundary derivatives at t_n; then update the solution via (3.45) to t_{n+1}; and finally prescribe the Dirichlet boundaries at t_{n+1}. A summary of the full algorithm is provided in Section 3.3.5.

3.3.2 Frequency space filters

In an explicit time formulation, numerical errors may give rise to instability. Since the FC derivatives are evaluated in the frequency domain, Fourier domain filters can be efficiently applied to effectively eliminate undesirable high-frequency error. The spectral filter used in the differentiation procedure employed in this thesis is given by

$$\sigma\left(\frac{2k}{N}\right) = \exp\left(-\alpha\left(\frac{2k}{N}\right)^{2p}\right), \tag{3.46}$$

and applied to a function $u(x)$ with Fourier coefficients $\hat{u}_n$ via the operation

$$\sum_{k=-\frac{N}{2}}^{\frac{N}{2}} \hat{u}_k \exp(ikx) \quad \longrightarrow \quad \sum_{k=-\frac{N}{2}}^{\frac{N}{2}} \sigma\left(2k/N\right) \hat{u}_k \exp(ikx). \tag{3.47}$$

The positive integer p controls the rate of decay of the filter coefficients, and the real parameter α determines the level of suppression: the highest-frequency modes are multiplied by $e^{-\alpha}$.

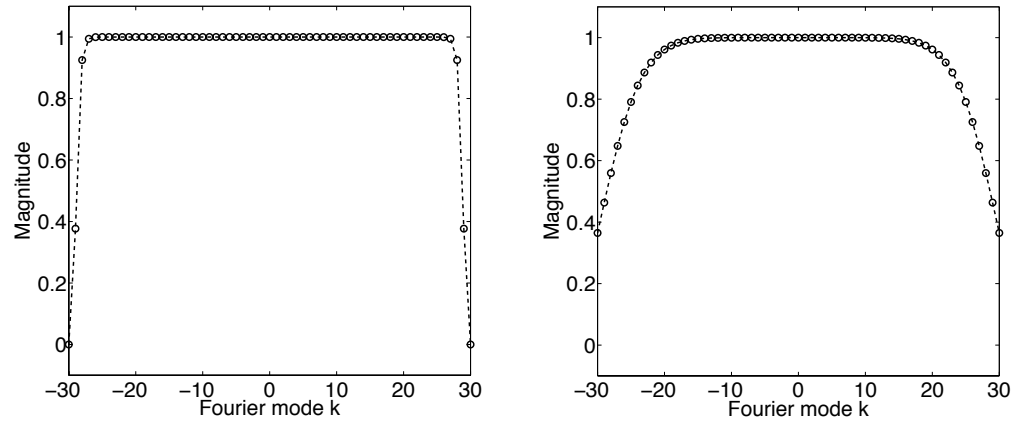


Figure 3.10: *Depictions of the exponential filter with parameter values given by* $(p, \alpha) = (3N/5, 16\log(10))$ *(left) and* $(p, \alpha) = (4, -c_L\Delta t/h_{min}\ln(10^{-2}))$ *(right).*

The filter with parameters $(p, \alpha) = (3N/5, 16\log(10))$ used in [4] (for the *viscous* Navier-Stokes equations) which is plotted in Figure 3.10, an absence of viscosity

has been found to create a large discontinuity in neighboring sub-patches in our parallel implementation and, as a result, to give rise to instability—as the Fourier series expansions in segments from different overlapping patches may be made overly inconsistent by the filtering procedure, and, thus, give rise to instability. This behavior, which is also cited in [30], is demonstrated in the left portion of Figure 3.11.

A milder choice of parameters (in the sense that the highest frequency terms aren't entirely discarded) is used in this thesis, namely

$$(p, \alpha) = (4, -c_L \Delta t / h_{min} \ln(10^{-2})), \tag{3.48}$$

where $c_L = \sqrt{(\lambda + 2\mu)/\rho}$ is the maximum (longitudinal) wave speed in the material, and h_{min} is the finest spatial step size throughout the computational domain. This particular choice of parameters, inspired by a similar choice in [30], was employed to incorporate the CFL condition in such a way that the filter approaches unity as $\Delta t \to 0$ for a fixed Δx. As seen in the right graph of Figure 3.10, this filter does not eliminate the highest modes completely but more smoothly forces the decay of the Fourier coefficients. This is the choice of filter that is adopted for all subsequent simulations in this work since it retains numerical stability. This is illustrated in the right portion of Figure 3.11: after 5,000 time-steps the left image shows some significant errors at interface boundaries. By 10,000 time-steps, the magnitude of these errors is orders of magnitude higher. With the parameters used to produce the right portion of the figure the errors remain as shown in the figure for 10,000 time-steps and, indeed, as far as we have checked, for arbitrarily long times thereafter—at least up to 500,000 time-steps.

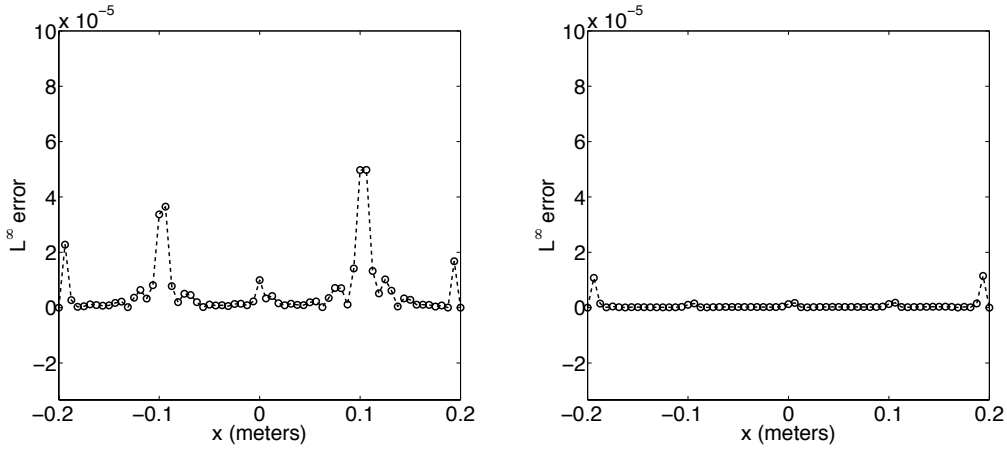


Figure 3.11: *Numerical errors along the x-axis after* 5000 *time-steps for plane waves in a thin plate of dimensions* $70mm \times 10mm \times 25mm$ *decomposed onto four sub-patches. Left: Errors for parameters values* $(p, \alpha) = (3N/5, 16\ln(10))$. *Right: Errors for parameters values* $(p, \alpha) = (4, -c_L \Delta t/h_{min} \ln(10^{-2}))$.

3.3.3 Treatment of traction boundaries in curvilinear coordinates

Numerically, traction boundary conditions can be applied by solving for the relevant coordinate derivatives in terms of tangential derivatives in the surface plane during the computation of the right hand side, and applying the Neumann FC operator introduced in Section 3.1.2. For example, the traction boundary condition on the face corresponding to $(0, r, s)$ and given by (3.36) can be solved for the out-plane-derivatives $\widetilde{u}_q, \widetilde{v}_q$ and $\widetilde{w}_q$, which yields

$$\begin{pmatrix} \widetilde{u}_q(0,r,s) \\ \widetilde{v}_q(0,r,s) \\ \widetilde{w}_q(o,r,s) \end{pmatrix} = \frac{1}{\kappa} \begin{pmatrix} B_{\widetilde{w}_q} C_{\widetilde{v}_q} - B_{\widetilde{v}_q} C_{\widetilde{w}_q} & A_{\widetilde{v}_q} C_{\widetilde{w}_q} - A_{\widetilde{w}_q} C_{\widetilde{v}_q} & A_{\widetilde{w}_q} B_{\widetilde{v}_q} - A_{\widetilde{v}_q} B_{\widetilde{w}_q} \\ B_{\widetilde{u}_q} C_{\widetilde{w}_q} - B_{\widetilde{w}_q} C_{\widetilde{u}_q} & A_{\widetilde{w}_q} C_{\widetilde{u}_q} - A_{\widetilde{u}_q} C_{\widetilde{w}_q} & A_{\widetilde{u}_q} B_{\widetilde{w}_q} - A_{\widetilde{w}_q} B_{\widetilde{u}_q} \\ B_{\widetilde{v}_q} C_{\widetilde{u}_q} - B_{\widetilde{u}_q} C_{\widetilde{v}_q} & A_{\widetilde{u}_q} C_{\widetilde{v}_q} - A_{\widetilde{v}_q} C_{\widetilde{u}_q} & A_{\widetilde{v}_q} B_{\widetilde{u}_q} - A_{\widetilde{u}_q} B_{\widetilde{v}_q} \end{pmatrix} \begin{pmatrix} D_1 \\ D_2 \\ D_3 \end{pmatrix} \tag{3.49}$$

where A, B, C, D and κ families of quantities are given by

$$A_{\widetilde{u}_q} = (2\mu+\lambda) J n_x q_x + \mu J n_y q_y + \mu J n_z q_z, \quad A_{\widetilde{v}_q} = \lambda J n_x q_y + \mu J n_y q_x, \quad A_{\widetilde{w}_q} = \lambda J n_x q_z + \mu J n_z q_x,$$

$$A_{\widetilde{u}_r} = (2\mu+\lambda) J n_x r_x + \mu J n_y r_y + \mu J n_z r_z, \quad A_{\widetilde{v}_r} = \lambda J n_x r_y + \mu J n_y r_x, \quad A_{\widetilde{w}_r} = \lambda J n_x r_z + \mu J n_z r_x,$$

$$A_{\widetilde{u}_s} = (2\mu+\lambda)Jn_xs_x + \mu Jn_ys_y + \mu Jn_zs_z, \quad A_{\widetilde{v}_s} = \lambda Jn_xs_y + \mu Jn_ys_x, \quad A_{\widetilde{w}_s} = \lambda Jn_xs_z + \mu Jn_zs_x,$$

$$B_{\widetilde{u}_q} = \mu Jn_xq_y + \lambda Jn_yq_x, \quad B_{\widetilde{v}_q} = \mu Jn_xq_x + (2\mu+\lambda)Jn_yq_y + \mu Jn_zq_z, \quad B_{\widetilde{w}_q} = \lambda Jn_yq_z + \mu Jn_zq_y,$$
$$B_{\widetilde{u}_r} = \mu Jn_xr_y + \lambda Jn_yr_x, \quad B_{\widetilde{v}_r} = \mu Jn_xr_x + (2\mu+\lambda)Jn_yr_y + \mu Jn_zr_z, \quad B_{\widetilde{w}_r} = \lambda Jn_yr_z + \mu Jn_zr_y,$$
$$B_{\widetilde{u}_s} = \mu Jn_xs_y + \lambda Jn_ys_x, \quad B_{\widetilde{v}_s} = \mu Jn_xs_x + (2\mu+\lambda)Jn_ys_y + \mu Jn_zs_z, \quad B_{\widetilde{w}_s} = \lambda Jn_ys_z + \mu Jn_zs_y,$$

$$C_{\widetilde{u}_q} = \mu Jn_xq_z + \lambda Jn_zq_x, \quad C_{\widetilde{v}_q} = \mu Jn_yq_z + \lambda Jn_zq_y, \quad C_{\widetilde{w}_q} = \mu Jn_xq_x + \mu Jn_yq_y + (2\mu+\lambda)Jn_zq_z,$$
$$C_{\widetilde{u}_r} = \mu Jn_xr_z + \lambda Jn_zr_x, \quad C_{\widetilde{v}_r} = \mu Jn_yr_z + \lambda Jn_zr_y, \quad C_{\widetilde{w}_r} = \mu Jn_xr_x + \mu Jn_yr_y + (2\mu+\lambda)Jn_zr_z,$$
$$C_{\widetilde{u}_s} = \mu Jn_xs_z + \lambda Jn_zs_x, \quad C_{\widetilde{v}_s} = \mu Jn_ys_z + \lambda Jn_zs_y, \quad C_{\widetilde{w}_s} = \mu Jn_xs_x + \mu Jn_ys_y + (2\mu+\lambda)Jn_zs_z,$$

$$D_1 = \widetilde{d}_1(0,r,s,t) - (A_{\widetilde{u}_r}\widetilde{u}_r + A_{\widetilde{v}_r}\widetilde{v}_r + A_{\widetilde{w}_r}\widetilde{w}_r + A_{\widetilde{u}_s}\widetilde{u}_s + A_{\widetilde{v}_s}\widetilde{v}_s + A_{\widetilde{w}_s}\widetilde{w}_s),$$
$$D_2 = \widetilde{d}_2(0,r,s,t) - (B_{\widetilde{u}_r}\widetilde{u}_r + B_{\widetilde{v}_r}\widetilde{v}_r + B_{\widetilde{w}_r}\widetilde{w}_r + B_{\widetilde{u}_s}\widetilde{u}_s + B_{\widetilde{v}_s}\widetilde{v}_s + B_{\widetilde{w}_s}\widetilde{w}_s),$$
$$D_3 = \widetilde{d}_3(0,r,s,t) - (C_{\widetilde{u}_r}\widetilde{u}_r + C_{\widetilde{v}_r}\widetilde{v}_r + C_{\widetilde{w}_r}\widetilde{w}_r + C_{\widetilde{u}_s}\widetilde{u}_s + C_{\widetilde{v}_s}\widetilde{v}_s + C_{\widetilde{w}_s}\widetilde{w}_s),$$

and

$$\kappa = A_{\widetilde{w}_q}B_{\widetilde{v}_q}C_{\widetilde{u}_q} - A_{\widetilde{v}_q}B_{\widetilde{w}_q}C_{\widetilde{u}_q} - A_{\widetilde{w}_q}B_{\widetilde{u}_q}C_{\widetilde{v}_q} + A_{\widetilde{u}_q}B_{\widetilde{w}_q}C_{\widetilde{v}_q} + A_{\widetilde{v}_q}B_{\widetilde{u}_q}C_{\widetilde{w}_q} - A_{\widetilde{u}_q}B_{\widetilde{v}_q}C_{\widetilde{w}_q}. \tag{3.50}$$

The out-of-plane derivatives $\widetilde{u}_q(0,r,s), \widetilde{v}_q(0,r,s)$ and $\widetilde{w}_q(0,r,s)$ can be obtained from (3.36) in terms of the in-plane derivatives $\widetilde{u}_r, \widetilde{u}_s, \widetilde{v}_r, \widetilde{v}_s, \widetilde{w}_r$ and $\widetilde{w}_s$ at t_n. Once they have been determined, every invocation of the FC algorithm for the relevant *first* derivatives in the right hand side of (3.33) at each time-step, i.e. in the computation of $\widetilde{u}_q(q,r,s), \widetilde{v}_q(q,r,s)$ and $\widetilde{w}_q(q,r,s)$ throughout the domain, employs the Neumann

operator of Section 3.1.2 using, for example, the d matching points given by

$$\begin{pmatrix} \widetilde{u}(q_d, r, s) \\ \widetilde{u}(q_{d-1}, r, s) \\ \vdots \\ \widetilde{u}(q_1, r, s) \\ \widetilde{u}_q(0, r, s) \end{pmatrix}, \tag{3.51}$$

where $\widetilde{u}_q(0, r, s)$ is found from above by (3.49). This portion of the algorithm is placed in the context of the overall FC methodology in the algorithm pseudo-code presented in Section 3.49.

Remark 5. *The use of the filtering procedure described in the previous section is restricted in our algorithm to the evaluation of the right hand side of the elastic wave equation* (3.33)*, and it is not used in the computation of the in-plane derivatives in the right hand side of* (3.49)*. The latter filtering procedure, which is not necessary, makes the traction boundary conditions more inconsistent with solution values in the interior volume of a computational domain, and gives rise to larger errors near the traction boundaries than can otherwise be obtained.*

3.3.4 Absorbing boundary conditions

Elastodynamics problems may require a patch that can effectively simulate the semi-infinite or infinite extent of a geometry without giving rise to pollution resulting from scattering from computational boundaries. (The physical experiments presented in Chapter 4 accomplish this in the laboratory by way of acoustic absorbers made of plasticine [53, 67]. This procedure allows one to isolate patterns that arise from various holes or cracks from boundary effects.)

An infinite sponge layer that absorbs outgoing waves and minimizes reflections from a physical boundary has been included as part of our overset algorithm. An

absorbing mesh inspired by the formulation [6] contains two key elements: a stretched component to slow down the advance of outgoing waves (effectively preventing them from reaching the farthest edge of the computational domain); and a damping term (or artificial dissipation) to compensate for the spurious reflections that may result from such a stretch. In a one-dimensional case, for example, one might consider a domain $\Omega = [x_0, x_1]$ parametrized by

$$x(q) = x_0 + (x_1 - x_0)q \quad q \in [0, 1]. \tag{3.52}$$

and discretized by N points. A new mapping can be constructed to preserve the discretization on a subset of the domain $[0, q_1]$ and stretch it on $[q_1, q]$. This is accomplished by first defining a partition of unity $P(q)$ [58] that smoothly ramps up from $P = 0$ to $P = 1$ as shown in Figure 3.12 and given by

$$P(q) = P(q, q_1, \Delta q) = \begin{cases} 0, & q \leq q_1 \\ 1 - \exp\left(\dfrac{2e^{-1/\tau}}{\tau - 1}\right), & q_1 + \Delta q \leq q, \quad \tau = \dfrac{q - q_1}{1 - q_1}. \end{cases} \tag{3.53}$$

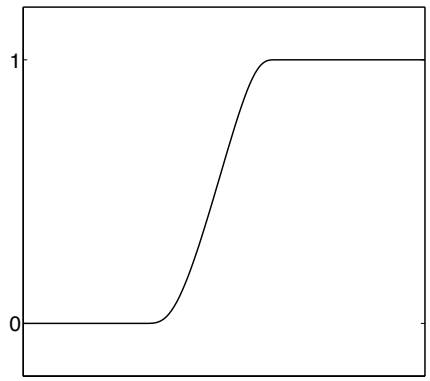


Figure 3.12: *An example of a stretching function $P(q)$.*

The absorbing patch is subsequently produced by the mapping given by

$$X(q) = \begin{cases} x(q), & q \leq q_1 \\ \dfrac{x(q)}{\gamma_0 + (1-\gamma_0)(1-P(q))}, & q_1 \leq q \leq 1, \end{cases} \tag{3.54}$$

where $\gamma_0 \in (0,1)$ is a so-called *stretching constant* and chosen in practice to be smaller than one. (Note that (3.54) can also be used to stretch for variable resolution or better-fitting overset arrangements.) For a fixed γ_0, the amount of physical stretch in our formulation (parameterized by $[q_1, 1]$ and determined by the slope of τ) varies by the selected number of points N in the absorbing layer. An example of a radially absorbing patch placed around a square plate is shown in Figure 3.13.

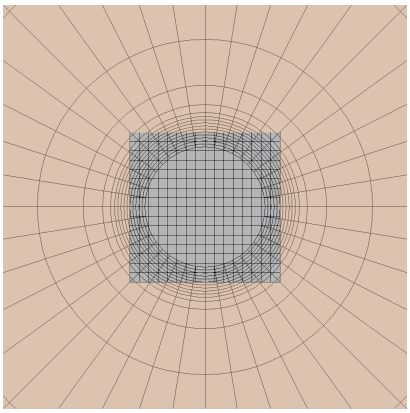


Figure 3.13: *An example of a square mesh overset by a radially absorbing patch;* $\gamma_0 = 0.10$.

