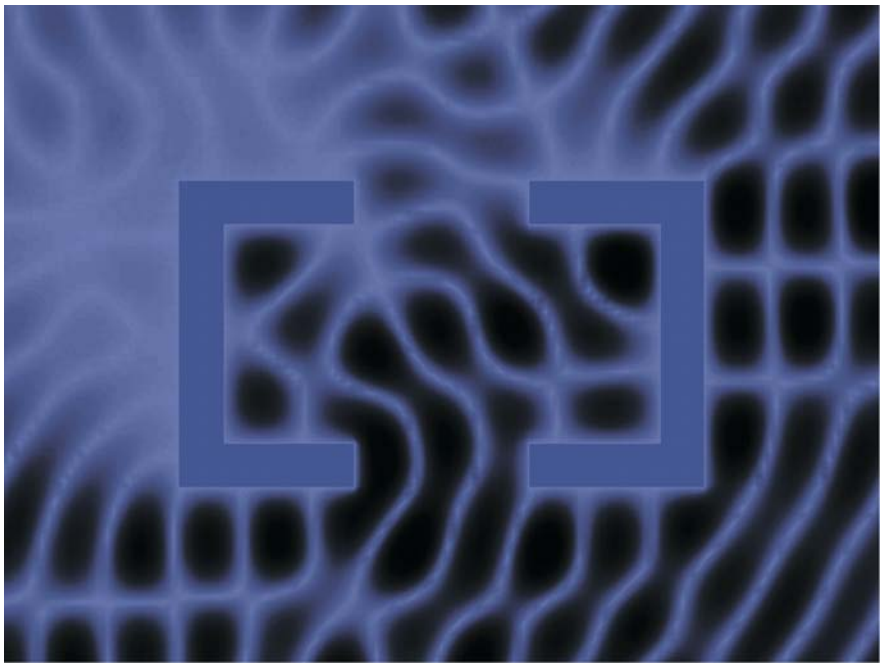
JYVÄSKYLÄ LICENTIATE THESES IN COMPUTING
10

Sanna Mönkölä

Spectral Element Method and Controllability Approach for Time-harmonic Wave Propagation



UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ LICENTIATE THESES IN COMPUTING 10

Sanna Mönkölä

Spectral Element Method and Controllability Approach for Time-harmonic Wave Propagation

UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2008

Spectral Element Method and Controllability Approach for Time-harmonic Wave Propagation

JYVÄSKYLÄ LICENTIATE THESES IN COMPUTING 10

Sanna Mönkölä

Spectral Element Method and Controllability Approach for Time-harmonic Wave Propagation

UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2008

Editor
Timo Männikkö
Department of Mathematical Information Technology, University of Jyväskylä

ISBN 978-951-39-3202-2 (PDF)

ISBN 978-951-39-3131-5 **(nid.)**

ISSN 1795-9713

Jyväskylä University Printing House, Jyväskylä 2008

ABSTRACT

Mönkölä, Sanna
Spectral element method and controllability approach for time-harmonic wave propagation
Jyväskylä: University of Jyväskylä, 2008, 112 p.
(Jyväskylä Licentiate Theses in Computing
ISSN 1795-9713; 10)
ISBN 978-951-39-3202-2 (PDF), 978-951-39-3131-5 (nid.)
Finnish summary

The objective of this work is to develop efficient numerical techniques for solving time-harmonic wave equations. In particular, we consider a controllability technique for the numerical solution of the Helmholtz and the Navier equations. The original time-harmonic equation is represented as an exact controllability problem for the time-dependent wave equation. This problem is then formulated as a least-squares optimization problem, which is solved by conjugate gradient method. Such an approach was first suggested and developed in the 1990s by French researchers and we introduce some improvements to its practical realization. We derive a new way to compute the gradient of the least-squares functional and use algebraic multigrid method for preconditioning the conjugate gradient algorithm. We discretize the wave equation in space domain with higher order spectral elements. The degrees of freedom associated with the basis functions are situated at the Gauss-Lobatto quadrature points of the elements, and the Gauss-Lobatto quadrature rule is used so that the mass matrix becomes diagonal. The method is combined with the second order central finite difference or the fourth order Runge-Kutta time discretization. As a consequence of these choices, only matrix-vector products are needed in time-dependent simulation, which makes this controllability method computationally efficient.

Keywords: exact controllability, linear wave equation, Helmholtz equation, Navier equation, spectral element method

Author

Sanna Mönkölä
Department of Mathematical Information Technology
University of Jyväskylä
Finland

Supervisors

Professor Tuomo Rossi
Department of Mathematical Information Technology
University of Jyväskylä
Finland

Docent Erkki Heikkola
Numerola Oy
Finland

Examiners

Professor Kazufumi Ito
Department of Mathematics
North Carolina State University
USA

Assistant professor Dirk Pauly
Department of Mathematics
University of Duisburg-Essen
Germany

ACKNOWLEDGEMENTS

This licentiate thesis is a result of studies and research at the Department of Mathematical Information Technology of University of Jyväskylä. During this work, funding from various sources was used to make full-time research and international contacts possible. Thus, the financial support from Finnish Foundation for Technology, Jyväskylä Graduate School in Computing and Mathematical Sciences, National Graduate School in Engineering Mechanics, Alfred Kordelin Foundation, Finnish Cultural Foundation, and Ellen and Artturi Nyyssönen Foundation is gratefully acknowledged.

I would like to express my gratitude to professor Tuomo Rossi, Dr. Erkki Heikkola and Dr. Timo Männikkö for their help and useful advice during my research. I am thankful to MSc. Anssi Pennanen and Dr. Janne Martikainen for providing the AMG solver. I would like to thank the examiners, professor Kazufumi Ito and Dr. Dirk Pauly, for carefully reviewing the thesis and giving valuable comments. I would also like to thank my colleagues for their inspiring company and friendship. Finally, I wish to take the opportunity to thank my family and friends.

Jyväskylä, January 2008
Sanna Mönkölä

NOTATIONS

Derivatives	$f'(t,y(t))$	$= \frac{df(t,y(t))}{dt}$
Partial derivatives	$f_{xy}(x,y,z)$	$= \frac{\partial}{\partial y}(\frac{\partial f(x,y,z)}{\partial x})$
	$f^{(i,j)}(x,y)$	$= \frac{\partial^j}{\partial y^j}(\frac{\partial^i f(x,y,z)}{\partial x^i})$
Partial integration	$\int_a^b f'g dx$	$= fg - \int_a^b g'f dx$
Nabla operator	∇	$= e_x\frac{\partial}{\partial x} + e_y\frac{\partial}{\partial y} + e_z\frac{\partial}{\partial z}$, where e_i is unit vector pointing in the direction i of the basis coordinates.
Gradient of u	∇u	$= e_x\frac{\partial}{\partial x} + e_y\frac{\partial}{\partial y} + e_z\frac{\partial}{\partial z}u$ $= \frac{\partial u}{\partial x}e_x + \frac{\partial u}{\partial y}e_y + \frac{\partial u}{\partial z}e_z$ $= (\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}, \frac{\partial u}{\partial z})$
Divergence operator	$\nabla\cdot$	
Divergence of u	$\nabla \cdot u$	$= \left(e_x\frac{\partial}{\partial x} + e_y\frac{\partial}{\partial y} + e_z\frac{\partial}{\partial z}\right) \cdot (u_x e_x + u_y e_y + u_z e_z)$ $= \frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y} + \frac{\partial u_z}{\partial z}$
Laplace operator	$\triangle$	$= \nabla \cdot \nabla$ $= \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$
Laplacian of u	$\nabla^2 u$	$= \nabla \cdot (\nabla u)$ $= \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}$
Double dot product of tensors u and v	$u : v$	$= \sum_i \sum_j u_{ij} v_{ji}$

From the divergence theorem $\int_\Omega \nabla \cdot f \; d\Omega = \int_{\partial\Omega} f \cdot n \; d\Omega$ and by use of the product rule $\nabla \cdot (v\nabla u) = \nabla v \cdot \nabla u + v\nabla^2 u$ we get the Green's first identity

$$\int_{\partial\Omega} v\frac{\partial u}{\partial n} \, dS = \int_\Omega \nabla v \cdot \nabla u \; d\Omega + \int_\Omega v\nabla^2 u \; d\Omega,$$

where $\frac{\partial u}{\partial n} = n \cdot \nabla u$ and n is the unit outward pointing normal vector on the boundary $\partial\Omega$. This is valid for any region Ω and any functions u and v.

LIST OF FIGURES

LIST OF TABLES

CONTENTS

1 INTRODUCTION

Wave equations are partial differential equations which describe the propagation of various types of waves such as acoustic, elastic and electromagnetic waves. They are important in the modelling of important processes and phenomena in almost any field of science and engineering. When small oscillation amplitudes are considered, linear models are allowed. In this study, we focus in particular on two time-harmonic linear wave equations: the scalar valued Helmholtz equation concerning the propagation of acoustic waves and the vector valued Navier equation describing the propagation of waves in an elastic medium. These fundamental equations for time-harmonic wave propagation occur in a number of physical applications such as underwater acoustics, medicine, and geophysics. They can also be used to model the scattering of time-harmonic acoustic waves by an obstacle. In this thesis, we concentrate on scattering problems but the same method can be used for other types of Helmholtz and Navier problems as well.

A wide range of numerical methods have been used for solving time-harmonic wave equations. These methods can be divided into boundary and domain based methods. We are especially interested to solve problems with varying material parameters. For such problems, boundary based methods are not directly applicable whereas domain based methods are more flexible in this respect. Thus, we focus our attention to domain based methods.

Domain based formulations can be discretized, for instance, by finite difference (FDM) or Galerkin finite element methods (FEM) (see, e.g., [48, 57]). Especially the FEM approximation and solution of time-harmonic wave equations has received much attention during the past two decades (see, e.g., [97]). Many efficient solution techniques have been developed for the finite element equations such as domain decomposition methods [7, 8, 17, 23, 30, 33, 34, 77], fictitious domain (domain embedding) methods [6, 32, 52, 53] and multigrid methods [12, 27, 45, 62, 98]. These methods are typically used when the solution is based directly on the complex valued time-harmonic equations and low order finite elements (see e.g. [10, 37, 46]).

In the FEM solution of the time-harmonic wave equation, the discretization mesh needs to be adjusted to the wavelength of the wave. Higher frequencies require finer meshes to reach sufficient accuracy and a typical rule is to keep a fixed number of grid points in a wavelength. This means keeping the quantity κh fixed, where κ is the wavenumber and h the mesh step size. Therefore, high frequency problems often lead to large-scale

linear systems to be solved for which conventional solution method can not be used.

In addition to approximation error, an important consideration in the finite element solution of time-harmonic wave problems is the so called pollution effect (see e.g. [4, 5, 57, 58] and references there in). In [58], it is shown that the relative error of the hp-version [1] of finite element solutions in the H^1 -seminorm consists of two parts. One of these is the approximation error, which is of order $\left(\frac{\kappa h}{2p}\right)^p$ and the other is the pollution error, which is of order $\kappa\left(\frac{\kappa h}{2p}\right)^{2p}$, where p is the order of the basis functions. Consequently, the relative error increases as the wavenumber increases, even if κh is kept constant. The pollution part becomes the dominant source of the relative error at high wavenumbers. It is known that the pollution effect can not be avoided in two- and three-dimensional problems [4]. Thus, fixed error level would require keeping the quantity $\kappa^2 h$ fixed, which leads to unacceptable computational costs for high frequency problems.

To reduce the pollution error, especially in large scale problems, modifications of the classical FEM are needed. These are, for instance, Galerkin generalized least-squares [47], discontinuous enrichment [105], and spectral collocation [20] methods. One way to decrease the pollution effect is to modify the polynomial basis of standard FEM so that the local basis will consist of nonpolynomial shape functions. This is done in discontinuous Galerkin method [21, 31, 35, 36, 60]. Ultra weak variational formulation (UWVF) [19, 55, 56] uses standard finite element meshes and a new kind of variational formulation on the interfaces between the elements. It reduces the memory requirement compared to the standard FEM, but might suffer from numerical instability. Also spectral [18, 86] and collocation methods [5] are used to reduce the pollution effect. Higher order approximations, which are used to reduce the influence of the pollution effect, are considered on a general level, for example, in [93]. We apply specifically the spectral element method (SEM), which is considered in [22, 65]. This method combines the geometric flexibility of finite elements with the high accuracy of spectral methods. The basis functions are higher order Lagrange interpolation polynomials, and the nodes of these functions are placed at Gauss-Lobatto (GL) collocation points. The integrals in the weak form of the equation are evaluated with the corresponding Gauss-Lobatto quadrature formulas. As a consequence of the choice, spectral element discretization leads to diagonal mass matrices which significantly improves the computational efficiency. Moreover, when using higher order elements, same accuracy is reached with less degrees of freedom than when using lower order finite elements.

The discretization and solution methods mentioned above are based on handling directly the time-harmonic equation. They all lead to large-scale discrete problems with indefinite linear equations for which it is difficult to develop efficient iterative methods. Furthermore, the dimension of the system increases rapidly as the wavenumber increases, which makes the accurate solution even more challenging. An alternative is to simulate the time-dependent equation with respect to time until time-harmonic solution is reached (asymptotic approach). However, this approach suffers from poor convergence at least in the case of large wavenumbers and complicated domains. In this thesis, we use the idea of Bristeau, Glowinski, and Périaux, presented in [13, 14, 15, 16, 42], to formulate the time-harmonic wave problem as an exact controllability problem for the time-dependent wave equation. In other words, we try to find such initial conditions that after one time-period

the solution and its time derivative coincide with the initial conditions. This controllability problem is reformulated after discretization as a least squares problem, which is solved with the preconditioned conjugate gradient (CG) algorithm. Computation of the gradient of the function to be minimized is an essential stage of the method. In [16], the gradient was derived on the continuous level, and the same formula was used also on the discrete level. We discretize first the wave equation and the function to be minimized. Then, we compute the gradient directly for the discretized problem.

Previously [50], we have used the central finite difference scheme for time discretization. That scheme is second order accurate and with a diagonal mass matrix also fully explicit in time; the values for each time step are determined from the values of the previous time steps. These are essential properties for computational efficiency. Only matrix-vector products are needed in time-dependent simulation, but the scheme needs to satisfy the CFL condition, which limits the length of the time step (see [22] for details). Now, we compare the central finite difference scheme with the fourth order accurate Runge-Kutta method. While using fourth order Runge-Kutta method, explicity of the method can be maintained with diagonal mass matrices, but still, the method is only conditionally stable (see also [51, 81]).

The rest of the thesis is organized as follows: First, the mathematical formulations of linear wave equations and some preliminaries are considered in Chapter 2. Then, we present Helmholtz and Navier equations for scattering problems in Chapter 4. The formulation of the exact controllability problem is considered in Chapter 5. The discretization of the exact controllability problem is described in Chapter 6. The time discretization is presented in Section 6.3. More precisely, for time discretization we use central finite differences in Section 6.3.1 and fourth order Runge-Kutta scheme in Section 6.3.2. In Chapter 7, we present the least-squares problem and consider its conjugate gradient solution in Chapter 8. Computation of the gradient of the functional, which is an essential point of the method, we have done with the adjoint state technique in Sections 7.2-7.2.2. Algebraic multigrid method [78] (see also [2]) is used for preconditioning the conjugate gradient algorithm (see Section 8.2). Finally, in Chapter 9, we show the performance of the method with numerical experiments.

2 LINEARIZED WAVE EQUATIONS

Assuming that changes in velocity, density and deformation are small, it is possible to describe the propagation of mechanical waves by linear wave equations. In this chapter, we introduce the mathematical model of linearized equations for acoustic and elastic waves in two dimensional isotropic media. Since the material is isotropic, material parameters are independent on the coordinate system and the material has identical properties in all directions at a point. First we derive the Helmholtz equation in Section 2.1 and then the Navier equation in Section 2.2.

2.1 Acoustic wave equation

Fluids are composed of moving and colliding molecules, but the fluid flow is assumed to be a macroscopic flow with properties expressed as averages of the molecular properties. Acoustic waves are small oscillations of pressure, which are associated with local motions of the particles of the fluid. That is why differential equation of acoustic pressures are used for modelling the fluid domain. The acoustic wave equation in fluid media can be derived as a simplification of the basic equations of fluid dynamics [102]. We concentrate on fluids which are incompressible and irrotational. We assume also the fluid to be isothermal, i.e. temperature is constant, which implies constant viscosity. The steady-state values of the velocity and density are denoted by $\mathbf{v}_0 = 0$ and ρ_0. Taking these assumptions into account, we obtain the equation of continuity

$$\frac{\partial \rho_f}{\partial t} + \nabla \cdot (\rho_0 \mathbf{v}_f) = 0, \tag{1}$$

where $\mathbf{v}_f$ is the velocity of fluid particles, t is time, and ρ_f denotes the density of fluid. This equation is known also as the equation of mass conservation telling that mass is neither created nor destroyed within a volume element.

Neglecting the gravity and supposing that spatial distributions of the pressure are negligible, the momentum conservation can be expressed as

$$\nabla p_f = -\rho_0 \frac{\partial \mathbf{v}_f}{\partial t}, \tag{2}$$

where p_f is the pressure. After dividing the equation (2) by ρ_f, approximating $\rho_f \approx \rho_0$, and taking divergence, we get the form

$$\nabla \cdot \left(\frac{1}{\rho_f} \nabla p_f \right) = -\nabla \cdot \frac{\partial \mathbf{v}_f}{\partial t}. \tag{3}$$

Acoustic equation of state in fluid can be written, with the adiabatic condensation $\varkappa = \frac{\partial \rho_f}{\rho_f \partial p_f} = \frac{1}{c^2 \rho_f}$ depending on the speed of sound c, in the form

$$\frac{\partial p_f}{\partial t} = \frac{1}{\varkappa \rho_0} \frac{\partial \rho_f}{\partial t}. \tag{4}$$

Then, we substitute the time derivative of density from equation (1) to the equation of state (4), and get

$$\frac{1}{\rho_f c^2} \frac{\partial p_f}{\partial t} = -\nabla \cdot \mathbf{v}_f. \tag{5}$$

By taking time derivative, we yield to

$$\frac{1}{\rho_f c^2} \frac{\partial^2 p_f}{\partial t^2} = -\nabla \cdot \frac{\partial \mathbf{v}_f}{\partial t}. \tag{6}$$

From equations (3) and (6), we form the wave equation. After adding a body force f, it reads

$$\frac{1}{\rho_f c^2} \frac{\partial^2 p_f}{\partial t^2} - \nabla \cdot \left(\frac{1}{\rho_f} \nabla p_f \right) = f. \tag{7}$$

Substituting $\mathbf{v}_f = \nabla \phi$ to equation (5) and eliminating the pressure in terms of velocity potential ϕ results in the another form of the acoustic wave equation

$$\frac{1}{c^2} \frac{\partial^2 \phi}{\partial t^2} - \nabla \cdot (\nabla \phi) = f_\phi. \tag{8}$$

We are going to solve the time-harmonic problem with spatial variable $\mathbf{x} = (x_1, x_2)^T \in \mathbb{R}^2$. The time-dependence of the pressure is in the form $\exp(i\omega t)$. Replacing $p_f(\mathbf{x}, t)$ by $p_f = P_f(\mathbf{x}) \exp(i\omega t)$ and $f(\mathbf{x}, t)$ by $f = F(\mathbf{x}) \exp(i\omega t)$ in equation (7), and making an assumption of time-harmonic values of frequency ω, we get the time-harmonic Helmholtz equation

$$-\frac{\kappa^2}{\rho_f} P_f - \nabla \cdot \left(\frac{1}{\rho_f} \nabla P_f \right) = F \tag{9}$$

where $\kappa = \frac{\omega}{c}$ is physical parameter, called wavenumber, which describes how many waves there is for a 2π unit. It characterizes the oscillatory behaviour of the solution and increases, if the frequency of the waves increases.

2.2 Elastic wave equation

The linear theory of elasticity [68] models mechanical properties of solids assuming small deformations. With this theory several phenomena, such as seismic waves in the earth and ultrasonic waves used to detect flaws in materials, can be described. In this section, we will present the fundamental equations of the theory of linearized elasticity assuming constant temperature and derive the Navier equation, which governs the propagation of time-harmonic waves in elastic solids [44, 67].

An elastic material responds to an applied force by deforming and returns to its original shape upon the removal of the applied force. Thus, there is no permanent deformation within elastic behaviour. Elasticity can be linear or non-linear. In the linear case, which we concentrate on, the deformation is proportional to the force used to make the deformation. Geometric deformation of the solid is called strain and forces that occur in the solid are described as stresses. In the case of thin solid, we have plane stress situation. If the solid is thick, we can examine plane strain.

In 1678, Robert Hooke, on the basis of experiments with springs, stated a rule between extension and force. This rule, commonly referred to as generalized Hooke's law, says that stress is a linear function of strain, where the stress assumes small displacements, and has the following form:

$$\sigma(\mathbf{u_s}) = C\epsilon(\mathbf{u_s}), \tag{10}$$

where vector field $\mathbf{u_s}$ denotes the displacement, $\sigma(\mathbf{u_s})$ is the stress tensor, C is the elastic moduli tensor and $\epsilon(\mathbf{u_s})$ is strain tensor. In isotropic media, the elastic moduli tensor C is invariant under rotations and reflections.

Diagonal components of $\sigma(\mathbf{u_s})$ present normal stresses and the other components present shear stresses [59]. The strain tensor is defined through the derivatives of the displacement vector $\mathbf{u_s}_i$ such that

$$\epsilon(\mathbf{u_s}_{ik}) = \frac{1}{2}\left(\frac{\partial \mathbf{u_s}_i}{\partial x_k} + \frac{\partial \mathbf{u_s}_k}{\partial x_i} + \frac{\partial \mathbf{u_s}_l}{\partial x_i}\frac{\partial \mathbf{u_s}_l}{\partial x_k}\right). \tag{11}$$

For infinitesimal strains (small deformations), $|\frac{\partial u_i}{\partial x_i}| \ll 1$, the strains are related to the displacements by the linearized strain tensor ϵ, which is defined by

$$\epsilon(\mathbf{u_s}) = \frac{1}{2}\left(\nabla \mathbf{u_s} + (\nabla \mathbf{u_s})^T\right) \tag{12}$$

$$= \begin{pmatrix} \frac{\partial \mathbf{u_s}_1}{\partial x_1} & \frac{1}{2}(\frac{\partial \mathbf{u_s}_1}{\partial x_2} + \frac{\partial \mathbf{u_s}_2}{\partial x_1}) \\ \frac{1}{2}(\frac{\partial \mathbf{u_s}_1}{\partial x_2} + \frac{\partial \mathbf{u_s}_2}{\partial x_1}) & \frac{\partial \mathbf{u_s}_2}{\partial x_2} \end{pmatrix}, \tag{13}$$

where $\nabla \mathbf{u_s}$ is the Jacobian matrix of $\mathbf{u_s}$. Normal strain $\epsilon_{x_i x_i} = \frac{\partial u_i}{\partial x_i}$ is the change of length in x_i-direction divided by the length in x_i-direction and shear strain $\epsilon_{x_i x_j} = \epsilon_{x_j x_i} = \frac{1}{2}(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i})$ tells that the angle between x_i and x_j axis would be diminished by $2\epsilon_{x_i x_j}$.

This approximation does not apply in situations of discontinuities (e.g. deformations involving large angle shears) in properties of the physical medium.

Tensor C is a fourth order tensor of elastic constants of the medium with components $C_{ijkl} = \lambda_s \delta_{ij}\delta_{kl} + \mu_s(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk})$, where δ_{ij} is the Kronecker delta and $i, j, k, l = 1, \ldots, 2$. In general, C_{ijkl} would have $3^4 = 81$ independent components in $\mathbb{R}^3$. Because of symmetry of stress and strain tensors $C_{ijkl} = C_{jikl} = C_{ijlk}$, and $C_{ijkl} = C_{klij}$ by energy considerations. This reduces the number of material constants to 21 in the three-dimensional case and to 6 in the two-dimensional case. For isotropic two dimensional material, the number of essential elastic constants reduces to two and the equation (10) can be written in the form

$$\sigma(\mathbf{u_s}) = \lambda_s(\nabla \cdot \mathbf{u_s})I + 2\mu_s\epsilon(\mathbf{u_s}) \tag{14}$$

$$= \begin{pmatrix} (2\mu_s + \lambda_s)\frac{\partial \mathbf{u_{s1}}}{\partial x_1} + \lambda_s \frac{\partial \mathbf{u_{s2}}}{\partial x_2} & \mu_s(\frac{\partial \mathbf{u_{s1}}}{\partial x_2} + \frac{\partial \mathbf{u_{s2}}}{\partial x_1}) \\ \mu_s(\frac{\partial \mathbf{u_{s1}}}{\partial x_2} + \frac{\partial \mathbf{u_{s2}}}{\partial x_1}) & (2\mu_s + \lambda_s)\frac{\partial \mathbf{u_{s2}}}{\partial x_2} + \lambda_s \frac{\partial \mathbf{u_{s1}}}{\partial x_1} \end{pmatrix}, \tag{15}$$

where $\mathcal{I}$ is identity matrix and λ_s and μ_s are the Lamé parameters. The Lamé parameters $\lambda_s = C_{1122}$ and $\mu_s = C_{1212}$ can be expressed also as

$$\mu_s = \frac{E}{2(1+\nu)},\ \lambda_s = \frac{E\nu}{(1+\nu)(1-2\nu)}, \tag{16}$$

where E is the Young modulus and ν is the Poisson ratio $0 < \nu < \frac{1}{2}$. Young modulus E is a measure of the stiffness of the solid. It describes how much force is needed to attain the given deformation. Poisson ratio is a measure of the compressibility of the solid. It is the ratio of lateral to longitudinal strain in a uniaxial tensile stress.