While the stretching described above in effect "slows down" outgoing waves, small errors in the form of reflected waves may result from the large metric derivatives and coarsened resolution in the stretched regions [6]. These are easily suppressed by damping the amplitude through the introduction of an artificial damping term of the

form

$$\begin{aligned}
\widetilde{u}_t &= \widetilde{g}_1 - \widetilde{\beta}(q)\widetilde{u}, & \widetilde{g}_{1t} &= \widetilde{F}_1(\widetilde{u}, \widetilde{u}_q, \widetilde{u}_r, \widetilde{u}_s, \widetilde{v}, \widetilde{v}_q, \widetilde{v}_r, \widetilde{v}_s, \widetilde{w}_q, \widetilde{w}_r, \widetilde{w}_s, t), \\
\widetilde{v}_t &= \widetilde{g}_2 - \widetilde{\beta}(q)\widetilde{b}, & \widetilde{g}_{2t} &= \widetilde{F}_2(\widetilde{u}, \widetilde{u}_q, \widetilde{u}_r, \widetilde{u}_s, \widetilde{v}, \widetilde{v}_q, \widetilde{v}_r, \widetilde{v}_s, \widetilde{w}_q, \widetilde{w}_r, \widetilde{w}_s, t), \\
\widetilde{w}_t &= \widetilde{g}_3 - \widetilde{\beta}(q)\widetilde{w}, & \widetilde{g}_{3t} &= \widetilde{F}_3(\widetilde{u}, \widetilde{u}_q, \widetilde{u}_r, \widetilde{u}_s, \widetilde{v}, \widetilde{v}_q, \widetilde{v}_r, \widetilde{v}_s, \widetilde{w}_q, \widetilde{w}_r, \widetilde{w}_s, t),
\end{aligned} \tag{3.55}$$

where the corresponding damping function $\widetilde{\beta}(q)$ with strength β_0 is given by

$$\widetilde{\beta}(q) = \beta_0 - \beta_0(\gamma_0 + (1-\gamma_0)(1-P(q))) \tag{3.56}$$

and scales with (3.54). By the time waves arrive at the edge of a sponge layer, their amplitudes should be damped enough to enable a zero displacement boundary condition ($u = v = w = 0$) at the farthest boundary line. The properties of the sponge layer methodology discussed above are illustrated in Figure 3.14 by means of an experiment concerning a radially absorbing patch placed around an aluminum square plate ($\lambda \approx 1.997 \times 10^7 \mathrm{N/m}^2, \mu \approx 1.024 \times 10^7 \mathrm{N/m}^2$). In practice, we have found that use of the parameters $N = 100, \gamma_0 = 0.075$ and $\beta_0 = 2$ gives rise to a performance similar to the one obtained for this experiment.

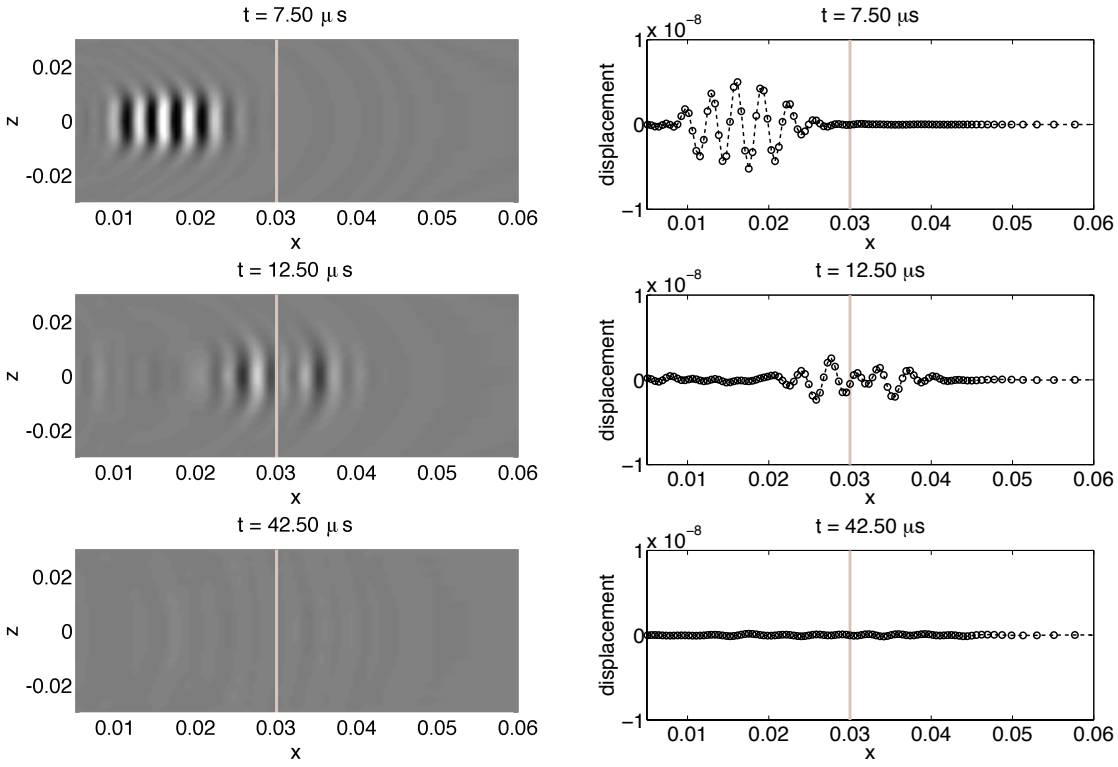


Figure 3.14: *A numerical simulation of the out-of-plane displacement resulting from right-going waves traveling into an absorbing layer of aluminum ($N_{sponge} = 100$, $\gamma_0 = 0.075$, $\beta_0 = 2$) whose boundary is demarcated by the vertical line. The right-hand column of figures represents the corresponding amplitude along the horizontal at $z = 0$.*

3.3.5 Overall algorithm pseudo-code

The previous sections in this chapter have presented the methodologies for the solution of the elastic wave equation in three-dimensional domains. The brief pseudo-code below (Algorithm 1) summarizes our solver.

3.4 Performance studies

Our complete solver for the fully three-dimensional variable coefficient elastic wave equation has been presented in Sections 3.1 through 3.3. In this section we present a variety of numerical experiments that demonstrate the qualities of the algorithm,

1: **for** Each time-step n **do**
2: **for** All patches Ω_i **do**
3: **for** All sub-patches $\omega_j \in \Omega_i$, each assigned to a processor **do**
4: Evaluate the directional derivatives needed to enable enforcement of the traction boundary conditions via (3.49) (if they exist in a given problem) at time $t = t_n$
5: Advance the solution throughout the domain ω_j and its boundary $\partial\omega_j$ to time t_{n+1} via Equations (3.45). (Traction conditions are enforced by using the directional derivatives with the FC Neumann operator in the computation of the right hand sides F_1, F_2, F_3.)
6: Update physical Dirichlet boundary values at t_{n+1} (enforced by direct injection of the given Dirichlet boundary values.)
7: **if** ω_j has neighboring sub-patches $\{\omega_k\} \in \Omega_m,\ m = i$ **then**
8: Exchange fringe points solution values with $\{\omega_k\}$ via MPI
9: **end if**
10: **if** ω_j has neighboring (sub)patches $\{\omega_k\} \in \Omega_m,\ m \neq i$ **then**
11: Interpolate and exchange values with $\{\omega_k\}$ via MPI
12: **end if**
13: **end for**
14: **end for**
15: **end for**

Algorithm 1: *Summary of the full FC-based numerical PDE solver for the elastic wave equation.*

including its convergence and dispersion properties, its stability and its parallel performance.

3.4.1 Convergence and code verification experiments

A verification of the correctness of the numerical scheme and its C++ implementation (which accounts for the full complexity inherent in equations (3.33) and (3.36)), can be performed by the method of manufactured solutions (MMS). This procedure allows us to confirm, in particular, that the algorithm achieves the theoretical order of accuracy (in our case, 5th-order)—similar verification procedures have been used extensively; see e.g. [66, 69, 77]. The method postulates a smooth solution and substitutes it into the governing equations and boundary conditions to determine the external forcing and boundary conditions that yield the desired solution. For our example we consider the solution

$$\begin{aligned} u &= \sin(2\pi f_1(x - c_1 t)) \sin(2\pi f_2(y - c_1 t)) \sin(2\pi f_3(z - c_1 t)), \\ v &= \sin(2\pi f_1(x - c_2 t)) \sin(2\pi f_2(y - c_2 t)) \sin(2\pi f_3(z - c_2 t)), \\ w &= \sin(2\pi f_1(x - c_3 t)) \sin(2\pi f_2(y - c_3 t)) \sin(2\pi f_3(z - c_3 t)), \end{aligned} \tag{3.57}$$

with parameters $f_1 = f_2 = f_3 = 12$Hz and $c_1 = c_2 = c_3 = \sqrt{5}$m/s on a cylinder of radius $r = 10$cm and length of 30cm filled with a solid of material constants $\rho = 1, \lambda = 1$ and $\mu = 2$. For the numerical computations the cylindrical domain is viewed as the union of two overlapping patches ω_1 (the rectangular box) and ω_2 (the cylinder)—as indicated in Figure 3.15, which also contains a snapshot of the v coordinate, $v = v(x, y, z, t)$, of the solution (u, v, w). Considering integer multiples of the coarsest spatial discretization used, which contains a total of $N = 30 \times 30 \times 30$ points in ω_1 and $N = 30 \times 30 \times 80$ points in ω_2, we advance the simulation on up to 512 processors for five thousand time-steps at a step size of $\Delta t = 0.1\mu s$, employing both Dirichlet and traction boundary conditions. The maximum absolute errors among all components of the vector solution (u, v, w) for all time-steps are displayed in

Table 3.2—where the fifth-order accuracy of the algorithm can easily be appreciated.

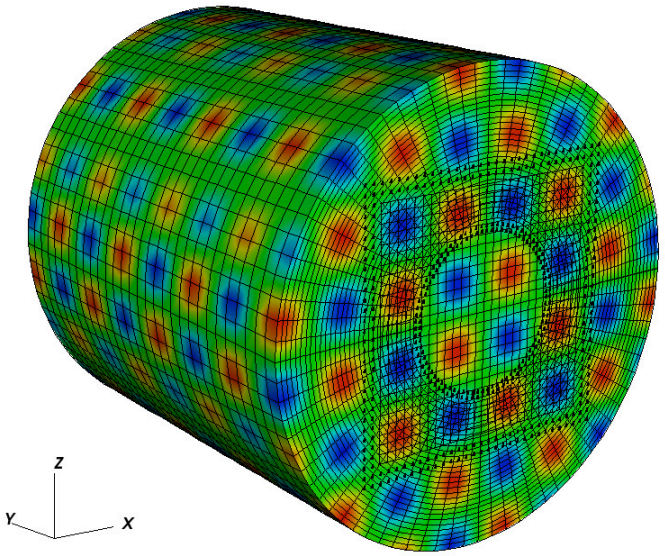


Figure 3.15: *Numerical values of the vertical displacement produced by the right-hand-side source (3.57). The bold dots indicate the layers of interpolation points used.*

N (patch ω_1)	N (patch ω_2)	L^{∞}_{err} Dirichlet	$O(L^{\infty})$	L^{∞}_{err} Traction	$O(L^{\infty})$
$30 \times 30 \times 30$	$30 \times 30 \times 80$	2.34e-03	—	2.82e-03	—
$60 \times 60 \times 60$	$60 \times 60 \times 160$	7.89e-05	4.89	8.29e-05	5.09
$90 \times 90 \times 90$	$90 \times 90 \times 240$	9.41e-06	5.24	9.37e-06	5.38
$120 \times 120 \times 120$	$120 \times 120 \times 320$	2.00e-06	5.38	2.00e-06	5.37
$150 \times 150 \times 150$	$150 \times 150 \times 400$	6.51e-07	5.03	6.51e-07	5.03

Table 3.2: *Maximum error in the elastic displacement over 5000 time-steps. A fine temporal discretization is used so that the overall error is dominated by errors arising from the spatial discretization.*

3.4.2 Dispersion and stability experiments

In Section 3.1 we highlighted two of the great strengths of the FC methodology: its nearly dispersionless character and its stability. In order to quantify these properties

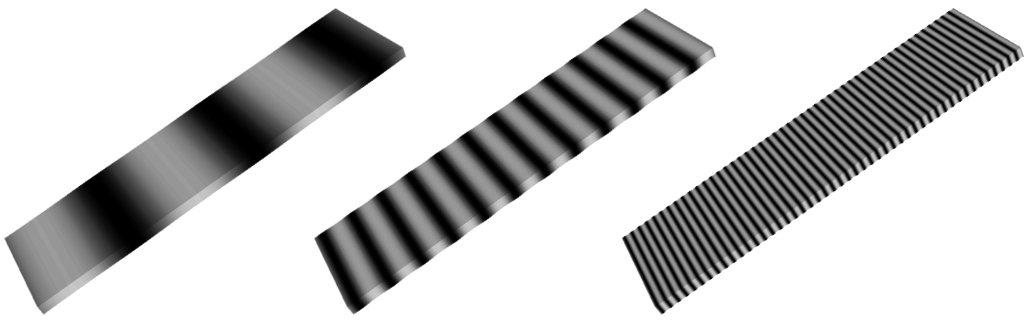


Figure 3.16: *From left to right: The numerical solution of equation* (3.58) *at time* $t = 0.1s$ *for configurations involving* $n = 3, 12$ *and* 48 *wavelengths along a thin aluminum plate.*

in the 3D block-decomposed methodology, it is useful to consider, as we did in the 1D wave equation example, numerical solutions to the elasticity equations for waves that propagate over long distances. Considering a 3D aluminum plate of dimensions 400mm $\times$ 10mm $\times$ 100mm and a Poisson's ratio of $\nu = .35$ (corresponding to material parameters $\lambda = 6.049 \times 10^{10}\text{N/m}^2$ and $\mu = 2.593 \times 10^{10}\text{N/m}^2$) using both Dirichlet and traction boundary conditions, our method is applied to advance the solution given by

$$u(x, y, z, t) = 0, \quad v(x, y, z, t) = \sin(2\pi n(x - c_L t)), \quad w(x, y, z, t) = 0, \tag{3.58}$$

over one period ($t = 0$ to $t = 1$) for various wavenumbers n discretized by fixed numbers 15 and 20 of points per wavelength, and where $c_L = \sqrt{\lambda + 2\mu}$ is the longitudinal wave speed in meters-per-second. An illustration of the computational domain and the corresponding solution can be seen in Figure 3.16 for wavenumbers $n = 3, 12$ and 48; similar results can be obtained for much larger wavenumbers. As shown in what follows (and as is the case for all cases we have ever considered in this regard, including cases involving much larger wavenumbers), the FC numerical errors are virtually independent of frequency as long as the numbers of points-per-wavelength are kept constant.

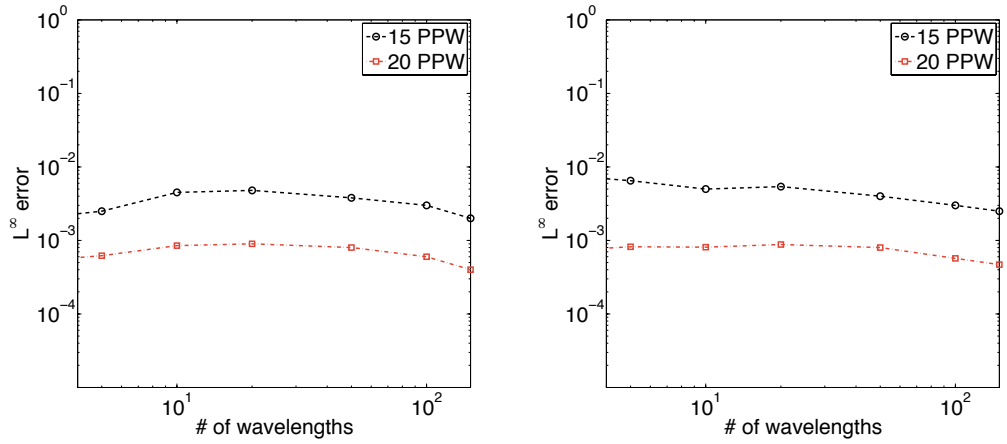


Figure 3.17: *Maximum numerical errors over all space and over one full temporal cycle of a plane wave solution with increasing number of wavelengths for Dirichlet (left) and traction (right) boundary conditions.*

The numerical results presented in this section, which result from simulations conducted using up to 512 processors for the highest wavenumbers n, were obtained by means of the domain decomposition and parallel implementation described in Section 3.2.3.2. The use of a block decomposition represents a key difference between the present implementation and that associated with the 1D example discussed earlier: for a number of p_{total} parallel processes, waves that travel along each dimension's line segmentation can cross Gram polynomial matching regions as many as $O(p_{total}^{1/3})$ times—as opposed to the $O(1)$ crossings in the 1D example. Consideration of Figures 3.17 and 3.18, which resulted from a run involving $600,000$ time-steps, suggests that the use of large numbers of sub-domains and interior sub-patches does not give rise to significant dispersion or stability degradation.

3.4.3 Parallelization experiments

The load-balancing methodologies introduced in Sections 3.2.3.2 and 3.2.3.3, respectively, enable an efficient parallel implementation of the FC solver by means of a workload distribution based on an overset decomposition of the computational domain

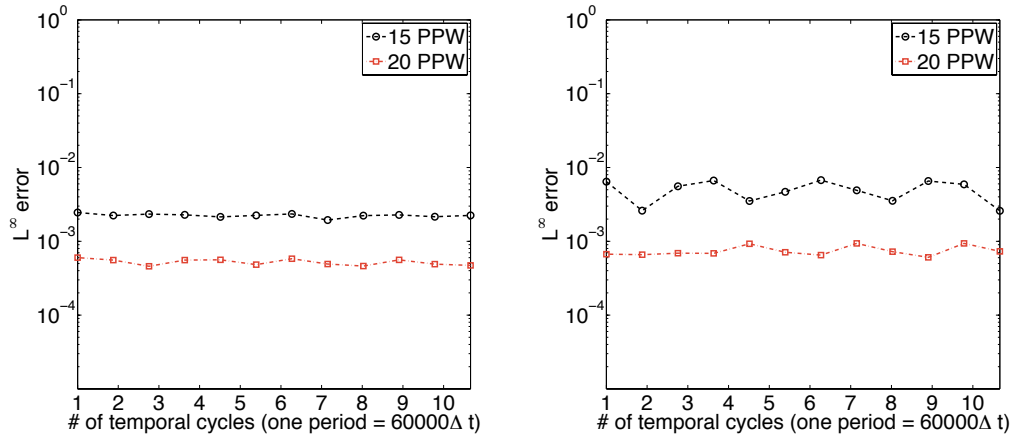


Figure 3.18: *Numerical errors over many temporal cycles for a plate ten-wavelengths in length demonstrating long-time stability for Dirichlet and Traction boundary conditions (left and right images, respectively).*

into a collection of curvilinear patches and sub-patches. By augmenting each sub-patch with a fringe region and assigning it to a single processor, our proposed methodology maintains accuracy throughout the full computational domain and additionally enables rapid computations of the needed Fourier transforms using fixed FFT sizes (see Appendix D.3). To demonstrate the parallel scalability, we conducted tests on a computing cluster (Appendix D.4) using 16 cores per node for up to 480 processing cores. The elasticity solver was advanced for 1,000 time-steps and each processor recorded the maximum absolute error in the solution in the corresponding sub-patch as well as the time spent in local computation—including the time spent in interpolation and communication but excluding the (extremely small) start-up and initialization times.

To analyze the scalability of the solver we first consider a thin-plate test problem with traction boundary conditions, and we record the number S of CPU-seconds required for the solver to advance a million spatial unknowns for one time-step; it is easy to check that S is given by the expression

$$S = \frac{(\#\text{ of processors}) \times (\text{total computation time})}{3 \times \left[(\#\text{ of spat. discret. pts.}) + (\#\text{ of traction bdry. pts.})\right]/10^6}, \tag{3.59}$$

where the factor of 3 stems from the total number of unknowns at each point, i.e. the displacements u, v and w.

Clearly, perfect scaling arises whenever S remains (essentially) constant as the discretization is refined and the number of assigned processors is increased. The S values for an experiment concerning the aforementioned thin plate containing a single patch are given in Table 3.3; these results illustrate the perfect scalability of our solver up to at least 480 processors/sub-patches, while maintaining fixed accuracy, for a single patch configuration (see Table 3.4 for corresponding results in a multi-patch setup).

Remark 6. *For typical applications of the FC elasticity solver, the expected computational load on each processor outweighs the communication load of the fringe regions, and hence it is not necessary to explicitly consider this factor in processor allocation—it is indirectly accounted for by the relatively equal block sizes enforced by the algorithm we put forth in Section 3.2.3.3.*

A second test was conducted to determine the scalability properties of our FC elasticity solver for configurations containing multiple patches—for which an additional computational cost associated with inter-patch interpolation occurs. Use of an appropriate inter-patch interpolation strategy increases the computing cost per million unknowns per processor *by a fixed amount*, that is, by an amount that is independent of the number of patches used. This useful property results from the fact that each donor and receiving sub-patch contains a number of interpolation points that does not grow with the overall discretization (as long as the sub-patch discretization sizes remain fixed, which is the recommended strategy for our solver), and, therefore, the total time spent by each processor in interpolation procedures is independent of the number of processors used. Furthermore, to mitigate the possible imbalance that may arise from the additional load incurred by sub-patches that require interpolation versus those that don't, all interpolation weights are precomputed on the donor patch during initialization so that all interpolations can be performed locally.

This additionally ensures that no stencils are communicated: each boundary sub-patch simply receives, at each time-step, the interpolated u, v and w values directly from each donor sub-patch and replaces those values in its solution. The corresponding results of an experiment conducted on a plate with a circular through-hole, whose geometry is composed of the six different curvilinear patches detailed in Chapter 4.2, is displayed in Table 3.4. Clearly, our methodology maintains the excellent parallel scaling and consistent accuracy observed in Table 3.3 for the single patch configuration, and only a small additional amount of computing time is required by the necessary inter-patch interpolation.

# grid pts	# processors	L^∞ error	$O(L^\infty)$	S
799,200	120	2.83e-3	—	1.20 sec
2,589,408	360	3.00e-4	5.54	1.23 sec
4,111,884	480	1.38e-4	5.04	1.30 sec

# grid pts	# processors	L^∞ error	S
4,111,884	120	2.90e-4	1.34 sec
—	360	3.00e-4	1.23 sec
—	480	3.06e-4	1.28 sec

Table 3.3: *CPU-seconds per million unknowns and errors for a domain consisting of a single curvilinear patch (no interpolation), indicating excellent scalability up to at least 480 processors. Upper table: weak convergence test, for which both the number of discretization points and the numbers of processors are increased simultaneously. Lower table: strong convergence test, wherein for a fixed number of grid points, the number of processors used is increased. Clearly, essentially perfect parallel efficiency is obtained under weak and strong convergence tests.*

# grid pts	# processors	L^∞ error	$O(L^\infty)$	S
377,460	120	1.70e-1	—	1.61 sec
3,033,360	360	7.98e-3	4.41	1.54 sec
10,252,980	480	1.04e-3	5.03	1.65 sec

# grid pts	# processors	L^∞ error	S
3,033,360	240	7.89e-3	1.51 sec
—	360	7.98e-3	1.55 sec
—	480	8.32e-3	1.45 sec

Table 3.4: *CPU-seconds per million unknowns and errors for a domain of a plate with a circular-through hole consisting of six different curvilinear meshes which interpolate from each other (see Section 4.2.3). Upper table: weak convergence test, for which both the number of discretization points and the numbers of processors are increased simultaneously. Lower table: strong convergence test, wherein for a fixed number of grid points, the number of processors used is increased. Clearly, essentially perfect parallel efficiency is obtained under weak and strong convergence tests.*

Chapter 4

Numerical examples and applications

The introduction section in Chapter 1 presents an assortment of motivating scientific and engineering applications for problems in elastodynamics, chief among them the propagation of seismic waves induced by earthquakes and techniques related to ultrasonic non-destructive evaluation in, e.g., airplane wings. In order to demonstrate the broad applicability and high resolution capabilities of our FC-based solver for complex elastic-wave propagation problems, this chapter presents in Section 4.1 examples of shear wave propagation and subsequent ground motion in simulated 3D geological structures; and in Section 4.2 a first-time comparison between experimental studies and the corresponding 3D numerical simulations for the scattering of guided ultrasonic waves in thin aluminum plates with defects.

4.1 Seismic response in 3D topographies

The first investigations of the effect on seismic wave motion by irregular topography were conducted a half-century ago (e.g. the studies of [37, 75]) and were spurred by artifacts found in seismograms recorded near rough patches of the Earth—which were later identified and confirmed both experimentally and semi-analytically [9, 13, 14] as scattered waves consisting mostly of Rayleigh surface waves. An additional observation

concerned high amplification of ground acceleration at topographic ridges, which was found near epicenters of strong earthquakes, and which manifested itself as intense shaking in elevated areas of the surface. (The 1994 Northridge, California earthquake, for example, recorded *at the top of a hill* one of the highest accelerations ever observed [71]). A full understanding of these observations, which can enable predictions as to the locations of the strongest expected ground motion, necessitates a fast solver that can resolve responses to seismic waves in large and complicated computational domains within a short computation time. In what follows we present results of applications of our elastic wave equation solver with its overset grid methodology to 3D geological structures impacted by incident shear waves. In particular, we will study wave amplification by a 180m hill, a situation proposed by [14] and treated previously by a variety of numerical methods [7, 45, 49]; this will serve both as a physical verification of our implementation of traction boundary conditions as well as an example of a problem of great practical interest. As noted in Section 4.1.1, in particular, the present solver produces high-definition solutions in short runs using 384 cores of a present-day computer by means of significantly coarser spatial discretizations than those used by other solvers. In Section 4.1.2 a significantly more complex extended version of this topography is presented, which is treated by our solver with no additional effort.

4.1.1 A gently-sloped mountain

The effect of three-dimensional topography has been repeatedly studied via consideration of a Gaussian hill of height 180m whose profile is parameterized by

$$y = y(x, z) = 1050 + 180 \exp\left[-\frac{x - 1040}{2 \cdot 250^2} - \frac{z - 1040}{2 \cdot 150^2}\right] \text{m} \tag{4.1}$$

for $(x, z) \in [0\text{km}, 2.08\text{km}]$, and whose computational domain extends to a depth of 1.05km. (The bottom of the lowermost computational patch lies on the plane $y = 0$.) For our treatment, this computational domain is covered by two overlapping patches as depicted in Figure 4.1, which contain $N = 400,000$ discretization points each: the patch containing the hill extends from a minimum height of $y = 550$m to the top

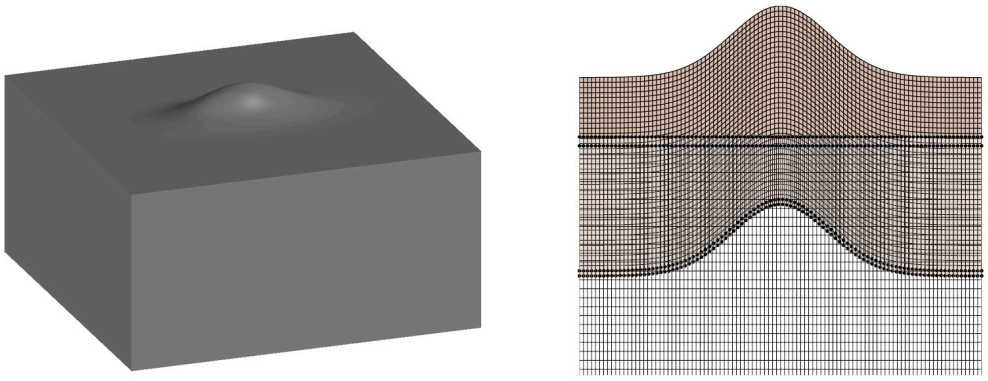


Figure 4.1: *Left: Computational domain corresponding to a* 180 *meter hill composed of two overset patches. Right: A two-dimensional close-up cut of the geometry and its discretization.*

of the hill at $y = 1230$m and spans the lateral dimensions of the domain given by $(x, z) \in [0, 2080] \times [0, 2080]$m, and the second patch covers the cubic domain given by $(x, y, z) \in [0, 2080] \times [0, 900] \times [0, 2080]$. A homogeneous ground medium is assumed, with a P-wave velocity of 3.2km/s, an S-wave velocity of 1.8475km/s and a density of 2200kg/m^3. A vertically incident S-wave (shear wave) of fundamental wavelength $\lambda = 180$m polarized along the short axis of the hill is considered; the time-dependent source is a Ricker wavelet at $y = 0$ that is described by the Dirichlet boundary conditions

$$\begin{aligned} u(x, 0, z, t) &= 0, \\ v(x, 0, z, t) &= 0, \\ w(x, 0, z, t) &= .5[2\pi^2 f_0^2 (t - t_0)^2 - 1] e^{-\pi^2 f_0^2 (t-t_0)^2}, \end{aligned} \tag{4.2}$$

centered at time $t_0 = .5$s with a frequency $f_0 = 10.2$Hz [14, 49]. Traction-free boundary conditions were imposed on the top surface of Figure 4.1 and periodic conditions were assumed along the lateral edges. The simulation was evolved up to time $t = 1.4$s at a step size of $\Delta t = 2.0 \times 10^{-4}$s.

The time responses (e.g., seismograms) for the various displacements measured at receivers along the minor axis of the hill are given in Figure 4.2, where we have

indicated the incident S-wave as well as the diffracted P- and Rayleigh waves. A strong amplification of the ground motion at and near the top of the hill is clearly observed, and the diffracted waves away from the topography are composed of a combination of a P-wave and a Rayleigh wave, both of which carry similar amplitudes in the vicinity of the hill but decay at different rates as functions of the distance from the hill. Snapshots of the solution for the in-plane ground displacement $w(x, y, z, t)$ over the entire surface are additionally presented in Figure 4.3, where, following previous contributions, it can be further observed that most of the waves diffracted away from the hill are generated in the summit. The amplification factor within the hill itself appears to exhibit a strong variability: the maximal amplitudes occur in small areas around the summit or at mid-slope when a de-amplification is exhibited in the peak. A strong directivity by the shape of the hill is also seen: most of the energy propagates in the direction along the minor axis of the hill, with a much less significant amount in the direction of the major axis. This is further confirmed in the bottom image of Figure 4.2, where results of the ground motion produced along the $x-$profile that runs along the major axis show that there is almost no diffracted wave field radiated in this direction.

The results of this simulation are in excellent agreement with solutions produced previously by means of the time-domain boundary element method in [45] (which, as noted in the conclusion section of that paper, becomes very costly for problems requiring large discretizations); with solutions found by the high-order spectral element method in [49], which employed a total of $N = 4,935,953$ discretization points for a time-step that scales with at least $1/\Delta x^2$; and, finally, with solutions found by the stable FD method in [7] which employed a total of $N = 109,808,412$ discretization points. For comparison, the diffracted P- and Rayleigh waves were accurately evaluated in our simulations using a total of $N = 800,000$ discretization points for a run-time of approximately 5.9 minutes on 384 processors.

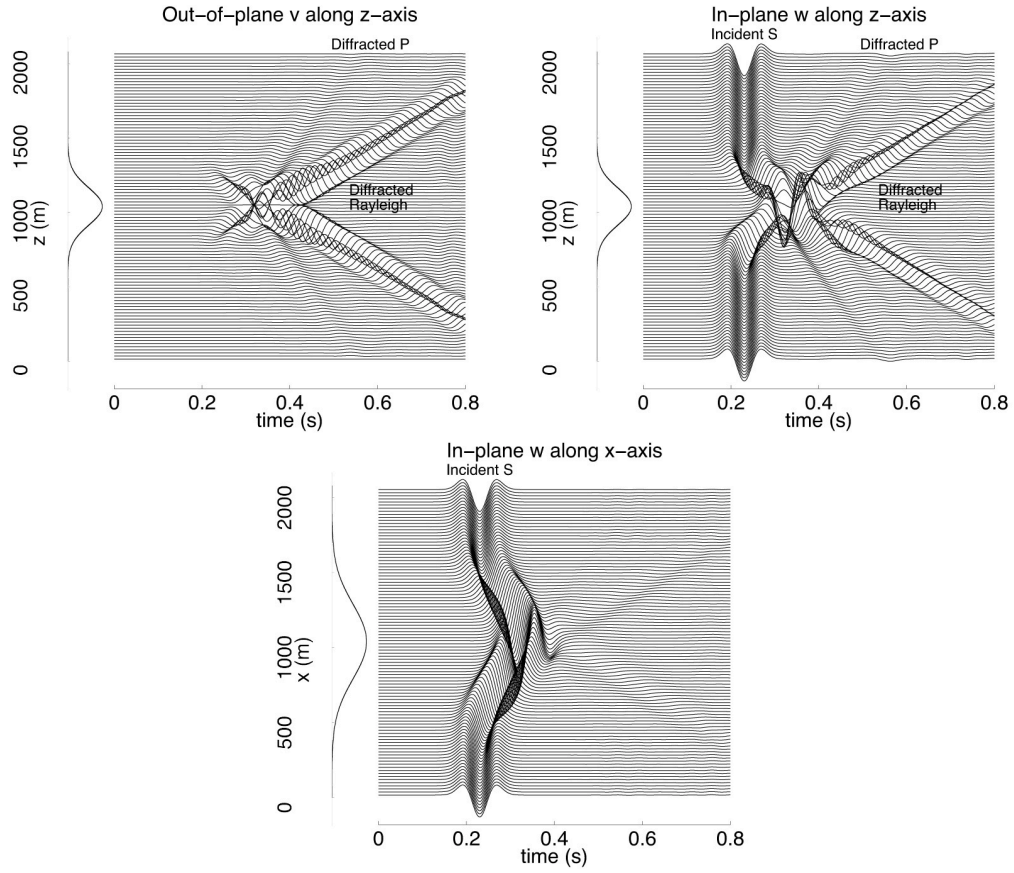


Figure 4.2: *Time responses of various displacement components on the surface for receivers placed along the z-axis (the minor axis of the mountain) and x-axis (the major axis). The incident S-wave is polarized along the minor (z-) axis of the* 180m *mountain.*

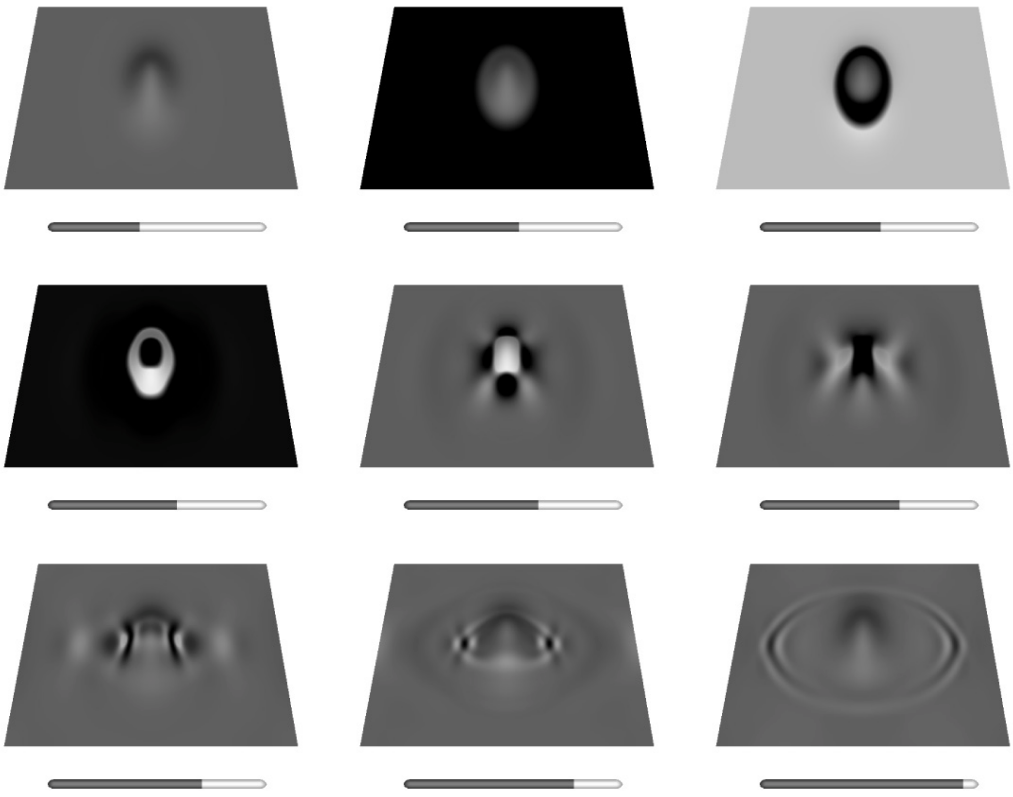


Figure 4.3: *Horizontal displacement $w(x, y, z, t)$ at various snapshots in time, where the z-coordinate direction contains the minor axis of the hill. An intense field concentration at the summit is clearly visible, as is the preferential propagation direction of the diffracted waves (in the direction of the short axis of the mountain) it generates. The time slider is given underneath each snapshot.*

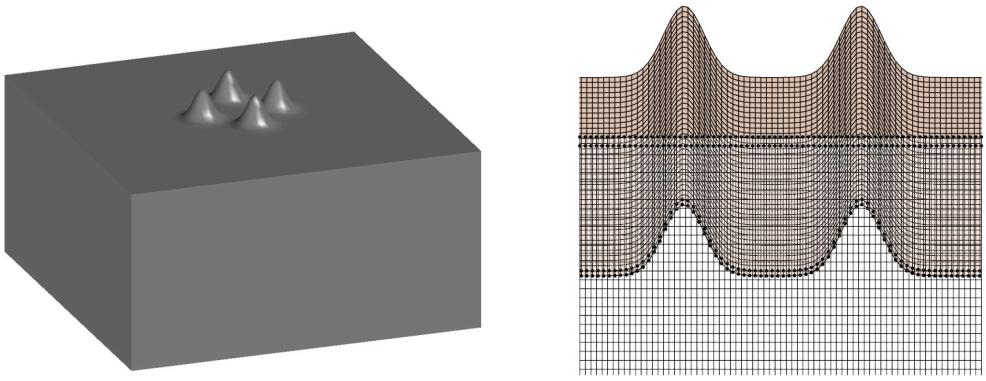


Figure 4.4: *Computational domain for a four steeply-sloped mountain system, and corresponding overset patches.*

4.1.2 A cluster of steeply-sloped mountains

To demonstrate the capabilities of our solver we extend the previous example by considering four sharp symmetric 180m tall mountains, each given by the respective parameterizations

$$y_i(x,z) = 1050 + 180\exp\left[-\frac{x-x_0^i}{2\cdot 50^2} - \frac{z-z_0^i}{2\cdot 50^2}\right]\text{m}, \quad i = 1,2,3,4 \tag{4.3}$$

centered at $(x_0^i, z_0^i) = (1240, 1040)$, $(800, 1040)$, $(1040, 1240)$,$(1040, 800)$ meters, respectively; once again, the full computational domain extends to a depth of 1.05km covered by two overlapping patches (Figure 4.4): the patch with the hills of size $N = 1,600,000$ points and the Cartesian patch underneath of size $N = 400,000$. We considered again an incident S-wave polarized along the z-axis with a source given by (4.2), and we ran the simulation up to a time of $t = 1.4$s at a step size of $\Delta t = 7\times 10^{-5}$s for a total computation time of just under 26 minutes on 384 processors.

The seismograms corresponding to receivers which, as in the previous example, were placed along the central z- and x-axes, are displayed in Figure 4.5. These images indicate how the wave structure that occurs in this case is significantly different from the one obtained for the single-hill problem. Although the largest amplifications can still be observed near each one of the hill summits, the diffracted waves do not have

a preferential propagation direction. This is likely due to the biaxial symmetry of the configuration: one observes similar ground motion in both displacements u and w in the cardinal directions of the surface plane. The overall energy in this case is diffracted into a much more complex wave pattern, and, unlike the observation in the single hill case, shaking in the neighborhood of the hills continues to exist well after impact. Snapshots of the solution for the in-plane ground displacement $w(x, y, z, t)$ over the entire surface are additionally presented in Figure 4.4, where it is similarly observed that the maximal amplitude occurs in small areas around the hill summits and within the general vicinity of the hill cluster. In this case, however, the diffracted energy continuously bounces between the free surfaces of the four sharp hills, exciting a long wake of low amplitude wavefronts radiating outward from the overall cluster so that displacement and ground motion remain elevated over much of the entire surface long after it does for the gentle hill considered in the previous section.