When the mass is constant, conservation of linear momentum can be presented in the form:

$$\rho_s\left(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v}\right) - \nabla \cdot \sigma(\mathbf{u_s}) = \mathbf{f}, \tag{17}$$

where $(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v})$ is the material derivative of velocity and $\mathbf{f}$ is the source function. For small deformations, we can assume, that $\mathbf{v} \cdot \nabla \mathbf{v} \approx 0$ and $\mathbf{v} \approx \frac{\partial \mathbf{u_s}}{\partial t}$. Supposing that the body is subject to a body force $\mathbf{f}$, the equation of motion can be presented in the form:

$$\rho_s \frac{\partial^2 \mathbf{u_s}}{\partial t^2} - \nabla \cdot \sigma(\mathbf{u_s}) = \mathbf{f}, \tag{18}$$

where component ij of the stress tensor $\sigma(\mathbf{u_s})$ is

$$\begin{aligned} \sigma_{ij} &= \frac{E}{1+\nu}\epsilon_{ij} + \frac{E\nu}{(1+\nu)(1-2\nu)}\epsilon_{kk}\delta_{ij} \\ &= 2\mu_s\epsilon_{ij} + \lambda_s\epsilon_{kk}\delta_{ij}. \end{aligned}$$

In general, we assume the medium to be heterogeneous. So, the partial derivatives in $\nabla \cdot \sigma$ apply to λ_s and μ_s as well as to the displacement. In homogeneous medium, the

formula is not so complicated, and the equation (18) can be presented in component form as follows:

$$\begin{cases} -(2\mu_s+\lambda_s)\frac{\partial^2 u_1}{\partial x_1^2} - \mu_s\frac{\partial^2 u_1}{\partial x_2^2} - (\mu_s+\lambda_s)\frac{\partial^2 u_2}{\partial x_1 \partial x_2} + \rho_s\frac{\partial^2 u_1}{\partial t^2} &= \mathbf{f}_1, \\ -(2\mu_s+\lambda_s)\frac{\partial^2 u_2}{\partial x_2^2} - \mu_s\frac{\partial^2 u_2}{\partial x_1^2} - (\mu_s+\lambda_s)\frac{\partial^2 u_1}{\partial x_1 \partial x_2} + \rho_s\frac{\partial^2 u_2}{\partial t^2} &= \mathbf{f}_2. \end{cases}$$

We will consider the propagation of time-harmonic waves with angular frequency $\omega > 0$ such that the time-dependence is $\exp(i\omega t)$. So we get the time-harmonic version of the equation (18) by replacing $\mathbf{u_s}(\mathbf{x},t)$ by $\mathbf{u_s} = \mathbf{U_s}(\mathbf{x})\exp(i\omega t)$ and $\mathbf{f}(\mathbf{x},t)$ by $\mathbf{f} = \mathbf{F}(\mathbf{x})\exp(i\omega t)$. Then, the stress tensor $\sigma(\mathbf{u_s})$ is

$$\sigma(\mathbf{u_s}) = (\lambda_s(\nabla\cdot\mathbf{U_s})I + \mu_s(\nabla\mathbf{U_s} + (\nabla\mathbf{U_s})^T))\exp(i\omega t) = \sigma(\mathbf{U_s})\exp(i\omega t), \quad (19)$$

or, in component form

$$\sigma_{jk}(\mathbf{u_s}) = \lambda_s\left(\frac{\partial u_1\exp(i\omega t)}{\partial x_1} + \frac{\partial u_2\exp(i\omega t)}{\partial x_2}\right)\delta_{ij} + \mu_s\left(\frac{\partial u_j\exp(i\omega t)}{\partial x_k} + \frac{\partial u_k\exp(i\omega t)}{\partial x_j}\right),$$

where $j = 1,2$. Now, we have got the time-harmonic elasticity equation

$$-\rho_s\omega^2\mathbf{U_s} - \nabla\cdot\sigma(\mathbf{U_s}) = \mathbf{F}. \quad (20)$$

which has the components

$$-\rho_s\omega^2\mathbf{U_s}_j + \sum_{k=1}^{2}\frac{\partial\sigma_{jk}(\mathbf{U_s})}{\partial x_k} = \mathbf{F}_j,\ j = 1,2. \quad (21)$$

Inserting the time-harmonic stress tensor as given by (19) into the time-harmonic equation of motion (20) and assuming constant material parameters we yield to the following form of the Navier equation:

$$-\rho_s\omega^2\mathbf{U_s} - \mu_s\nabla^2\mathbf{U_s} - (\mu_s+\lambda_s)\nabla(\nabla\cdot\mathbf{U_s}) = \mathbf{F}, \quad (22)$$

which can be written in the component form

$$\begin{cases} -(2\mu_s+\lambda_s)\frac{\partial^2 u_1}{\partial x_1^2} - \mu_s\frac{\partial^2 u_1}{\partial x_2^2} - (\mu_s+\lambda_s)\frac{\partial^2 u_2}{\partial x_1 \partial x_2} - \rho_s\omega^2\mathbf{U_s}_1 &= \mathbf{F}_1, \\ -(2\mu_s+\lambda_s)\frac{\partial^2 u_2}{\partial x_2^2} - \mu_s\frac{\partial^2 u_2}{\partial x_1^2} - (\mu_s+\lambda_s)\frac{\partial^2 u_1}{\partial x_1 \partial x_2} - \rho_s\omega^2\mathbf{U_s}_2 &= \mathbf{F}_2. \end{cases} \quad (23)$$

The wave equation in fluid media can also be derived as a special case of the elasticity equation (18). The displacement vector in fluid domain $\mathbf{u_f}$ can be replaced by the corresponding pressure $p_f = -\lambda_f\nabla\cdot\mathbf{u_f}$. Then we have to take account, that in the solid media $\mu_s > 0$ whereas in the fluid media $\mu_f = 0$. Taking the divergence of equation (18) and substituting equation (14) into equation (18) yields to the wave equation in fluid media

$$\frac{1}{\rho_f c^2}\frac{\partial^2 p_f}{\partial t^2} - \nabla \cdot \left(\frac{1}{\rho_f}\nabla p_f\right) = f, \tag{24}$$

where wave speed $c = \sqrt{\frac{\lambda_f}{\rho_f}}$ and source term $f = -\nabla \cdot \mathbf{f}_f$.

3 EXTERIOR DOMAIN PROBLEMS

Acoustic and elastic wave problems are formulated in unbounded, i.e. exterior, two-dimensional domains. The mathematical formulation of exterior problems, including the conditions of existence and uniqueness of the solution, are presented, for instance, in [24, 69, 88, 100, 101, 103]. In this chapter, we briefly describe this theory in the case of acoustic waves, which are modeled by the Helmholtz equation. The elastic wave problem can be treated in a similar manner.

We consider the scattering of a time-harmonic acoustic plane wave by a bounded, open obstacle $\Theta \subset \mathbb{R}^N$. We denote the boundary of the scattering obstacle by Γ_0. Furthermore, we define the weighted function spaces in the exterior domain $G = \mathbb{R}^N \setminus (\Theta \cup \Gamma_0)$ for $a, b \in \mathbb{R}$ and $k \in \mathbb{N}$ as follows:

$$
\begin{aligned}
L^2_{\mathrm{loc}}(G) &= \{f \in L^2(D) \text{ for all compact sets } D \subset G\},\\
L^2_a(G) &= \{f \in L^2_{\mathrm{loc}}(G) \text{ such that } \rho^a f \in L^2(G)\},\\
L^2_{<a}(G) &= \bigcap_{b<a} L^2_b(G), \qquad L^2_{>a}(G) = \bigcup_{b>a} L^2_b(G),\\
H^k_{\mathrm{loc}}(G) &= \{f \in H^k(D) \text{ for all compact sets } D \subset G\},\\
H^k_a(G) &= \{f \in H^k_{\mathrm{loc}}(G) \text{ such that } f, f', \ldots, f^{(k)} \in L^2_a(G)\},\\
\mathring{H}^k_a(G) &= \{\text{the closure of } C^\infty_0(G) \text{ in the norm of } H^k_a\},\\
H^k_{<a}(G) &= \bigcap_{b<a} H^k_b(G), \qquad \mathring{H}^k_{<a}(G) = \bigcap_{b<a} \mathring{H}^k_b(G),
\end{aligned}
$$

where $\rho(\mathbf{x}) = (1 + r^2)^{1/2}$ is a weight function with $r = \|\mathbf{x}\|_2$. Then, we discuss the exterior Dirichlet problem for the Helmholtz equation with constant coefficients

$$
-\kappa^2 P_f(\mathbf{x}) - \nabla^2 P_f(\mathbf{x}) = F, \qquad \text{in } G, \tag{25}
$$

$$
P_f(\mathbf{x}) = 0, \qquad \text{on } \Gamma_0, \tag{26}
$$

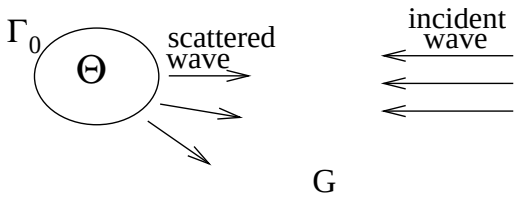


FIGURE 1 Scattering by the obstacle Θ in the exterior domain $G = \mathbb{R}^N \setminus (\Theta \cup \Gamma_0)$.

where $F \in L^2_{\text{loc}}(G)$ and $\kappa > 0$. In addition, we need the Sommerfeld radiation condition

$$-i\kappa P_f + \frac{\partial P_f}{\partial r} \in L^2_{>-1/2}(G) \tag{27}$$

where i is the imaginary unit, and the radial derivative is defined by

$$\frac{\partial}{\partial r} = \frac{\mathbf{x} \cdot \nabla}{r}. \tag{28}$$

Definition 3.1 The outgoing solution to the exterior Dirichlet problem for the equation (25) with $F \in L^2_{\text{loc}}(G)$ is the function $P_f \in \mathring{H}^1_{<-1/2}(G)$ which satisfies the condition (27) and

$$\int_G \left(\nabla P_f \cdot \nabla \bar{\phi} - \kappa^2 P_f \bar{\phi} \right) dx = \int_G F \bar{\phi}\, dx, \quad \forall\, \phi \in C_0^\infty(G). \tag{29}$$

The spectrum of the negative Laplace operator, $-\nabla^2$, for an exterior domain G is given by the set $[0, \infty)$. By self-adjointness of the operator and using Rellich's estimate, it can be seen that this spectrum is continuous. Thus, the operator $-\left(\kappa^2 + \nabla^2\right)$ is injective and its range is dense in $L^2(G)$, but its inverse is not continuous. Since there does not exist a solution for all $F \in L^2(G)$, we need to both reduce the set of admissible right-hand sides and extend the solution space.

It is known that the exterior Dirichlet problems for Helmholtz equation are uniquely solvable if $\kappa^2 \in \mathbb{C} \setminus [0, \infty)$. Since all the complex numbers with nonzero imaginary part belong to this resolvent set (complement of the spectrum), there exists a unique solution for (25)-(26), in $\mathring{H}^1(G)$ for all $F \in L^2(G)$, where κ^2 is replaced by $(\kappa^2 + i\tau) \in \mathbb{C}$ with $\tau > 0$. Thus, we put the Helmholtz equation in the form

$$-\left(\kappa^2 + i\tau\right) P_{f_\tau} - \nabla^2 P_{f_\tau} = F, \tag{30}$$

and use the limiting absorption principle [25] to show that $P_f = \lim_{\tau \to 0} P_{f_\tau}$ exists in some weaker topology. This principle is well defined for all $F \in L^2_{>1/2}(G)$, and we obtain a unique solution to the Dirichlet problem for the Helmholtz equation for any $\kappa > 0$.

Although the elastic wave problem is more complicated, it can be treated basically in the same way as the Helmholtz problem (see, e.g., [101]). Proving the existence and

uniqueness of the solution for the exterior problems with varying coefficients, such as

$$\begin{aligned} -\frac{\kappa(\mathbf{x})^2}{\rho_f(\mathbf{x})}P_f - \nabla\cdot\left(\frac{1}{\rho_f(\mathbf{x})}\nabla P_f\right) &= F, && \text{in } G,\\ \mathbf{U_s} &= 0, && \text{on } \Gamma_0, \end{aligned}$$

and

$$\begin{aligned} -\omega^2\rho_s(\mathbf{x})\mathbf{U_s} - \nabla\cdot\sigma(\mathbf{U_s}) &= \mathbf{F}, && \text{in } G,\\ \mathbf{U_s} &= 0, && \text{on } \Gamma_0, \end{aligned}$$

is more challenging than for the problem with constant coefficients, but it can be done by using an approach related to the methology described above.

To solve the scattering problem numerically, we need an artificial boundary condition that is an approximation of the Sommerfeld radiation condition. This boundary condition is set on the exterior artificial boundary, and it ensures that the solution approximates the restriction of the solution in the original unbounded region. In what follows, we consider acoustic and elastic problems with varying coefficients in bounded domains.

4 TIME-HARMONIC EQUATIONS WITH BOUNDARY CONDITIONS

The equations need to be completed by boundary conditions to get a well-posed and physically meaningful problem. The boundary conditions can be divided to the conditions on physical boundaries and artificial boundaries. Dirichlet and Neumann boundary conditions are examples of the physical boundary conditions. Because of their simplicity, they are also the most common boundary conditions imposed on a model problem. Dirichlet boundary conditions give the value of the unknown at the given boundary whereas Neumann boundary conditions give the value of the gradient of the unknown at the boundary. Neumann boundary condition models a free boundary, where no external forces are acting. Robin boundary condition is a combination of Dirichlet and Neumann boundary conditions. Absorbing boundary condition is an example of the artificial boundary conditions (see, e.g. [3, 9, 10, 28, 39, 46, 64]). These conditions are used to truncate the original unbounded domain.

4.1 Helmholtz problem in bounded domains

The Helmholtz equation is a fundamental equation for time-harmonic wave propagation. It occurs in a number of physical applications such as underwater acoustics, medicine, and geophysics. It can also be used to model the scattering of time-harmonic acoustic waves by an obstacle.

We consider the scattering of a time-harmonic acoustic plane wave by a bounded obstacle in a two-dimensional domain. The scattering can be modelled by the Helmholtz equation with an absorbing boundary condition

$$-\frac{\kappa(\mathbf{x})^2}{\rho_f(\mathbf{x})}P_f - \nabla\cdot\left(\frac{1}{\rho_f(\mathbf{x})}\nabla P_f\right) = F, \qquad \text{in } \Omega, \tag{31}$$

$$P_f = 0, \qquad \text{on } \Gamma_0, \tag{32}$$

$$-i\kappa(\mathbf{x})P_f + \frac{\partial P_f}{\partial \mathbf{n}} = Y_{\text{ext}}, \qquad \text{on } \Gamma_{\text{ext}}, \tag{33}$$

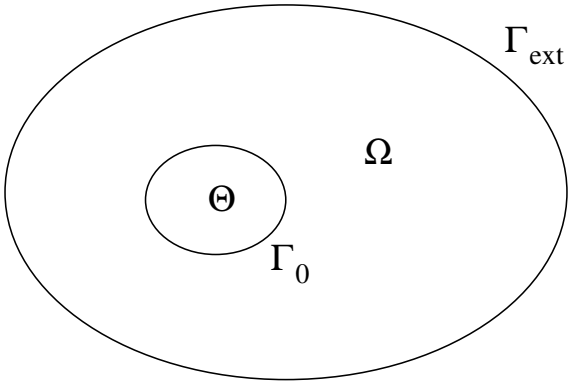


FIGURE 2 Obstacle Θ, domain Ω, and the two parts of the boundary $\partial\Omega = \Gamma_0 \cup \Gamma_{\text{ext}}$ of the domain Ω.

where $P_f(\mathbf{x})$ denotes the (complex-valued) total acoustic pressure field. The total field is sum of the scattered wave $P_{f_{\text{scat}}}$ and the incident plane wave $P_{f_{\text{inc}}}$. The Dirichlet boundary condition sets the value for the pressure on Γ_0, and F and Y_{ext} are source terms due to the incident plane wave.

The problem setting is illustrated in Figure 2, where $\Theta \subset \mathbb{R}^2$ denotes the obstacle and $\Omega \subset \mathbb{R}^2$ is the domain between the obstacle and the absorbing boundary Γ_{ext}. The boundary of the obstacle is denoted by Γ_0. On the absorbing boundary Γ_{ext}, we impose the conventional absorbing boundary condition [28]. This is the simplest alternative and not accurate in approximating the Sommerfeld radiation condition. However, it is sufficient for the presentation of the controllability method of this thesis. Vector $\mathbf{n}$ denotes the outward normal vector to Ω, and wavenumber and density of the material are denoted by $\kappa(\mathbf{x})$ and $\rho_f(\mathbf{x})$, respectively. The wavenumber is related to the angular frequency ω and to the speed of sound $c(\mathbf{x})$ by the formula $\kappa(\mathbf{x}) = \frac{\omega}{c(\mathbf{x})}$. The corresponding wavelength is given by $\ell(\mathbf{x}) = \frac{2\pi}{\kappa(\mathbf{x})}$.

The time-harmonic incident plane wave is given by $P_{f_{\text{inc}}}(\mathbf{x}) = \exp(i\boldsymbol{\omega} \cdot \mathbf{x})$, where i is the imaginary unit and the vector $\boldsymbol{\omega}$ gives the propagation direction. The angular frequency equals the euclidean norm of $\boldsymbol{\omega}$, that is $\omega = \|\boldsymbol{\omega}\|_2$. Then, the functions F and Y_{ext}, in the equations above, are of the form

$$F = -\frac{\kappa(\mathbf{x})^2}{\rho_f(\mathbf{x})} P_{f_{\text{inc}}}(\mathbf{x}) - \nabla \cdot \left(\frac{1}{\rho_f(\mathbf{x})} \nabla P_{f_{\text{inc}}}(\mathbf{x}) \right), \tag{34}$$

$$Y_{\text{ext}} = -i\kappa(\mathbf{x}) P_{f_{\text{inc}}}(\mathbf{x}) + \frac{\partial P_{f_{\text{inc}}}(\mathbf{x})}{\partial \mathbf{n}}. \tag{35}$$

4.2 Navier problem in bounded domains

In an elastic and isotropic body $\Omega \subset \mathbb{R}^2$ with density $\rho_s(\mathbf{x})$, the propagation of time-harmonic waves with angular frequency ω is governed by the Navier equation

$$-\omega^2 \rho_s(\mathbf{x})\mathbf{U_s} - \nabla \cdot \sigma(\mathbf{U_s}) = \mathbf{F} \qquad \text{in } \Omega, \tag{36}$$

where $\mathbf{U_s}$ denotes the displacement field $\mathbf{U_s}(\mathbf{x}) = (u_1(\mathbf{x}), u_2(\mathbf{x}))^T$, which depends on the spatial variable $\mathbf{x} = (x_1, x_2)^T \in \mathbb{R}^2$.

The boundary of the domain Ω, marked as $\partial\Omega$, is divided into two distinct parts (see Figure 2). The boundary Γ_0 is assumed to be rigid, whereas on the artificial boundary Γ_{ext} we impose an absorbing boundary condition

$$\mathbf{U_s} = 0 \qquad \text{on } \Gamma_0, \tag{37}$$

$$i\omega\rho_s(\mathbf{x})\mathbf{B}\mathbf{U_s} + \sigma(\mathbf{U_s})\mathbf{n} = \mathbf{G}_{\text{ext}} \qquad \text{on } \Gamma_{\text{ext}}, \tag{38}$$

where $\mathbf{B}$ is a matrix [28, 87] such that

$$\mathbf{B} = \begin{pmatrix} n_1 & n_2 \\ n_2 & -n_1 \end{pmatrix} \begin{pmatrix} c_p & 0 \\ 0 & c_s \end{pmatrix} \begin{pmatrix} n_1 & n_2 \\ n_2 & -n_1 \end{pmatrix}. \tag{39}$$

The vector $\mathbf{n} = (n_1, n_2)^T$ is the outward pointing normal vector on Γ_{ext}. Coefficients c_p and c_s represent the speed of the pressure waves (P-waves) and the speed of the shear waves (S-waves), respectively, and are formulated as

$$c_p = \sqrt{\frac{\lambda_s(\mathbf{x}) + 2\mu_s(\mathbf{x})}{\rho_s(\mathbf{x})}}, \quad c_s = \sqrt{\frac{\mu_s(\mathbf{x})}{\rho_s(\mathbf{x})}}. \tag{40}$$

The P-waves move in a compressional motion, while the motion of the S-waves is perpendicular to the direction of wave propagation [99].

If the time-harmonic incident plane wave is given by $\mathbf{U_{s\text{inc}}}(\mathbf{x}) = \exp(i\boldsymbol{\omega} \cdot \mathbf{x})$, the functions $\mathbf{F}$ and $\mathbf{G}_{\text{ext}}$, in the equations above, are of the form

$$\mathbf{F} = -\omega^2 \rho_s(\mathbf{x})\mathbf{U_{s\text{inc}}}(\mathbf{x}) - \nabla \cdot \sigma(\mathbf{U_{s\text{inc}}}(\mathbf{x})), \tag{41}$$

$$\mathbf{G}_{\text{ext}} = i\omega\rho_s(\mathbf{x})\mathbf{B}\mathbf{U_{s\text{inc}}}(\mathbf{x}) + \sigma(\mathbf{U_{s\text{inc}}}(\mathbf{x}))\mathbf{n}. \tag{42}$$

5 EXACT CONTROLLABILITY PROBLEM

Exact controllability is well-known and extensively researched topic within classical wave equations [16]. The basic idea of exact controllability is to have preassigned initial and final states such that beginning from the initial state, the final state can be achieved by some control. Exact controllability can be boundary [26, 40, 83, 89], internal [49, 74, 75] or pointwise controllability depending on where the control has been set. Since pointwise controllability can be handled as a special case of internal controllability near one point [29], both internal and pointwise controllability are also known as distributed controllability [72].

With exact controllability concept, control can be applied to initial conditions. Thus, it is possible to find time periodic solutions to wave equations without solving the Helmholtz or Navier problem. This can be done by applying a method, introduced in [13], to the original time-dependent wave equation. The main idea of this method is ispired by the Hilbert Uniqueness Method (HUM) which was introduced by Lions [71] as a systematic method to address controllability problems for partial differential equations.

By exact controllability of the problem we mean that there is a system in a given initial state $(\mathbf{u}(0), \frac{\partial \mathbf{u}}{\partial t}(0))$ and a control $\mathbf{e} = (\mathbf{e}_0, \mathbf{e}_1)$ such that the given final state $(\mathbf{u}(T), \frac{\partial \mathbf{u}}{\partial t}(T))$ can be achieved [73]. To achieve the time-harmonic solution, we minimize the difference between inital conditions and the corresponding variables after one time period. Proceeding this way, the problem of time-harmonic wave scattering can be handled with time-dependent equations as a least squares problem, which can be solved by a conjugate gradient algorithm. The exact controllability approach has been shown to be robust but quite CPU time demanding since the solution of forward and backward wave equations and preconditioning are required at each iteration [16]. There is, however, strong experimental evidence that the minimization process is independent of certain problem parameters which leads to optimal order of computational cost in corresponding problem configurations. Numerical examples indicate this feasibility of the method in Chapter 9.

5.1 Helmholtz problem

Instead of solving directly the time-harmonic equation, we return to the corresponding time-dependent equation and look for time-periodic solution. Direct time-integration of the wave equation can be used to reach the time-periodic case, but convergence is usually too slow to be useful in practice. As the results in [16] indicate, the convergence can be vastly improved by control techniques.

Solution of the time-harmonic equation (31)-(33) is equivalent to finding a periodic solution for the corresponding time-dependent wave equation. The period T corresponding to the angular frequency ω is given by $\frac{2\pi}{\omega}$, and the T-periodic solution can be achieved by controlling the initial conditions such that the solution at time T coincides with the initial conditions. To describe the approach in detail, we introduce the Hilbert space Z_f for the initial conditions $\mathbf{e} = (\mathbf{e}_0, \mathbf{e}_1) \in Z_f$ by

$$Z_f = V \times L^2(\Omega), \tag{43}$$

where

$$V = \{v \in H^1(\Omega) \text{ such that } v = 0 \text{ on } \Gamma_0\}. \tag{44}$$

Then, we have the following exact controllability problem: Find initial conditions $\mathbf{e} \in Z_f$ such that equations

$$\frac{1}{\rho_f(\mathbf{x})c(\mathbf{x})^2}\frac{\partial^2 p_f}{\partial t^2} - \nabla \cdot \left(\frac{1}{\rho_f(\mathbf{x})}\nabla p_f\right) = f, \qquad \text{in } Q = \Omega \times [0,T], \tag{45}$$

$$p_f = 0, \qquad \text{on } \gamma_0 = \Gamma_0 \times [0,T], \tag{46}$$

$$\frac{1}{c(\mathbf{x})}\frac{\partial p_f}{\partial t} + \frac{\partial p_f}{\partial \mathbf{n}} = y_{\text{ext}}, \qquad \text{on } \gamma_{\text{ext}} = \Gamma_{\text{ext}} \times [0,T], \tag{47}$$

$$p_f(\mathbf{x},0) = \mathbf{e}_0 \qquad \text{in } \Omega, \tag{48}$$

$$\frac{\partial p_f}{\partial t}(\mathbf{x},0) = \mathbf{e}_1 \qquad \text{in } \Omega, \tag{49}$$

$$p_f(\mathbf{x},T) = \mathbf{e}_0 \qquad \text{in } \Omega, \tag{50}$$

$$\frac{\partial p_f}{\partial t}(\mathbf{x},T) = \mathbf{e}_1 \qquad \text{in } \Omega, \tag{51}$$

hold with

$$f(\mathbf{x},t) = -\frac{\kappa(\mathbf{x})^2}{\rho_f(\mathbf{x})}p_{f_{\text{inc}}}(\mathbf{x},t) - \nabla \cdot \left(\frac{1}{\rho_f(\mathbf{x})}\nabla p_{f_{\text{inc}}}\right), \tag{52}$$

$$y_{\text{ext}}(\mathbf{x},t) = \frac{\partial p_{f_{\text{inc}}}(\mathbf{x},t)}{\partial \mathbf{n}} - \text{Re}(i\kappa \mathbf{U}_{\text{inc}} \exp(-i\omega t)), \tag{53}$$

where $p_{f_{\text{inc}}}(\mathbf{x},t) = \text{Re}(\mathbf{U}_{\text{inc}}(\mathbf{x})\exp(-i\omega t))$.

The spectral element discretization of the problem is based on the weak formulation of the classical wave equation (45)-(47): Find p_f satisfying $p_f(t) \in V$ for any $t \in [0,T]$ and

$$\int_\Omega \frac{1}{\rho_f(\mathbf{x})c(\mathbf{x})^2}\frac{\partial^2 p_f}{\partial t^2} v\,dx + \int_\Omega \frac{1}{\rho_f(\mathbf{x})}\nabla p_f \cdot \nabla v\,dx + \int_{\Gamma_{\text{ext}}} \frac{1}{c(\mathbf{x})\rho_f(\mathbf{x})}\frac{\partial p_f}{\partial t} v\,ds \qquad (54)$$
$$= \int_\Omega f v\,dx + \int_{\Gamma_{\text{ext}}} \frac{1}{\rho_f(\mathbf{x})} y_{\text{ext}} v\,ds$$

for any $v \in V$ and $t \in [0,T]$.

5.2 Navier problem

Solving of the time-harmonic equation (36),(37)-(38) is equivalent to finding a time-periodic solution for the corresponding time-dependent wave equation

$$\rho_s(\mathbf{x})\frac{\partial^2 \mathbf{u_s}}{\partial t^2} - \nabla\cdot\sigma(\mathbf{u_s}) = \mathbf{f}, \qquad \text{in } Q = \Omega\times[0,T], \qquad (55)$$
$$\mathbf{u_s} = 0, \qquad \text{on } \gamma_0 = \Gamma_0\times[0,T], \qquad (56)$$
$$\mathbf{B}\frac{\partial \mathbf{u_s}}{\partial t} + \sigma(\mathbf{u_s})\mathbf{n} = \mathbf{g}_{\text{ext}}, \qquad \text{on } \gamma_{\text{ext}} = \Gamma_{\text{ext}}\times[0,T], \qquad (57)$$

where $\mathbf{u_s} = (U_1, U_2)^T$, $\mathbf{f} = (F_1, F_2)$, $\mathbf{g}_0 = (G_{01}, G_{02})$, and $\mathbf{g}_{\text{ext}} = (G_{\text{ext1}}, G_{\text{ext2}})^T$. In addition to the system (55)-(57), we take into account the initial conditions

$$\mathbf{u_s}(\mathbf{x},0) = \mathbf{e}_0, \qquad \text{in } \Omega, \qquad (58)$$
$$\frac{\partial \mathbf{u_s}(\mathbf{x},0)}{\partial t} = \mathbf{e}_1, \qquad \text{in } \Omega. \qquad (59)$$

For the weak formulation of the problem (55)-(59), we introduce the function space $\mathbf{V}$ by $\mathbf{V} = \{\mathbf{y} \in H^1(\Omega)\times H^1(\Omega) \text{ such that } \mathbf{y} = \mathbf{0} \text{ on } \Gamma_0\}$. After introducing the Hilbert space $\mathbf{Z_s}$ for the initial conditions $\mathbf{e} = (\mathbf{e}_0, \mathbf{e}_1)^T \in \mathbf{Z_s}$ by

$$\mathbf{Z_s} = \mathbf{V}\times(L^2(\Omega)\times L^2(\Omega)), \qquad (60)$$

we formulate the exact controllability problem as follows: Find initial conditions $\mathbf{e} = (\mathbf{e}_0, \mathbf{e}_1)^T$ such that equations (55)-(59) hold with the terminal conditions

$$\mathbf{u_s}(\mathbf{x},T) = \mathbf{e}_0 \qquad \text{in } \Omega, \qquad (61)$$
$$\frac{\partial \mathbf{u_s}(\mathbf{x},T)}{\partial t} = \mathbf{e}_1 \qquad \text{in } \Omega. \qquad (62)$$

The time period corresponding to the angular frequency ω is given by $T = \frac{2\pi}{\omega}$, and the T-periodic solution can be achieved by controlling the initial conditions such that the terminal conditions (61)-(62) are equal to the initial conditions (58)-(59) at the end of the computation.

By multiplying the equation (55) with any test function $\mathbf{v}$ in the space $\mathbf{V}$, using the Green's formula and substituting the boundary conditions we get the following weak formulation: Find $\mathbf{u_s}$ satisfying $\mathbf{u_s}(t) \in \mathbf{V}$ for any $t \in [0, T]$ and

$$\int_{\Omega} \rho_s(\mathbf{x}) \frac{\partial^2 \mathbf{u_s}}{\partial t^2} \cdot \mathbf{v}\, dx + \int_{\Omega} \sigma(\mathbf{u_s}) : \epsilon(\mathbf{v})\, dx + \int_{\Gamma_{\text{ext}}} \mathbf{B} \frac{\partial \mathbf{u_s}}{\partial t} \cdot \mathbf{v}\, ds = \int_{\Omega} \mathbf{f} \cdot \mathbf{v}\, dx + \int_{\Gamma_{\text{ext}}} \mathbf{g}_{\text{ext}} \cdot \mathbf{v}\, ds \tag{63}$$

for any $\mathbf{v} \in \mathbf{V}$ and $t \in [0, T]$.

6 DISCRETIZATION

This section is about the numerical approximation of the wave equation. In order to produce an approximate solution of partial differential equations, the given domain is discretized into a collection of elements. The elements are associated with a mesh, which defines the geometry of the domain. In two-dimensional domain, the mesh is built, for example, from triangular or quadrilateral elements. In three-dimensional domain, which is the most natural domain to solve physical phenomena, volume elements such as tetrahedra or hexahedra are used. The accuracy can be increased by using appropriate elements [38]. The size of the used elements depends on the problem and different sizes of elements can be used in a domain. The error of the approximation decreases as the size of elements decreases while the number of elements increases. Neighboring elements are connected by particular points, which are called nodes. The elements and nodes are numbered both locally and globally and geometric properties, such as coordinates, are generated.

Basis functions consist of sets of polynomials and are used to give the discrete values of the approximated solution. In particular, the geometry of the elements is described by a mapping of a reference element onto a possible deformed element in the physical coordinates. When the mapped basis functions coincide with the geometry functions, the element is referred to as isoparametric.

Discretization methods play a big role in efficiency of the controllability method. The key factor in developing efficient solution methods is the use of high order approximations without computationally demanding matrix inversions. We have made a move towards meeting these requirements by using the spectral element method [22] for the spatial discretization of the wave equation. It allows convenient treatment of complex geometries and varying material properties. The basis functions are higher order Lagrange interpolation polynomials, and the nodes of these functions are placed at Gauss-Lobatto collocation points. The integrals in the weak form of the equation are evaluated with the corresponding Gauss-Lobatto quadrature formulas. As a consequence of the choice, spectral element discretization leads to diagonal mass matrices which significantly improves the computational efficiency of the explicit time-integration used. Moreover, when using higher order elements, same accuracy is reached with less degrees of freedom than when using lower order finite elements.

Since the spectral element method can be described as a finite element method, in which higher order spectral method is used within each element, we first briefly consider

the finite element method in Section 6.1. Another reason for presenting some background of the finite element method is the fact that the special case of the lowest order spectral element corresponds to the bilinear finite element. In Section 6.2, the spectral element method is applied to the weak form of the equations in the space domain. Then, finite difference and Runge-Kutta schemes are used to advance the system in time in Section 6.3.

6.1 Finite element method

The Finite element method (FEM) is a general technique for the numerical solution of PDEs in structural engineering [106], which originates in the 1950's in the field of structural analysis in aircraft industry. It was thought to be a generalization of methods used earlier for beams, frames and plates by subdividing the structure into smaller parts. FEM has become one of the main computing tools of scientists and engineers. It has been used in solid and fluid mechanics to solve, for instance, elasticity, plasticity and fluid-structure interaction problems. In a higher order finite element method, which is called p-FEM, the basis functions are higher order Lagrange polynomials, and the nodes of these functions are placed at regular uniform nodal points in each element. According to [70], with this kind of distribution of the nodal points, the accuracy is not as good as the one achieved with the spectral element method. Also mass lumping (i.e. creating diagonal mass matrices) might be a computationally inefficient operation with FEM.

6.2 Spectral element method

The spectral element method (SEM) was pioneered in the mid 1980's by Anthony Patera [85] and Yvon Maday [76], and it combines the geometric flexibility of finite elements [59, 66] with the high accuracy of spectral methods [18]. When using SEM, the physical domain is typically divided into nonoverlapping quadrilateral elements, but also triangular elements can be used [63, 11]. Contrary to quadrilateral spectral elements, mass matrices are not generally diagonal with triangular elements [95]. Whether mass matrices are diagonal or not, the computational effort is larger on triangular elements than on quadrilateral elements. The reason for this is that triangles are not tensor-product elements, and hence the computation of the derivatives involves all collocation point values on elements. Consequently, the cost of computing derivatives is higher on triangles than on quadrilaterals. Moreover, accuracy is slightly better on quadrilaterals than on triangles, and condition number of the stiffness matrices grows faster for triangles than quadrilaterals [84]. At present, it seems that triangle based SEM is competitive with the quadrilateral one only if the domain Ω has a curved shape. These are the reasons why we have chosen to use quadrilateral elements and the associated polynomial spectral basis. A detailed comparison of SEM on quadrilaterals and triangles is made in [84], and quadrature formulas needed for quadrilateral and triangle based methods are recently presented, for instance,

in [22] and [96], respectively.

After the domain is decomposed into elements, a local polynomial basis is introduced in each element. These basis functions are explained in the next section. The degrees of freedom associated with the basis functions are situated at the Gauss-Lobatto quadrature points of the quadrilateral. This is the main difference between SEM and p-FEM.

The computational efficiency of the method is based on the use of the Gauss-Lobatto quadrature rule in the computation of the finite element matrices. It provides lumped mass matrices without reducing the order of accuracy and leads to efficient time-dependent simulation.

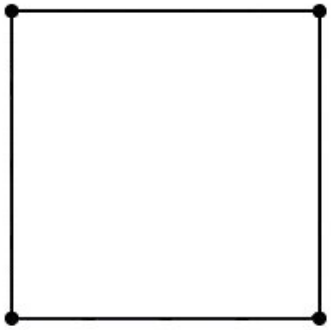


FIGURE 3 Spectral element of order $r = 1$.

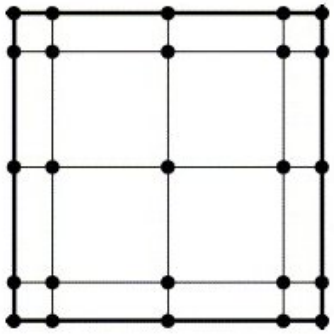


FIGURE 4 Spectral element of order $r = 4$.

6.2.1 Basis functions

The spectral element method has polynomial basis functions of degree r in each spatial dimension, and integrals over the elements are evaluated using numerical quadrature. On a reference element $[0,1]^2$, we introduce a set of local basis functions consisting of Lagrange polynomials of degree r. More precisely, we first map each element back to the reference domain using the invertible local mapping $\mathcal{G}_i : \Omega_{ref} = [0,1]^2 \rightarrow \Omega_i$.

In each direction of the reference element, we introduce a set of Gauss-Lobatto (GL) points $\xi_i, i = 1, \ldots, r+1, \in [0,1]$. The rth order GL quadrature points in the one-dimensional reference element $[0,1]$ are the zeroes of $x(1-x)L_r^{'}(2x-1),\ x \in [0,1]$, where

$$L_r^{'}(x) = \frac{2r-1}{r}\left(L_{r-1}(x) + xL_{r-1}^{'}(x)\right) - \frac{r-1}{r}L_{r-2}^{'}(x) \tag{64}$$

is the derivative of rth degree Legendre polynomial L_r. The sequence of polynomials L_r is given by the recursion formula

$$\begin{cases} L_0(x) &= 1, \\ L_1(x) &= x, \\ L_r(x) &= \frac{2r-1}{r} x L_{r-1}(x) - \frac{r-1}{r} L_{r-2}(x),\ r > 1. \end{cases} \tag{65}$$

The rth order basis functions on the interval $[0,1]$, constructed as a set of Lagrange interpolants,

$$\hat{\varphi}_j(\xi_i) = \prod_{p=1, p\neq j}^{r+1} \left(\frac{\xi_i - \xi_p}{\xi_j - \xi_p} \right), \quad j = 1, \ldots, r+1 \tag{66}$$

are uniquely determined by the requirement that each function has the value one at one Gauss-Lobatto quadrature point and zero at the remaining quadrature points

$$\hat{\varphi}_j(\xi_i) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j. \end{cases} \tag{67}$$

Quadrature points and the set of basis functions of the reference element in higher dimensions are achieved by products of the $(r+1)$ one-dimensional Lagrange interpolants $\hat{\varphi}_j(\xi_i), j = 1, \ldots, r+1$. For example in the two-dimensional case, the Lagrange interpolant associated with the ijth grid node, is defined as

$$\hat{\varphi}_{ij}(\xi, \zeta) = \hat{\varphi}_i(\xi)\hat{\varphi}_j(\zeta) = \prod_{p=1, p\neq i}^{r+1} \left(\frac{\xi - \xi_p}{\xi_i - \xi_p} \right) \prod_{q=1, q\neq j}^{r+1} \left(\frac{\zeta - \xi_q}{\xi_j - \xi_q} \right), \quad j = 1, \ldots, r+1.$$

The coordinate system in the reference element is formed by ξ and ζ. The coordinate of the ith grid node in the direction of ξ is marked as ξ_i, whereas ζ_j is the coordinate of the jth node in the direction of ζ.

By definition, these polynomials have the fundamental property that they vanish at all but one of the GL points. The mapping between the reference element and ith element is defined such that $\mathcal{G}_i(\xi, \zeta) = x \in \Omega_i$.

6.2.2 Discrete weak formulation

The spectral element method (as well as the finite element method) adds contributions from all individual elements to the global solution. Therefore, each element is calculated independently and needs a local numbering. The first discretization step consists in dividing the domain Ω into N_e disjoint quadrilateral elements. We denote the elements by Ω_i, $i = 1, \ldots, N_e$, and assume that $\Omega = \bigcup_{i=1}^{N_e} \Omega_i$, i.e., the mesh coincides with the domain exactly. After the domain Ω is divided into a finite number of elements, each element is associated with a finite number of nodes. Each of N_e elements is individually mapped to a reference element $\Omega_{\text{ref}} = [0,1]^2$. The use of reference elements is based on affine mappings $\mathcal{G}_i : \Omega_{\text{ref}} \to \Omega_i$ such that $\mathcal{G}_i(\Omega_{\text{ref}}) = \Omega_i$. We make use of this mapping to make transformations from the physical domain to the reference domain, and on the contrary.

The spectral element method is obtained from the weak formulations (54) and (63) by restricting the problem in the infinite dimensional spaces V and $\mathbf{V}$ into finite dimensional subspaces $V_h^r \subset V$ and $\mathbf{V}_h^r \subset \mathbf{V}$, respectively. First, we define the finite dimensional space V_h^r by

$$V_h^r = \{v_h \in V \text{ such that } v_h|_{\Omega_i} \circ \mathcal{G}_i \in Q^r\}, \tag{68}$$

where

$$Q^r(\Omega_i) = \{v(\xi,\zeta) = \sum_{p=0}^{r}\sum_{q=0}^{r} a_{pq}\xi^p\zeta^q, \quad a_{pq} \in \mathbb{R}\} \tag{69}$$

is the set of polynomials of order r in $\mathbb{R}^2$. The quadrilateral mesh is assumed to satisfy the usual regularity assumptions for a finite element mesh [66].

The basis functions φ_n for the space V_h^r are constructed with the help of the basis functions $\hat{\varphi}_{jk}$, $j,k = 1,\ldots,r+1$, on the reference element Ω_{ref}. These functions are Lagrange interpolants of the Gauss-Lobatto integration points in Ω_{ref} and can be written as a product of two one dimensional basis functions which are polynomials of order r. Then, for each basis function φ_n for V_h^r we can identify a basis function $\hat{\varphi}_{jk}$ such that $\varphi_n|_{\Omega_i} \circ \mathcal{G}_i = \hat{\varphi}_{jk}$ (see [22] for details). Based on these definitions we can write the semidiscrete weak formulation of the wave equation (45)-(47): Find p_{f_h} satisfying $p_{f_h}(t) \in V_h^r$ for any $t \in [0,T]$ and

$$\begin{aligned}\int_\Omega \frac{1}{\rho_f(\mathbf{x})c(\mathbf{x})^2}\frac{\partial^2 p_{f_h}}{\partial t^2}v_h\,dx + \int_\Omega \frac{1}{\rho_f(\mathbf{x})}\nabla p_{f_h}\cdot\nabla v_h\,dx + \int_{\Gamma_{\text{ext}}} \frac{1}{c(\mathbf{x})\rho_f(\mathbf{x})}\frac{\partial p_{f_h}}{\partial t}v_h\,ds \\ = \int_\Omega f v_h\,dx + \int_{\Gamma_{\text{ext}}} \frac{1}{\rho_f(\mathbf{x})}y_{\text{ext}}v_h\,ds,\end{aligned} \tag{70}$$

for all $v_h \in V_h^r$ and $t \in [0,T]$. The dimension of the space V_h^r is the number of Gauss-Lobatto points of the quadrilateral mesh and we denote this number by N_{dof}.

The finite element subspace $\mathbf{V}_h^r$ of $\mathbf{V}$ is given by

$$\mathbf{V}_h^r = \{\mathbf{y} = (y_1,y_2) \in \mathbf{V} \text{ such that } y_k|_{\Omega_i} \circ \mathcal{G}_i \in Q^r, k = 1,2\}, \tag{71}$$

where Q^r is the set of polynomials of order r in each variable in space. The semidiscrete weak formulation of the wave equation (55)-(57): Find $\mathbf{u}_{\mathbf{s}h}$ satisfying $\mathbf{u}_{\mathbf{s}h}(t) \in \mathbf{V}_h^r$ for any $t \in [0,T]$ and

$$\begin{aligned}\int_\Omega \rho_s(\mathbf{x})\frac{\partial^2 \mathbf{u}_{\mathbf{s}h}}{\partial t^2}\cdot\mathbf{v}_h\,dx + \int_\Omega \sigma(\mathbf{u}_{\mathbf{s}h}) : \epsilon(\mathbf{v}_h)\,dx + \int_{\Gamma_{\text{ext}}} \mathbf{B}\frac{\partial \mathbf{u}_{\mathbf{s}h}}{\partial t}\cdot\mathbf{v}_h\,ds \\ = \int_\Omega \mathbf{f}\cdot\mathbf{v}_h\,dx + \int_{\Gamma_{\text{ext}}} \mathbf{g}_{\text{ext}}\cdot\mathbf{v}_h\,ds\end{aligned} \tag{72}$$

for any $\mathbf{v}_h \in \mathbf{V}_h^r$ and $t \in [0,T]$. The dimension of the space $\mathbf{V}_h^r$ is two times the number of Gauss-Lobatto points of the quadrilateral mesh, i.e. $2N_{\text{dof}}$.

6.2.3 Semidiscretized equation

After the spectral element discretization of the discrete variational problems (70) and (72), the both wave equations can be presented as semidiscrete equations of the form

$$\mathcal{M}\frac{\partial^2 \mathbf{u}}{\partial t^2} + \mathcal{S}\frac{\partial \mathbf{u}}{\partial t} + \mathcal{K}\mathbf{u} = \mathcal{F}, \tag{73}$$

where $\mathcal{M}$ is the mass matrix, $\mathcal{S}$ is a damping term due to the absorbing boundary condition, $\mathcal{K}$ is the stiffness matrix, and $\mathcal{F}$ the source term. Only terms along the boundary Γ_{ext} contribute to the nonzero entries of the matrix $\mathcal{S}$. The time-dependent vector of unknowns $\mathbf{u}$ determines the global solution at each Gauss-Lobatto point in the mesh.

Acoustic wave equation

In the case of the acoustic wave equation, we denote by $\mathbf{u} \in \mathbb{R}^{N_{\text{dof}}}$ the vector containing the values of the function p_{f_h} (total pressure) at the Gauss-Lobatto points of the quadrilateral mesh. The entries of the $N_{\text{dof}} \times N_{\text{dof}}$ matrices $\mathcal{M}$, $\mathcal{S}$, and $\mathcal{K}$ are given by the formulas

$$\mathcal{M}_{ij} = \int_{\Omega} \frac{1}{\rho_f(\mathbf{x})c(\mathbf{x})^2}\varphi_i\varphi_j dx, \tag{74}$$

$$\mathcal{S}_{ij} = \int_{\Gamma_{\text{ext}}} \frac{1}{\rho_f(\mathbf{x})c(\mathbf{x})}\varphi_i\varphi_j ds, \tag{75}$$

$$\mathcal{K}_{ij} = \int_{\Omega} \frac{1}{\rho_f(\mathbf{x})}\nabla\varphi_i \cdot \nabla\varphi_j dx, \tag{76}$$

and the components of the vector $\mathcal{F}$ are of the form

$$\mathcal{F}_i = \int_{\Omega} f\varphi_i dx + \int_{\Gamma_{\text{ext}}} \frac{1}{\rho_f(\mathbf{x})} y_{\text{ext}}\varphi_i ds. \tag{77}$$

Elastic wave equation

When the elastic wave equation is considered, $\mathbf{u} \in \mathbb{R}^{2N_{\text{dof}}}$ contains the nodal values of the displacement $\mathbf{u_s}(\mathbf{x}, t)$ at time t. Then, $\mathcal{M}$, $\mathcal{S}$, and $\mathcal{K}$ are 2×2 block matrices and $\mathcal{F}$ is a block vector as follows:

$$\mathcal{M} = \begin{pmatrix} \mathcal{M}_{11} & \mathbf{0} \\ \mathbf{0} & \mathcal{M}_{22} \end{pmatrix}, \quad \mathcal{S} = \begin{pmatrix} \mathcal{S}_{11} & \mathcal{S}_{12} \\ \mathcal{S}_{21} & \mathcal{S}_{22} \end{pmatrix}, \quad \mathcal{K} = \begin{pmatrix} \mathcal{K}_{11} & \mathcal{K}_{12} \\ \mathcal{K}_{21} & \mathcal{K}_{22} \end{pmatrix}, \quad \mathcal{F} = \begin{pmatrix} \mathcal{F}_1 \\ \mathcal{F}_2 \end{pmatrix}.$$

The components of these block forms are

$$(\mathcal{M}_{11})_{ij} = \int_{\Omega} \rho_s(\mathbf{x})\varphi_j\varphi_i \, dx, \tag{78}$$

$$(\mathcal{M}_{22})_{ij} = \int_{\Omega} \rho_s(\mathbf{x})\varphi_j\varphi_i \, dx, \tag{79}$$

$$(\mathcal{S}_{11})_{ij} = \int_{\Gamma_{ext}} \left(c_p n_1^2 + c_s n_2^2\right) \varphi_j\varphi_i \, ds, \tag{80}$$

$$(\mathcal{S}_{12})_{ij} = \int_{\Gamma_{ext}} (c_p - c_s) n_1 n_2 \varphi_j\varphi_i \, ds \tag{81}$$

$$(\mathcal{S}_{21})_{ij} = \int_{\Gamma_{ext}} (c_p - c_s) n_1 n_2 \varphi_j\varphi_i \, ds \tag{82}$$

$$(\mathcal{S}_{22})_{ij} = \int_{\Gamma_{ext}} \left(c_p n_2^2 + c_s n_1^2\right) \varphi_j\varphi_i \, ds, \tag{83}$$

$$(\mathcal{K}_{11})_{ij} = \int_{\Omega} \left(\lambda_s(\mathbf{x})\frac{\partial\varphi_j}{\partial x_1}\frac{\partial\varphi_i}{\partial x_1} + 2\mu_s(\mathbf{x})\left(\frac{\partial\varphi_j}{\partial x_1}\frac{\partial\varphi_i}{\partial x_1} + \frac{1}{2}\frac{\partial\varphi_j}{\partial x_2}\frac{\partial\varphi_i}{\partial x_2}\right)\right) dx, \tag{84}$$

$$(\mathcal{K}_{12})_{ij} = \int_{\Omega} \left(\lambda_s(\mathbf{x})\frac{\partial\varphi_j}{\partial x_2}\frac{\partial\varphi_i}{\partial x_1} + \mu_s(\mathbf{x})\frac{\partial\varphi_j}{\partial x_1}\frac{\partial\varphi_i}{\partial x_2}\right) dx, \tag{85}$$

$$(\mathcal{K}_{21})_{ij} = \int_{\Omega} \left(\lambda_s(\mathbf{x})\frac{\partial\varphi_j}{\partial x_1}\frac{\partial\varphi_i}{\partial x_2} + \mu_s(\mathbf{x})\frac{\partial\varphi_j}{\partial x_2}\frac{\partial\varphi_i}{\partial x_1}\right) dx, \tag{86}$$

$$(\mathcal{K}_{22})_{ij} = \int_{\Omega} \left(\lambda_s(\mathbf{x})\frac{\partial\varphi_j}{\partial x_2}\frac{\partial\varphi_i}{\partial x_2} + 2\mu_s(\mathbf{x})\left(\frac{1}{2}\frac{\partial\varphi_j}{\partial x_1}\frac{\partial\varphi_i}{\partial x_1} + \frac{\partial\varphi_j}{\partial x_2}\frac{\partial\varphi_i}{\partial x_2}\right)\right) dx, \tag{87}$$

$$(\mathcal{F}_1)_i = \int_{\Omega} \mathbf{f}_1\varphi_i \, dx + \int_{\Gamma_{ext}} \mathbf{g}_{ext1}\varphi_i \, ds, \tag{88}$$

$$(\mathcal{F}_2)_i = \int_{\Omega} \mathbf{f}_2\varphi_i \, dx + \int_{\Gamma_{ext}} \mathbf{g}_{ext2}\varphi_i \, ds, \tag{89}$$

where $i, j = 1, \ldots, N_{\text{dof}}$. By N_{dof} we denote the number of degrees of freedom (DOF) in each space variable, which is also the total number of Gauss-Lobatto points in the space discretization.

The computation of element matrices and vectors involves the integration over elementwise subregions. Evaluating these integrals analytically is usually complicated, even impossible. That is why numerical integration is used. In practice, we replace integrals by finite sums, in which we use Gauss-Lobatto weights and nodal points. The values of these sums are computed element by element with the Gauss-Lobatto integration rule. Collocation points are now the nodes of the spectral element. All but one of the shape functions will be zero at a particular node. Thus, $\mathcal{M}_{ij} = 0$ for $i \neq j$, which means that the matrix $\mathcal{M}$ is diagonal.

6.3 Time discretization

The time interval $[0, T]$ is divided into N timesteps, each of length $\Delta t = T/N$. After replacing the time derivatives in the semidiscretized form (73) by the appropriate approximations and taking into account the initial conditions we obtain the fully discrete state equation.

The time discretization of the semi-discrete equation is performed with the central finite differences (CD) in Section 6.3.1 and with the fourth order Runge-Kutta (RK) method in Section 6.3.2. The CD method is second order accurate, while the RK method is fourth order accurate, with respect to the timestep Δt. Both methods lead to an explicit time stepping scheme. These properties are essential for computational efficiency. Only matrix-vector products are needed in time-dependent simulation, but the scheme needs to satisfy the stability condition, which limits the length of the time step.

6.3.1 Central finite difference method

The time discretization is based on second order centered finite difference method such that the time derivatives are, for $i = 0, \ldots, N$, replaced by the following approximations:

$$\frac{\partial \mathbf{u}}{\partial t} = \frac{\mathbf{u}^{i+1} - \mathbf{u}^{i-1}}{2\Delta t} \tag{90}$$

$$= \frac{\mathbf{u}(\mathbf{x}, t_{i+1}) - \mathbf{u}(\mathbf{x}, t_{i-1})}{t_{i+1} - t_{i-1}}, \tag{91}$$

$$\frac{\partial^2 \mathbf{u}}{\partial t^2} = \frac{\mathbf{u}^{i+1} - 2\mathbf{u}^{i} + \mathbf{u}^{i-1}}{\Delta t^2} \tag{92}$$

$$= \frac{\mathbf{u}(\mathbf{x}, t_{i+1}) - 2\mathbf{u}(\mathbf{x}, t_i) + \mathbf{u}(\mathbf{x}, t_{i-1})}{(\frac{1}{2}(t_{i+1} - t_{i-1}))^2}, \tag{93}$$

where $t_i = i\Delta t$.

After time discretization and taking into account the initial conditions (48)-(49) or (58)-(59), we get

$$\mathcal{M}\frac{\mathbf{u}^{i+1} - 2\mathbf{u}^{i} + \mathbf{u}^{i-1}}{\Delta t^2} + \mathcal{S}\frac{\mathbf{u}^{i+1} - \mathbf{u}^{i-1}}{2\Delta t} + \mathcal{K}\mathbf{u}^{i} = \mathcal{F}^{i}, \tag{94}$$

$$\mathbf{u}^0 = \mathbf{e}_0 \tag{95}$$

$$\frac{\mathbf{u}^1 - \mathbf{u}^{-1}}{2\Delta t} = \mathbf{e}_1. \tag{96}$$

With central finite differences, the fully discrete state equation can be represented in the matrix form

$$\begin{pmatrix} \mathcal{I} & & & & & \\ \mathcal{C}_0 & \mathcal{M} & & & & \\ \mathcal{B} & \mathcal{C} & \mathcal{D} & & & \\ & \ddots & \ddots & \ddots & & \\ & & \mathcal{B} & \mathcal{C} & \mathcal{D} & \\ & & & \mathcal{B} & \mathcal{C} & \mathcal{D} \end{pmatrix} \begin{pmatrix} \mathbf{u}^0 \\ \mathbf{u}^1 \\ \vdots \\ \vdots \\ \mathbf{u}^N \\ \mathbf{u}^{N+1} \end{pmatrix} - \begin{pmatrix} \mathcal{I} & 0 \\ 0 & \Delta t\,\mathcal{B} \\ 0 & 0 \\ \vdots & \vdots \\ \vdots & \vdots \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{e}_0 \\ \mathbf{e}_1 \end{pmatrix} - \Delta t^2 \begin{pmatrix} 0 \\ \frac{1}{2}\mathcal{F}^0 \\ \mathcal{F}^1 \\ \vdots \\ \vdots \\ \mathcal{F}^N \end{pmatrix} = 0, \tag{97}$$

where $\mathbf{u}^i$ and $\mathcal{F}^i$ are the vectors $\mathbf{u}$ and $\mathcal{F}$ at the time instant $i\Delta t$, and $\mathbf{e}_0$ and $\mathbf{e}_1$ are the initial conditions. The matrix blocks $\mathcal{C}_0$, $\mathcal{B}$, $\mathcal{C}$ and $\mathcal{D}$ are given by the formulas

$$\mathcal{C}_0 = \frac{\Delta t^2}{2}\mathcal{K} - \mathcal{M}, \tag{98}$$

$$\mathcal{D} = \mathcal{M} + \frac{\Delta t}{2}\mathcal{S}, \tag{99}$$

$$\mathcal{C} = \Delta t^2\mathcal{K} - 2\mathcal{M}, \tag{100}$$

$$\mathcal{B} = \mathcal{M} - \frac{\Delta t}{2}\mathcal{S}, \tag{101}$$

while $\mathcal{I}$ is the identity matrix. The form (97) is further used to derive the adjoint state equation in Section 7.