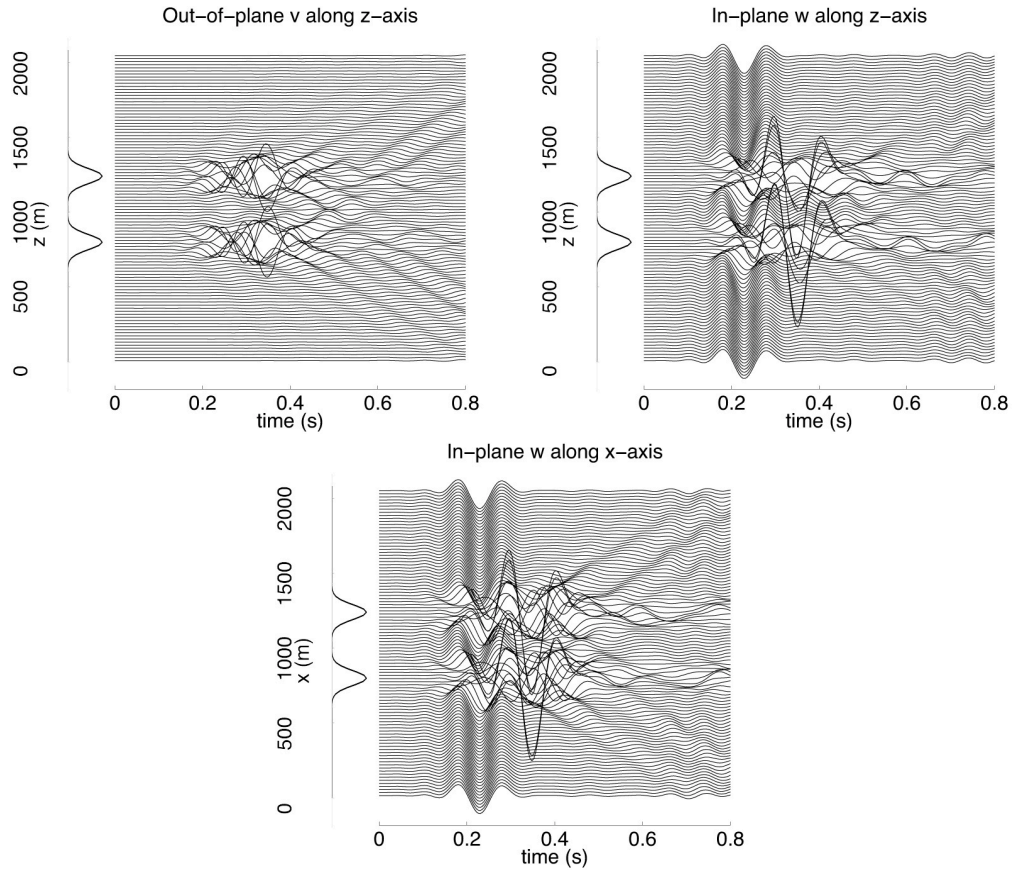


Figure 4.5: *Time responses of various displacement components on the surface for receivers placed along the z-axis and x-axis for a topography consisting of a cluster of four sharp mountains.*

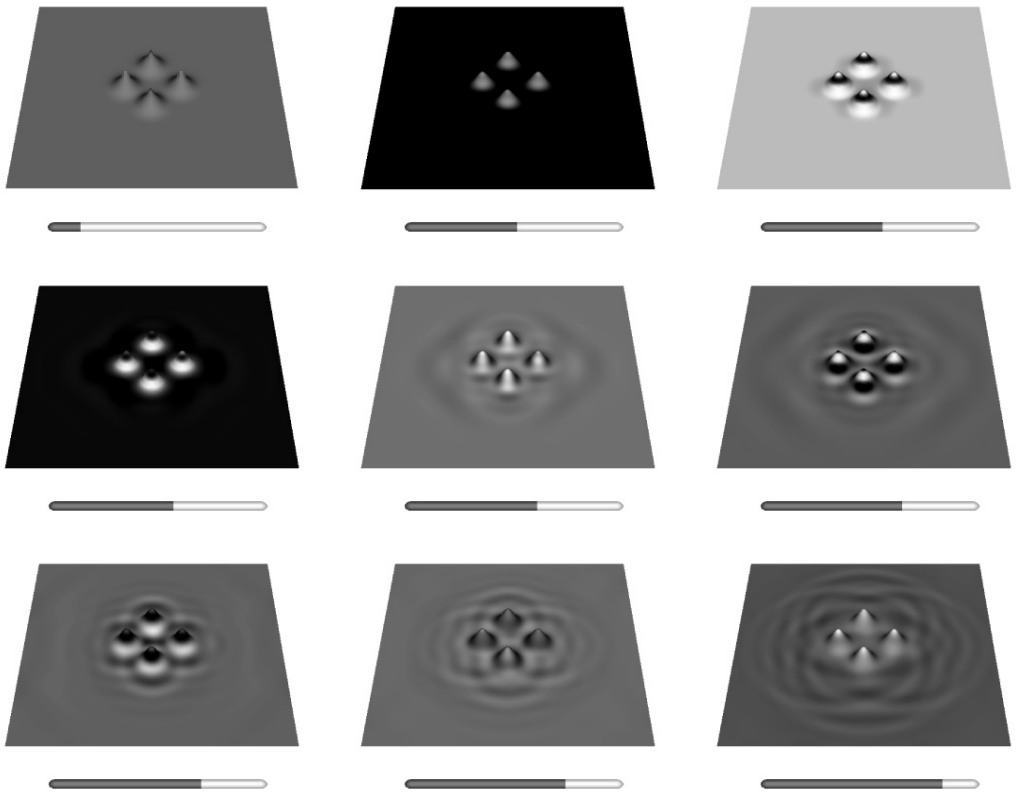


Figure 4.6: *Horizontal displacement $w(x, y, z, t)$ at various snapshots in time. Much of the diffracted energy remains concentrated within the cluster of hills long after impact.*

4.2 The scattering of waveguides in plates with defects

The concept of non-destructive evaluation (NDE) of materials through propagation of specifically excited elastic modes was first introduced in the 1960s [80, 22, 65, 78]. The propagation characteristics of ultrasonic wave guides (frequencies of $f \approx 1$ MHz to $f \approx 10$ MHz) in particular are directly related to the micro-structure and mechanical properties of a material, and have hence amassed increased attention and wide use in the field of characterization and testing of structures—including plates and beams—in an effort to detect defects by the properties of the scattered waves (see e.g. [34, 28, 51, 63, 40, 68] as well as the two excellent reviews [24, 74] on guided-wave NDE). The ultrasonic techniques in use in laboratories today have matured into a powerful tool to study the integrity of a variety of plate-like structures arising in industry, spanning from aircraft wings to oil pipelines to bridges. Ultrasonic waves induced on an effectively infinite plate interact with the top and bottom plate surfaces in the manner described in Chapter 2 of this thesis: sufficiently far from the source the elastic field amounts to a superposition of guided Lamb and/or Rayleigh wave modes. The specific nature and number of modes produced, the propagation of each one of which is governed by the plate dispersion equations (2.25), is determined by the details of the physical configuration. Exciting a certain combination of P- and S-wave modes by carefully selected temporal and spatial frequencies (which, in the work [53] is achieved through use of a sophisticated wedge-like structure to mediate the wave-transfer), for example, can enable the propagation of a single guided wave mode.

4.2.1 Generation of mechanical waves by surface tractions

In experimental applications, mechanical waves in solid structures are excited either through contact—employing, e.g., piezoelectric transducers—or through non-

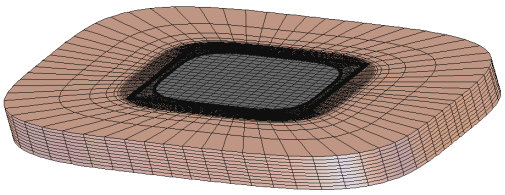


Figure 4.7: *Model of an aluminum plate overset with a sponge layer.*

contact methods such as electromagnetic acoustic transducers, air coupled devices and lasers [34]. The specific laboratory experiments considered in the next sections employ a classical contact method known as the wedge method (or *prismatic coupling block method*) which enables the launch of directional, nearly planar waves with very good efficiency and very good control of the acoustic wave temporal profile. The specific nature of the excitation technique is based on a well-known methodology for conversion of longitudinal waves into Rayleigh or Lamb modes. This is accomplished by elastically coupling a plastic prism or wedge atop whose sloping surface a piezo-electric transducer is placed which produces a plane longitudinal wave that impinges at an angle θ to the boundary between the structure of interest and the wedge [78]. This results in the creation of a periodic perturbation between the boundary of the solid and the wedge with a spatial period equal to the corresponding Rayleigh wavelength in the material sample. The perturbation thus excites a Rayleigh wave that propagates along the surface of the material.

4.2.2 Elastic wave propagation on a three-dimensional plate: preliminary example

Before considering the specific experimental setup described in the previous section, here we present results of a preliminary case study concerning generation of guided waves at ultrasonic frequencies under a simpler mathematical setup. As a preliminary study we thus consider a rectangular aluminum plate of dimensions 300mm×10mm×150mm and material parameters $\lambda = 2.24 \times 10^6$ N/m^2 and $\mu =$

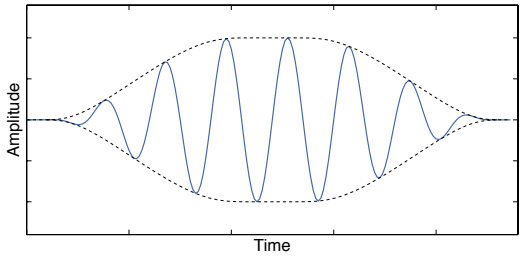


Figure 4.8: *An example of the windowing function S (black) given by (3.53) and the resulting sinusoidal waveform (blue) for the incident source. The center of the vertical axis represents zero amplitude.*

9.600×10^6 N/m^2, discretized by a number $N = 140 \times 33 \times 100$ of points and overset with a sponge layer of size $N_{sponge} = 100, \gamma_0 = 0.075$ and $\beta_0 = 2$ (with 250 points in the radial dimension)—as illustrated in Figure 4.7.

A simple surface-wave source was produced by means of a sinusoidal-in-time traction boundary condition which is smoothly ramped-up in time and space over a small area $A = L_x L_z$ on the surface $y = h = .005$mm, where L_x is the width in x and L_z the width in z (the choice $L_z > L_x$ simulates a sort of "thick" line source). Mathematically the non-zero traction boundary condition of the form

$$\begin{pmatrix} \sigma_{11} & \sigma_{21} & \sigma_{31} \\ \sigma_{12} & \sigma_{22} & \sigma_{32} \\ \sigma_{13} & \sigma_{23} & \sigma_{33} \end{pmatrix} \begin{pmatrix} n_x \\ n_y \\ n_z \end{pmatrix} = \begin{pmatrix} 0 \\ h \\ 0 \end{pmatrix}, \tag{4.4}$$

was applied on the top surface of the plate, where

$$h = S(x - x_0, c_x, L_x) S(z - z_0, c_z, L_z) S\left(t - L_t/2, c_t, L_t\right) \sin\left(2\pi f t\right), \tag{4.5}$$

and where S is the smoothly varying windowing function constructed on the basis of the function (3.53). The time window of the source is given by $L_t = n_{cycles} f$, where f is the frequency and n_{cycles} is the number of cycles for which the source is activated. An example of the time-evolution for a periodic force activated for a few cycles is given in Figure 4.8.

By choosing an excitation frequency of $f = 1$MHz in the ultrasound regime

activated for 250 cycles, we applied the force to the left edge of the plate and evolved our simulation at a time-step of $\Delta t = 1$ ns for $200,000$ time-steps and employed a filter of parameters $(p, \alpha) = (4, -\log\left(10^{-2}\right))$. Figure 4.9 displays snapshots in time of the magnitude of the resulting out-of-plane displacement solution $v(x, .005, z, t)$, where the higher-frequency P-waves can be observed as well as the surface waves that follow. Figure 4.10 additionally illustrates an initial wave-train excited by two different point source frequencies, $f = 1.8$MHz and $f = .9$MHz, corresponding to the different wavelengths that are observed. The experiments in this section demonstrate the dispersion relations contained in the Rayleigh-Lamb frequency equations and a first physically-motivated experiment employing the sponge layer.

4.2.3 Applications to ultrasonic non-destructive evaluation

The thin-plate scattering applications considered in this section are motivated by a collaboration with our co-authors in [53]; corresponding laboratory experiments were performed by these colleagues at the University of Vigo in Spain. The contribution [53] presents a quantitative characterization of defects in aluminum plates by comparison between experimental and simulated (numerical) scattering patterns of narrow-band guided waves with circular and rectangular holes—albeit with a simplified mathematical model based on use of scalar waves on a two-dimensional domain. A suggestion that two-dimensional scalar models could be used in this context, which predates our contribution, is not justified by detailed consideration of physical processes but, rather, by analogy. The two-dimensional mathematical model and numerical methodology used in that contribution are presented in Sections 2.3.1 and Appendix B, respectively. A comparison between the experimental and the two-dimensional numerical results presented in [53] shows approximate structural agreement, but, in fact, significant quantitative differences between experiment and theory in important quantities such as the reflection coefficient, which in some cases amount to departures of as much as 100%, have been observed. The corresponding three-dimensional study, conducted

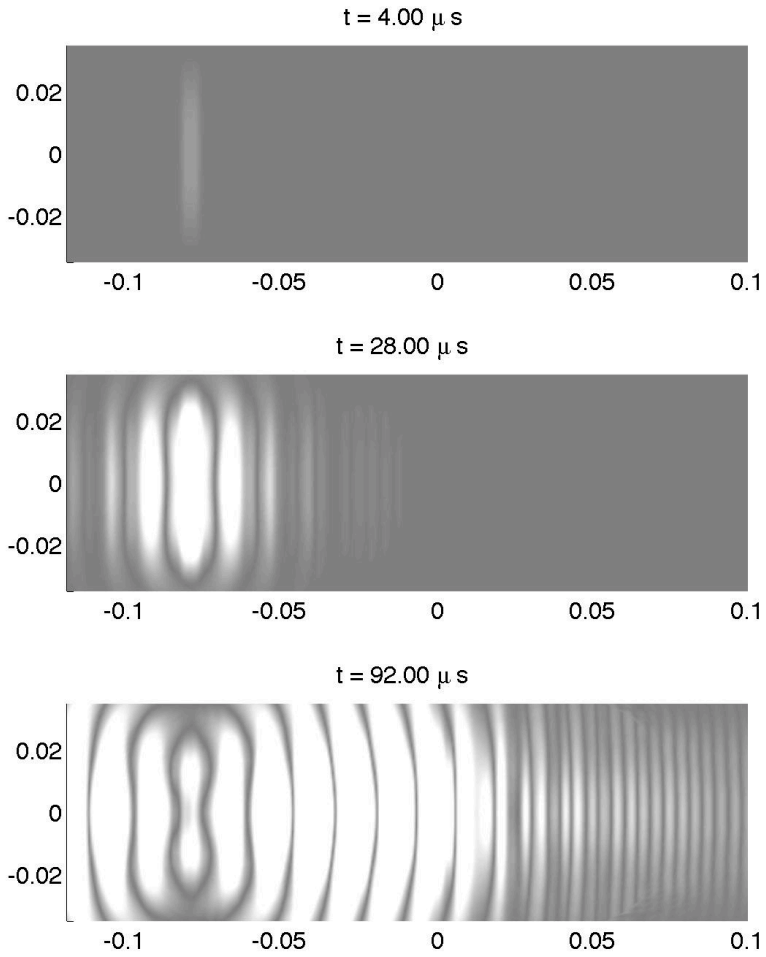


Figure 4.9: *The amplitude of the numerical displacement v as a response to a periodic source near the left edge of the displayed region.*

by means of our elasticity solver and presented below, addresses this deficiency by considering the full complexity of the elastic wave propagation in three-dimensional space. An example of the experimentally measured scattering by a source activated for 99 cycles is given in Figure 4.11.

In the present section we revisit this problem, but we tackle the numerical simulation problem by means of our full three-dimensional elastic wave solver. The incident waves in these laboratory experiments were generated by means of the wedge method described in Section 4.2.1: the longitudinal wave emitted by a piezoelectric transducer was coupled to the surface of a thin plate through a prismatic coupling block of angle

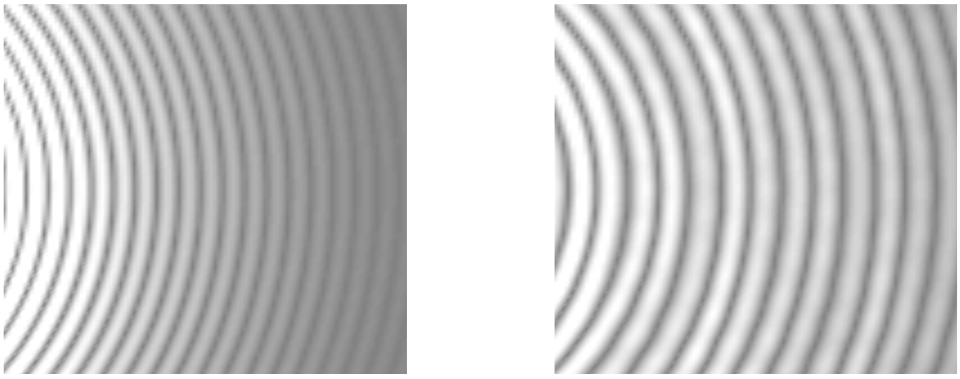


Figure 4.10: *A snapshot of the P-wave-train displacement field arising from a normally incident source at frequencies $f = 1.8$ MHz (left) and $f = 0.9$ (right).*

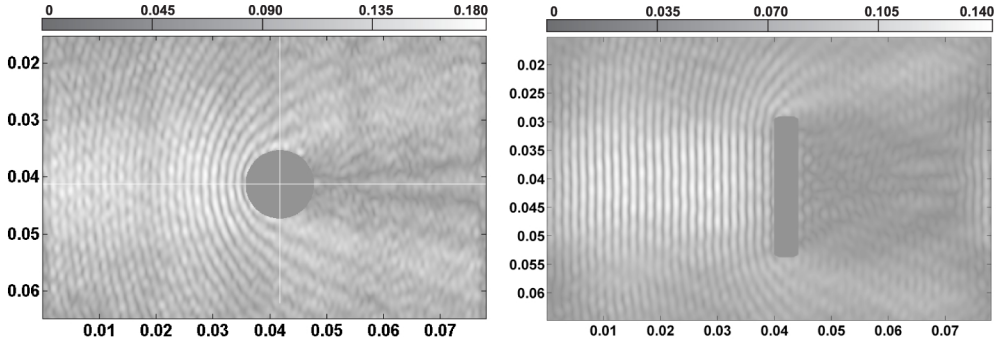


Figure 4.11: *Experimental values for the modulus of the complex out-of-plane displacement as depicted in [53] for the scattering of thin waveguides in plates with through-hole defects.*

$\theta_w = 65$ degrees, as depicted in the full experimental setup presented in Figure 4.12. In the experiments under consideration, the piezoelectric source was excited in such a way that the guided wave trains had a frequency of $f = 1$ MHz and were quasi-monochromatic with wavelength $\lambda = 2.96$mm, yielding a Rayleigh phase velocity $c = \lambda f = 2960$ m/s. The instantaneous out-of-plane displacement field $v(x, y, z, t)$ due to the propagation of the guided wave-train were obtained by means of a novel double-pulsed TV holography system—details can be found in [53, 67, 34]. The resulting processed optical data rendered an optical phase-change map $\Phi(x, z, t)$ that is proportional to the instantaneous out-of-plane displacement $v(x, y, z, t)$.