Next, we denote by $\hat{n}$ the number of diagonal elements in the matrix $\mathcal{M}$ and consider the time complexity of computing the state equation. When solving the state equation (97), inversion of the mass matrix $\mathcal{M}$ and some matrix-vector products are needed at each timestep. Because of the diagonality of the matrix $\mathcal{M}$, its inverse $\mathcal{M}^{-1}$ is obtained simply by inverting each diagonal element. Since also the matrix $\mathcal{S}$ contains only diagonal blocks, a matrix-vector product with matrix $\mathcal{M}$, $\mathcal{M}^{-1}$, $\mathcal{S}$, or $\mathcal{S}^{-1}$ (or some linear combination of these) takes only order of $\hat{n}$ operations. The matrix-vector multiplication involving the sparse stiffness matrix $\mathcal{K}$ is more demanding. Even if only non-zero matrix entries are multiplied, the operation count is proportional to the number of nonzero elements in $\mathcal{K}$, which is of order $r^2\hat{n}$. This means that the computational cost for each timestep of the state equation with the CD time stepping is of order $O(r^2\hat{n})$.

It is worth mentioning that the spatial discretization with SEM is very accurate since it is based on high-degree polynomials. If the time discretization is of low order, it reduces the global accuracy of the scheme. In this respect, we present also a Runge-Kutta method which is fourth order accurate.

6.3.2 Fourth order Runge-Kutta method

The state equation of the form (73) can be presented as a system of differential equations

$$\mathbf{y}'(t) = f(t, \mathbf{y}(t)), \tag{102}$$

where $\mathbf{y}'(t) = \left(\frac{\partial \mathbf{u}}{\partial t}, \frac{\partial^2 \mathbf{u}}{\partial t^2}\right)^T$, $\mathbf{y} = (\mathbf{u}, \mathbf{v})^T$ is a vector of time stepping variables $\mathbf{u}$ and $\mathbf{v}$, and the function $f(t, \mathbf{y}(t)) = (f_1(t, \mathbf{u}, \mathbf{v}), f_2(t, \mathbf{u}, \mathbf{v}))^T$ has components

$$f_1(t, \mathbf{u}, \mathbf{v}) = \mathbf{v}, \tag{103}$$

$$f_2(t, \mathbf{u}, \mathbf{v}) = -\mathcal{M}^{-1}\left(\mathcal{S}\mathbf{v} + \mathcal{K}\mathbf{u} - \mathcal{F}\right). \tag{104}$$

The fourth order Runge-Kutta method is a Taylor series method. In general, Taylor series methods have a feature to keep the errors small, but there is also a disadvantage of requiring the evaluation of higher derivatives of the function $f(t, \mathbf{y}(t))$. The advantage of Runge-Kutta method is that explicit evaluations of the derivatives of the function $f(t, \mathbf{y}(t))$ are not required, but linear combinations of values of $f(t, \mathbf{y}(t))$ are used to approximate $\mathbf{y}(t)$.

The fourth order Runge-Kutta algorithm is

$$\mathbf{y}^{i+1} = \mathbf{y}^i + \frac{1}{6}\left(\begin{pmatrix} k_{11} \\ k_{12} \end{pmatrix} + 2\begin{pmatrix} k_{21} \\ k_{22} \end{pmatrix} + 2\begin{pmatrix} k_{31} \\ k_{32} \end{pmatrix} + \begin{pmatrix} k_{41} \\ k_{42} \end{pmatrix}\right) \tag{105}$$

where $\mathbf{u}^i$ and $\mathbf{v}^i$ are the first and second components of $\mathbf{y}^i = \left(\mathbf{u}^i, \frac{\partial \mathbf{u}^i}{\partial t}\right)^T$, $i = 1, \ldots, N$. The initial condition is set as $\mathbf{y}^0 = (\mathbf{e}_0, \mathbf{e}_1)^T$, and $k_j = (k_{j1}, k_{j2})^T, j = 1, 2, 3, 4$, are the gradient estimates (one for each substep) as follows:

$$\begin{pmatrix} k_{11} \\ k_{12} \end{pmatrix} = \begin{pmatrix} \Delta t f_1(t_i, \mathbf{u}^i, \mathbf{v}^i) \\ \Delta t f_2(t_i, \mathbf{u}^i, \mathbf{v}^i) \end{pmatrix} \tag{106}$$

$$\begin{pmatrix} k_{21} \\ k_{22} \end{pmatrix} = \begin{pmatrix} \Delta t f_1(t_i + \frac{\Delta t}{2}, \mathbf{u}^i + \frac{k_{11}}{2}, \mathbf{v}^i + \frac{k_{12}}{2}) \\ \Delta t f_2(t_i + \frac{\Delta t}{2}, \mathbf{u}^i + \frac{k_{11}}{2}, \mathbf{v}^i + \frac{k_{12}}{2}) \end{pmatrix} \tag{107}$$

$$\begin{pmatrix} k_{31} \\ k_{32} \end{pmatrix} = \begin{pmatrix} \Delta t f_1(t_i + \frac{\Delta t}{2}, \mathbf{u}^i + \frac{k_{21}}{2}\, \mathbf{v}^i + \frac{k_{22}}{2}) \\ \Delta t f_2(t_i + \frac{\Delta t}{2}, \mathbf{u}^i + \frac{k_{21}}{2}\, \mathbf{v}^i + \frac{k_{22}}{2}) \end{pmatrix} \tag{108}$$

$$\begin{pmatrix} k_{41} \\ k_{42} \end{pmatrix} = \begin{pmatrix} \Delta t f_1(t_i + \Delta t, \mathbf{u}^i + k_{31}, \mathbf{v}^i + k_{32}) \\ \Delta t f_2(t_i + \Delta t, \mathbf{u}^i + k_{31}, \mathbf{v}^i + k_{32}) \end{pmatrix}. \tag{109}$$

In other words, at each timestep i the function f is evaluated four times by using the formulas (103)-(104) to get the gradient estimates (106)-(109), and then the successive approximation of $\mathbf{y}$ is calculated by the formula (105). Because of the diagonality of the matrix $\mathcal{M}$, the only matrix inversion needed in time stepping (e.g $\mathcal{M}^{-1}$ in equation (104)) is computed simply by inverting each diagonal element in the matrix $\mathcal{M}$. This requires only $\hat{n}$ floating point operations, which is the number of diagonal elements in the matrix $\mathcal{M}$, and known as the number of degrees of freedom in the space discretization. Since the matrix $\mathcal{S}$ contains only diagonal blocks, also the operation count of the matrix-vector product $\mathcal{S}v$ is of order $\hat{n}$. In the matrix-vector multiplication involving the sparse stiffness matrix $\mathcal{K}$, only non-zero matrix entries are multiplied, which requires order of $r^2\hat{n}$ operations. Besides these, $2\hat{n}$ additions and $3\hat{n}$ multiplications are needed for a single evaluation of the function f. According to (105), computation of $\mathbf{y}^i$ needs $14\hat{n}$ floating point operations. Thus, the computational cost for each timestep of the state equation is of order $O(r^2\hat{n})$ also with RK time stepping. Although the computational demand is of the same order for both CD and RK time steppings, the number of floating point operators needed for RK is nearly four times that of CD.

In practice, the solution $\mathbf{y}^i$ at $t = i\Delta t$ is achieved by first solving $k = (k_1, k_2, k_3, k_4)^T$ from the equation

$$\begin{pmatrix} \mathcal{I} & & & \\ \hat{D} & \mathcal{I} & & \\ & \hat{D} & \mathcal{I} & \\ & & 2\hat{D} & \mathcal{I} \end{pmatrix} \begin{pmatrix} k_1 \\ k_2 \\ k_3 \\ k_4 \end{pmatrix} + \begin{pmatrix} 2\hat{D} \\ 2\hat{D} \\ 2\hat{D} \\ 2\hat{D} \end{pmatrix} \mathbf{y}^{i-1} - \begin{pmatrix} \mathcal{H}^{i-1} \\ \mathcal{H}^{i-\frac{1}{2}} \\ \mathcal{H}^{i-\frac{1}{2}} \\ \mathcal{H}^{i} \end{pmatrix} = 0, \tag{110}$$

where $\mathcal{I}$ is the identity matrix, $\mathcal{F}^i$ is the vector $\mathcal{F}$ at time $t = i\Delta t$, and the matrix block $\hat{D}$ and the vector blocks $\mathcal{H}^i$, are given by the formulas

$$\hat{D} = \begin{pmatrix} 0 & -\frac{\Delta t}{2}\mathcal{I} \\ \frac{\Delta t}{2}\mathcal{M}^{-1}\mathcal{K} & \frac{\Delta t}{2}\mathcal{M}^{-1}\mathcal{S} \end{pmatrix}, \tag{111}$$

$$\mathcal{H}^i = \begin{pmatrix} \Delta t\mathcal{M}^{-1}\mathcal{F}^i \\ 0 \end{pmatrix}. \tag{112}$$

Then, $\mathbf{y}^i$ is solved from the equation

$$\mathbf{y}^i = \mathbf{y}^{i-1} - \begin{pmatrix} \hat{R} & 2\hat{R} & 2\hat{R} & \hat{R} \end{pmatrix} \begin{pmatrix} k_1 \\ k_2 \\ k_3 \\ k_4 \end{pmatrix}, \tag{113}$$

with

$$\hat{R} = \begin{pmatrix} -\frac{1}{6}\mathcal{I} & 0 \\ 0 & -\frac{1}{6}\mathcal{I} \end{pmatrix}. \tag{114}$$

When using the fourth order Runge-Kutta method, the state equation can be represented in the simplified form:

$$\begin{pmatrix} \mathcal{I} & & & & \\ \mathcal{N} & \mathcal{I} & & & \\ & \ddots & \ddots & & \\ & & \mathcal{N} & \mathcal{I} & \\ & & & \mathcal{N} & \mathcal{I} \end{pmatrix} \begin{pmatrix} \mathbf{y}^0 \\ \mathbf{y}^1 \\ \vdots \\ \mathbf{y}^{N-1} \\ \mathbf{y}^N \end{pmatrix} - \begin{pmatrix} \mathcal{I} \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix} \mathbf{y}^0 - \begin{pmatrix} 0 \\ \hat{F}^1 \\ \vdots \\ \hat{F}^{N-1} \\ \hat{F}^N \end{pmatrix} = 0, \tag{115}$$

where $\mathcal{N} = -(\hat{B}\hat{A}^{-1}\hat{C} + \mathcal{I})$ and $\hat{F}^i = -\hat{B}\hat{A}^{-1}\hat{H}$ such that

$$\hat{A} = \begin{pmatrix} \mathcal{I} & & & \\ \hat{D} & \mathcal{I} & & \\ & \hat{D} & \mathcal{I} & \\ & & 2\hat{D} & \mathcal{I} \end{pmatrix}, \quad \hat{C} = \begin{pmatrix} 2\hat{D} \\ 2\hat{D} \\ 2\hat{D} \\ 2\hat{D} \end{pmatrix},$$

$$\hat{B} = \begin{pmatrix} \hat{R} & 2\hat{R} & 2\hat{R} & \hat{R} \end{pmatrix}, \quad \hat{H}^T = \Delta t\mathcal{M}^{-1} \begin{pmatrix} \mathcal{F}^{i-1} & 0 & \mathcal{F}^{i-\frac{1}{2}} & 0 & \mathcal{F}^{i-\frac{1}{2}} & 0 & \mathcal{F}^{i} & 0 \end{pmatrix}.$$

In the next section when describing the control algorithm we use for the state equation the short form

$$s(\mathbf{e}, \mathbf{u}(\mathbf{e})) = 0, \tag{116}$$

where $\mathbf{e} = (\mathbf{e}_0, \mathbf{e}_1)^T$ contains the initial values and $\mathbf{u}$ the vectors $\mathbf{u}^i$. We denote the state equation by $s_0(\mathbf{e}, \mathbf{u}(\mathbf{e})) = 0$ in the special case with $\mathcal{F}^i = 0$ for all i.

7 CONTROL PROBLEM

The exact controllability problem for computing $T-$periodic solution for the wave equation involves finding such initial conditions $\mathbf{e}_0$ and $\mathbf{e}_1$ that the solution $\mathbf{u}$ and its time derivative $\frac{\partial \mathbf{u}}{\partial t}$ at time T would coincide with the initial conditions. For the numerical solution, this problem is formulated as a least-squares optimization problem with the cost function

$$J(\mathbf{e}, \mathbf{u}(\mathbf{e})) = \frac{1}{2}\left(\int_{\Omega} \left|\frac{\partial \mathbf{u}(\mathbf{x}, T)}{\partial t} - \mathbf{e}_1\right|^2 + \left|\nabla(\mathbf{u}(\mathbf{x}, T) - \mathbf{e}_0)\right|^2 dx\right) \tag{117}$$

where $\mathbf{e} = (\mathbf{e}_0, \mathbf{e}_1) \in Z$ and $\mathbf{u}$ is the solution of the initial value problem [16]. After the spatial discretization, the cost function to be minimized is of the form

$$J(\mathbf{e}, \mathbf{u}(\mathbf{e})) = \frac{1}{2}\left(\mathbf{u}^N - \mathbf{e}_0\right)^T \mathcal{K}\left(\mathbf{u}^N - \mathbf{e}_0\right) + \frac{1}{2}\left(\frac{\partial \mathbf{u}^N}{\partial t} - \mathbf{e}_1\right)^T \mathcal{M}\left(\frac{\partial \mathbf{u}^N}{\partial t} - \mathbf{e}_1\right), \tag{118}$$

where $\frac{\partial \mathbf{u}^N}{\partial t}$ are replaced by appropriate approximations and $\mathbf{u}^i$ are given by equation (97) or (115). When we find $\mathbf{e} \in Z$ such that $J(\mathbf{e}, \mathbf{u}(\mathbf{e})) = 0$ the conditions (50) and (51) in the case of Helmholtz equation, or (61) and (62) in the case of Navier equation, are also satisfied and we have the time-harmonic solution.

The least squares formulation is given by

$$\min_{\mathbf{e} \in Z} J(\mathbf{e}). \tag{119}$$

The solution of the minimization problem is equivalent to finding such $\mathbf{e}^* \in Z$ that $\nabla J(\mathbf{e}^*, \mathbf{u}(\mathbf{e}^*)) = 0$, which is actually a linear system because J is a quadratic functional. We use the preconditioned conjugate gradient (CG) method for this system (see, e.g., [41, 42, 104]). Each iteration step involves computation of the gradient of J, which is an essential stage of the algorithm.

In [16], the gradient was derived for the continuous cost function (117) and the same formulas were used also on the discrete level. This approach does not lead exactly to the

gradient of the function to be minimized. That is why we proceed in different order and discretize the problem before deriving the gradient formulas (see Section 7.2). To compare the two ways to compute the gradient, we represent in Section 7.1 the formulas in [16] by using the same discerete notation as in Section 7.2. However, our experiments indicate that the two ways to compute the gradient lead to practically the same convergence for the CG method (see also [107]).

7.1 Gradient of the continuous cost function with central finite difference method

Starting from the continuous cost function (117), the state equation is just the one expressed in (97), and the adjoint state equation can be presented in the matrix form

$$\begin{pmatrix} \mathcal{D} & \mathcal{C} & \mathcal{B} & & & \\ & \mathcal{D} & \mathcal{C} & \mathcal{B} & & \\ & & \ddots & \ddots & \ddots & \\ & & & \mathcal{D} & \mathcal{C} & \mathcal{B} \\ & & & & \mathcal{M} & \mathcal{C}_N \\ & & & & & \mathcal{I} \end{pmatrix} \begin{pmatrix} \mathbf{p}^{-1} \\ \mathbf{p}^0 \\ \vdots \\ \mathbf{p}^{N-2} \\ \mathbf{p}^{N-1} \\ \mathbf{p}^N \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ a_{N-1} \\ a_N \end{pmatrix}, \tag{120}$$

where

$$\mathcal{C}_N = \mathcal{C}_0 + \Delta t \mathcal{S}\mathcal{M}^{-1}, \tag{121}$$

$$a_N = \frac{\partial \mathbf{u}^N}{\partial t} - \mathbf{e}_1, \tag{122}$$

$$a_{N-1} = \Delta t \mathcal{B}\mathcal{M}^{-1}\mathcal{K}(\mathbf{u}^N - \mathbf{e}_0). \tag{123}$$

Solutions of the adjoint state equation (120) are $\mathbf{p}^0$ and $\frac{\partial \mathbf{p}^0}{\partial t} = \frac{\mathbf{p}^1 - \mathbf{p}^{-1}}{2\Delta t}$. Gradient components are then computed by the formulas

$$\frac{dJ(\mathbf{e}, \mathbf{u}(\mathbf{e}))}{d\mathbf{e}_0} = \mathcal{K}(\mathbf{e}_0 - \mathbf{u}^N) - \mathcal{M}\frac{\partial \mathbf{p}^0}{\partial t} + \mathcal{S}\mathbf{p}^0, \tag{124}$$

$$\frac{dJ(\mathbf{e}, \mathbf{u}(\mathbf{e}))}{d\mathbf{e}_1} = \mathcal{M}\left(\mathbf{p}^0 + \mathbf{e}_1 - \frac{\partial \mathbf{u}^N}{\partial t}\right). \tag{125}$$

which can be compared to the formulas (132)-(133) below.

7.2 Gradient of the discretized cost function

State equations (97) and (115) can be represented in the residual form (116), and by the adjoint equation technique we see that

$$\frac{dJ(\mathbf{e},\mathbf{u}(\mathbf{e}))}{d\mathbf{e}_k} = \frac{\partial J(\mathbf{e},\mathbf{u})}{\partial \mathbf{e}_k} - \mathbf{p}^T \frac{\partial s(\mathbf{e},\mathbf{u})}{\partial \mathbf{e}_k}, \quad k = 0, 1, \tag{126}$$

where $\mathbf{p}$ is the solution of the adjoint equation

$$\left(\frac{\partial s(\mathbf{e},\mathbf{u})}{\partial \mathbf{u}}\right)^T \mathbf{p} = \left(\frac{\partial J(\mathbf{e},\mathbf{u})}{\partial \mathbf{u}}\right)^T. \tag{127}$$

The state equation (116) is also called the forward equation because it is solved by advancing forward in time. The adjoint equation (127) requires advancing backward in time, so it is called the backward equation [16].

7.2.1 Adjoint equation with the central finite difference method

In the matrix form, the adjoint equation corresponding to the state equation (97) is given by

$$\begin{pmatrix} \mathcal{I} & \mathcal{C}_0 & \mathcal{B} & & & \\ & \mathcal{M} & \mathcal{C} & \mathcal{B} & & \\ & & \mathcal{D} & \ddots & \ddots & \\ & & & \ddots & \ddots & \mathcal{B} \\ & & & & \mathcal{D} & \mathcal{C} \\ & & & & & \mathcal{D} \end{pmatrix} \begin{pmatrix} \mathbf{p}^0 \\ \mathbf{p}^1 \\ \vdots \\ \vdots \\ \mathbf{p}^N \\ \mathbf{p}^{N+1} \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \frac{\partial J}{\partial \mathbf{u}^{N-1}} \\ \frac{\partial J}{\partial \mathbf{u}^N} \\ \frac{\partial J}{\partial \mathbf{u}^{N+1}} \end{pmatrix}, \tag{128}$$

where

$$\frac{\partial J}{\partial \mathbf{u}^{N-1}} = \frac{1}{2\Delta t}\mathcal{M}(\mathbf{e}_1 - \frac{\partial \mathbf{u}^N}{\partial t}), \tag{129}$$

$$\frac{\partial J}{\partial \mathbf{u}^{N+1}} = \frac{1}{2\Delta t}\mathcal{M}(\frac{\partial \mathbf{u}^N}{\partial t} - \mathbf{e}_1), \tag{130}$$

$$\frac{\partial J}{\partial \mathbf{u}^N} = \mathcal{K}(\mathbf{u}^N - \mathbf{e}_0). \tag{131}$$

The gradient components are then the following:

$$\frac{dJ(\mathbf{e},\mathbf{u}(\mathbf{e}))}{d\mathbf{e}_0} = \mathcal{K}(\mathbf{e}_0 - \mathbf{u}^N) + \mathbf{p}^0, \tag{132}$$

$$\frac{dJ(\mathbf{e},\mathbf{u}(\mathbf{e}))}{d\mathbf{e}_1} = \mathcal{M}\left(\mathbf{e}_1 - \frac{\partial \mathbf{u}^N}{\partial t}\right) + \Delta t \mathcal{B}\mathbf{p}^1. \tag{133}$$

7.2.2 Adjoint equation with the fourth order Runge-Kutta method

The adjoint equation corresponding to the state equation (115) is

$$\begin{pmatrix} \mathcal{I} & \mathcal{N}^T & & & \\ & \mathcal{I} & \mathcal{N}^T & & \\ & \ddots & \ddots & & \\ & & & \mathcal{I} & \mathcal{N}^T \\ & & & & \mathcal{I} \end{pmatrix} \begin{pmatrix} \mathbf{q}^0 \\ \mathbf{q}^1 \\ \vdots \\ \mathbf{q}^{N-1} \\ \mathbf{q}^N \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \frac{\partial J(\mathbf{e},\mathbf{u}(\mathbf{e}))}{\partial \mathbf{y}^N} \end{pmatrix}, \tag{134}$$

where $\mathbf{q}^i = \left(\mathbf{p}^i, \frac{\partial \mathbf{p}^i}{\partial t}\right)$, $i = N-1,\ldots,0$, the transpose of the matrix $\mathcal{N}$ is $\mathcal{N}^T = -(\hat{C}^T(\hat{A}^T)^{-1}\hat{B} + \mathcal{I})$, and

$$\frac{\partial J(\mathbf{e},\mathbf{u}(\mathbf{e}))}{\partial \mathbf{y}^N} = \begin{pmatrix} \frac{\partial J(\mathbf{e},\mathbf{u}(\mathbf{e}))}{\partial \mathbf{u}^N} \\ \frac{\partial J(\mathbf{e},\mathbf{u}(\mathbf{e}))}{\partial \left(\frac{\partial \mathbf{u}^N}{\partial t}\right)} \end{pmatrix} = \begin{pmatrix} \mathcal{K}(\mathbf{u}^N - \mathbf{e}_0) \\ \mathcal{M}(\frac{\partial \mathbf{u}^N}{\partial t} - \mathbf{e}_1) \end{pmatrix}. \tag{135}$$

So, the starting value for $\mathbf{q}^i$, at time $t = T = N\Delta t$, is

$$\mathbf{q}^N = \begin{pmatrix} \mathcal{K}(\mathbf{u}^N - \mathbf{e}_0) \\ \mathcal{M}(\frac{\partial \mathbf{u}^N}{\partial t} - \mathbf{e}_1) \end{pmatrix}.$$

At each time step, $k = (k_1, k_2, k_3, k_4)^T$ is solved from the equation

$$\begin{pmatrix} \mathcal{I} & \hat{D}^T & & \\ & \mathcal{I} & \hat{D}^T & \\ & & \mathcal{I} & 2\hat{D}^T \\ & & & \mathcal{I} \end{pmatrix} \begin{pmatrix} k_1 \\ k_2 \\ k_3 \\ k_4 \end{pmatrix} = \begin{pmatrix} -\hat{R} \\ -2\hat{R} \\ -2\hat{R} \\ -\hat{R} \end{pmatrix} \mathbf{q}^{i+1}, \tag{136}$$

and $\mathbf{q}^i$ is computed by the formula

$$\mathbf{q}^i = \mathbf{q}^{i+1} - \begin{pmatrix} 2\hat{D}^T & 2\hat{D}^T & 2\hat{D}^T & 2\hat{D}^T \end{pmatrix} \begin{pmatrix} k_1 \\ k_2 \\ k_3 \\ k_4 \end{pmatrix}. \tag{137}$$

After this, we get the gradient components, which are

$$\frac{dJ(\mathbf{e},\mathbf{u}(\mathbf{e}))}{d\mathbf{e}_0} = \mathcal{K}(\mathbf{e}_0 - \mathbf{u}^N) + \mathbf{p}^0, \tag{138}$$

$$\frac{dJ(\mathbf{e},\mathbf{u}(\mathbf{e}))}{d\mathbf{e}_1} = \mathcal{M}\left(\mathbf{e}_1 - \frac{\partial \mathbf{u}^N}{\partial t}\right) + \frac{\partial \mathbf{p}^0}{\partial t}. \tag{139}$$

8 OPTIMIZATION ALGORITHM

The purpose is to minimize functional J, which depends on the initial conditions both directly and indirectly through the solution of the wave equation. Since vector $\mathbf{u}$ depends linearly on the initial conditions $\mathbf{e}_0$ and $\mathbf{e}_1$, J is a quadratic function. Hence, (118) can be minimized by solving the linear system $\nabla J(\mathbf{e}_0, \mathbf{e}_1) = 0$. We solve this linear system with an optimization method, which requires the gradient of the functional J with respect to the control variables $\mathbf{e}_0$ and $\mathbf{e}_1$. Computation of the initial approximation for the method is described in Section 8.1.

Conjugate gradient (CG) method is a numerical method for solution of systems of linear equations with a symmetric and positive definite matrix. If the unpreconditioned CG algorithm is used, solution of the least-squares problem is slow, and the number of iterations grows rapidly with the order of elements [76]. For preconditioning the conjugate gradient algorithm, we use a modification of Kickinger's [61] algebraic multigrid (AMG), which is introduced in [78]. That is why we first present a brief overview of multigrid methods in Section 8.2, and then consider the preconditioned conjugate gradient algorithm in Section 8.3.

8.1 Computation of the initial approximation

It is important to have smooth initial approximations for $\mathbf{e}_0$ and $\mathbf{e}_1$, which satisfy the boundary conditions. In [16], a special procedure suggested by Mur in [82, p. 950] was used, which leads to faster convergence to the time-harmonic solution. We apply the same procedure, and first define a smooth transition function $\theta(t)$, which increases from zero to one in the time interval $[0, T_{\mathrm{tr}}]$:

$$\theta(t) = \begin{cases} \left(2 - \sin\left(\pi t / 2T_{\mathrm{tr}}\right)\right) \sin\left(\pi t / 2T_{\mathrm{tr}}\right), & \text{if } 0 \le t \le T_{\mathrm{tr}}, \\ 1, & \text{if } t \ge T_{\mathrm{tr}}. \end{cases} \tag{140}$$

The length of the time interval should be chosen as a multiple of the period T, i.e., $T_{\mathrm{tr}} = nT$ with n a positive integer. Then, for example in the case of the acoustic wave equation, we solve the following initial value problem:

$$\frac{1}{\rho_f(\mathbf{x})c(\mathbf{x})^2}\frac{\partial^2 w}{\partial t^2} - \nabla\cdot\left(\frac{1}{\rho_f(\mathbf{x})}\nabla w\right) = \theta(t)f, \qquad \text{in } Q = \Omega\times[0,T_{\text{tr}}], \tag{141}$$

$$w = 0, \qquad \text{on } \gamma_0 = \Gamma_0\times[0,T_{\text{tr}}], \tag{142}$$

$$\frac{1}{c(\mathbf{x})}\frac{\partial w}{\partial t} + \frac{\partial w}{\partial \mathbf{n}} = \theta(t)y_{\text{ext}}, \qquad \text{on } \gamma_{\text{ext}} = \Gamma_{\text{ext}}\times[0,T_{\text{tr}}], \tag{143}$$

$$w(\mathbf{x},0) = 0 \qquad \text{in } \Omega, \tag{144}$$

$$\frac{\partial w}{\partial t}(\mathbf{x},0) = 0 \qquad \text{in } \Omega. \tag{145}$$

The initial approximations for the control variables $\mathbf{e}_0$ and $\mathbf{e}_1$ are now the solution w and its time derivative at time T_{tr}. If the obstacle Θ of the scattering problem is convex, there are no interacting reflections, and already this initial procedure may converge rapidly to the time-harmonic solution. However, in general the convergence is slow and we need to continue with the control algorithm.

8.2 Multigrid methods

Multigrid methods [45, 98] are well known as efficient solution techniques for solving linear systems of the form

$$\mathcal{K}\tilde{\mathbf{w}} = \tilde{\mathbf{g}}, \tag{146}$$

where $\tilde{\mathbf{w}}$ is an unknown vector, and $\tilde{\mathbf{g}}$ and $\mathcal{K}$ are known vector and matrix, respectively. These methods can be used as stand-alone solvers or as preconditioners for iterative solvers. The key idea of multigrid methods is to accelerate the solution by transfering the residuals from fine to coarse grids, computing corrections on a coarser grid and then interpolating them back to the fine grid problem. This procedure is applied recursively such that a hierarchy of fine and coarser grid levels is used to seek the solution for the finest mesh. In practice, the equation (146) is solved iteratively from an approximation $\tilde{\mathbf{w}}_0$. For a particular level l, the residual is given by $\tilde{r}_l = \tilde{\mathbf{g}} - \mathcal{K}\tilde{\mathbf{w}}_l$. This is used as the basis of a correction equation $\tilde{\mathbf{w}}_{l+1} = \tilde{\mathbf{w}}_l + \tilde{e}_l$. The error $\tilde{e}_l$ is related to the residual by $\mathcal{K}\tilde{e}_l = \tilde{r}_l$.

In algebraic multigrid (AMG) methods [79, 90, 91, 94], the central idea is to develop a sequence of problems

$$\mathcal{K}_l\tilde{\mathbf{w}}_l = \tilde{\mathbf{g}}_l, \tag{147}$$

corresponding to grid levels $l = 0,\ldots,\tilde{k}$, where $\tilde{k}$ represents the coarsest level. Contrary to classical geometric multigrid methods, actual coarsenig of the given mesh is not needed for finding coarser grid levels. This requirement is loosened in graph based algebraic multigrid methods (see e.g. [61, 78]) so that the finest discretization mesh is used to

construct the coarser level components. Instead of actual values stored in the matrix $\mathcal{K}$, the topological graph of this matrix is used to form a coarser graph. This coarsening process operates in a geometric fashion by sequentially choosing a coarse level node and eliminating the neighboring nodes of the graph. We use a modification of Kickinger's [61] algebraic multigrid method which is introduced in [78]. In selecting the unknowns for coarser levels, the primary criterion is to take the node with minimum degree when eliminations have taken into account. The secondary criterion is to follow the original node numbering.

The use of AMG methods for spectral elements has recently been studied in [54]. The number of connections between unknowns of the problem increases when higher order elements are used. In that case, the coarsening strategy described above leads to unacceptably coarse systems and convergence factor of AMG degrades rapidly as the order of the approximation polynomials increases. We have overcome this problem by employing a graph that is constructed so that unknowns are connected to each other as if low order finite elements would have been used in the discretization process. Only the unknowns corresponding to the nearest neighbouring Gauss-Lobatto points are connected to each other. Additionally, in vector valued problems it is necessary to prevent mixture of various types of unknowns also on coarser levels. This is achieved by giving the method an initial graph where the sets of graph nodes corresponding to different types of unknowns are not interconnected. These settings ensure fast computation of coarser level components and constitute an essential part of graph based coarsening for higher order elements.

Matrices, which are used at multigrid levels, are set as an initialization step of the AMG algorithm by using restriction operators. The restriction operator $\widetilde{\mathbf{R}}_l^{l+1}$ from fine level l to coarse level $l+1$ is

$$\widetilde{\mathbf{R}}_l^{l+1} = \begin{pmatrix} \mathbf{R}_l^{l+1} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_l^{l+1} \end{pmatrix}, \tag{148}$$

where the components of the restriction matrices $\mathbf{R}_l^{l+1}$ are

$$\left(\mathbf{R}_l^{l+1}\right)_{ij} = \begin{cases} 1 & \text{for a fine grid point } j \text{ which is a coarse grid point } i, \\ \frac{1}{k} & \text{for a fine grid point } j \text{ which is a neighbor of coarse grid} \\ & \text{point } i \text{ and has } k \text{ neighboring coarse grid points,} \\ 0 & \text{otherwise.} \end{cases} \tag{149}$$

The algebraic multigrid, which we use, employs a Galerkin projection for the construction of the coarse grid equations. When the fine level matrix $\mathcal{K}_l$ is known, the coarse grid operator is given by the Galerkin formula $\mathcal{K}_{l+1} = \widetilde{\mathbf{R}}_l^{l+1} \mathcal{K}_l (\widetilde{\mathbf{R}}_l^{l+1})^T$. The finest level matrix $\mathcal{K}_0$ is actually the matrix $\mathcal{K}$, and the minimum size of the coarse system is predetermined.

By coarsening the fine grid, the low oscillation error on the fine grid is interpolated into high frequency error on the coarse grid. The key to dealing with high oscillation error is the smoothing process. As a smoother of the AMG we have used successive over relaxation (SOR), with overrelaxation parameter 1.2, unless other mentioned. One iteration of SOR is used for both pre- and post-smoothing. Additionally, in the beginning of

every multigrid iteration, four iterations of SOR are used to smooth the solution initially. Through successive coarsening levels, the low oscillation error increases, which makes possible to identify and remove the error at a fast rate. Three main cycling strategies of AMG are V-cycle, W-cycle, and full multigrid scheme [45, 92]. At this stage, we utilize the W-cycle until the residual norm of the solution is smaller than 10^{-6}.

Each AMG iteration starts with the finest level matrix $\mathcal{K}_0$, right hand side vetor $\tilde{\mathbf{g}}_0$ and an approximation $\tilde{\mathbf{w}}_0$. For a particular level l, the residual is given by $\tilde{r}_l = \tilde{\mathbf{g}}_l - \mathcal{K}_l\tilde{\mathbf{w}}_l$. This is used as the basis of a correction equation $\tilde{\mathbf{w}}_l = \tilde{\mathbf{w}}_l + \tilde{e}_l$. The error $\tilde{e}_l$ is related to the residual by $\mathcal{K}_l\tilde{e}_l = \tilde{r}_l$. After pre-smoothing the approximation, the residual is computed and mapped from the fine level l to a coarser level $l+1$ by using the restriction operator $\widetilde{\mathbf{R}}_l^{l+1}$. At the coarsest level $\tilde{k}$, a coarse level problem $\mathcal{K}_{\tilde{k}}\tilde{e}_{\tilde{k}} = \tilde{r}_{\tilde{k}}$ is solved with LU-factorization. The error $\tilde{e}_{l+1}$ of the coarse level $l+1$ is mapped to a finer level l by the prolongation operator $\widetilde{\mathbf{P}}_{l+1}^{l}$, which is chosen to be the transpose of the restriction operator, i.e.

$$\widetilde{\mathbf{P}}_{l+1}^{l} = (\widetilde{\mathbf{R}}_l^{l+1})^T. \tag{150}$$

Prolongated errors are added to the approximate solution to correct the solution on a finer grid. This procedure is known as coarse grid correction.

On the way to the finest level, post-smoothing is utilized, and the W-cycle ends with post-smoothing on the finest level. AMG cycles are repeated until the solution converges, i.e. the residual of the approximation of $\tilde{\mathbf{w}}$ is sufficiently small.

8.3 Preconditioned conjugate gradient algorithm

To improve the convergence rate of the CG algorithm, we use the block-diagonal preconditioner

$$\mathcal{L} = \begin{pmatrix} \mathcal{K} & 0 \\ 0 & \mathcal{M} \end{pmatrix}, \tag{151}$$

where the first and second blocks are associated with the first and second terms in (118), respectively. Solution of a linear system with the preconditioner requires the solution of systems with the stiffness matrix $\mathcal{K}$ and the diagonal mass matrix $\mathcal{M}$. Efficient solution of linear systems with the matrix $\mathcal{K}$ is critical for the overall efficiency of the control method. At this stage, we use the algebraic multigrid method, which is presented in Section 8.2.

Our practical realization of the preconditioned CG algorithm differs from the one in [16] with respect to the spatial discretization, the preconditioning method, and the gradient computation. Each CG iteration requires computation of the gradient ∇J, which involves the solution of the state equation (116) and the corresponding adjoint equation (127). Also solution of one linear system with the preconditioner $\mathcal{L}$ and some matrix-vector operations are needed. Values of the control variables $\mathbf{e}$ at the ith iteration are denoted by $\mathbf{e}_0^i$ and $\mathbf{e}_1^i$. Solution of the adjoint state equation is $\mathbf{p}$, and the gradient variable

is $\mathbf{g} = (g_0, g_1)$. By $s_0(\mathbf{e}, \mathbf{u}(\mathbf{e})) = 0$ we denote the state equation (97), where $\mathcal{F}^i = 0$ for all i. Then, the CG algorithm for solving the least-squares problem is the following:

Algorithm 1 *Preconditioned CG algorithm*

Use method of Section 8.1 to compute the initial values $\mathbf{e}_0^0$ *and* $\mathbf{e}_1^0$.
Solve the state equation $s(\mathbf{e}^0, \mathbf{u}(\mathbf{e}^0)) = 0$.
Solve the adjoint state equation $\left(\frac{\partial s(\mathbf{e}^0, \mathbf{u}(\mathbf{e}^0))}{\partial \mathbf{u}(\mathbf{e}^0)}\right)^T \mathbf{p} = \left(\frac{\partial J(\mathbf{e}^0, \mathbf{u}(\mathbf{e}^0))}{\partial \mathbf{u}(\mathbf{e}^0)}\right)^T$.
Compute the gradient vectors g_0 *and* g_1 *by the formulas* (132)-(133) *or* (138)-(139).
Solve linear system with the preconditioner $\mathcal{L}\mathbf{w} = -\mathbf{g}$.
Set $c_0 = -(\mathbf{w}, \mathbf{g})$, $c = c_0$ *and* $i = 1$.
Repeat until $\sqrt{\frac{c}{c_0}} < \varepsilon$
  Solve the state equation $s_0(\mathbf{w}, \mathbf{u}(\mathbf{w})) = 0$.
  Solve the adjoint state equation $\left(\frac{\partial s(\mathbf{w}, \mathbf{u}(\mathbf{w}))}{\partial \mathbf{u}(\mathbf{w})}\right)^T \mathbf{p} = \left(\frac{\partial J(\mathbf{w}, \mathbf{u}(\mathbf{w}))}{\partial \mathbf{u}(\mathbf{w})}\right)^T$.
  Compute the gradient updates v_0 *and* v_1 *by the formulas* (132)-(133) *or* (138)-(139).
  Compute $\rho = \frac{c}{(\mathbf{w}, \mathbf{v})}$.
  $\mathbf{e}^i = \mathbf{e}^{i-1} + \rho \mathbf{w}$.
  $\mathbf{g} = \mathbf{g} + \rho \mathbf{v}$.
  Solve linear system with the preconditioner $\mathcal{L}\mathbf{v} = -\mathbf{g}$.
  $\gamma = \frac{1}{c}$, $c = -(\mathbf{v}, \mathbf{g})$, $\gamma = c\gamma$.
  $\mathbf{w} = \mathbf{v} + \gamma \mathbf{w}$, $i = i + 1$.

As discussed in Section 6.3, solving the state equation needs $O(r^2 \hat{n})$ floating point operations at each timestep in CD and RK time steppings. From this, we can conclude that the computational demand for computing the solution for the state equation with N timesteps is $O(Nr^2 \hat{n})$. The order of computational operations needed for solving the adjoint state equation is of the same order than for solving the state equation. On the whole, the computational cost for one iteration of the CG algorithm is of order $O(Nr^2 \hat{n})$. Assuming that the number of timesteps N is fixed, number of iterations is approximately constant, and the element order r has small integer values, the computational demand for the overall CG algorithm is $O(\hat{n})$.

9 NUMERICAL EXPERIMENTS

The main goal of these numerical experiments is to study the accuracy of the spatial discretization and its effect on computational complexity. The overall accuracy of the discrete solution given by the controllability method depends on the following factors:

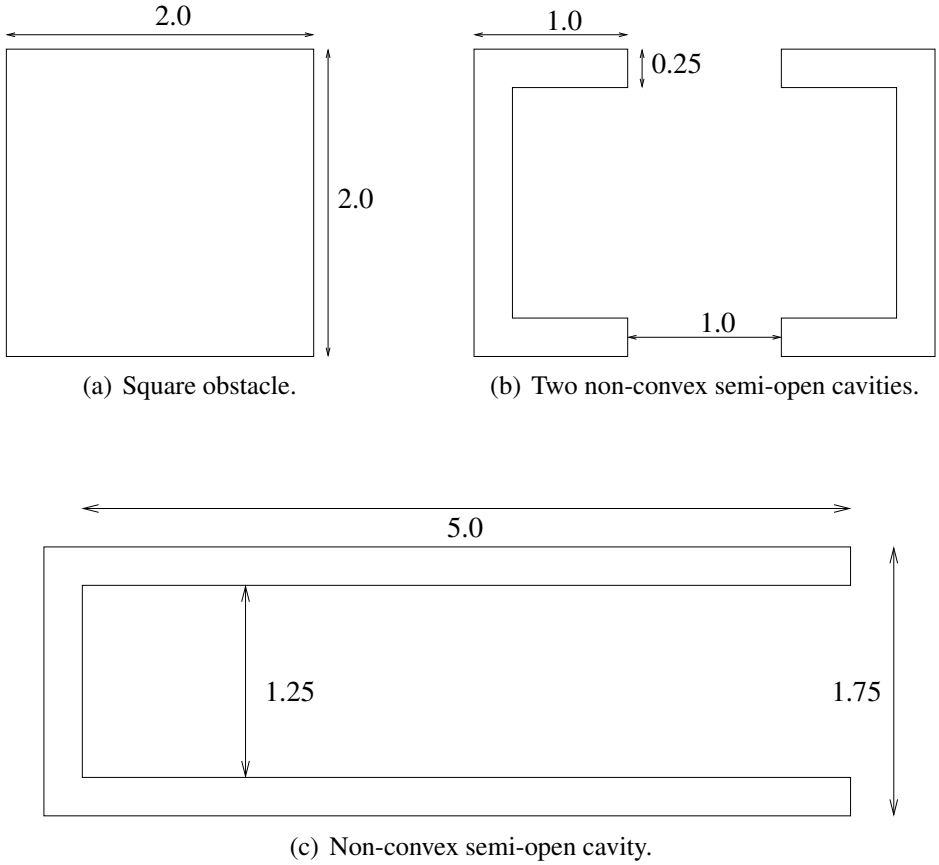
- spatial discretization, which is performed by the spectral element method with mesh density h and element order r,
- time discretization, which is performed by central finite differences or the fourth order Runge-Kutta scheme with timestep Δt,
- stopping criterion ε of the CG method,
- approximation of the geometrical boundaries,
- approximation of the radiation condition.

In what follows, we describe how the numerical experiments are defined to eliminate and isolate the error factors.

Numerical experiments are divided into four parts. First, we consider the acoustic wave equation with incident plane wave $p_{f_{\text{inc}}}(\mathbf{x},t) = \cos(\boldsymbol{\omega}\cdot\mathbf{x})\cos(\omega t) + \sin(\boldsymbol{\omega}\cdot\mathbf{x})\sin(\omega t)$ in Sections 9.1 and 9.2. The two ways to compute the gradient are compared in Section 9.1, while the comparison between CD and RK time discretizations is presented in Section 9.2. The last two sections are devoted to solving the elastic wave equation with incident plane wave

$$\mathbf{u}_{\mathbf{s}\text{inc}} = \begin{pmatrix} \omega_1 \cos(\omega t - \frac{\omega}{c_p}\mathbf{x}\cdot\boldsymbol{\omega}) + \omega_2 \cos(\omega t - \frac{\omega}{c_s}\mathbf{x}\cdot\boldsymbol{\omega}) \\ \omega_2 \cos(\omega t - \frac{\omega}{c_p}\mathbf{x}\cdot\boldsymbol{\omega}) - \omega_1 \cos(\omega t - \frac{\omega}{c_s}\mathbf{x}\cdot\boldsymbol{\omega}) \end{pmatrix}. \tag{152}$$

In Section 9.3, elastic wave equation is solved with CD time discretization, and RK time discretization is applied to the elasticity solver in Section 9.4. In majority of the examples, we use 300 timesteps per one time period $[0, 2\pi/\omega]$, and the stopping criterion works with the relative norm $\sqrt{c/c_0}$ and $\varepsilon = 10^{-5}$. Unless other mentioned, we have set the propagation direction $\boldsymbol{\omega} = \omega(-\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2})$, densities $\rho_f(\mathbf{x}) = 1$ and $\rho_s(\mathbf{x}) = 1$, and speed of the sound $c(\mathbf{x}) = 1$. The numerical model of Algorithm 1 is implemented in Fortran 95. The tests presented in Section 9.1 have been performed on an HP 9000/785/J5600 workstation at 552MHz PA-RISC 8600 CPU. Computations reported in Sections 9.2 and



(a) Square obstacle.

(b) Two non-convex semi-open cavities.

(c) Non-convex semi-open cavity.

FIGURE 5 Geometrical shapes of the obstacles.

9.3 have been carried out on a PC with an AMD Athlon XP 2200+ CPU at 1.8 GHz, and an AMD Opteron 885 at 2.6 GHz have been used to get the results which are presented in Section 9.4.

In all the experiments, the infinite domain is truncated by the boundary Γ_{ext}. Mesh generator provided by Numerola Oy is used to divide the computational domain into square elements, each having a side length h. Since geometries with curved boundaries can not be represented exactly by a rectangular mesh, we use only polygonal boundaries (see Figures 5(a)-5(c)). The domain Ω with a square obstacle with side length 2 (see Figure 5(a)) is defined such that the surrounding boundary Γ_{ext} coincided with the border of the square $[0,0] \times [4,4]$. Scattering by two semi-open cavities (see Figure 5(b)) is solved in a domain with the artificial boundary Γ_{ext} coinciding with the perimeter of the rectangle $[0,5] \times [0,4]$. Internal width and height of each cavity is $3/4$ and $5/4$. Thickness of the wall is $1/4$, and distance between cavities is 1. In simulations with one non-convex semi-open cavity (see Figure 5(c)), the lower left corner of the rectangular computational domain surrounding the obstacle is at the point $(0,0)$ and the upper right corner is at the point $(7.25, 3.75)$. Internal width and height of the cavity are 5 and $5/4$, and thickness of the wall is $1/4$. All the obstacles are surrounded by the boundary Γ_0, centered in the computational domain Ω, and located at perpendicular distance of 1 from the boundary Γ_{ext}. Thus, the lower left corner of the cavity is at the point $(1,1)$ in each cases. With these geometries, we present also examples of varying speed of sound

$c(\mathbf{x})$. For this, we use coated obstacles, in which $c(\mathbf{x})$ is different than in the surrounding domain. Thickness of the coating material parallel to the surface of the obstacle is $1/4$.

In essence, outlines of Sections 9.1-9.4 are similar to each other. The first test problem in each section is chosen to test the accuracy of the approximation. For this purpose we construct an artificial problem, the solution of which is known to be the plane wave. This type of test problem could be constructed with any test geometry described above, but we choose to use the square geometry (see Figure 5(a)) with the associated outer boundary. In the acoustic case, we introduce function $\hat{y} \in H^1(\Omega)$ such that $p_{f_{\text{inc}}}(\mathbf{x},t) = \cos(\omega t - \boldsymbol{\omega}\cdot\mathbf{x})$, $\hat{y}|_{\Gamma_0} = p_{f_{\text{inc}}}$, $\hat{y}|_{\Gamma_{\text{ext}}} = \frac{\partial \hat{y}}{\partial \mathbf{n}}|_{\Gamma_{\text{ext}}} = 0$, and $y_{\text{ext}} = \frac{\partial p_{f_{\text{inc}}}}{\partial t} + \frac{\partial p_{f_{\text{inc}}}}{\partial \mathbf{n}}$. Then, the function $\hat{p}_f$ defined by $\hat{p}_f = p_{f_{\text{inc}}} - \hat{y}$ satisfies equation (45) with the nonzero right-hand side $f = -\frac{\partial^2 \hat{y}}{\partial t^2} + \nabla^2 \hat{y}$ as well as equations (46) and (47). After solving $\hat{p}_f$ we get the actual solution by $p_{f_{\text{inc}}} = \hat{p}_f + \hat{y}$, which approximates the plane wave. This modification eliminates the error caused by the absorbing boundary condition, and allows us to study the effect of the spatial discretization. For the elastic case, we can construct a corresponding test problem by introducing an auxiliary function $\hat{\mathbf{g}} \in H^1(\Omega)\times H^1(\Omega)$ such that

$$
\begin{aligned}
\rho\frac{\partial^2 \hat{\mathbf{u}}_s}{\partial t^2} - \nabla\cdot\sigma(\hat{\mathbf{u}}_s) &= -\rho\frac{\partial^2 \hat{\mathbf{g}}}{\partial t^2} + \nabla\cdot\sigma(\hat{\mathbf{G}}), && \text{in } Q = \Omega\times[0,T],\\
\hat{\mathbf{u}}_s &= 0, && \text{on } \gamma_0 = \Gamma_0\times[0,T],\\
\mathbf{B}\frac{\partial \hat{\mathbf{u}}_s}{\partial t} + \sigma(\hat{\mathbf{u}}_s)\mathbf{n} &= \mathbf{g}_{\text{ext}} && \text{on } \gamma_{\text{ext}} = \Gamma_{\text{ext}}\times[0,T],\\
\hat{\mathbf{u}}_s(\mathbf{x},0) &= \mathbf{u}_{\mathbf{s}\text{inc}}(\mathbf{x},0) - \hat{\mathbf{g}}(\mathbf{x},0), && \text{in } \Omega,\\
\frac{\partial \hat{\mathbf{u}}_s(\mathbf{x},0)}{\partial t} &= \frac{\partial \mathbf{u}_{\mathbf{s}\text{inc}}(\mathbf{x},0)}{\partial t} - \frac{\partial \hat{\mathbf{g}}(\mathbf{x},0)}{\partial t}, && \text{in } \Omega,
\end{aligned}
$$

$$
\hat{\mathbf{g}}|_{\Gamma_0} = \mathbf{u}_{\mathbf{s}\text{inc}}, \quad \hat{\mathbf{g}}|_{\Gamma_{\text{ext}}} = 0, \quad \frac{\partial \hat{\mathbf{g}}}{\partial \mathbf{n}}|_{\Gamma_{\text{ext}}} = 0,
$$

the solution of which is $\hat{\mathbf{u}}_s = \mathbf{u}_{\mathbf{s}\text{inc}} - \hat{\mathbf{g}}$. Accuracy tests (in Sections 9.1.1, 9.2.1, 9.3.1 and 9.4.1) consist mainly of results discussing approximation error of the spatial discretization and pollution effect arising with frequency. When RK time discretization is considered (in Sections 9.1.1 and 9.4.1), we also show how to define the size of time step which eliminates the temporal error. After accuracy tests we present wave propagation and scattering problems (in Sections 9.1.2, 9.2.2, 9.3.2, 9.4.2, and 9.4.3). In acoustic scattering experiments, we consider the ordinary scattering problem (45)-(48), where $f = 0$ and $y_{\text{ext}} = \frac{\partial p_{f_{\text{inc}}}}{\partial \mathbf{n}} + \frac{\partial p_{f_{\text{inc}}}}{\partial t}$. Respectively, for elastic scattering problems $\mathbf{f} = 0$ and $\mathbf{g}_{\text{ext}} = \mathbf{B}\frac{\partial \mathbf{u}_{\mathbf{s}\text{inc}}}{\partial t} + \sigma(\mathbf{u}_{\mathbf{s}\text{inc}})\mathbf{n}$. These problems are solved with three different scatterers (see Figures 5(a)-5(c)). Elastic wave propagation, discussed in Section 9.4.3, is simulated in the rectangle $[0,0]\times[4,4]$.

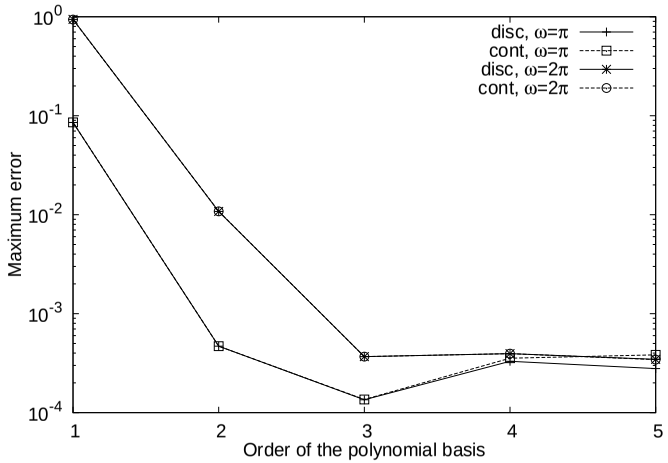


FIGURE 6 Maximum error with respect to the order of the polynomial basis for two different values of the mesh resolution such that $h = 1/4$.

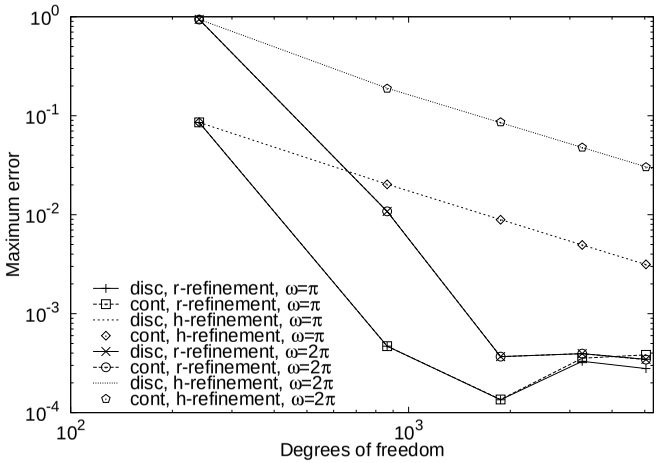


FIGURE 7 Maximum errors of h- and r-refinements with respect to degrees of freedom.