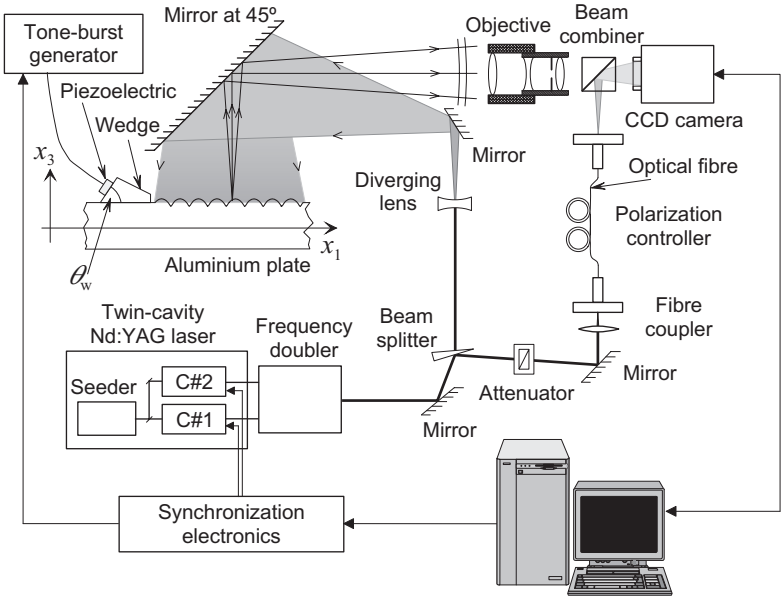


Figure 4.12: *Experimental setup as depicted in [53] for generation and recording of surface waves in a thin plate.*

For our numerical experiments we use analytical representations provided by our collaborators for the experimentally observed Dirichlet boundary condition on the left edge of the computational domain. These representations are provided in the form given by

$$
\begin{aligned}
u(0,y,z) &= \left(\frac{\lambda_0}{8\pi}\right) \frac{\left(-(2-\xi^2)e^{k_r\sqrt{1-\xi^2}(y-h)} + 2e^{k_R\sqrt{1-\alpha^2\xi^2}(y-h)}\right)\left(i\sqrt{1-\xi^2}k_R\right)}{\left(-(2-\xi^2)+2\sqrt{1-\xi^2}\sqrt{1-\alpha^2\xi^2}\right)\left(\sqrt{(2e_1z+e_2)^2+(k_R)^2}\right)}\Phi(z,t),\\
v(0,y,z) &= \left(\frac{\lambda_0}{8\pi}\right) \frac{-(2-\xi^2)e^{k_r\sqrt{1-\xi^2}(y-h)} + 2\sqrt{1-\xi^2}\sqrt{1-\alpha^2\xi^2}e^{k_R\sqrt{1-\alpha^2\xi^2}(y-h)}}{-(2-\xi^2)+2\sqrt{1-\xi^2}\sqrt{1-\alpha^2\xi^2}}\Phi(z,t),\\
w(0,y,z) &= \left(\frac{\lambda_0}{8\pi}\right) \frac{\left(-(2-\xi^2)e^{k_r\sqrt{1-\xi^2}(y-h)} + 2e^{k_R\sqrt{1-\alpha^2\xi^2}(y-h)}\right)\left(i\sqrt{1-\xi^2}(2e_1z+e_2)\right)}{\left(-(2-\xi^2)+2\sqrt{1-\xi^2}\sqrt{1-\alpha^2\xi^2}\right)\left(\sqrt{(2e_1z+e_2)^2+(k_R)^2}\right)}\Phi(z,t),
\end{aligned}
\tag{4.6}
$$

where $\alpha = \dfrac{c_T}{c_L}$, $\quad \xi = \dfrac{c_R}{c_T}$, $\quad c_R = \lambda_R f$, $\quad k_R = \dfrac{2\pi}{\lambda_R}$, and where $\lambda_0 = 5.32\times10^{-8}$m is the wavelength of the optical laser used for measurements in the TV holography system. The function Φ that appears in these expressions is the so-called complex optical

phase-change field and is analytically given by a least squares fit of experimental data (at the left boundary $x = x_0$) to the function given by

$$\Phi(z,t) = \Phi(x_0,z,t) = Ae^{i\phi}g(t)h(z)B_1(t)B_2(z) \tag{4.7}$$

where the component functions are given by

$$\begin{aligned}
&B_1(t) = b_1 e^{-\left(\frac{t-b_2}{b_3}\right)^2} e^{i(c_1 t + c_2)},\\
&B_2(z) = \left(\frac{p_1}{z^2 + q_1 z + q_2}\right) e^{i(e_1 z^2 + e_2 z + e_3)},\\
&g(t) = P(t, t_1, t_2, \delta_t),\\
&h(z) = P(z, z_1, z_2, \delta_z),\\
&P(\xi,\xi_1,\xi_2,\delta) = \begin{cases}
0 & -\infty < \xi < \xi_1 - \dfrac{\delta}{2},\\
\cos^2\left(\dfrac{\pi}{2}\left(\dfrac{\xi-\xi_1}{\delta} - \dfrac{1}{2}\right)\right) & \xi_1 - \dfrac{\delta}{2} \le \xi \le \xi_1 + \dfrac{\delta}{2},\\
1 & \xi_1 + \dfrac{\delta}{2} < \xi < \xi_2 - \dfrac{\delta}{2},\\
1 - \cos^2\left(\dfrac{\pi}{2}\left(\dfrac{\xi-\xi_2}{\delta} - \dfrac{1}{2}\right)\right) & \xi_2 - \dfrac{\delta}{2} \le \xi \le \xi_2 + \dfrac{\delta}{2},\\
0 & \xi_2 + \dfrac{\delta}{2} < \xi < \infty,
\end{cases}
\end{aligned} \tag{4.8}$$

and where the corresponding parameters for each experimental case considered are given in Appendix C. The corresponding Dirichlet conditions on the displacements u, v and w in terms of the measured optical phase-change field Φ, given in Equation (4.6), assumes a quasi-Rayleigh wave-like dependence in the thickness mode that decays to the bottom of the plate—a feature characterized by the incident source applied only to the top surface and the relative thickness of the plate ($2h = 10$mm) against the induced wavelength ($\lambda_R = 2.96$mm). As will be seen, this formulation for our elasticity solver results in simulations that are in very close *quantitative* agreement with experiments, but a Lamb wave formulation for the thickness, as mentioned in Section 2.3, may also capture the behavior of the experimental fields: the excited waves can be described by the zeroth symmetric and anti-symmetric Lamb modes displayed

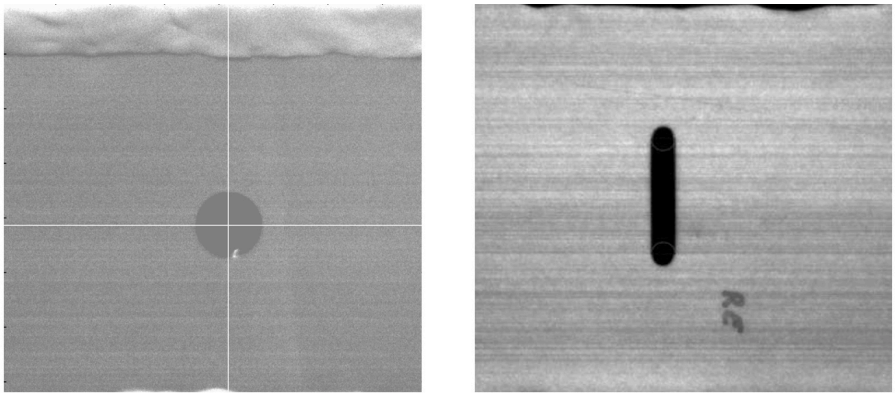


Figure 4.13: *Photographs of experimental samples including holes of circular (left) and slot (right) shapes.*

in Figure 2.2 and governed by the Rayleigh-Lamb frequency equations (2.25).

4.2.4 Three-dimensional computational simulations

Our computational simulations concern experiments performed at the University of Vigo for small wave-trains incident on circular holes of diameters $4, 6, 8$ and 12mm centered in plates of thickness $2h = 10$mm; additional experiments are currently ongoing for geometries containing slot-shaped holes as well as half-through holes. We have constructed the corresponding computational geometries, to the physical specifications of the experimental regions of interest shown in Figure 4.13, by means of our overlapping grid strategy—the corresponding computational domains including an illustration of the overset patches is given in Figure 4.14.

As discussed in Chapter 3 and evidenced by Figure 4.14, the edges of the hole boundaries require particular care in modeling—they must be smooth to avoid singularities in our high-order numerical method, but they must be sufficiently sharp so as to provide an adequate approximation of the experimental sample. We perform a rounding of such edges by means of a treatment that has been briefly mentioned earlier and that is presented in detail in Appendix A—where we make use of transfinite interpolation coupled with a parametric representation of a superellipse to construct two-dimensional slices of the holes that are subsequently rotated to produce the

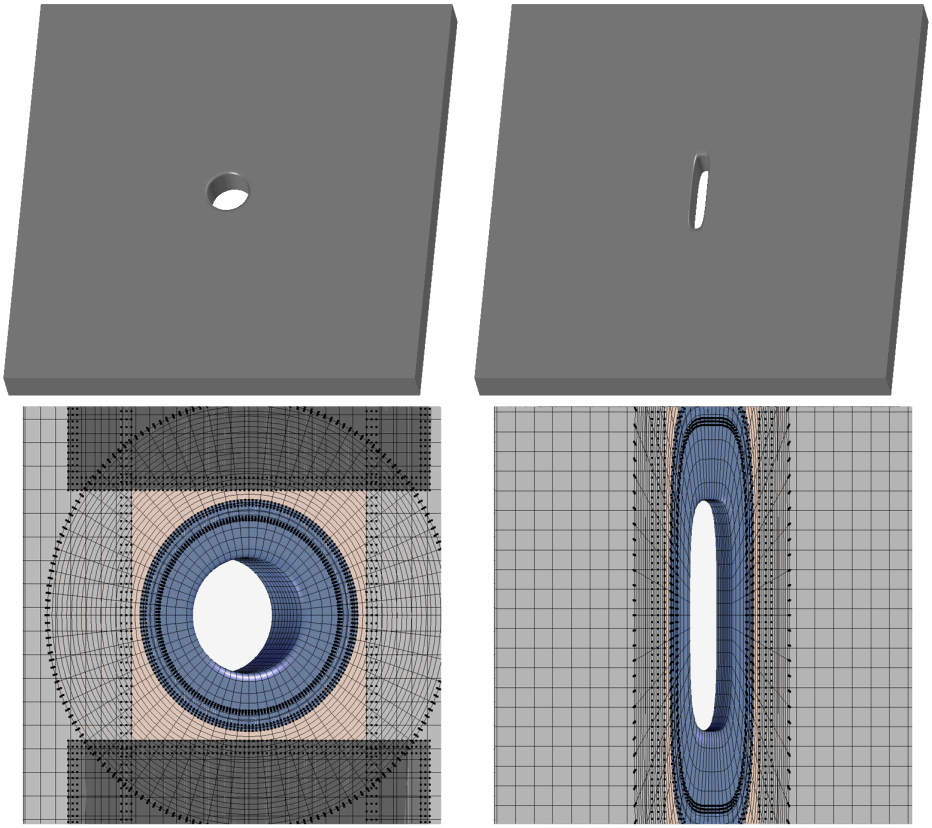


Figure 4.14: *Computational models corresponding to the experimental geometries presented in Figure 4.13. The models are composed of six overlapping patches (bottom row).*

corresponding three-dimensional geometries. We have found that use of the rounding parameter $\eta = 20$ in equation (A.1) provides adequate approximations—the solver produces solutions in very good agreement with the experimentally measured fields, as will be demonstrated later in this section. Examples of the corresponding rounded geometries—both for the circularly shaped through-holes and the rectangularly shaped through-hole—are illustrated in Figure 4.15.

4.2.5 Numerical results and comparison to experiments

A total of $N = 5,933,561$ discretization points were used for the six sub-patches describing the 12mm hole case, and similar numbers were used for the $4, 6$ and 8mm holes. Traction boundary conditions were enforced at the top and bottom of the

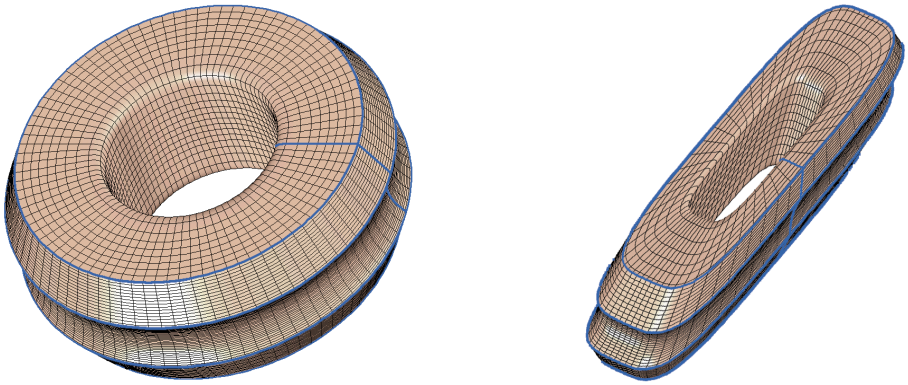


Figure 4.15: *Meshes for a rounded circular and rectangular scatterer that are constructed by revolving two-dimensional grids produced by means of the transfinite interpolation described in Appendix A.*

plate as well as the hole boundaries, and the solution was advanced at a time-step of $\Delta t = 0.5$ns for a total number of $50,000$ steps. Simulations were carried out on a computing cluster using 464 processors and using a number of four fringe points in each dimension for the subsequent sub-patches for a total run-time of just under two hours each.

A depiction of the solution values for the scattering by a 12mm hole is given in Figures 4.16 and 4.17 as snapshots of the out-of-plane displacement solution v for both the experiments and our numerical simulations. The images displayed demonstrate very good agreement between measured field values and the three-dimensional simulation employing the approximate Dirichlet boundary condition given by the expressions (4.6)—a mutual validation of our numerical methodology as well as the cutting-edge measurement techniques employed by our collaborators. Figure 4.18 additionally illustrates profiles of the out-of-plane displacement v along the centered vertical and horizontal lines that overlay Figure 4.16 and Figure 4.17 to the left of the hole. The agreement between the experimental and numerical profiles further supports the assumption of the approximate incident field given by (4.6). (The oscillations in the experimental curves are attributed to experimental error: cf. the second row of images in Figure 4.16 which indeed seems to suggest existence of experimental error

of relatively high frequency.)

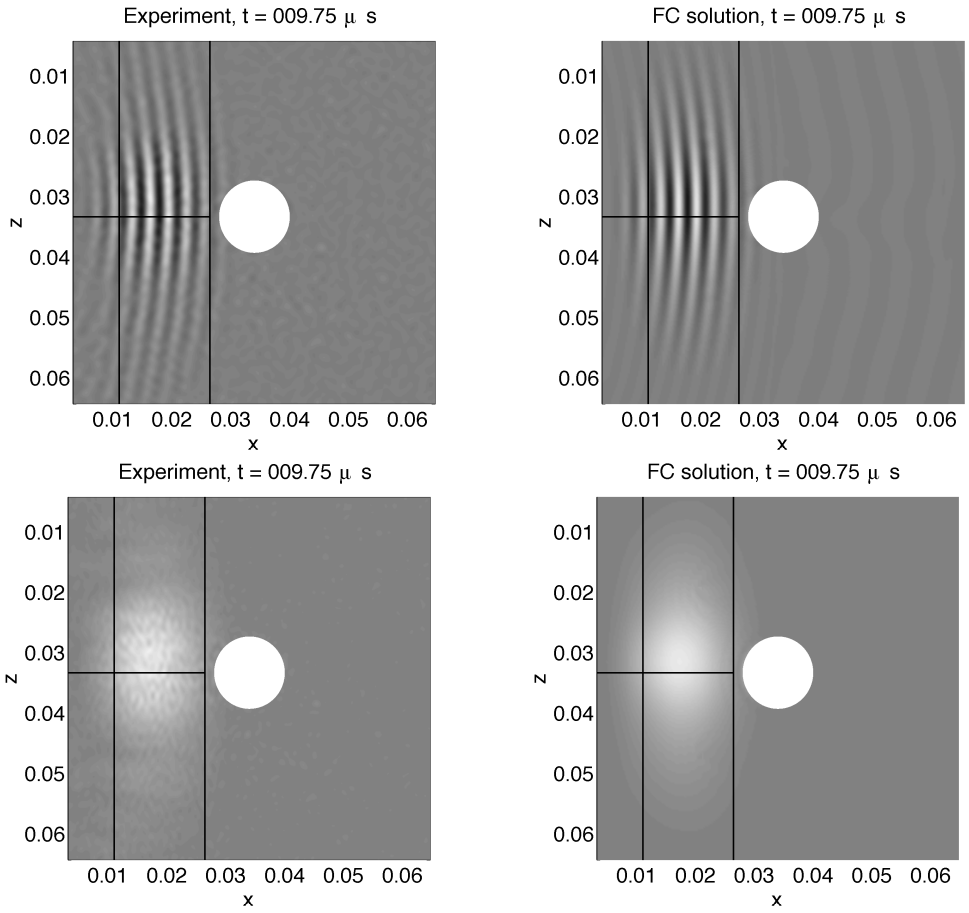


Figure 4.16: *Experimental (left) and simulated (right) snapshots of the incident waves for the* 12*mm diameter circular hole. Top: real part of the out-of-plane displacement* $v(x, y, z, t)$*. Bottom: corresponding complex amplitude.*

In order to provide an additional quantitative comparison in the present context, we evaluate the "backscattering reflection coefficient" from the fields measured along the horizontal line given in Figure 4.16 and compare it to the corresponding experimental quantity as provided by our collaborators. The right plot of Figure 4.19 displays the x coordinate of the location of the peak modulus v_{max} of the complex out-of-field displacement as a function of time along this line, where three distinct temporal intervals we call Zones $1, 2$ and 3 are demarcated by the dashed lines. In Zone 1,

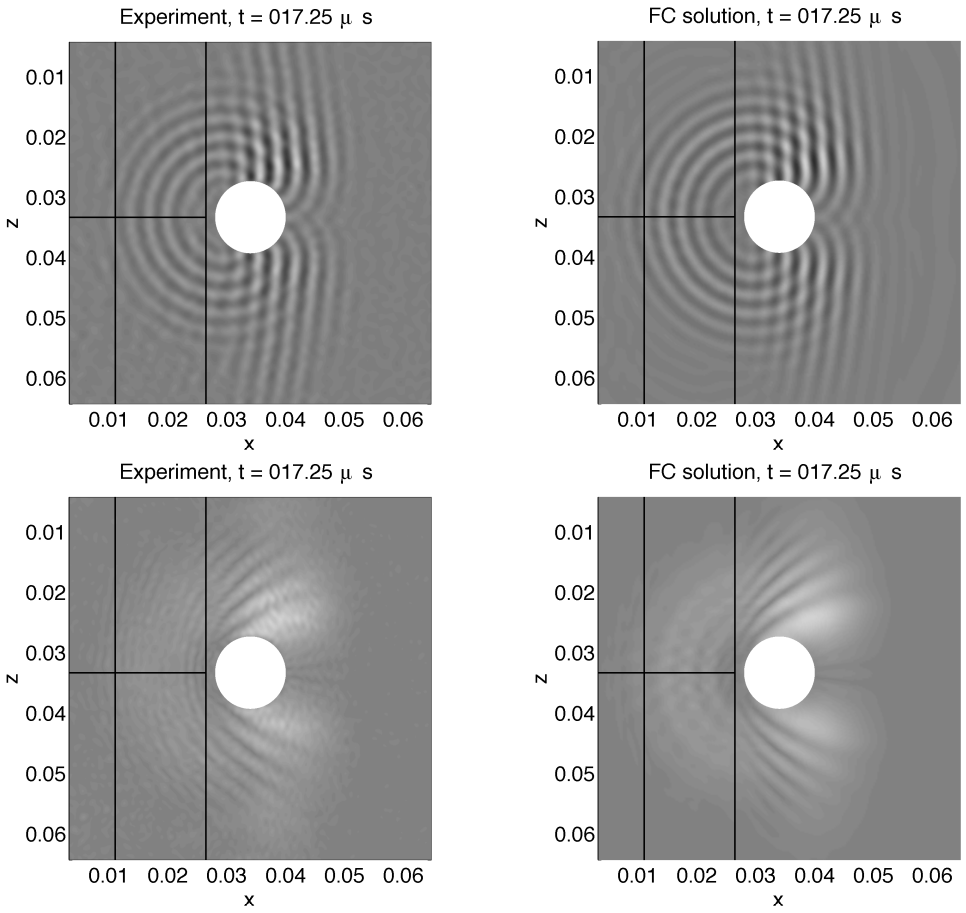


Figure 4.17: *Experimental (left) and simulated (right) snapshots of the scattered waves for the* $12mm$ *diameter circular hole. Top: The real part of the out-of-plane displacement* $v(x, y, z, t)$*. Bottom: The corresponding complex amplitude.*

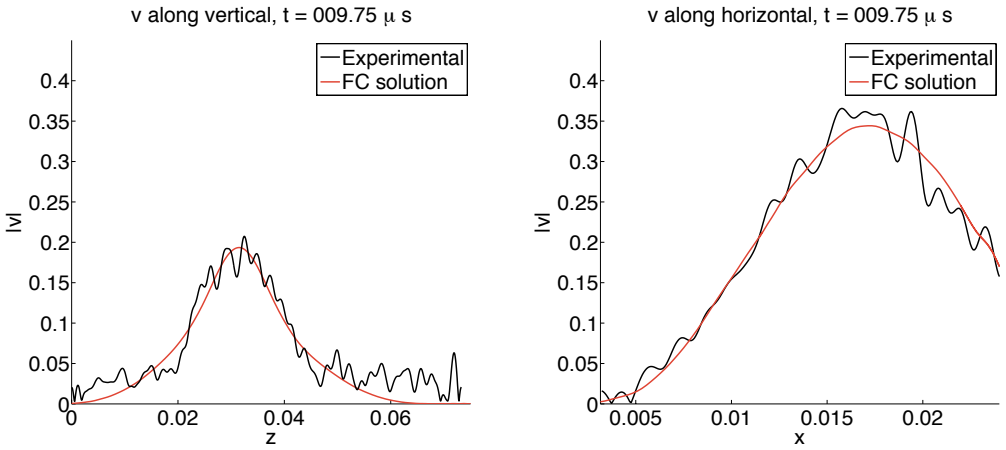


Figure 4.18: *Horizontal and vertical cross-sections of the fields displayed in Figure 4.16.*

the peak of the pulse has fully developed. In the next interval, the waves interact with the hole and the maximum is produced by a superposition of the incident and backscattered fields and is located close the hole border. In the final Zone 3, the maximum of the pulse is produced only by the backscattered field and travels back towards the left. In order to characterize the attenuation observed in the propagation of both the incident and backscattered fields (the left image of Figure 4.19), we analyze the maximum in Zone 1 as a function of its distance to the virtual point source x_0 associated with the incident field and the maximum in Zone 3 as a function of its distance to the virtual point source associated with the backscattered field—given by the center of the hole x_h. (The associated locations of both sources are given in the tables of Appendix C.) A nonlinear least squares curve fitting procedure is applied in each zone to the corresponding maxima v_{max} as a function of their respective distances to the incident and scattered source centers, employing a power law given by $g_j(x) = a_j x^{b_j}$ for both zones $j = 1$ and $j = 3$. This enables us to estimate the value of the incident field at the hole border–that is, at a distance r from the center of the hole—by the expression given by

$$g_1(r) = a_1(|x_h - r - xo|)^{b_1} \tag{4.9}$$

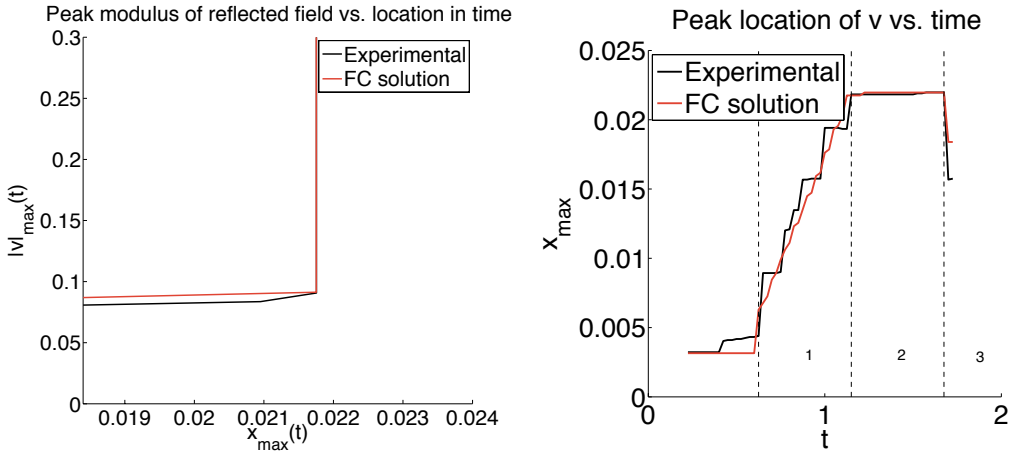


Figure 4.19: *Left: Peak modulus value of the v component of the backscattered field at point x using time as a parameter. Time advances right to left. Right: Location of the peak modulus as a function of time including the incident (Zone* 1*) and backscattered (Zone* 3*) fields.*

and the value of the backscattered field at the hole border by the expression given by

$$g_3(r) = a_3 r^{b_3}. \tag{4.10}$$

The resulting reflection coefficient, which we will denote R, can then be defined as the ratio of the incident field contribution at the hole border to the scattered field contribution at the hole border. It is simply given by the expression

$$R = \frac{g_3(r)}{g_1(r)}. \tag{4.11}$$

The values of R calculated from the corresponding laboratory experiments, from the time-dependent FEM scalar model employed in [67], and from the solutions found by our solver, are given in Table 4.1 for each of the $4, 6, 8$ and 12mm hole cases.