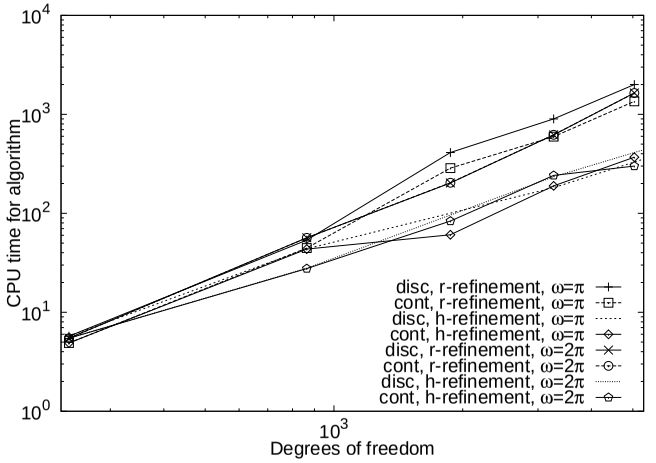


FIGURE 8 CPU time of h- and r-refinements in seconds.

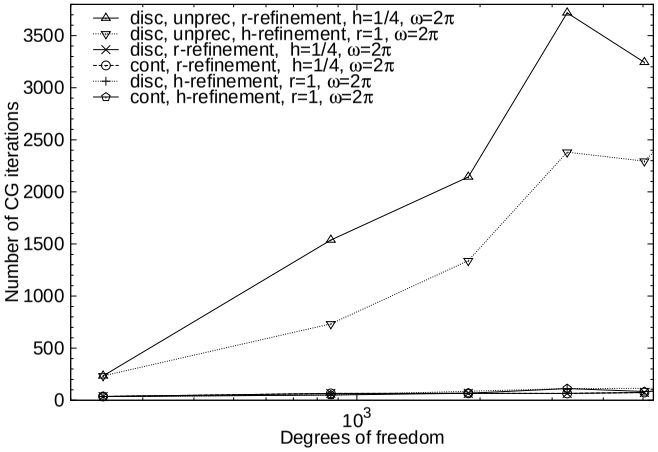


FIGURE 9 Number of CG iterations of h- and r-refinements.

9.1 Comparison between the two ways to compute the gradient

In this section, we present results of the acoustic scattering problem with central finite difference time discretization. To show that with stopping criterion $\varepsilon = 10^{-5}$ the two ways to compute the gradient, presented in Sections 7.1 and 7.2, lead essentially to same results, computations are carried out with gradient of the discretized cost function and with gradient of the continuous cost function. On the other hand, with smaller stopping criteria the discrete approach gives more precise minimization result than the continuous one. We also compare the conventional finite element method with higher order spectral element discretizations. In the end of the section, we show illustrations of acoustic scattering solutions.

9.1.1 Accuracy of space discretization

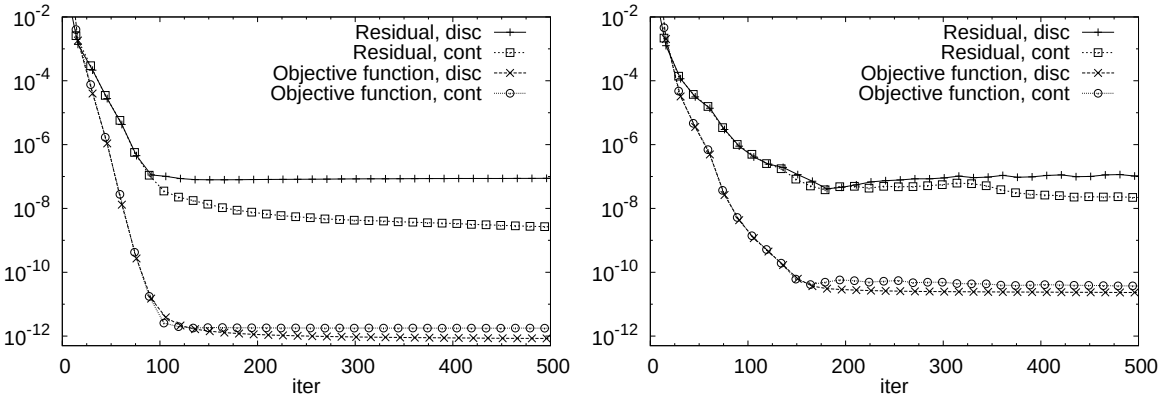
These tests concentrate mainly on the accuracy of the spatial discretization. First, we use a mesh with constant resolution ℓ/h, where $\ell = 2\pi c(\mathbf{x})/\omega$. Then, several mesh resolutions are tested to find a good combination of element order and mesh stepsize. In the tests considering the pollution effect, we keep ωh constant, run the tests with several wavenumbers and use lower mesh resolution with higher order elements.

Constant resolution

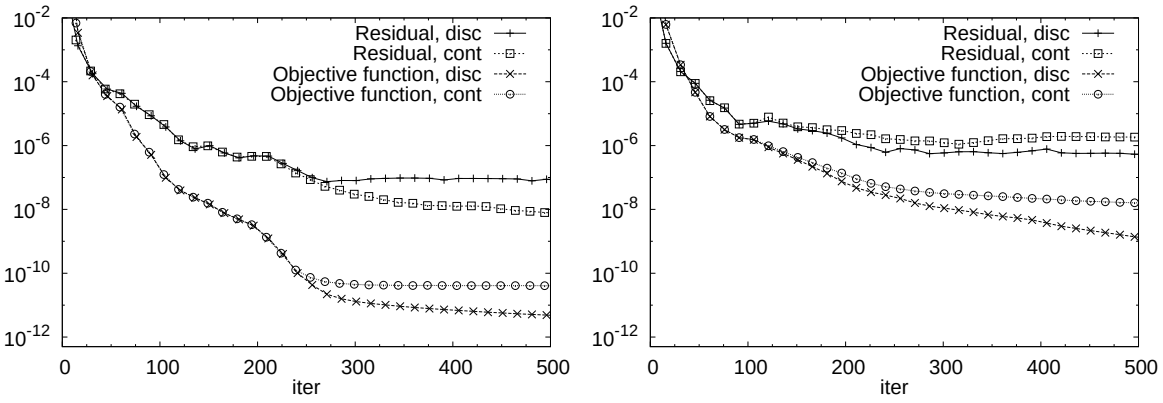
In the first experiment, we construct a mesh with $h = 1/4$ in the computational domain. This mesh is used to solve the test problem with wavenumbers $\omega = \pi$ and $\omega = 2\pi$, which implies that the mesh resolution has values 8 and 4. Figure 6 shows the error when the order of the spectral basis is increased. As the order increases, the error decreases until the error level of the time discretization or the stopping criterion is achieved. Results achieved by using the preconditioned CG algorithm with gradient of the discretized and continuous cost functions are marked in Figure 6 as disc and cont, respectively. The two ways to compute the gradient lead to practically the same errors.

Since the number of optimization variables is two times the number of degrees of freedom (DOF), we extend the first experiment by studying the error in terms of DOF. The results of Figure 6 are repeated in Figure 7 as spectral basis order refinement (r-refinement). The error curves of the mesh step refinement (h-refinement) corresponding to the classical FEM discretization are obtained by keeping the order fixed ($r = 1$) and using mesh stepsizes $h = 1/4i$, where $i = 1, \ldots, 5$. As the order of the polynomial basis increases, the maximum error between the numerical solution and the analytical solution decreases until the error of the time discretization or the stopping criterion is achieved. The error becomes smaller also with mesh step refinement, but the convergence rate is higher for r-refinement than for h-refinement. Based on the results, it seems clear that it is better to increase the order than the resolution to improve efficiency.

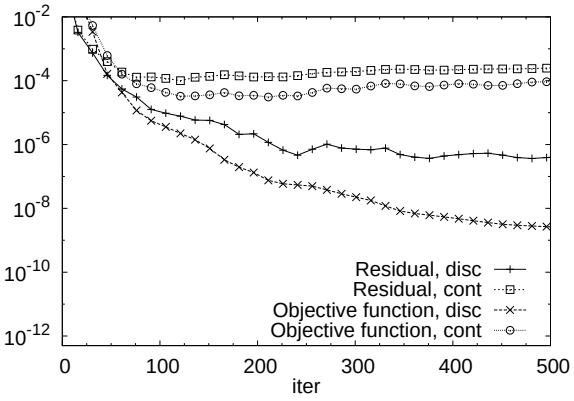
This conclusion is further supported by Figure 8, which shows the CPU times for these experiments. Naturally the CPU time increases as the resolution or the order is increased, but it also seems to depend linearly on the number of degrees of freedom. The conclusion, that total CPU time for the SEM is much less than the total CPU time for the



(a) r=1 (b) r=2



(c) r=3 (d) r=4



(e) r=5

FIGURE 10 The values of residual and objective function J as functions of iteration number with gradient of discrete and continuous cost function.

FEM for same accuracy, follows from these findings.

Preconditioners play an important role in accelerating the convergence rate of the CG method. To show the benefit of preconditioning, computations corresponding to h- and r-refinement with $\omega = 2\pi$ are repeated without preconditioning, i.e. setting $\mathcal{L} = \mathcal{I}$ in Algorithm 1. The number of CG iterations required grows with the number of degrees of freedom in the unpreconditioned case (see Figure 9) while in the preconditioned case, the number of iterations remains approximately constant. In these computations, CPU time required by the AMG preconditioner is only a few percents of the CPU time for the whole algorithm. Thus, significant savings result from the AMG preconditioner.

To show how r-refinement affects the convergence history, we have computed the values of residual and cost function by using the two ways to compute the gradient. In this computation, we have used $h = 1/10$, $\omega = 2\pi$ and $\Delta t = 1/320$ which implies 10 elements per wavelength and 320 timesteps per time period. The stopping criterion was $\varepsilon = 10^{-9}$, and the maximum number of iterations was limited to be 500. The results for spectral orders $r = 1,\ldots,5$, in cases of discrete and continuous level gradient, are presented with respect to iteration number in Figure 10. With small iteration numbers the graphs for a particular element order seems to be almost identical whether the discrete or continuous approach is used. When considering larger iteration numbers, we can observe that the discrete version converges to smaller value of cost function. This means also that with high number of iterations, the solution computed with the discrete approach is more accurate than the one with the continuous approach. This was not recognized during the earlier tests, since the stopping criterion $\varepsilon = 10^{-5}$ of the CG method set a lower bound for the error and stopped the simulations before smaller values of the residual were reached. Now, we can say that using the continuous version of the gradient leads to yet another error component setting the new lower bound for the error, which is noted only with strict stopping criteria. Thus, using the discrete approach gives exactly the gradient of the function to be minimized and leads to more precise minimization result.

Varying resolution

As we have seen in Figure 8, consumption of the CPU time increases linearly with respect to the number of degrees of freedom. Moreover, we have observed rapid decrease in error with respect to the order of the polynomial basis until the error level of time discretization becomes dominant (see Figures 6 and 7). This is why we have tried to find a good combination of order of the element and mesh stepsize by making computations with several resolutions of the mesh for each order of the element. This experiment has been carried out by keeping the wavenumber constant 4π. Various mesh stepsizes have been used such that ωh had values $\pi 2^{r-4}$, $\pi 2^{r-5}$, and $\pi 2^{r-6}$ for each order of the element r.

Figure 11 presents the maximum error in the solution with respect to the number of degrees of freedom. The purpose of this experiment is to show that certain error level can be reached by smaller number of degrees of freedom by increasing the order of the basis and adjusting the mesh stepsize accordingly. The difference is obvious when moving from $r = 1$ to higher orders. The reduction of the error is blocked by the fixed stopping criterion of the CG method and the fixed time discretization, which makes the comparison of higher orders difficult.

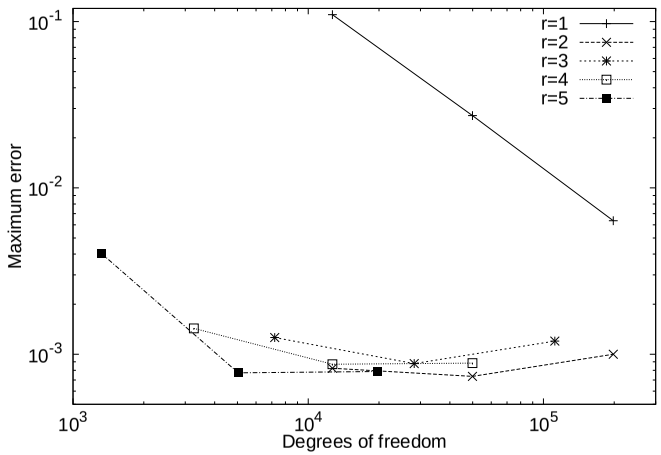


FIGURE 11 Maximum error with respect to the number of degrees of freedom in the case of varying resolution.

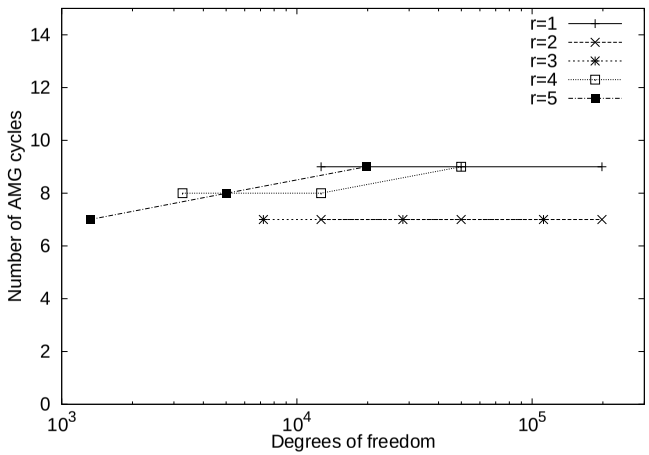


FIGURE 12 Number of AMG cycles with respect to the number of degrees of freedom in the case of varying resolution.

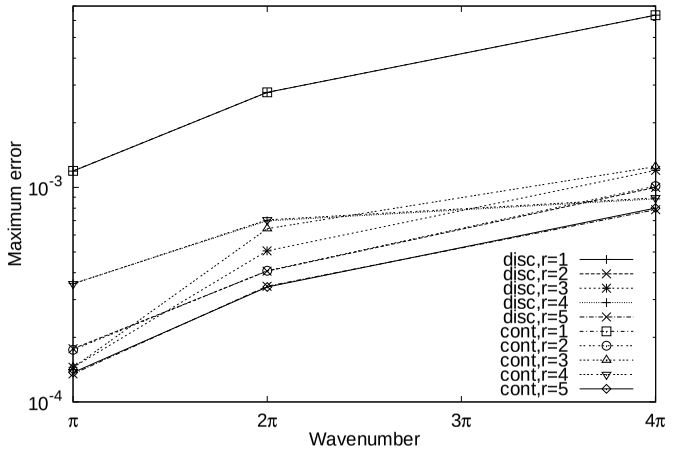


FIGURE 13 Behaviour of the error with respect to the wavenumber for different orders of the polynomial basis such that $\omega h = \pi 2^{r-6}$.

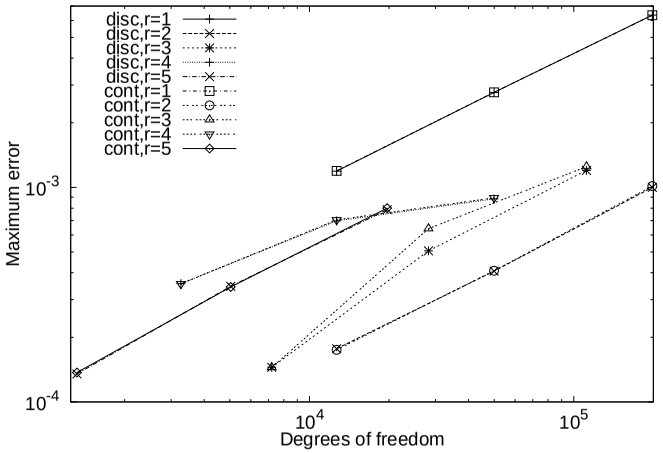


FIGURE 14 Behaviour of the error with respect to degrees of freedom such that $\omega h = \pi 2^{r-6}$.

The results of Figure 11 are repeated in Figure 12, but attention is focused on the number of AMG cycles. Since the number of AMG cycles stays approximately constant for different mesh resolutions, the preconditioner is not restricting the mesh stepsize. Because the resolution with $\omega h = \pi 2^{r-6}$ gives good results for all orders of the basis, we use it in next accuracy tests, when considering the situation with constant resolution and changing wavenumbers.

Pollution effect

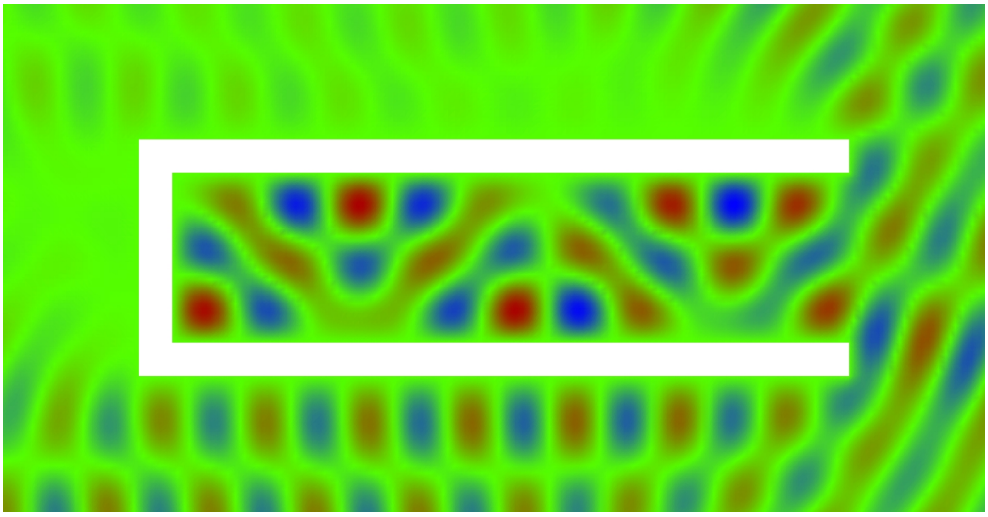
We performed another set of experiments by varying the resolution of the mesh with the order of the basis. More specifically we used lower resolution with higher orders according to the equation $\omega h = \pi 2^{r-6}$. Figure 13 shows the error with respect to increasing wavenumber for orders 1-5. Once again, we can notice that the difference between the two ways to compute the gradient does not have remarkable effect on the results. The effect of the pollution term is clearly visible in the error curves. We expected to see a more pronounced reduction in the pollution effect with higher orders. Now it is almost similar with all orders. Perhaps the difference could be observed by extending the test to higher wavenumbers. Figure 14 shows the same errors in terms of the number of degrees of freedom. These results support the conclusion that certain error level is reached more efficiently by applying higher order spatial discretization.

9.1.2 Acoustic scattering

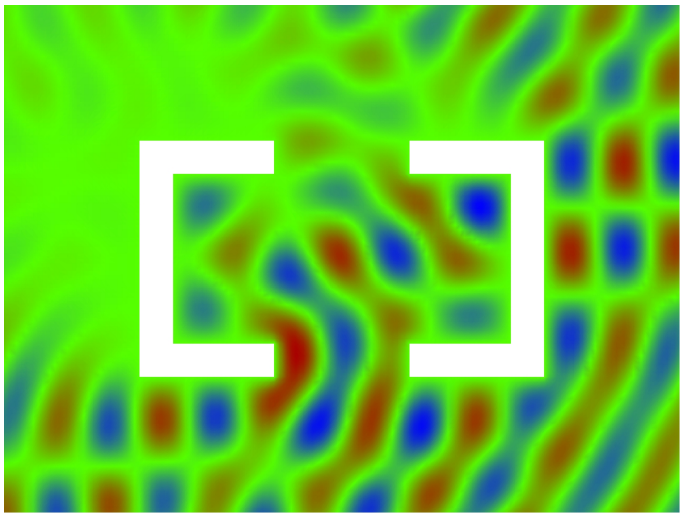
We use the propagation direction $\boldsymbol{\omega} = \omega\big(-\frac{\sqrt{3}}{2}, \frac{1}{2}\big)$, angular frequency $\omega = 3\pi$, and mesh stepsize $h = 1/8$. There are slightly over five elements per wavelength. To guarantee demands for accuracy also for higher orders, we have chosen to use 600 timesteps per one time period $[0, T]$, and the stopping criterion works instead of the relative norm $\sqrt{c/c_0}$ with the absolute norm $\sqrt{c}$ and $\varepsilon = 10^{-3}$. Number of iterations with different scatterers in both discrete and continuous cases of the gradient are shown in Table 1.

TABLE 1 The number of degrees of freedom (DOF) and the number of iterations (iter) of the preconditioned CG algorithm in the case of discrete (disc) and continuous (cont) level gradient with different scatterers.

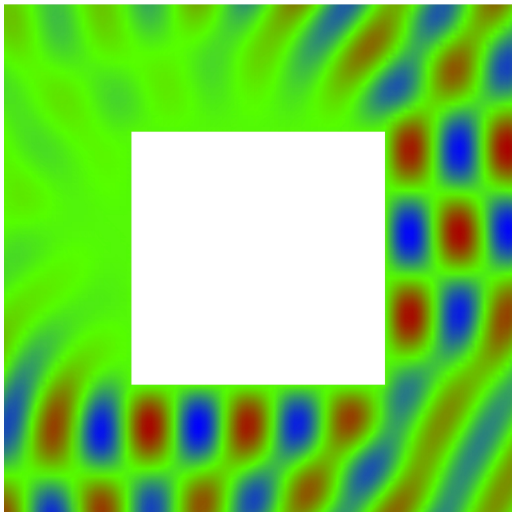
		r				
Type of the obstacle		1	2	3	4	5
non-convex semi-open cavity	DOF	2112	8064	17856	31488	48960
	iter disc	152	123	122	122	122
	cont	163	116	116	115	115
two non-convex semi-open cavities	DOF	1221	4635	10241	18039	28029
	iter disc	90	95	95	95	95
	cont	90	96	95	95	96
convex obstacle (square)	DOF	864	3264	7200	12672	19680
	iter disc	45	65	45	45	41
	cont	45	65	45	45	40



(a) Non-convex semi-open cavity.



(b) Two non-convex semi-open cavities.



(c) Square obstacle.

FIGURE 15 Scattering solutions with discrete version of gradient.

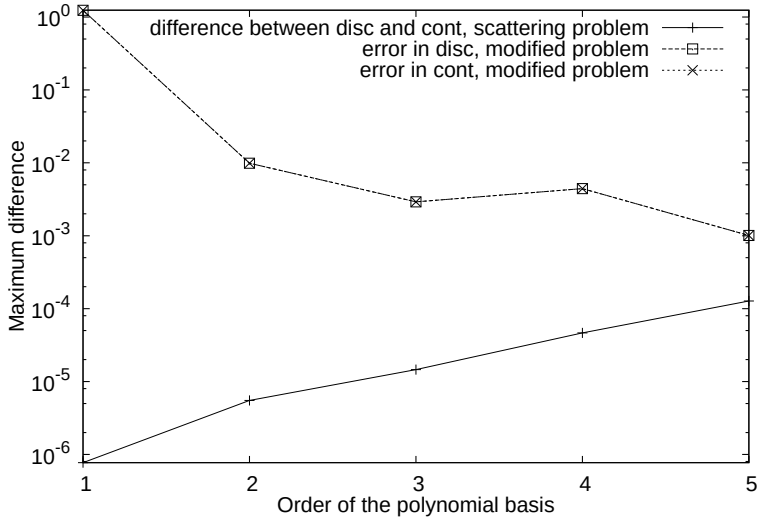


FIGURE 16 Maximum difference between scattering solutions corresponding to continuous and discrete versions of gradient and maximum errors of the solutions corresponding to continuous and discrete versions of gradient in the case of square scatterer.

As we can see, the number of iterations is substantially less in the case of convex square scatterer than in the cases of non-convex scatterers. In all the experiments it appears that preconditioning keeps the number of CG iterations bounded with respect to r. Contour plots of the numerical solutions with discrete version of gradient and $r = 3$ are in Figures 15(a)-15(c).

From Table 1 we find also that the number of iterations fluctuates between discrete and continuous cases of the gradient only in the experiment with a non-convex semi-open cavity. In the other experiments, the two ways to compute the gradient lead to practically the same number of iterations. Because of that, also the computational effort is of the same order of magnitude in both two cases of computing the gradient.

In Figure 16, the difference is depicted with respect to the order of element in the case of square scatterer. We have also compared the magnitude of this difference with the error between exact solution and the computed one. This is why we have plotted in Figure 16 also maximum errors with respect to the order of the polynomial basis for two cases of computing the gradient. Although maximum difference seems to grow with the order of polynomial basis, difference between the two solutions is substantially less than the maximum error with the exact solution.

We illustrate the performance of the controllability method with varying speed of sound by using two coated non-convex semi-open cavities as reflectors. Now, the time interval $[0, 1]$ is divided into 300 timesteps of equal size and angular frequency is $\omega = 4\pi$. In the first test, the speed of sound $c(\mathbf{x})$ is varying such that it is equal to one outside the obstacle and 0.5 in the coating, implying that outside the obstacle wavelength $\ell(\mathbf{x}) = 0.5$ and in the coating $\ell(\mathbf{x}) = 0.25$. Absorbing boundary is located at a distance of 1.5ℓ from the obstacle. Since rectangular mesh with element width $h = 1/16$ is used, there are 8 elements per wavelength outside the obstacle and 4 in the coating. In the second test case, parameters are the same, expect $c(\mathbf{x}) = 0.25$ in the coating of the right hand obstacle.

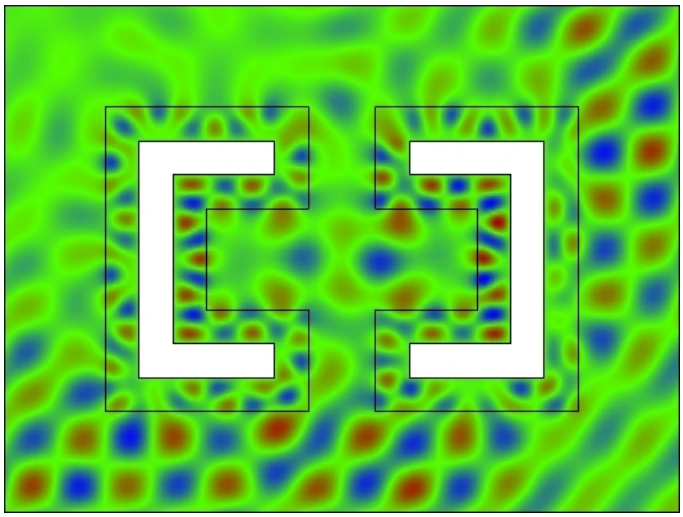


FIGURE 17 Solution of the first test problem with varying speed of sound.

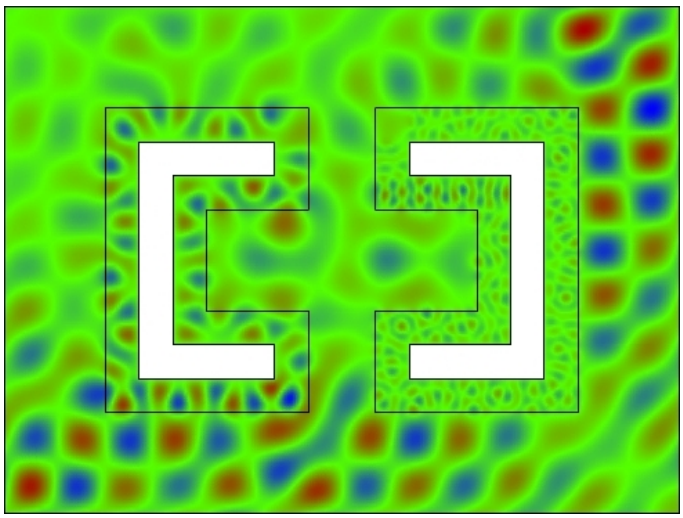


FIGURE 18 Solution of the second test problem with varying speed of sound.

TABLE 2 Number of iterations which is needed to reduce the relative euclidean norm of the gradient of the functional J below 10^{-4}.

order	DOF	number of iterations test 1	test 2
1	4635	372	579
2	18039	289	695
3	40211	292	594
4	71151	292	596

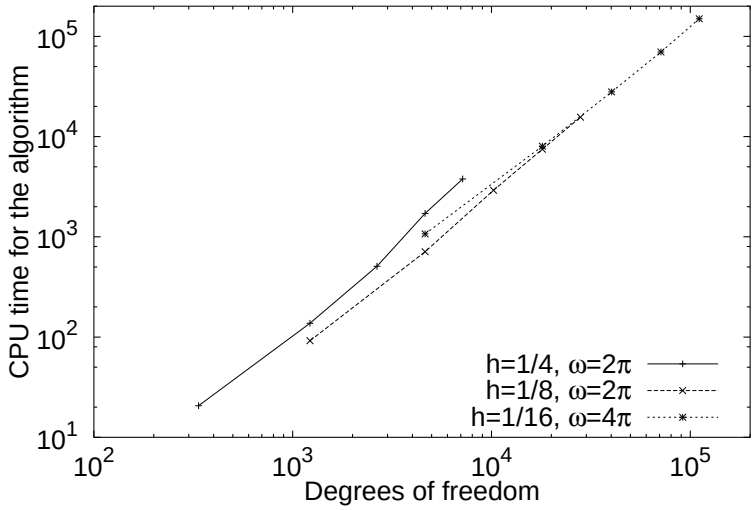


FIGURE 19 CPU time in seconds with respect to degrees of freedom, when $c(\mathbf{x})$ is 0.5 in both coatings and 1 outside the obstacles.

Both test examples are solved by increasing the order of the spectral element basis. Contours representing the total field p_f are plotted in Figure 17 and Figure 18 with order of the spectral element basis equal to 4. The number of CG iterations needed to solve the control problem in the case of discrete level gradient are given in Table 2, which also shows the number of degrees of freedom (DOF) in the spectral element mesh. These results show that, if the spatial discretization is accurate enough, the number of iterations remains nearly constant while the number of optimization variables (i.e. two times DOF) increases. CPU time (in seconds) used for solving the first test problem is depicted in Figure 19, where DOF increases as the order of the spectral element basis increases from 1 to 5. The computational effort of the method seems to depend linearly on DOF.

9.2 Comparison between central finite difference and Runge-Kutta time discretizations for the acousticity equation

We study the accuracy of the temporal discretization by comparing the method with Runge-Kutta time discretization with the one with central finite difference time discretization. In this context, we use abbreviation RK meaning the controllability method with Runge-Kutta time discretization and CD standing for the method with central finite difference time discretization.

9.2.1 Accuracy of approximation

The test problem is solved with mesh stepsize $h = 1/4$ and angular frequencies $\omega = \pi$ and $\omega = 2\pi$ with both Runge-Kutta (RK) and central finite difference (CD) time discretization. The relaxation parameter of SOR is 1.4 in preconditioning. The time interval $[0, T]$ is divided into 300 timesteps in the case of CD time discretization and into 150 timesteps in the case of RK time discretization. Figure 20 shows the behaviour of

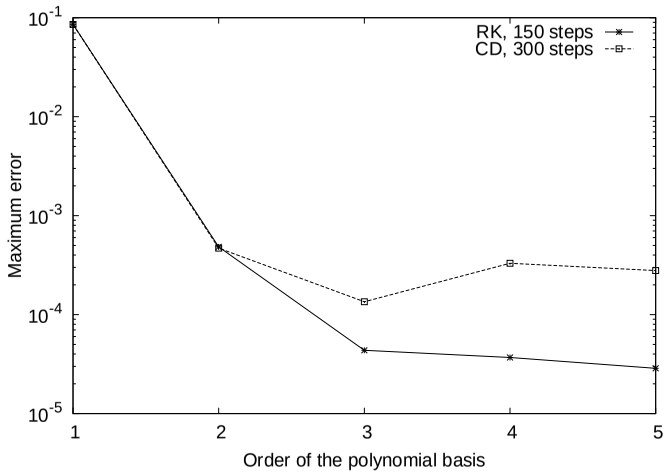


FIGURE 20 L_∞-norm of the difference between analytic solution and numerical solution with respect to the spectral order.

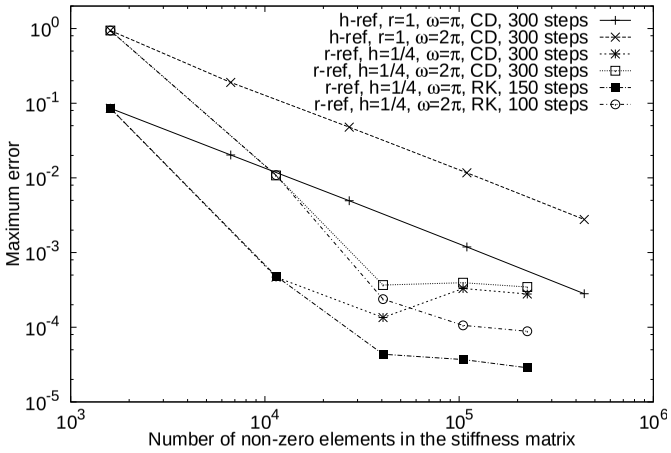


FIGURE 21 Maximum errors of h- and r-refinements with respect to the number of non-zero elements in the stiffness matrix.

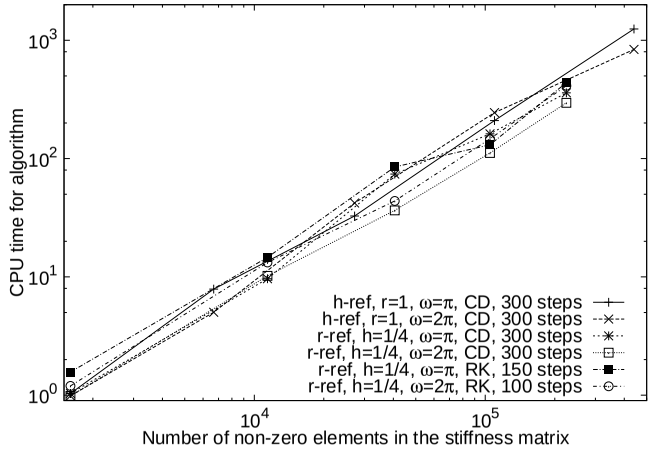


FIGURE 22 CPU time of h- and r-refinements in seconds.

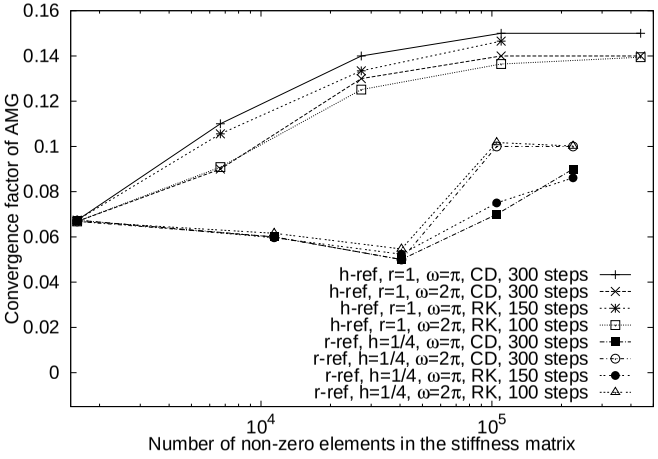


FIGURE 23 Convergence factor of AMG, with respect to the number of non-zero elements in the stiffness matrix.

error when the order of spectral basis increases. With both time discretization schemes, error of spatial discretization dominates with low order elements. Thus, the difference between maximum errors is negligible for spectral orders $r = 1$ and $r = 2$. When higher order elements are used, results computed with RK version of the algorithm are more accurate than the ones computed with the CD version.

The number of non-zero entries in the stiffness matrix is essential for computational efficiency, since the time stepping scheme involves mainly matrix-vector multiplications. This is why the comparison between h- and r-refinements is presented in terms of the number of nonzero matrix entries in Figure 21. The error curves of the r-refinement are achieved when the order of the spectral basis r is increased from 1 to 5 with mesh stepsize $h = 1/4$. The h-refinement is obtained by keeping the basis order fixed ($r = 1$) and doubling the resolution of the mesh, given by ℓ/h, consecutively.

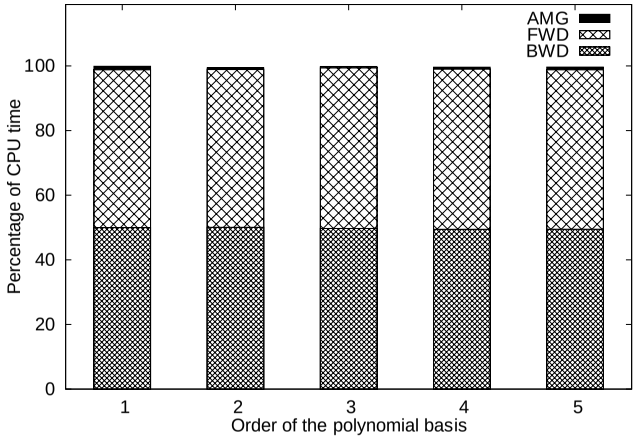
In conjunction with higher order elements, results computed with RK version of the algorithm are more accurate than the ones computed with the CD version (see Figure 21). This happens, because RK is higher order time scheme than CD. Apparently, the error of time discretization limits the accuracy with basis orders $r \geq 3$ in the case of CD time discretization, whereas the stopping criterion causes the limiting error in the RK case. Since the error of spatial discretization dominates with low order elements, the difference between errors is insignificant for spectral orders $r = 1$ and $r = 2$.

When the order of the polynomial basis increases or the mesh stepsize becomes smaller, systems to be solved become larger, which causes the increase in CPU time. When Δt is constant, the computational cost needed for one iteration is proportional to the number of non-zero elements in the stiffness matrix. To be more precise, the computational effort of the method seems to depend linearly on number of non-zero elements in the stiffness matrix (see Figure 22). According to Figure 22, the number of iterations varies such that the CPU time required for the two refinements corresponding to SEM and FEM are of the same order of magnitude. Convergence factor of AMG within these computations is shown in Figure 23.

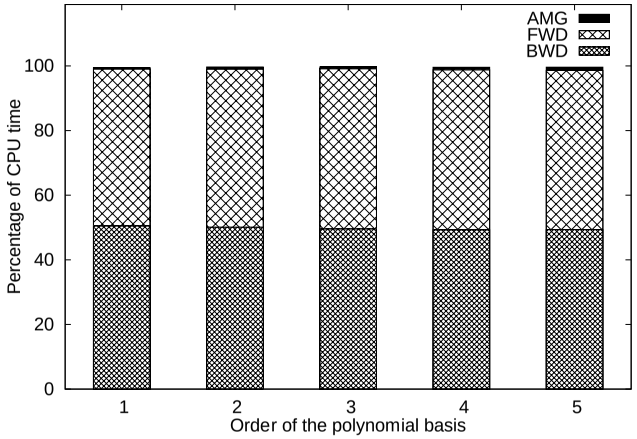
Most of the CPU time is used for solving state and adjoint state equations, and only small amount of time is consumed for preconditioning. The proportion of CPU time required by AMG per one iteration is shown in Figure 24. Figure 24 shows also the computational efforts of state (i.e. forward, FWD) and adjoint state (i.e. backward, BWD) equations in one CG iteration. Besides these computations, almost negligible amount of CPU time is used for matrix-vector multiplications at each iteration.

Accuracy of time discretization

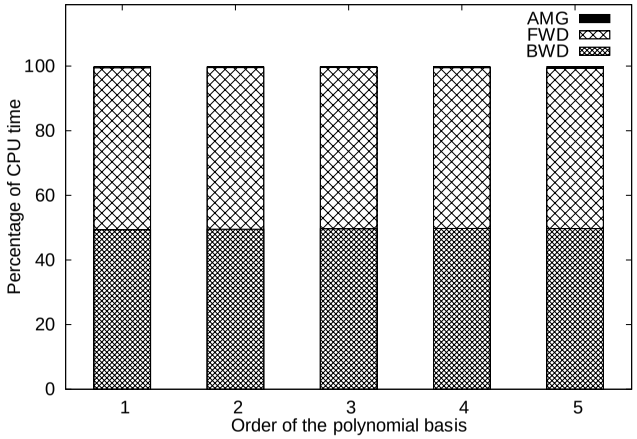
When higher order elements are used, good efficiency with high accuracy can be achieved by using sufficiently large mesh stepsize [1]. This is why we have performed another set of experiments by using coarser mesh with higher element order. In these computations, the resolution of the spatial discretization, i.e. degrees of freedom per wavelength, is approximately constant ($r\ell/h \approx 40$). Since the accuracy of the state equation depends only on spatial and temporal discretizations, we have first computed the accuracy of the solution of the state equation with various lengths of timestep. Computations have been carried out with the state equations (97) and (115) corresponding to the wave problem



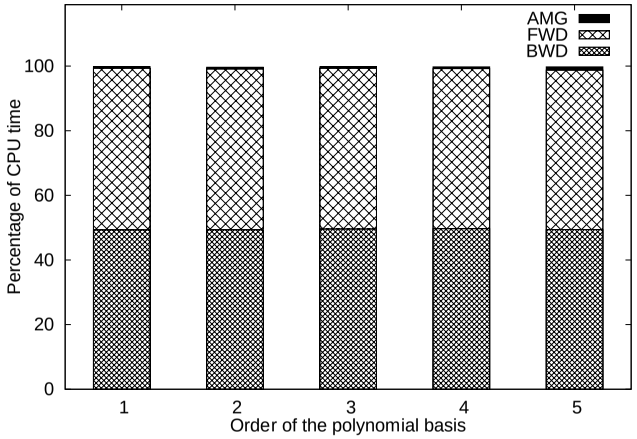
(a) CD version of the algorithm with $\omega = \pi$ and 300 timesteps.



(b) CD version of the algorithm with $\omega = 2\pi$ and 300 timesteps.



(c) RK version of the algorithm with $\omega = \pi$ and 150 timesteps.



(d) RK version of the algorithm with $\omega = 2\pi$ and 100 timesteps.

FIGURE 24 Proportion of CPU time (in percent) required by the AMG cycles and computing state (FWD) and adjoint state (BWD) in one CG iteration.

(45)-(49). The right hand side functions are

$$f = 0 \tag{153}$$

$$y_{\text{ext}} = \omega \sin(\boldsymbol{\omega} \cdot \mathbf{x}) + \frac{\partial \cos(\omega t - \boldsymbol{\omega} \cdot \mathbf{x})}{\partial \mathbf{n}}, \tag{154}$$

and the initial conditions are

$$\mathbf{e}_0 = \cos(\boldsymbol{\omega} \cdot \mathbf{x}) \tag{155}$$

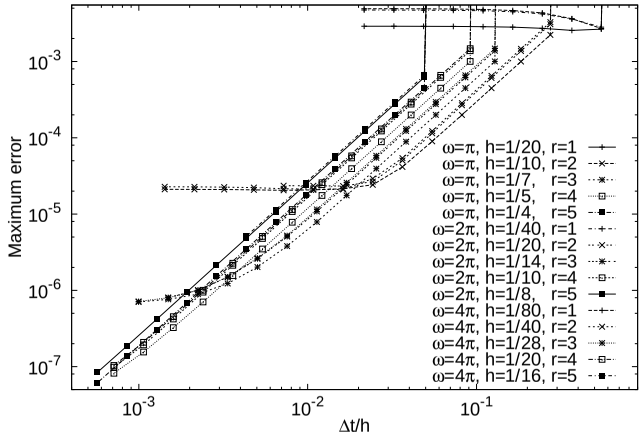
$$\mathbf{e}_1 = \omega \sin(\boldsymbol{\omega} \cdot \mathbf{x}). \tag{156}$$

We have started computations with a length of the timestep corresponding to 50 timesteps per one time period ($T = 2\pi/\omega$). In the case of CD time discretization the number of timesteps is multiplied by 1.5, whereas in the case of RK time discretization it is multiplied by 1.25. Proceeding this way, we have achieved a series of numerical results with various lengths of the timestep. Numerical solution is compared with the analytical one, which is $p_f = \cos(\omega t - \boldsymbol{\omega} \cdot \mathbf{x})$. Maximum errors between numerical and analytical solutions with $\omega = \{\pi, 2\pi, 4\pi\}$ in the case of CD time discretization are seen in Figure 25(a). Since we notice from Figure 25(a) that the curve with a particular spectral order has its characteristic shape, in the RK case only results with $\omega = \pi$ are shown in Figure 25(b). Results with $\omega = 2\pi$ and $\omega = 4\pi$ have the similar behaviour with respect to the length of timestep.

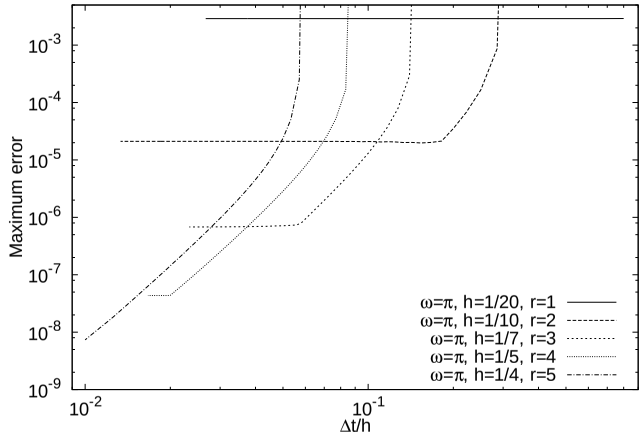
From results shown in Figures 25(a) and 25(b), we can determine, for instance, the stability conditions and orders of temporal and spatial accuracy. Vertical lines of the curves represent the unstable situation. To examine the stability condition, we have made also some computations with small numbers of timesteps. These computations are carried out with $10i$, for $i = 1, 2, 3, \ldots$, timesteps per time period until a stable solution is achieved. The first number of timesteps which leads to the stable solution, and the corresponding value of $\Delta t/h$, are shown in Table 3.

For large timesteps the error of temporal discretization dominates. Then, RK gives more accurate results with larger timestep than CD (see Figures 25(a) and 25(b)). On the left of Figures 25(a) and 25(b), curves turn to horizontal lines. The maximum error with respect to the length of timestep is not decreasing even if smaller timesteps are used. This means that the temporal error is eliminated. In the both Figures, the same error level is achieved for a particular order of the element r (with constant h) when the temporal error is eliminated. This is something that we expected, since the error remaining, after the temporal error is eliminated, is the error of spatial discretization. Every curve represents computations with a particular spectral order which has a characteristic discretization error. Naturally, the order of the space discretization error decreases when higher order elements are used.

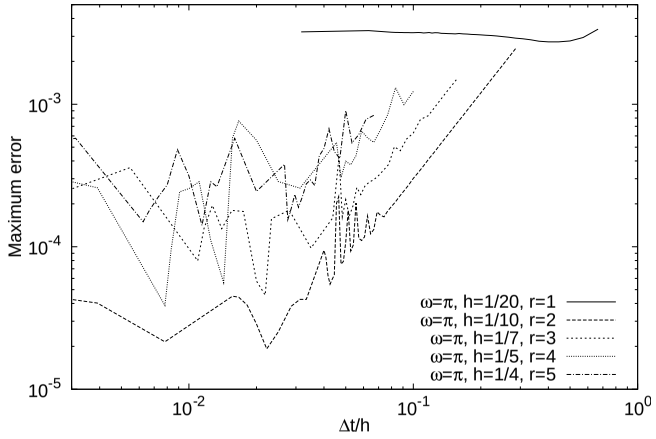
According to the results shown in Figures 25(a) and 25(b), we can choose the timestep to examine the spatial discretization such that the error of time discretization is negligible. When using the control algorithm that small timesteps are not reasonable, especially when concerning higher order elements. The reason for this is that we are using



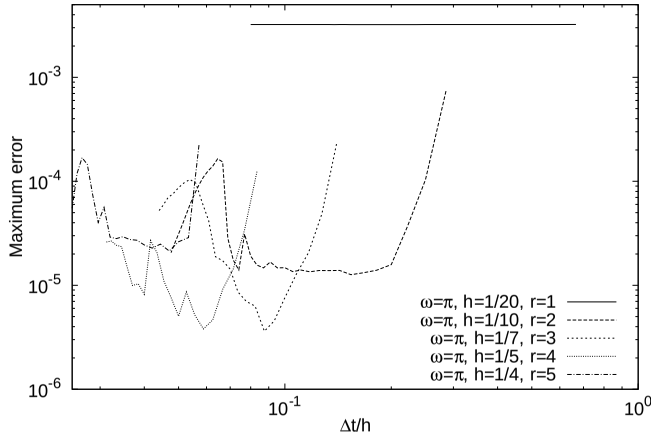
(a) Error in the case of state equation and CD time discretization.



(b) Error in the case of state equation and RK time discretization.



(c) Error in the case of control algorithm and CD time discretization.



(d) Error in the case of control algorithm and RK time discretization.

FIGURE 25 Maximum error with various lengths of timestep for each spectral order.

TABLE 3 Mesh stepsizes for different angular frequencies, stability conditions, and number of timesteps for different spectral orders.

	r	1	2	3	4	5
Mesh stepsizes for different angular frequencies	$\omega = \pi$	1/20	1/10	1/7	1/5	1/4
	2π	1/40	1/20	1/14	1/10	1/8
	4π	1/80	1/40	1/28	1/20	1/16
Number of timesteps to satisfy the stability condition	CD	60	70	90	100	120
	RK	60	70	90	120	140
Number of timesteps used in computations	CD	90	270	300	320	320
	RK	60	100	140	150	150

a constant stopping criterion ε, which limits the accuracy of the solution computed with Algorithm 1. So, our aim is to choose the length of timestep for each spectral order so that the temporal error is smaller than the error due to the stopping criterion of the algorithm.

To show how the stopping criterion of the algorithm influences accuracy, we have solved the artificial problem presented in Section 9.2.1 for each spectral order with various lengths of the timestep. These results are shown in Figures 25(c) and 25(d). Oscillations in maximum error are noticed, when the error of stopping criterion becomes dominant. Number of timesteps, chosen by a procedure discussed above, are shown in Table 3.

Pollution effect

The behaviour of the error with respect to the wavenumber, when time steps shown in Table 3 are used, can be seen in Figure 26. In the case of classical finite element discretization, i.e. $r = 1$, the error increases to considerably large level as the wavenumber increases. Error increases with wavenumber also for higher spectral orders. Thus, the pollution effect is not eliminated with higher orders, but results are more accurate than with $r = 1$.

Maximum error is plotted with respect to CPU time in Figure 27 with both time discretizations. Better accuracy is achieved with less work, when higher order elements are used. In the case of finite element discretization, the error is a little bit larger with RK than with CD time discretization. With higher order elements the error in the CD case seems to be an order of magnitude larger than the error in the RK case with the same CPU time consumption.

From Figures 26 and 27 we can notice that CPU time for algorithm grows with wavenumber. The reason for this is the increase in number of CG iterations (see Figure 28). Since amounts of CPU time used for AMG and for the whole algorithm grow nearly at the same rate, the proportion of CPU time used for AMG at each iteration is almost constant (see Figure 29). The percentage of CPU time required by the AMG cycles in one CG iteration and number of CG iterations are presented in Figures 30 and 31 with respect to the number of non-zero elements in the stiffness matrix.

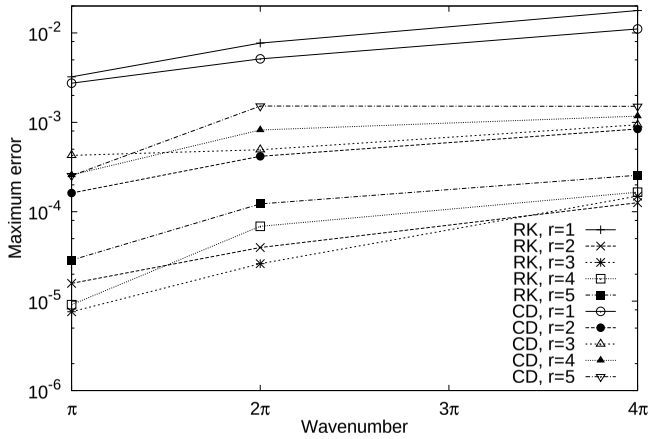


FIGURE 26 Behaviour of the error with respect to the wavenumber for different orders of the polynomial basis such that $r\ell/h \approx 40$.

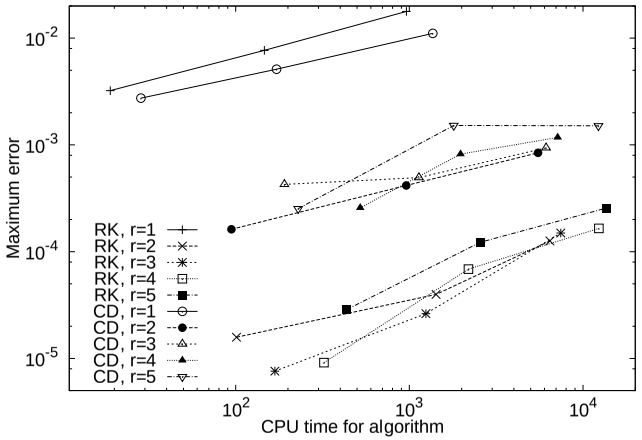


FIGURE 27 Behaviour of the error with respect to CPU time such that $r\ell/h \approx 40$.

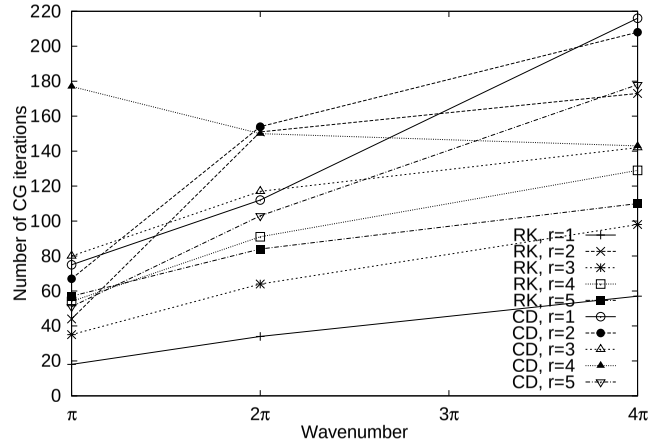


FIGURE 28 Number of CG iterations with respect to the wavenumber such that $r\ell/h \approx 40$.