As can be observed, the coefficients of the fully three-dimensional elastic wave simulation are in much closer agreement to the experimentally measured values than the corresponding two-dimensional scalar wave equation produced by means of a finite element method (whose R values can deviate by more than 100%). The inability

	Scalar FEM	Experimental	Our solver
4mm	0.914	0.383	0.310
6mm	0.950	0.466	0.435
8mm	0.931	0.443	0.418
12mm	0.984	0.458	0.470

Table 4.1: *Reflection coefficients R for each hole diameter.*

of the scalar model to capture the true behavior of the backscattered fields can be explained by the neglect in the two-dimensional scalar approximation of the modal conversion that occurs between the various displacement components. Our solver, which solves the full elastic wave equation, does not make this approximation and hence better captures the full elastodynamics of the problem. Corresponding temporal snapshots of each displacement solution for the 12mm hole case are provided in Figure 4.20 where, as can be expected, the modal conversion is quite clear in both the horizontal components u and w. In particular, the energy conversion into diffracted waves traveling through the thickness of the plate, which is not accounted for by the scalar approximation, is also evident from the observed waves propagating downwards over the lateral sides of the hole—which we further observe in our simulations to reflect at the bottom of the surface and propagate back upwards. The corresponding solution snapshots for the $4, 6$ and 8mm circular hole cases are additionally provided in Figure 4.21.

Experimental measurements and subsequent processing for other hole geometries are currently being pursued by our collaborators, and we thus considered for a preliminary numerical experiment the geometry of a corresponding experimental case, namely, a rounded-rectangular hole. The computational domain, illustrated in Figure 4.14, contains a hole of lengths 24mm and 4mm along the major and minor axes, respectively; the rounded rectangular geometry was constructed by the prescription provided in Appendix A. Snapshots of the corresponding numerical values of the three displacement components are depicted in Figure 4.22, where much less significant mode conversion to the w component (displacement in the major axis of the hole) is

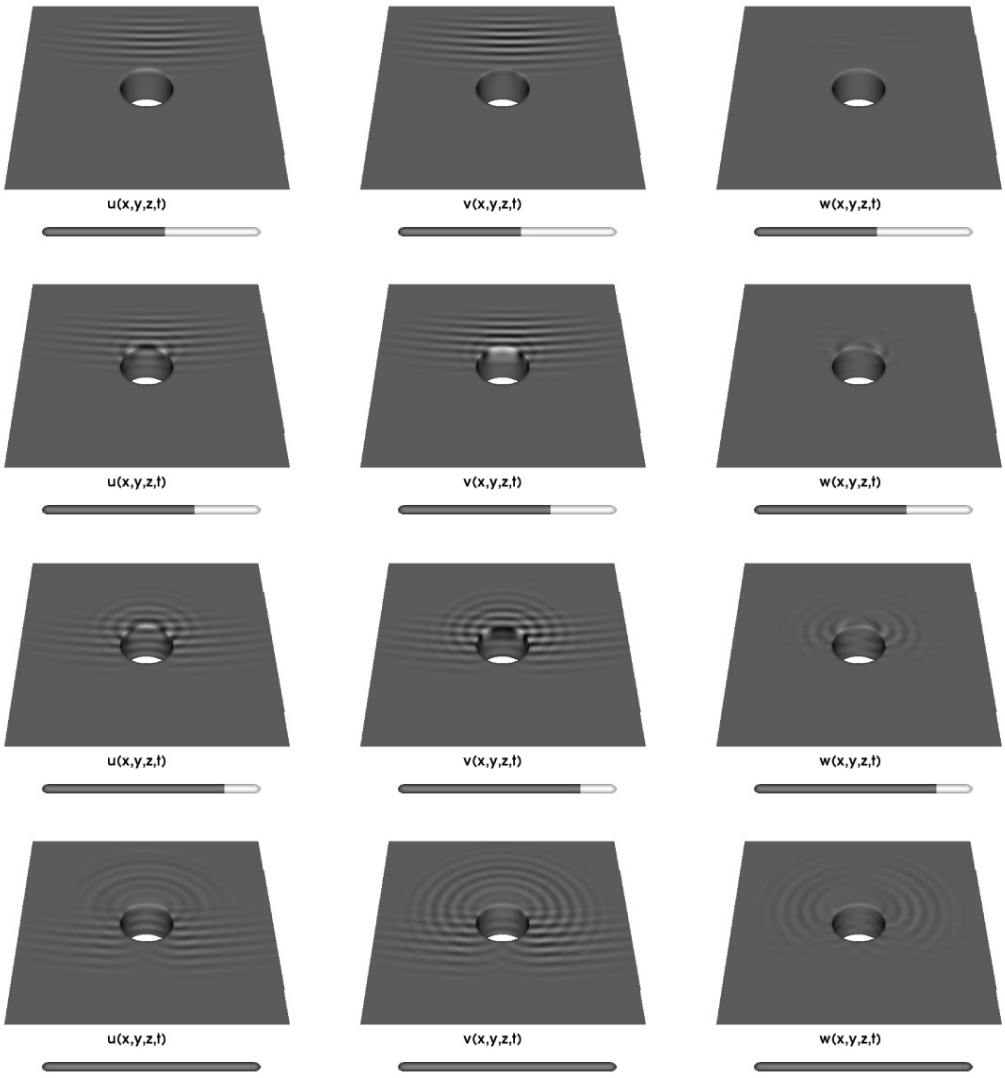


Figure 4.20: *Close-up of the* 12*mm diameter circular hole boundary for each displacement component* u*,* v *and* w *(from left to right, respectively)—clearly visible are the waves flowing from the top surface to the bottom along the hole thickness and the mode conversion between the in-plane components* u *and* w*.*

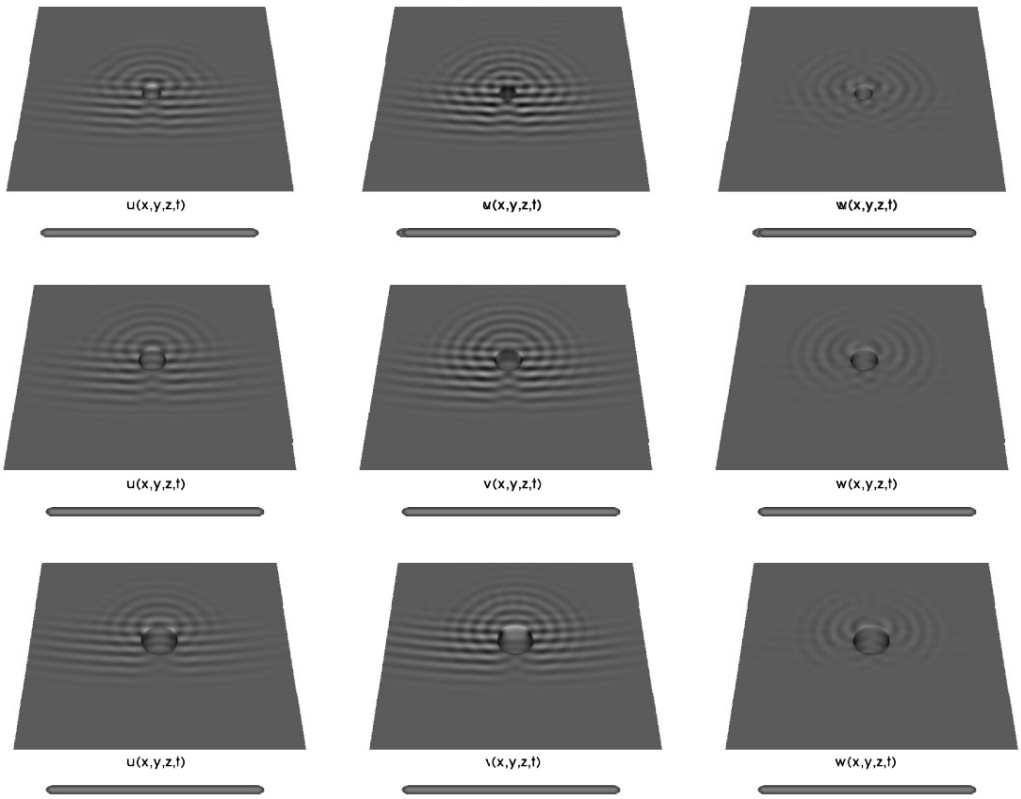


Figure 4.21: *Scattering results for the* $4, 6$ *and* 8*mm diameter circular hole configurations (from top to bottom, resp.) for the displacement components* u*,* v *and* w*.*

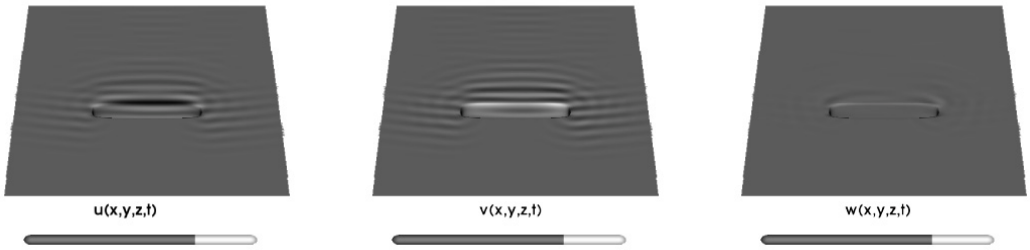


Figure 4.22: *Snapshots of wave scattering by a* 24*mm long rectangular slot.*

observed in the rightmost figure when compared with the circular hole geometries.

Chapter 5

Conclusions and future work

In this thesis we have introduced a new high-order methodology, based on a Fourier continuation approach, for the solution of elastodynamics problems in complex three-dimensional geometries. The methodology represents a first realization of a general three-dimensional FC-based solver for a variable-coefficient system of Partial Differential Equations. Through a combination of several key elements, including a new modified FC operator for the treatment of elastic boundary conditions and an overset strategy for the treatment of general three-dimensional geometries including unbounded media, the solver developed in this work effectively resolves linear elastic wave propagation problems in physically-relevant three-dimensional computational domains. Our new solver provides fast, accurate solutions that are nearly dispersionless, have significantly less restrictive CFL conditions than certain competing numerical techniques, and that are implemented on distributed-memory computing clusters by means of a parallelization strategy that has excellent scalability properties. The versatility of the techniques presented herein was demonstrated through a variety of realistic applications in seismology and non-destructive evaluation of materials with defects.

As with any newly-developed numerical methodology, the algorithms we have presented invite a number of interesting possible objectives for future work. For example, the explicit time-marching treatment we considered as part of our methodology—while carrying CFL conditions that scale only linearly with spatial discretization sizes—is

burdened by the stiff material properties inherent to most real-world elastic solids and manifests itself as a restrictive CFL constant. A method of temporal subcycling (based on polynomial temporal interpolation, which we have implemented successfully but have not employed for the numerical examples considered in this work), can be used to enable local time-stepping (e.g. for the finely discretized scatterers in Section 4.2.3) and faster overall computational times. Additionally, our solver could benefit from hybridization with other solvers that would afford localized implicit temporal formulations in regions of the computational domain wherein fine meshes need to be used. Our proposed framework for the treatment of complex geometries, finally, could be further supplemented by a proper hole-cutting algorithm that allows flexibility in the selection of the minimum segmented sizes for inclusion in an FC algorithm, as well as the ability to perform implicit inter-patch interpolation (where a point in a corresponding stencil could itself be an interpolation point for a patch). This would further enable simpler (and possibly coarser) constructions of patches for use in an overset strategy.

Other interesting applications also suggest themselves from the solver we have presented in this work:

- An extension of our solver could further permit treatment of material interfaces, including layered solid media, such as the models used for earthquakes or the coupling of the prismatic block used for non-destructive evaluation. This would entail two possible challenges: the numerical representation of such interfaces and the subsequent implementation of their corresponding physical boundary conditions. For example, fault lines in earthquakes can be described by the sliding of two layers of differing material properties —it would be of great interest to investigate the application of friction laws on a fault or a complex system of faults. The overset methodology we present here may enable high-order representations of any interface topology by the inclusion of boundary conforming curvilinear grids.

- The high-order and stable numerical method we have developed in this work has been thus far applied to acoustic wave-type problems. Our numerical scheme could also be coupled with shock-capturing methods for the investigation of physical problems that involve supersonic waves as well as the fine acoustic signature of the mechanical fields away from the shocks. Applications include supershear earthquakes, where the nucleation and propagation of rupture fronts along fault lines carry speeds in excess of the shear wave velocity, and underground damage generated by point blast explosions.

- Other PDE formulations, including the thermoelastic equations (for, e.g., the modeling of laser ablation used to excite modes for non-destructive evaluation) and viscoelastic equations (for, e.g., solid-fluid interactions), may also benefit from the strategies we have proposed in this work.

Appendix A

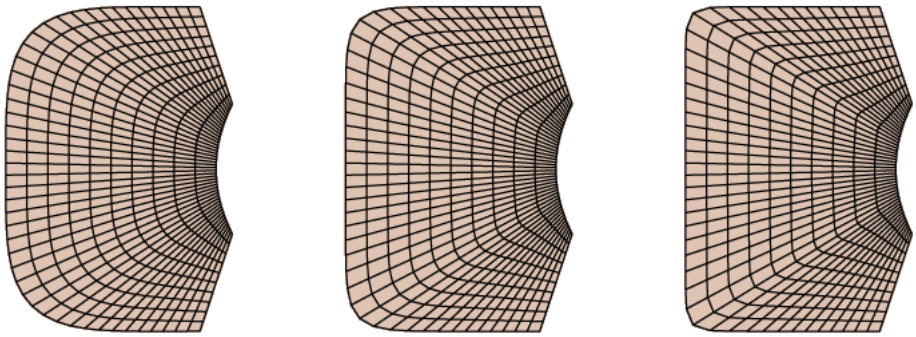
Algebraic mesh generation for holes with sharp edges

Solution singularities occur at points where sharp corners or edges of a geometry exist. Such singularities require special treatment; in our case such treatment is provided in the form of a smooth but sharp approximation of edges and corners for numerical treatment which is used in conjunction with a correspondingly fine computational mesh. A particularly simple procedure to produce useful geometries in our context is based on use of a superellipse—that is, the set of points satisfying the implicit equation

$$\left(\frac{x}{a}\right)^{\eta} + \left(\frac{y}{b}\right)^{\eta} = 1, \tag{A.1}$$

where η is any real number. It is easy to check that this curve tends to a rectangle as $\eta \to \infty$); a, b are the respective sizes of the major and minor super-ellipse axes. This is a particular case of the superquadric class of shapes that are often used in geometric modeling; they create—in rather clean, parametric forms—a wide variety of geometries with rounded edges and corners [44]. Used in conjunction with other (open or closed) curves and shapes to construct volumes that are structurally similar to cubes, the corresponding coordinate lines for the curvilinear mappings can be forged numerically (hyperbolic grid generators, elliptic grid generators) or algebraically (transfinite interpolation, Swan grids) [31, 33, 73].

An aluminum plate with a through-hole serves as a motivating example for de-



$\eta = 4$ $\eta = 8$ $\eta = 16$

Figure A.1: *Left to right: examples of rounded edges constructed via transfinite interpolation between two lines, a quadratic Bézier curve, and a polar parameterization of the superellipse (A.1).*

scribing the transfinite interpolation (TFI) used for rounded mesh generation: the four particular curves of the rounded configuration illustrated in Figure A.1 are used extensively in Chapter 4.2 and are parameterized at the top (a superellipse) by

$$\begin{aligned} x_t(q) &= \frac{-r_0\cos(\theta(q))}{(\cos(\theta(q))^\eta + \sin(\theta(q))^\eta)^{1/\eta}}, \quad 0 \le q \le 1, \quad \theta(q) = (9\pi/8)q - 9\pi/16, \\ y_t(q) &= \frac{r_0\sin(\theta(q))}{(\cos(\theta(q))^\eta + \sin(\theta(q))^\eta)^{1/\eta}}, \quad 0 \le q \le 1, \quad \theta(q) = (9\pi/8)q - 9\pi/16, \end{aligned} \tag{A.2}$$

at the bottom (a quadratic Bézier curve) by

$$\begin{aligned} x_b(q) &= (1-q)^2 x_0 + 2(1-q)q x_1 + q^2 x_2, \quad 0 \le q \le 1, \\ y_b(q) &= (1-q)^2 y_0 + 2(1-q)q y_1 + q^2 y_2, \quad 0 \le q \le 1, \end{aligned} \tag{A.3}$$

at the left by

$$\begin{aligned} x_\ell(r) &= x_b(0) + (x_t(0) - x_b(0))r, \quad 0 \le r \le 1, \\ y_\ell(r) &= y_b(0) + (y_t(0) - y_b(0))r, \quad 0 \le r \le 1, \end{aligned} \tag{A.4}$$

and at the right by

$$\begin{aligned} x_r(r) &= x_b(1) + (x_t(1) - x_b(1))r, \quad 0 \le r \le 1, \\ y_r(r) &= y_b(1) + (y_t(1) - y_b(1))r, \quad 0 \le r \le 1, \end{aligned} \tag{A.5}$$

so that the curves meet at the corners $x_b(0) = x_l(0), x_b(1) = x_r(0), x_r(1) = x_t(1), x_l(1) = x_t(0)$ and a grid can be constructed by a linear interpolation

$$\begin{aligned} x(q,r) &= (1-r)x_b(q) + rx_t(q) + (1-q)x_l(r) + qx_r(r) \\ &\quad - \left[qrx_t(1) + q(1-r)x_b(1) + r(1-q)x_t(0) + (1-q)(q-r)x_b(0)\right], \\ y(q,r) &= (1-r)y_b(q) + ry_t(q) + (1-q)y_l(r) + qy_r(r) \\ &\quad - \left[qry_t(1) + q(1-r)y_b(1) + r(1-q)y_t(0) + (1-q)(q-r)y_b(0)\right], \end{aligned} \tag{A.6}$$

on the rectangular space $[q,r] \in [0,1]^2$. Of immediate note in Figure A.1 is the non-orthogonality of the coordinate lines with the surface; this behavior (or lack thereof) may be undesirable for some computational schemes but can be corrected numerically via a hyperbolic or elliptic grid generator. The curvilinear formulations of the elastic wave equation we employ does not require coordinate lines that are orthogonal to the surface.

The above example of an algebraic mesh generation is two-dimensional in nature, but TFI is easily generalized to three dimensions (where six parametric surfaces are required, and where each surface itself can be constructed by its own TFI or elliptic generator). For the framework of plates with through-holes, a two-dimensional slice is sufficient; the third dimension z is established by revolving each surface by a rotation matrix tailored to, for example, a circular or rectangular hole (*scatterer*). One such operator is derived once again from a superquadric and is given by

$$\begin{pmatrix} \widetilde{x} \\ \widetilde{y} \\ \widetilde{z} \end{pmatrix} = \left(\frac{1}{(\cos(\theta))^\alpha + (\sin(\theta))^\alpha} \right)^{1/\alpha} \begin{pmatrix} r_x \cos(\theta) & 0 & r_z \sin(\theta) \\ 0 & 1 & 0 \\ -r_x \sin(\theta) & 0 & r_z \cos(\theta) \end{pmatrix} \begin{pmatrix} x \\ y \\ 0 \end{pmatrix}, \tag{A.7}$$

where r_x and r_z are scaling for the x- and z-axis in the rotation, respectively. The parameter α determines the specific geometry of the boundary curve: a value of $\alpha = 2$

represents a circular or elliptic hole (Figure 4.15, left), and a value of $\alpha = 12$ will create a rounded rectangle (Figure 4.15, right). The final construction of the full plate-with-hole geometry is then easily accomplished through the overset strategy described in Section 3.2.3 by embedding each scatterer in an overset arrangement with a Cartesian grid, as is described for the specific experiments modeled in Chapter 4.

Appendix B

Two-dimensional guided wave motion

This appendix presents a brief overview of the high-performance numerical method introduced in [19] which we employed to solve the scalar model presented in Chapter 2 that was published with our co-authors in [53].

B.1 Numerical methodology for solving the Helmholtz equation

If Γ denotes the boundary of the 2D through-thickness defect and $\hat{v}$ denotes complex amplitude associated to the out-of-plane component $v(x, y = h, z)$ of the scattered field, we can model our problem by means of the Partial Differential Equation

$$\begin{cases} \nabla^2 \hat{v} + k^2 \hat{v} = 0, \quad \text{outside } \Gamma \\ \left.\dfrac{\partial \hat{v}}{\partial n}\right|_{\Gamma} = g, \end{cases} \tag{B.1}$$

where g is a function defined on Γ. Stress-free conditions at the defect boundary imply conditions on the spatial derivatives of the scattered field as a function of the spatial derivatives of the incident field. For this reason a Neumann Boundary condition is used.

In order to efficiently produce accurate solutions of problem (B.1), we utilize a modified version of a highly efficient and accurate numerical methodology introduced recently [19], which we describe briefly in what follows. As it is well known, the (unique) solution of the Neumann Problem (B.1) can be expressed as a double-layer potential

$$\hat{v}(\mathbf{r}) = \int_{\Gamma} \frac{\partial G_k(\mathbf{r}, \mathbf{r}')}{\partial \mathbf{n}'_{\mathbf{r}}} \mu(\mathbf{r}') dl'. \tag{B.2}$$

Here G_k denotes the Hankel function

$$G_k(\mathbf{r}, \mathbf{r}') = iH_0^1(k|\mathbf{r} - \mathbf{r}'|) \tag{B.3}$$

and, letting N denote the hypersingular operator

$$N(\mu)(\mathbf{r}) = \lim_{z \to 0} \frac{\partial}{\partial \mathbf{n}_{\mathbf{r}}} \int_{\Gamma} \frac{\partial G_k(\mathbf{r}, \mathbf{r}'_z \mathbf{n}_{\mathbf{r}'})}{\partial \mathbf{n}'_{\mathbf{r}}} \mu(\mathbf{r}') dl', \tag{B.4}$$

the density μ is the unique solution of the first kind integral equation

$$N(\mu) = g. \tag{B.5}$$

Once the unknown surface density μ has been obtained, the solution $\hat{v}$ of Equation (B.1) can be produced at any point outside Γ by applying numerical quadrature to Equation (B.2).

Our numerical solver produces approximate solutions μ of Equation (B.5) for a given right- hand side g by seeking a set of values $\mu_j \approx \mu(\mathbf{r}_j)$ of the unknown μ at a set of points $\mathbf{r}_j, (j = 1, ..., n)$ on the curve Γ. The algorithm relies on a highly accurate approximation of the integral Equation (B.5) which can be obtained by appealing to an expression of the right-hand side of Equation (B.4) that only uses tangential derivatives, in conjunction with interpolation of the values μ_j by trigonometric polynomials and exact differentiation and integration of trigonometric monomials. In order to resolve

high curvatures while taking advantage of the excellent properties of trigonometric interpolation, the method utilizes smooth changes of variables that map an equi-spaced grid in the interval $[0, 2\pi]$ to a grid on Γ that contains a high density of discretization points $\mathbf{r}_j$ in high curvature portions of Γ. Note, in particular, that as a result of this discretization strategy, the density of discretization points $\mathbf{r}_j$ varies smoothly along Γ. The method then proceeds by constructing a linear system of equations for the quantities μ_j, which arises as the discretized version of the left-hand-side in Equation (B.5) is set to equal the right hand side of that equation at each point $\mathbf{r}_j, j = 1, ..., n$. The algorithm is then completed by solving this linear system by means of a numerical implementation of the Gaussian elimination method. It was verified through a variety of numerical experiments, including comparisons with exact solutions, that the solutions $\hat{v}$ produced by this methodology are highly accurate, and that the associated errors decay rapidly as discretizations are refined. Numerical considerations and experiments relevant to the specific geometries considered for the through-holes in this thesis follow.

B.2 Geometric considerations

In order to construct the non-circular boundaries such as those of the rectangular hole geometry (also called a "slot" in [53]), we employ a piecewise parametric geometry of Figure B.1, where w, h and r denote the slot width, height and corner radius, respectively. Parameterized as $(x(t), y(t))$ for $t \in [0, 2\pi]$, this geometry is piecewise continuous with respect to t. In order to smooth the parameterization, we employ the C^∞ partition of unity $P(t, t_0)$ given in (3.53) (introduced in Section 3.3.4) centered at a piecewise junction point t_0. For example, if $x_1(t)$ is vertical line segment given by (1) in Figure (B.1) and $x_2(t)$ the circular arc segment given by (2) in Figure (B.1), then a combined parameterization that is C^∞ in the neighborhood of t_0 connecting $x_1(t)$ to $x_2(t)$ smoothly is constructed simply as

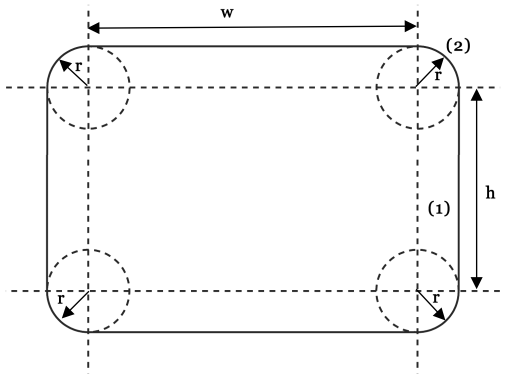


Figure B.1: *The 2D parametric slot geometry for use in the Helmholtz solver.*

$$\begin{aligned} x_{12}(t) &= x_1(t)P_1(t,t_0) + x_2(t)P_2(t,t_0) \\ &= x_1(t)P_1(t,t_0) + x_2(t)(1 - P_1(t,t_0)), \end{aligned} \tag{B.6}$$

and similarly for all other piecewise junctions (for a total of eight applications of the functions constructed on the basis of a partition of unity.)

B.3 Physical verification and a convergence study

As a physical verification of the slot geometry and other code modifications, we consider the behavior of the far field as we thin the height h of the slot and successively sharpen the corners—the geometry and subsequent solution approaches that of an open-arc strip. For the Dirichlet problem with an incident plane wave of frequency $k = 2\pi/\lambda = 1$ in the $\theta = \pi/2$ direction (measured from the positive x-axis), we considered geometries of slot width $w = 2.0$ and decreasing values of h, r and the numbers N of discretization points used to represent the boundary of the hole. The results we obtain are given in Figure B.2. Since the strip is not a closed curve and is hence treated by a completely different algorithm and code than that of the closed slot, the behavior observed in Figure B.2 is a strong indicator of the correctness of both the slot simulations and the strip.

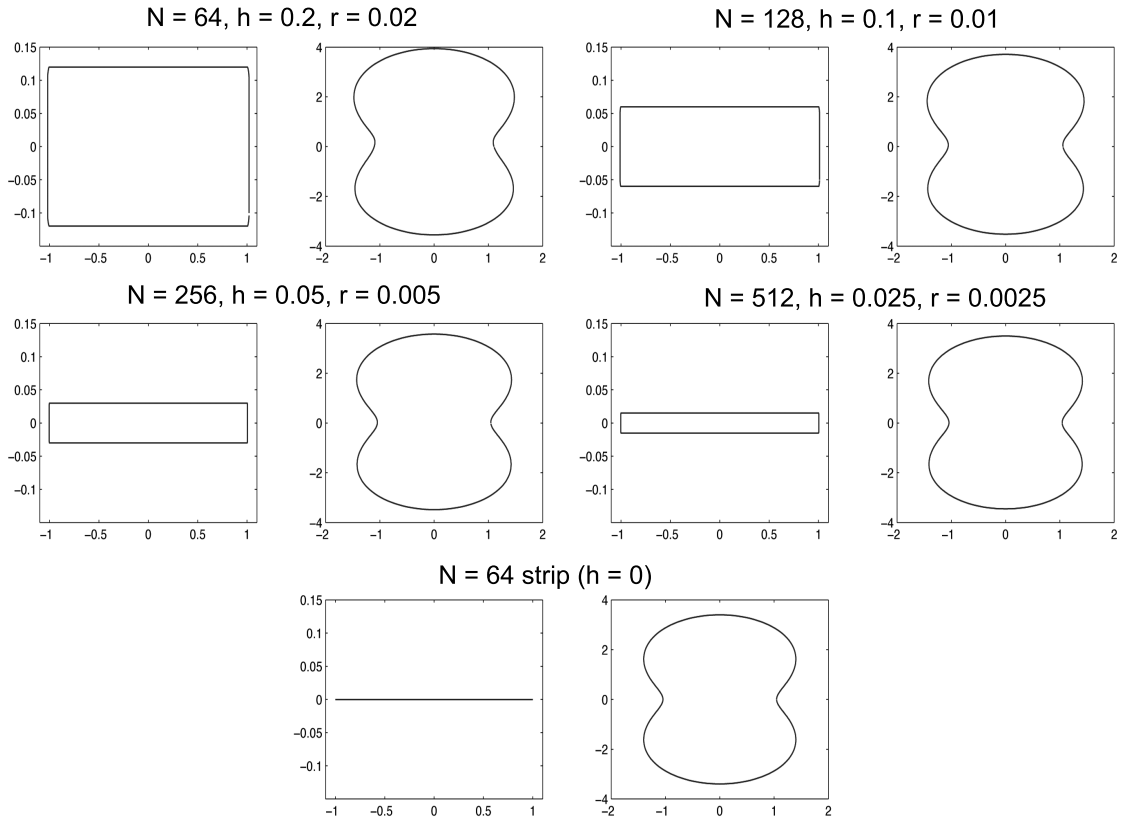


Figure B.2: *Results of the far field (right) as the slot height (left) is reduced and the corners are successively sharpened. The last strip is that of an open arc.*

For a numerical convergence test of the smoothed slot, we also solved a Neumann problem with specifications $w = 25, h = 5, r = 1$ and an incident wave of frequency $k = 2\pi/\lambda = 2.0$ and a direction that is perpendicular to the the base of the slot. The corresponding solution using $N = 2048$ discretization points is given in Figure B.3. By considering the "exact" solution to be a very fine solution with $N = 2048$ discretization points solved by using a GMRES tolerance of 10^{-8}, the errors in the surface density for various grid sizes N are given in Table B.1 and confirm the high-order convergence expected by [19].

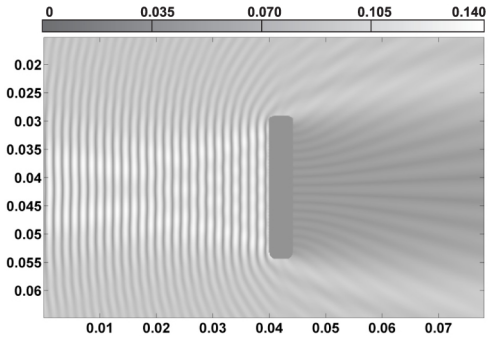


Figure B.3: *Time-harmonic scattering in 2D using the acoustic scalar equation and Neumann boundary conditions.*

N	Iterations	Surface Density Error	$O(L^\infty)$	Farfield Error	$O(L^\infty)$
256	31	3.17e-1	–	3.92e-1	–
512	28	1.5e-4	11.04	1.76e-5	14.44
1024	34	3.95e-7	8.56	3.77e-9	12.19

Table B.1: *Surface density errors for the slot-through hole in 2D relative to an"exact" solution using $N = 2048$ discretization points. All errors are absolute errors of the normalized solutions (*$\max v(\mathbf{r}) = 1$*.)*

Appendix C

Parameters for the NDE applications

This section provides the relevant parameters employed both experimentally and numerically in the ultrasonic NDE study of Section 4.2. They are provided most generously by our collaborators at the University of Vigo.

The regions of interest, including locations of the centers for the $4, 6, 8$ and 12mm diameter holes and locations of the "virtual point sources" used in the computation of the reflection coefficients, are given in Table C.1. The parameters for the incident field boundary condition (applied at the left end of the plate domains) are determined by least squares fitting to the function in (4.7) from experimentally measured wave fields—the corresponding parameters are given in Table C.2.

4mm hole		
Hole center (m)	Incident center (m)	Plate dimensions (m)
$x = +0.0352848170783922$	$x = -0.0920467705004080$	$x \in [+0.003115, +0.144261]$
$y = +0.0000000000000000$	$y = +0.0000000000000000$	$y \in [-0.005000, +0.005000]$
$z = +0.0368944522207174$	$z = +0.0355228789057980$	$z \in [-0.033679, +0.107468]$

6mm hole		
Hole center (m)	Incident center (m)	Plate dimensions (m)
$x = +0.0349233069781789$	$x = -0.0980871739352740$	$x \in [+0.003184, +0.142885]$
$y = +0.0000000000000000$	$y = +0.0000000000000000$	$y \in [-0.005000, +0.005000]$
$z = +0.0367112495679541$	$z = +0.0368581952699390$	$z \in [-0.033139, +0.106561]$

8mm hole		
Hole center (m)	Incident center (m)	Plate dimensions (m)
$x = +0.0350530250350388$	$x = -0.0943896100069270$	$x \in [+0.003109, +0.143321]$
$y = +0.0000000000000000$	$y = +0.0000000000000000$	$y \in [-0.005000, +0.005000]$
$z = +0.0345209925415568$	$z = +0.0359534882442510$	$z \in [-0.035585, +0.104627]$

12mm hole		
Hole center (m)	Incident center (m)	Plate dimensions (m)
$x = +0.0316906421926259$	$x = -0.0911821508197890$	$x \in [+0.003153, +0.138714]$
$y = +0.0000000000000000$	$y = +0.0000000000000000$	$y \in [-0.005000, +0.005000]$
$z = +0.0333739830636900$	$z = +0.0321707608700040$	$z \in [-0.034407, +0.101155]$

Table C.1: *Domains for circular hole geometries. The boundary condition for the incident field is applied to the left end of the interval for x given in the third column.*

4mm hole		
Param	Val	Units
λ_0	5.320000000000000e-7	m
A	2.000000000000000	$1/rad$
ϕ	-0.799360797413403	rad
b_1	0.459412330133725	rad
b_2	4.887614084715763e-6	s
b_3	2.820497732761213e-6	s
c_1	-6.160880183183645e6	rad/s
c_2	1.536282143904974	rad
p_1	3.492694217354267e-5	$rad \cdot m^2$
q_1	-0.072367764107921	m
q_2	0.001379307613937	m^2
e_1	8.748372259554402e3	rad/m^2
e_2	-6.072331920950221e2	rad/m
e_3	-1.281700922092168	rad
z_1	0.017136100000000	m
z_2	0.055311000000000	m
δ_z	0.02000000000000	m
t_1	9.589705300000000e-7	s
t_2	8.805463000000000e-6	s
δ_t	3.000000000000000e-6	s

6mm hole		
Param	Val	Units
λ_0	5.320000000000000e-7	m
A	2.46000000000000	$1/rad$
ϕ	-2.356194490192345	rad
b_1	0.372518544795628	rad
b_2	5.101300095547966e-6	s
b_3	2.777474091707908e-6	s
c_1	-6.164594602460224e6	rad/s
c_2	3.143491407154799	rad
p_1	3.428715293151852e-5	$rad \cdot m^2$
q_1	-0.074630510172381	m
q_2	0.001476691328839	m^2
e_1	8.857125408559656e3	rad/m^2
e_2	-6.513080868238072e2	rad/m
e_3	1.772661853946239	rad
z_1	0.017943100000000	m
z_2	0.056249900000000	m
δ_z	0.020000000000000	m
t_1	1.228012900000000e-6	s
t_2	8.960743800000000e-6	s
δ_t	3.000000000000000e-6	s

8mm hole		
Param	Val	Units
λ_0	5.320000000000000e-7	m
A	4.225000000000000	$1/rad$
ϕ	-1.958433953662837	rad
b_1	0.300505414282102	rad
b_2	5.800367091934857e-6	s
b_3	2.838108720820895e-6	s
c_1	-6.131563983879754e6	rad/s
c_2	7.041752163228258	rad
p_1	3.349255095589419e-5	$rad \cdot m^2$
q_1	-0.073553204209878	m
q_2	0.001493681123397	m^2
e_1	8.882591086403834e3	rad/m^2
e_2	-6.298101881291292e2	rad/m
e_3	0.642415443185962	rad
z_1	0.016958100000000	m
z_2	0.058668100000000	m
δ_z	0.02000000000000	m
t_1	1.926480500000000e-6	s
t_2	9.620282100000000e-6	s
δ_t	3.000000000000000e-6	s

12mm hole		
Param	Val	Units
λ_0	5.320000000000000e-7	m
A	2.55000000000000	$1/rad$
ϕ	-0.916297857297023	rad
b_1	0.388231783246028	rad
b_2	4.972084318183897e-6	s
b_3	2.703572308541960e-6	s
c_1	-6.112965306422946e6	rad/s
c_2	1.483813897899920	rad
p_1	2.748965532303669e-5	$rad \cdot m^2$
q_1	-0.062987375715764	m
q_2	0.0010631149558029	m^2
e_1	8.444680754430321e3	rad/m^2
e_2	-5.359665196814000e2	rad/m
e_3	3.202536087625343	rad
z_1	0.010000000000000	m
z_2	0.055000000000000	m
δ_z	0.02000000000000	m
t_1	0.000000000000000	s
t_2	9.000000000000000e-6	s
δ_t	3.000000000000000e-6	s

Table C.2: *Parameters for* (4.7) *fitting the experimental incident fields.*

Appendix D

Technical notes

D.1 On the continuation precomputation:

As described in Section 3.1.1, Fourier continuations are precomputed as extensions of a Gram polynomial basis for use in an FC-based solver. Although most code for the numerical experiments of this work is developed in the C++ environment, MATLAB with its Variable Precision Arithmetic (VPA) library of the Symbolic Toolbox is used to construct both the Dirichlet and Neumann operators with 256 digits in order to address the possible numerical loss of orthogonality inherent to ill-conditioned systems. To avoid pre-constructing Fourier continuation information at the beginning of a simulation, a library for a wide range of potential discretizations and numbers of matching points d are created and stored in file to enable a simulation to just load from file the requisite operators on the basis of the discretization size of the computational domain.

D.2 On overlapping patches and interpolation:

Given a point $(x_0, y_0, z_0) \in \Omega_i$, an adjacent donor patch Ω_j is found that contains (x_0, y_0, z_0) at its most interior, and a corresponding $n_s \times n_s \times n_s$ stencil is located (stored by its lower left-hand corner index) in patch Ω_j to have the corresponding interpolation point in patch Ω_i as close to its center as possible. The interpolation point and its stencil

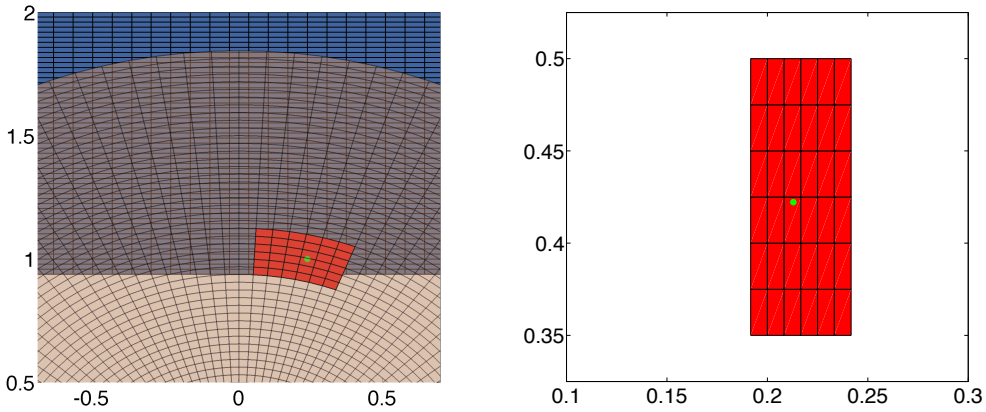


Figure D.1: *Left: Two-dimensional example of an overlapping patch in the x-y plane, where the red grid represents the interpolation stencil in an annular patch and the green dot represents an interpolation point in the rectangular patch. Right: The stencil and interpolation point in the corresponding q-r plane of the annular donor patch where the computation is performed.*

in the donor patch is then recast in the local curvilinear coordinates $(q, r, s) \in [0, 1]^3$ of the donor Ω_j, where the corresponding coordinates of the interpolation point (x_0, y_0, z_0) are found in the donor coordinate system via a multidimensional Newton-Raphson Method [25] applied as

$$\begin{pmatrix} q_{n+1} \\ r_{n+1} \\ p_{n+1} \end{pmatrix} = \begin{pmatrix} q_n \\ r_n \\ p_n \end{pmatrix} - \left(\mathcal{J}_j(q_n, r_n, s_n)\right)^{-1} \begin{pmatrix} x(q_n, r_n, s_n) - x_0 \\ y(q_n, r_n, s_n) - y_0 \\ z(q_n, r_n, s_n) - z_0 \end{pmatrix}, \tag{D.1}$$

for coordinates (x, y, z) and the Jacobian $\mathcal{J}_j$ corresponding to the curvilinear mapping of Ω_j. Interpolation is then simply performed (locally) in the uniformly discretized parameter space (q, r, s) of patch Ω_j via polynomial interpolation applied dimension-by-dimension through a Neville's algorithm [62]. A visual interpretation of this interpolation scheme in both coordinate systems is given by Figure D.1.

A mask corresponding to every point in the mesh is also output, with a value of 1 symbolizing a solution point, a value of -1 an interpolation point, and value of 0 an

unused point in the mesh (i.e. if a hole is cut in the grid). The elasticity solver inputs the mask for each curvilinear grid, and determines the size and starting/ending indices of each segmented FC block for computation. In practice, a size of 10 to 15 points is necessary for stability and the lack of control for this parameter that is intrinsic to most extant hole-cutting algorithms needs to be addressed.

D.3 On the FFT:

The FFTW3 library [35] is called to compute FFTs in code. Although shown to have relatively good complexity for any length N of the discrete FFT, sizes containing very small prime factors have been demonstrated to give a (sometimes significant) improvement on the number of CPU-cycles needed to execute an FFT [4]. Because the Fourier continuation algorithm of biased-order works by blending the left and right sides of a continued block (where each side is respectively continued to zero), an arbitrary number of zeros–with little effort–can be added to the continuation regions for blending to allow an FFT of arbitrary size. As such, each FFT block (of $N + C$ points, where N is the number of points used for the original function and C is the number of continuation points) is padded with enough zeros in the continuation regime that the overall block size for the FFT respects this optimized requirement. That is, one can look for two powers p_1, p_2 such that

$$\min_{p_1,p_2} 2^{p_1} 3^{p_2} \geq N + C$$

so that each length is padded by

$$l_{pad} = (N + C) - 2^{p_1} 3^{p_2}.$$

In general, for most simulations / clusters described in this work, FC blocks tend to be of size $(N+C) < 64$ and hence, in general, pad lengths are such that $N+C+l_{pad} = 64$.

Remark 7. *Our implementation of the elasticity equations involves the* real-*valued displacements given by u, v and w. As such, we have found it convenient to work with real coefficients and values only in code, and hence we subsequently construct our FC(Gram) operators on the basis of an alternative form of* (3.1) *given by*

$$f^c(x) = \sum_{k=-M}^{M} a_k \cos(2\pi kx) + b_k \sin(2\pi kx), \tag{D.2}$$

and compute FFTs of the corresponding continued function by means of the "halfcomplex" real-to-real Discrete Fourier Transform (DFT) provided by the FFTW library. This enables faster and more convenient Fourier transforms by restricting the output of an FFT call to the non-redundant half of the complex output for a real-input DFT–that is, the real and imaginary parts of a transform are returned in a real-valued array thus allowing for derivative computations to be conducted by real-valued coefficients.

D.4 On parallel implementation:

The developed code employs a custom data structure to store displacements (u, v, w) on a three dimensional mesh that enables contiguous blocks of memory to be accessed and packaged for communication in both the domain decomposition and interpolation routines (the latter requires a preprocessing step which we described in Section 3.4.3).

A majority of the numerical experiments for this work were run on a Poweredge cluster maintained by the group of Prof Oscar Bruno and consisting of 32 compute nodes, each equipped with two eight-core Intel Xeon 2.4 GHz processors (such that there are 16 cores per node) and 64 GB of RAM. Intel compilers were used on this system to take advantage of the architecture; this includes the Intel MPI and the Intel Math Kernel libraries (for FFTW3).

Other experiments were carried out on the Shared Heterogeneous Cluster (SHC) at Caltech's Center of Advanced Computing Research. The (floating) specifications of the system consist of 300-400 AMD Opteron 2.2 GHz cores for a total of more

than 700+ GB of RAM. Both this cluster and the one above operate on an Infiniband networking fabric that provides rapid and efficient communication between nodes.

Bibliography

[1] Abarbanel, S., Gottlieb, D., and Carpenter, M. H. On the removal of boundary errors caused by Runge–Kutta integration of nonlinear partial differential equations. *SIAM Journal on Scientific Computing 17*, 3 (1996), 777–782.

[2] Achenbach, J. D. *Wave Propagation in Elastic Solids*. Elsevier, 1974.

[3] Achenbach, J. D. *Reciprocity in Elastodynamics*. Cambridge, UK: Cambridge University Press, 2003.

[4] Albin, N., and Bruno, O. P. A spectral FC solver for the compressible Navier-Stokes equations in general domains i: Explicit time-stepping. *J. Comput. Phys. 230*, 16 (July 2011), 6248–6270.

[5] Albin, N., Bruno, O. P., Cheung, T. Y., and Cleveland, R. O. Fourier continuation methods for high-fidelity simulation of nonlinear acoustic beams. *The Journal of the Acoustical Society of America 132* (2012), 2371.

[6] Appelo, D., and Colonius, T. A high-order super-grid-scale absorbing layer and its application to linear hyperbolic systems. *J Comput Phys 228*, 11 (06 2009), 18–18.

[7] Appelö, D., and Petersson, N. A. A stable finite difference method for the elastic wave equation on complex geometries with free surfaces. *Communications in Computational Physics 5*, 1 (2009), 84–107.

[8] Babuska, I. M., and Sauter, S. A. Is the pollution effect of the FEM avoidable for the Helmholtz equation considering high wave numbers? *SIAM Review 42*, 3 (09 2000), 451–484.

[9] Bannister, S., Husebye, E., and Ruud, B. Teleseismic P coda analyzed by three-component and array techniques: deterministic location of topographic P-to-Rg scattering near the NORESS array. *Bulletin of the Seismological Society of America 80*, 6B (1990), 1969–1986.

[10] Bashforth, F., and Adams, J. *An Attempt to Test the Theories of Capillary Action: By Comparing the Theoretical and Measured Forms of Drops of Fluid. With an Explanation of the Method of Integration Employed in Constucting the Tables which Give the Theoretical Forms of Such Drops.* Cambridge, UK: Cambridge University Press, 1883.

[11] Birks, A., and Green, R. Ultrasonic Testing (Nondestructive Testing Handbook). *American Society for Nondestructive Testing* (1991), 223–224.

[12] Bond, L. J. Numerical techniques and their use to study wave propagation and scattering-a review. *Elastic Waves and Ultrasonic Nondestructive Evaluation* (1990), 17–27.

[13] Boore, D. M. A note on the effect of simple topography on seismic SH waves. *Bulletin of the Seismological Society of America 62*, 1 (1972), 275–284.

[14] Bouchon, M., Schultz, C. A., and Toksöz, M. N. Effect of three-dimensional topography on seismic motion. *Journal of Geophysical Research 101*, B3 (1996), 5835–5846.

[15] Boyd, J. P., and Ong, J. R. Exponentially-convergent strategies for defeating the Runge phenomenon for the approximation of non-periodic functions, part i: single-interval schemes. *Commun. Comput. Phys. 5*, (2-4) (2009).

[16] Brown, D. L., Henshaw, W. D., and Quinlan, D. J. Overture: An object-oriented framework for solving partial differential equations on overlapping grids. *Object Oriented Methods for Interoperable Scientific and Engineering Computing, SIAM* (1999), 245–255.

[17] Bruno, O. P. Fast, high-order, high-frequency integral methods for computational acoustics and electromagnetics. *Topics in Computational Wave Propagation Direct and Inverse Problems Series, Lecture Notes in Computational Science and Engineering 31* (2003).

[18] Bruno, O. P., Han, Y., and Pohlman, M. M. Accurate, high-order representation of complex three-dimensional surfaces via Fourier continuation analysis. *J. Comput. Phys. 0021-9991 227*, 2 (2007), 1094–1125.

[19] Bruno, O. P., and Lintner, S. K. Secondkind integral solvers for TE and TM problems of diffraction by open arcs. *Radio Science 47*, 6 (2012).

[20] Bruno, O. P., and Lyon, M. High-order unconditionally stable FC-AD solvers for general smooth domains i. basic elements. *Journal of Computational Physics 229*, 6 (2010), 2009 – 2033.

[21] Bruno, O. P., and Prieto, A. Spatially dispersionless, unconditionally stable fc–ad solvers for variable-coefficient pdes. *Journal of Scientific Computing* (2013), 1–36.

[22] Bushnell, J. C., and McCloskey, D. J. Thermoelastic stress production in solids. *J. Appl. Phys. 39*, 12 (1968), 5541–5546.

[23] Carpenter, M. H., Gottlieb, D., Abarbanel, S., and Don, W.-S. The theoretical accuracy of Runge–Kutta time discretizations for the initial boundary value problem: a study of the boundary error. *SIAM Journal on Scientific Computing 16*, 6 (1995), 1241–1252.

[24] Chimenti, D. Guided waves in plates and their use in materials characterization. *Applied Mechanics Reviews 50* (1997), 247.

[25] Dahlquist, G., and Björck, Å. *Numerical Methods*. Dover, 1974.

[26] Davis, J. L. Wave propagation in solids and fluids. *New York, Springer-Verlag, 1988, 396 p. 1* (1988).

[27] De La Puente, J., Käser, M., Dumbser, M., and Igel, H. An arbitrary highorder discontinuous Galerkin method for elastic waves on unstructured meshes–iv. anisotropy. *Geophysical Journal International 169*, 3 (2007), 1210–1228.

[28] Deschamps, M. *Ultrasonic wave propagation in non homogeneous media*, vol. 128. Springer, 2009.

[29] Dumbser, M., and Käser, M. An arbitrary highorder discontinuous galerkin method for elastic waves on unstructured meshes–ii. the threedimensional isotropic case. *Geophysical Journal International 167*, 1 (2006), 319–336.

[30] Elling, T. *GPU-accelerated Fourier-continuation solvers and physically exact computational boundary conditions for wave scattering problems*. PhD thesis, California Institute of Technology, 2012.

[31] Eriksson, L. Practical three-dimensional mesh generation using transfinite interpolation. *SIAM Journal on Scientific and Statistical Computing 6*, 3 (1985), 712–741.

[32] Faccioli, E., Maggio, F., Paolucci, R., and Quarteroni, A. 2d and 3d elastic wave propagation by a pseudo-spectral domain decomposition method. 237–251.

[33] Farrashkhalvat, M., and Miles, J. P. *Basic Structured Grid Generation: With an introduction to unstructured grid generation*. Elsevier Science, 2003.

[34] Fernández, J., Doval, Á. F., Trillo, C., Deán, J. L., and López, J. Video ultrasonics by pulsed tv holography: A new capability for non-destructive testing of shell structures. *International Journal of Optomechatronics 1*, 2 (2007), 122–153.

[35] Frigo, M., and Johnson, S. G. The design and implementation of fftw3. *Proceedings of the IEEE 93*, 2 (Feb. 2005), 216–231.

[36] Fung, Y.-c. *Foundations of solid mechanics*. Prentice Hall, 1965.

[37] Gilbert, F., and Knopoff, L. Seismic scattering from topographic irregularities. *Journal of Geophysical Research 65*, 10 (1960), 3437–3444.

[38] Gockenbach, M. . *Partial Differential Equations: Analytical and Numerical Methods, Second Edition*. Society for Industrial and Applied Mathematics (SIAM, 3600 Market Street, Floor 6, Philadelphia, PA 19104), 2010.

[39] Graff, K. F. *Wave Motion in Elastic Solids*. Dover Publications, Inc., 1975.

[40] Green Jr, R. E. Non-contact ultrasonic techniques. *Ultrasonics 42*, 1 (2004), 9–16.

[41] H., M., Saidu, I., and Waziri, M. Y. A simplified derivation and analysis of fourth order Runge–Kutta method. *International Journal of Computer Applications 9*, 8 (November 2010), 51–55. Published By Foundation of Computer Science.

[42] Harris, J. G. *Linear Elastic Waves*. Cambridge University Press, 2001.

[43] Henshaw, W. D., and Schwendeman, D. W. Parallel computation of three-dimensional flows using overlapping grids with adaptive mesh refinement. *Journal of Computational Physics 227*, 16 (2008), 7469–7502 0021–9991.

[44] Jaklic, A., Leonardis, A., and Solina, F. *Segmentation and Recovery of Superquadrics*. Springer, 2000.

[45] Janod, F., and Coutant, O. Seismic response of threedimensional topographies using a timedomain boundary element method. *Geophysical Journal International 142*, 2 (2000), 603–614.

[46] Käser, M., and Dumbser, M. An arbitrary highorder discontinuous galerkin method for elastic waves on unstructured meshes–i. the twodimensional isotropic case with external source terms. *Geophysical Journal International 166*, 2 (2006), 855–877.

[47] Käser, M., Dumbser, M., De La Puente, J., and Igel, H. An arbitrary highorder Discontinuous Galerkin method for elastic waves on unstructured meshes–iii. viscoelastic attenuation. *Geophysical Journal International 168*, 1 (2007), 224–242.

[48] Knupp, P., and Steinberg, S. *Fundamentals of Grid Generation*. Ann Arbor, MI: CRC Press, 1994.

[49] Komatitsch, D., and Vilotte, J.-P. The spectral element method: An efficient tool to simulate the seismic response of 2D and 3D geological structures. *Bulletin of the Seismological Society of America 88*, 2 (1998), 368–392.

[50] Kozdon, J. E., Dunham, E. M., and Nordström, J. Simulation of dynamic earthquake ruptures in complex geometries using high-order finite difference methods. *Journal of Scientific Computing* (2013), 1–33.

[51] Laguerre, L., Grimault, A., and Deschamps, M. Ultrasonic transient bounded-beam propagation in a solid cylinder waveguide embedded in a solid medium. *The Journal of the Acoustical Society of America 121* (2007), 1924.

[52] Lamb, H. On waves in an elastic plate. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character 93*, 648 (1917), 114–128.

[53] López-Vázquez, J. C., Deán-Ben, X. L., Trillo, C., Doval, Á. F., Fernández, J., Amlani, F., and Bruno, O. P. Numerical modeling and measurement by pulsed television holography of ultrasonic displacement maps in plates with through-thickness defects. *Optical Engineering 49*, 9 (09 2010), 095802–095802.

[54] Lyon, M., and Bruno, O. P. High-order unconditionally stable FC-AD solvers for general smooth domains ii. elliptic, parabolic and hyperbolic PDEs; theoretical considerations. *J. Comput. Phys. 229*, 9 (May 2010), 3358–3381.

[55] Mindlin, R. D. Mathematical theory of vibrations of elastic plates. *Proceedings of the 11th Annual Symposium on Frequency Control, Army Signal Engineering Laboratories* (1957), 1–40.

[56] Mindlin, R. D. Waves and vibrations in isotropic elastic plates. *Structural Mechanics* (1960), 199–232.

[57] Mindlin, R. D. *An Introduction to the Mathematical Theory of Vibrations of Elastic Plates*. Singapore: World Scientific, 2006.

[58] Monro, J. A. *A super-algebraically convergent, windowing-based approach to the evaluation of scattering from periodic rough surfaces*. PhD thesis, California Institute of Technology, 2008.

[59] Moser, F., Jacobs, L. J., and Qu, J. Modeling elastic wave propagation in waveguides with the finite element method. *NDT & E International 32*, 4 (1999), 225–234.

[60] Mullen, R., and Belytschko, T. Dispersion analysis of finite element semidiscretizations of the twodimensional wave equation. *International Journal for Numerical Methods in Engineering 18*, 1 (1982), 11–29.

[61] Nakoriakov, V., Pokousaev, B. G., reiber, I. R., and Bergles, A. *Wave Propagation in Gas Liquid Media*. CRC Press, 1993.

[62] Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*, 3 ed. Cambridge University Press, New York, NY, USA, 2007.

[63] Raghavan, A., and Cesnik, C. E. Review of guided-wave structural health monitoring. *Shock and Vibration Digest 39*, 2 (2007), 91–116.

[64] Rayleigh, L. On the free vibrations of an infinite plate of homogeneous isotropic elastic matter. *Proceedings of the London Mathematical Society 20* (1889).

[65] Ready, J. F. Effects due to absorption of laser radiation. *J. Appl. Phys. 36*, 2 (1965), 462–468.

[66] Roache, P. J. Code verification by the method of manufactured solutions. *Transactions-American Society of Mechanical Engineers Journal of Fluids Engineering 124*, 1 (2002), 4–10.

[67] Rodríguez-Gómez, P., López-Vázquez, J. C., Trillo, C., Doval, Á. F., and Fernández, J. L. Transient elastic wave propagation and scattering in plates: Comparison between pulsed TV–holography measurements and finite element method predictions, 2012.

[68] Rose, J. L. Guided wave nuances for ultrasonic nondestructive evaluation. *Ultrasonics, Ferroelectrics and Frequency Control, IEEE Transactions on 47*, 3 (2000), 575–583.

[69] Roy, C. J., Smith, T. M., and Ober, C. C. Verification of a compressible CFD code using the method of manufactured solutions. *Energy 2* (2002), s2.

[70] Scruby, C. B., and Drain, L. E. *Laser Ultrasonics: Techniques and Applications*. CRC Press, 1990.

[71] Shakal, A. CSMIP strong-motion records from the Northridge, California earthquake of January 17 1994.

[72] Sjögreen, B., and Petersson, N. A. A fourth order accurate finite difference scheme for the elastic wave equation in second order formulation. *Journal of Scientific Computing 52*, 1 (2012), 17–48.

[73] Spekreijse, S. P. Elliptic grid generation based on Laplace equations and algebraic transformations. *Journal of Computational Physics 118*, 1 (1995), 38 – 61.

[74] Su, Z., Ye, L., and Lu, Y. Guided Lamb waves for identification of damage in composite structures: A review. *Journal of Sound and Vibration 295*, 3 (2006), 753–780.

[75] Tatel, H. E. Note on the nature of a seismogram: II. *Journal of Geophysical Research 59*, 2 (1954), 289–294.

[76] Timoshenko, S. P., and Goodier, J. Theory of elasticity. *International Journal of Bulk Solids Storage in Silos 1*, 4 (2011), 567–567.

[77] Vedovoto, J. M., Silveira Neto, A. D., Mura, A., and Figueira da Silva, L. F. Application of the method of manufactured solutions to the verification of a pressure-based finite-volume numerical scheme. *Computers & Fluids 51*, 1 (2011), 85–99.

[78] Viktorov, I. A. *Rayleigh and Lamb Waves: Physical Theory and Applications*, vol. 147. Plenum press New York, 1967.

[79] Virieux, J. P-SV wave propagation in heterogeneous media: Velocity–stress finite–difference method. *Geophysics 51*, 4 (1986), 889–901.

[80] White, R. M. Elastic wave generation by electron bombardment or electromagnetic wave absorption. *J. Appl. Phys. 34*, 7 (1963), 2123–2124.

[81] Whitham, G. B. *Linear and Nonlinear Waves*, vol. 42. John Wiley & Sons, 2011.