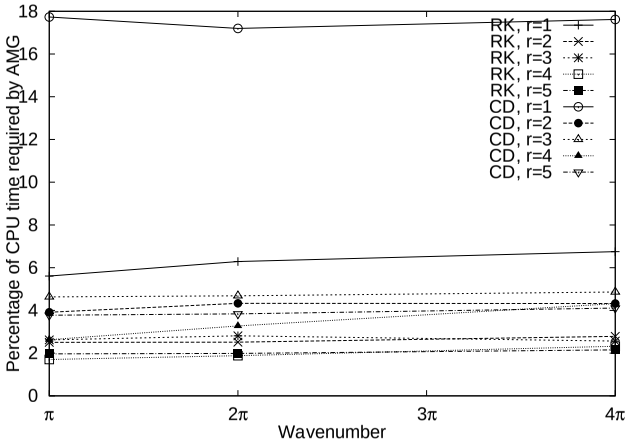


FIGURE 29 Proportion of CPU time (in percent) required by the AMG cycles in one CG iteration with respect to the wavenumber such that $r\ell/h \approx 40$.

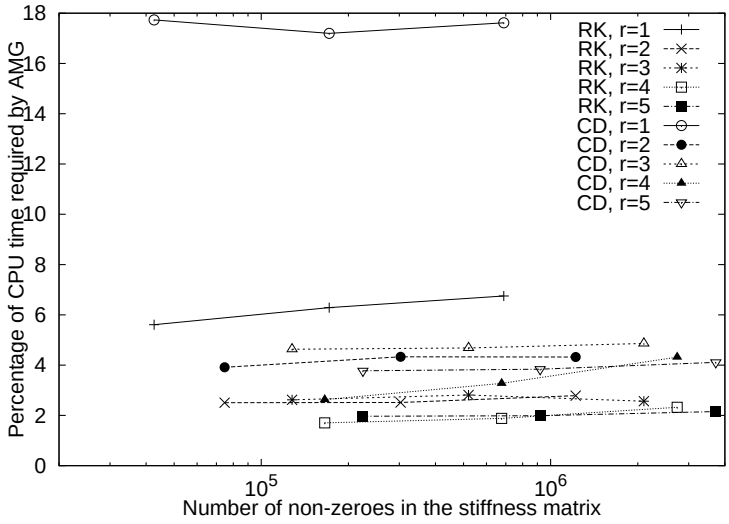


FIGURE 30 Proportion of CPU time (in percent) required by the AMG cycles in one CG iteration with $r\ell/h \approx 40$.

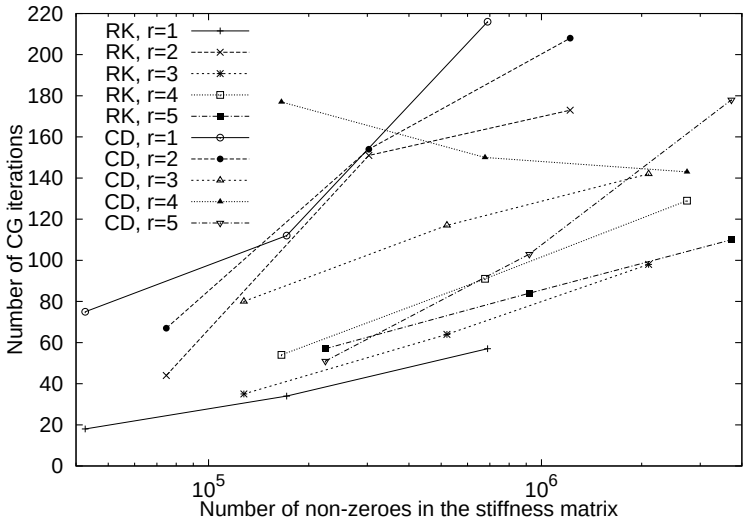


FIGURE 31 Number of CG iterations with respect to the number of non-zero elements in the stiffness matrix such that $r\ell/h \approx 40$.

9.2.2 Acoustic scattering

In these experiments, we have used the angular frequency $\omega = 4\pi$, which implies that the artificial boundary is located at distance 2ℓ from the scatterer. Mesh stepsizes and number of timesteps are chosen as in the previous example, except for coated RK tests. Because of stability conditions for RK, we need to use more timesteps when RK is used with varying material parameters (see Table 4).

TABLE 4 Mesh stepsizes and number of timesteps for different spectral orders with $\omega = 4\pi$.

r			1	2	3	4	5
	Mesh stepsize		$1/80$	$1/40$	$1/28$	$1/20$	$1/16$
Number of timesteps	CD		90	270	300	320	320
	RK	non-coated	60	100	140	150	150
	RK	coated	120	200	280	300	350

TABLE 5 Number of elements per wavelength for different spectral orders.

	r	1	2	3	4	5
Number of elements per wavelength	$c = 1$	40	20	14	10	8
	$\frac{3}{4}$	30	15	10.5	7.5	6
	$\frac{1}{2}$	20	10	7	5	4

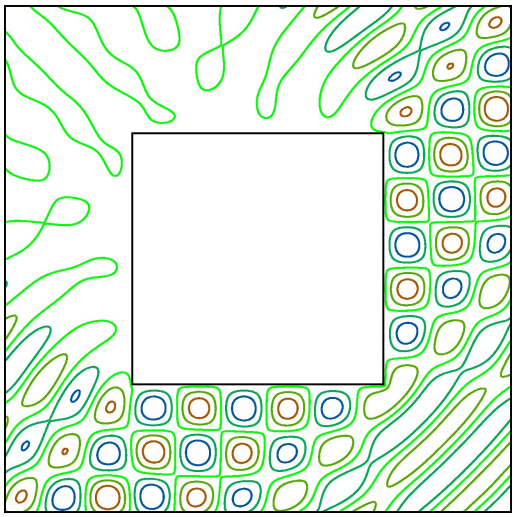


FIGURE 32 Contourplot of scattering by a convex obstacle with $r = 3$ and $h = 1/28$.

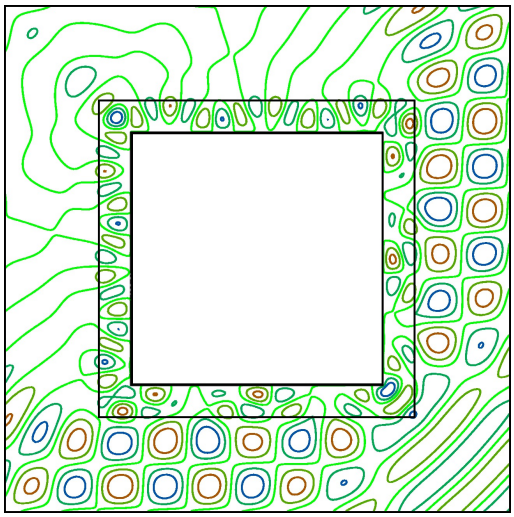


FIGURE 33 Contourplot of scattering by a coated convex obstacle with $r = 3$ and $h = 1/28$.

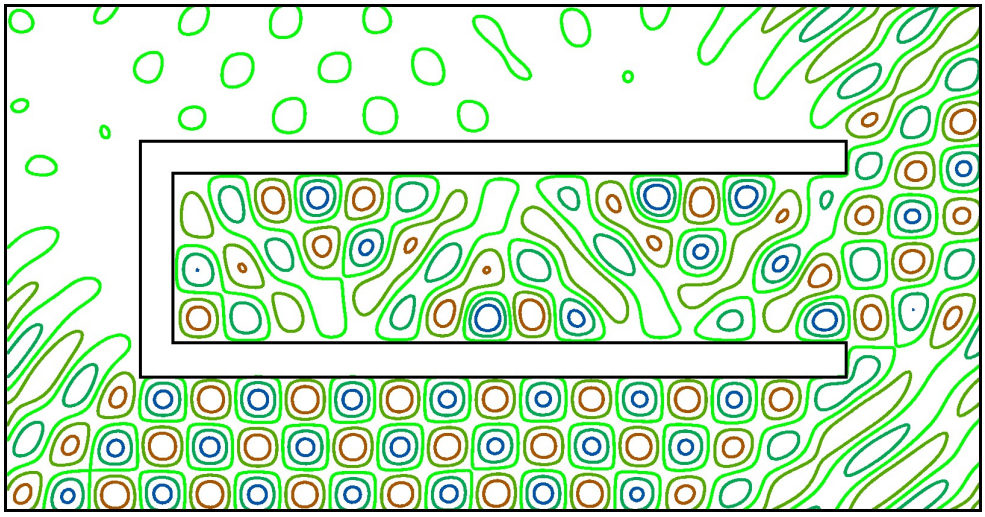


FIGURE 34 Contourplot of scattering by a non-convex semi-open cavity with $r = 3$ and $h = 1/28$.

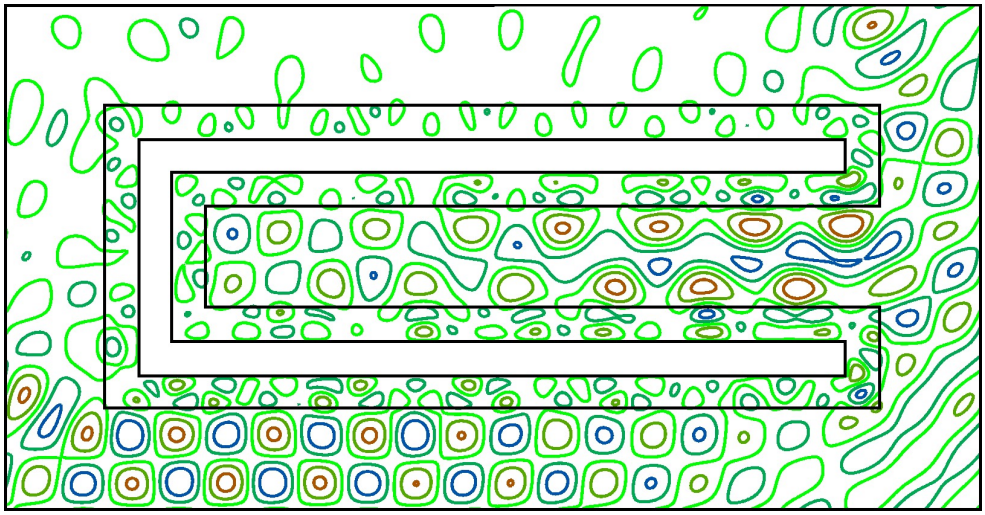


FIGURE 35 Contourplot of scattering by a coated non-convex semi-open cavity with $r = 3$ and $h = 1/28$.

The speed of sound $c(\mathbf{x})$ is equal to one outside the obstacle and $\frac{1}{2}$ in the coatings of square scatterer and non-convex semi-open cavity, implying that outside the obstacle wavelength $\ell(\mathbf{x}) = \frac{1}{2}$ and in the coating $\ell(\mathbf{x}) = \frac{1}{4}$. Since rectangular mesh with element width h is used, there are $1/2h$ elements per wavelength outside the obstacle and $1/4h$ in the coating (see Table 5). In the case of scattering by two non-convex semi-open cavities, $c(\mathbf{x}) = \frac{1}{4}$ in the coating of the left hand obstacle, and $c(\mathbf{x}) = \frac{3}{4}$ in the coating of the right hand obstacle. This implies that in the coating of left hand obstacle $\ell(\mathbf{x}) = \frac{1}{4}$, and there are $1/4h$ elements per wavelength, whereas in the coating of right hand obstacle $\ell(\mathbf{x}) = \frac{3}{8}$, which means $3/8h$ elements per wavelength (see Table 5). Numerical solutions of these scattering problems with $r = 3$ and $h = 1/28$ are shown in Figures 32-37. The Helmholtz equation is solved with non-convex semi-open cavities of similar shape e.g. in [13, 16, 43].

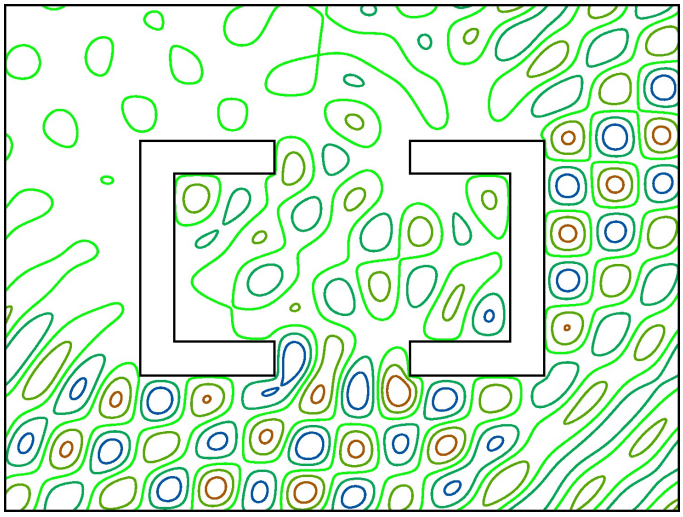


FIGURE 36 Contourplot of scattering by two non-convex semi-open cavities with $r = 3$ and $h = 1/28$.

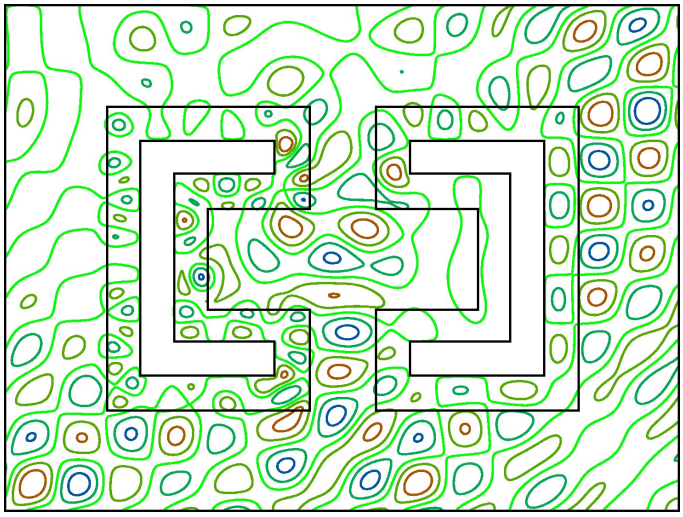


FIGURE 37 Contourplot of scattering by coated non-convex semi-open cavities with $r = 3$ and $h = 1/28$.

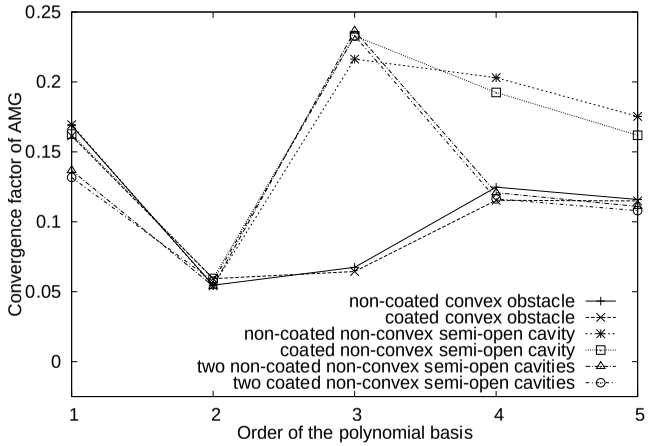
Number of iterations with coated and non-coated scatterers is compared in Table 6. When considering the algorithm with RK time discretization, we notice that computations with coated scatterers need two and a half times the number of iterations needed with non-coated scatterers. Convex obstacle is the simplest scatterer, and with it the smallest number of iterations is needed. For solving the scattering problem with two non-convex cavities, the number of iterations is twice as large as in the the case of convex scatterer. More reflections are produced inside the obstacle with one non-convex cavity than with the system of two non-convex cavities. That is why twice the number of iterations is needed to solve the problem with one non-convex cavity than with two non-convex cavities. Hence, the number of iterations depends strongly on the geometry of the scatterer.

TABLE 6 The number of iterations of the preconditioned CG algorithm in the case of CD and RK time discretization with different scatterers.

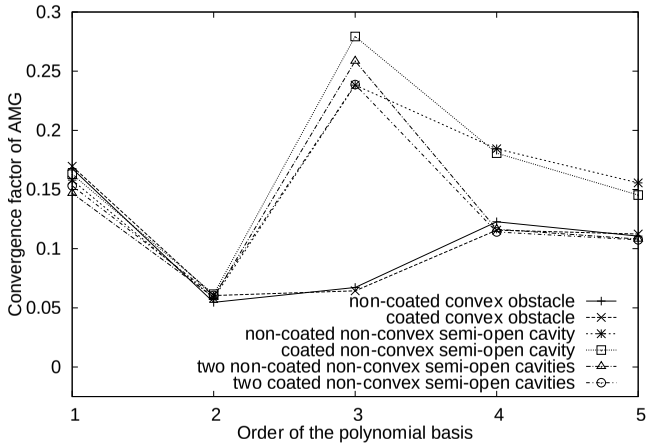
Type of the obstacle			1	2	3	4	5
convex obstacle (square)	CD	non-coated	216	208	142	143	178
	CD	coated	174	169	172	167	168
	RK	non-coated	59	75	74	76	75
	RK	coated	172	178	178	177	177
non-convex semi-open cavity	CD	non-coated	217	208	188	229	332
	CD	coated	589	732	732	727	727
	RK	non-coated	211	300	301	300	299
	RK	coated	851	738	736	735	735
two non-convex semi-open cavities	CD	non-coated	238	252	186	286	268
	CD	coated	374	340	341	338	338
	RK	non-coated	123	146	145	145	145
	RK	coated	367	347	347	347	347

The performance of the preconditioner for different scatterers is presented, with respect to the order of the polynomial basis, in Figure 38. It is shown that the preconditioner works similarly with the both time discretization schemes. The small number of AMG cycles means that also the convergence factor is small and the preconditioner is efficient. There is not significant difference between different kind of scatterers for $r = 1$ and $r = 2$. The best performance of the preconditioner is achieved with $r = 2$. In the cases of non-convex scatterers, the element order $r = 3$ seems to need a remarkably large number of AMG cycles, whereas in the case of convex scatterers the preconditioning is done more efficiently.

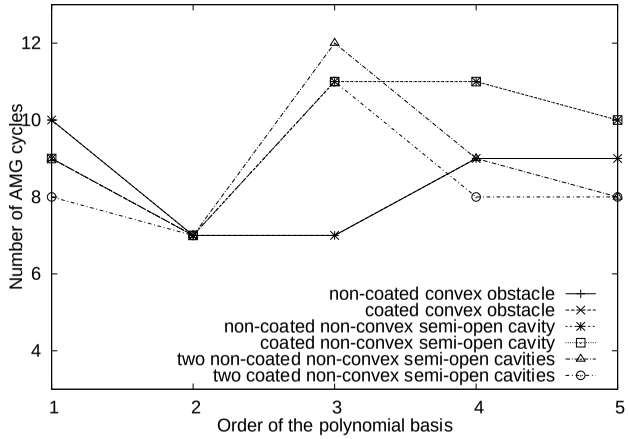
The comparison of CPU consumption between CD and RK time steppings at the first CG iteration is presented in Figures 39(a)-39(b). When the element order grows, the percentage of CPU time required by AMG behaves similarly for all the geometries. In the case of RK time stepping, this percentage is smaller and also the variations of in the proportion of CPU time are smaller between different values of the element order. The reason for this is that CPU time for the algorithm is larger in the case of RK time stepping. This is seen also in Figures 39(c)-39(d), where CPU times of the scattering examples at the first CG iteration are shown with respect to the number of non-zero elements in the stiffness matrix.



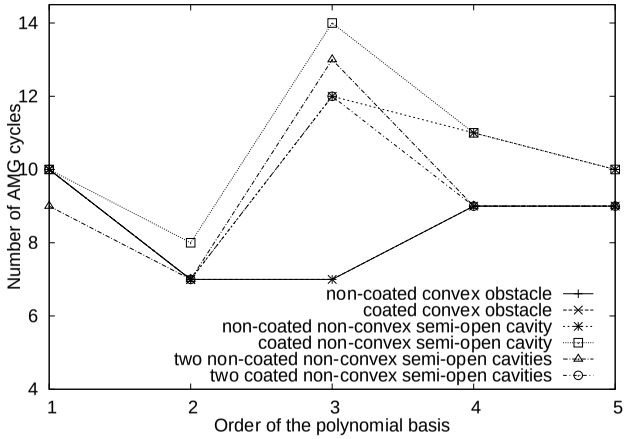
(a) Convergence factor of AMG in the case of RK time discretization.



(b) Convergence factor of AMG in the case of CD time discretization.

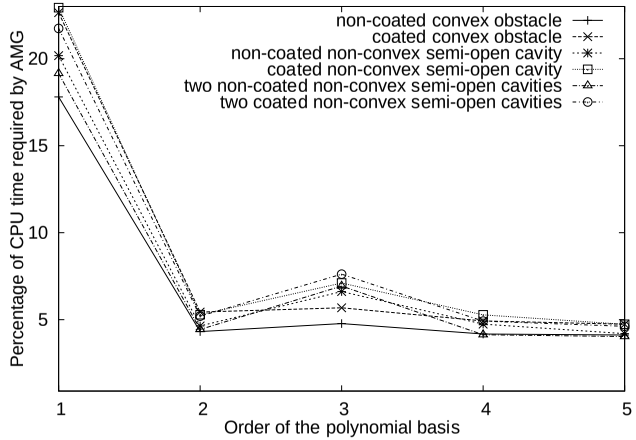


(c) Number of AMG cycles in the case of RK time discretization.

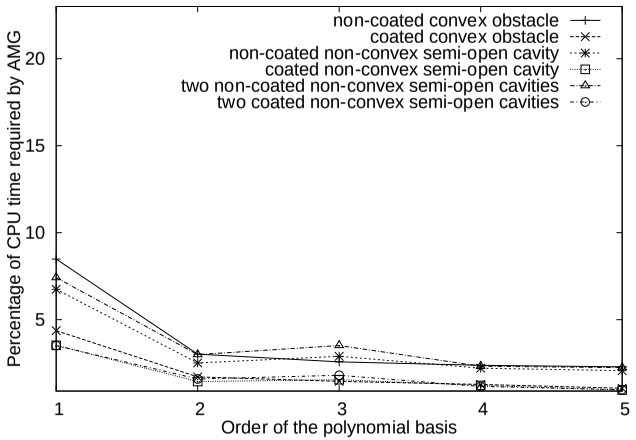


(d) Number of AMG cycles in the case of CD time discretization

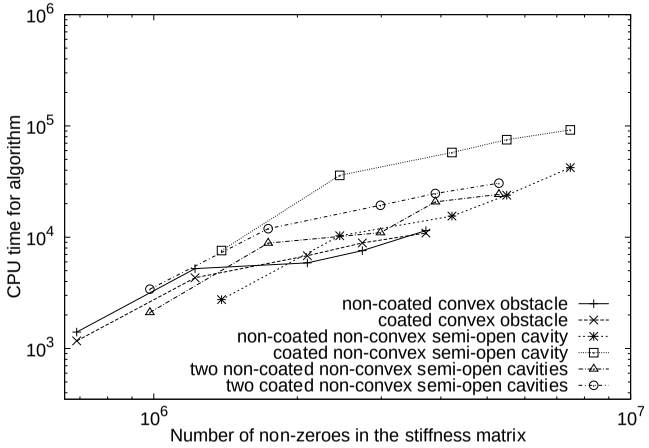
FIGURE 38 Convergence factor of AMG and number of AMG cycles at the first CG iteration with respect to the spectral order.



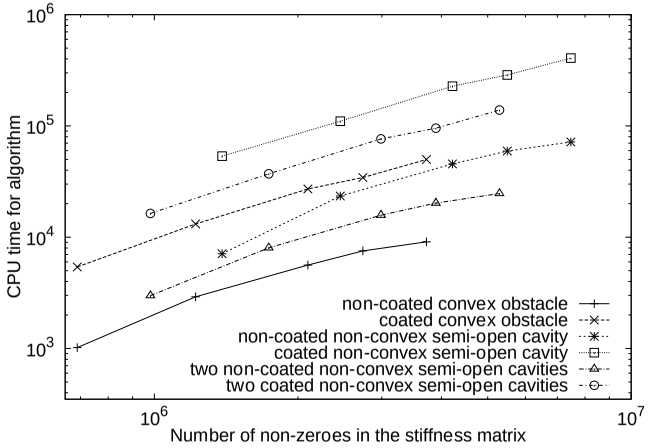
(a) Proportion of CPU time (in percent) required by the AMG cycles in the case of CD time discretization



(b) Proportion of CPU time (in percent) required by the AMG cycles in the case of RK time discretization.



(c) CPU time with respect to the number of non-zero elements in the stiffness matrix in the case of CD time discretization.



(d) CPU time with respect to the number of non-zero elements in the stiffness matrix in the case of RK time discretization.

FIGURE 39 Comparison of CPU consumption between CD and RK time steppings at the first CG iteration.

9.3 Elasticity solver with central finite difference time discretization

In this section, we consider the elasticity problem in an isotropic homogeneous elastic medium with $E = 5.75$, $\nu = 0.45$, and $\rho_s = 1$. For angular frequency we use $\omega = 4\pi$, unless other mentioned.

9.3.1 Accuracy of space discretization

Accuracy of h- and r-refinements, computed with $\omega = 4\pi$, is shown in Figure 40, and percentage of CPU time required by the AMG cycles in one CG iteration is shown in Figure 41. These results are in line with the results presented for acoustic wave equation in previous sections. Again, most of the computation time is consumed for solving the state and adjoint state equations. Although only small amount of time is used for preconditioning with AMG, this proportion is larger than what we discovered with the acoustic wave equation.

To show the benefit of preconditioning, computations with mesh step size $h = 1/4$ are performed both with and without preconditioning. CPU times in seconds for prenditioned and unpreconditioned algorithms are depicted in Figure 42, where DOF increases

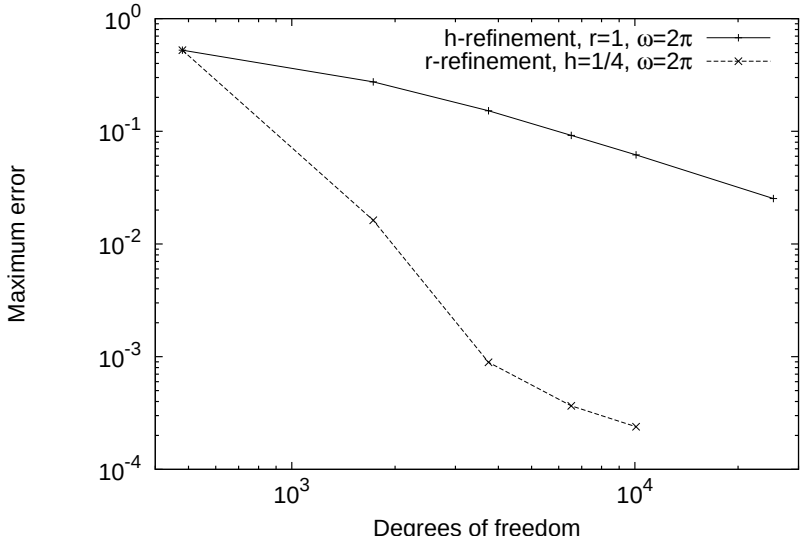


FIGURE 40 Maximum error versus degrees of freedom.

TABLE 7 Number of timesteps for different spectral orders.

r	1	2	3	4	5
Number of timesteps	420	520	460	360	260

as the order of the spectral element basis increases from 1 to 5. The preconditioned minimization seems to be many times faster than the unpreconditioned one. CPU time required by the AMG cycles in one CG iteration is only a small amount of the CPU time for the CG iteration.

We performed test showing the pollution effect also for the elasticity problem. Once again, we used coarser mesh with higher order elements according to the equation $\omega h = \pi 2^{r-6}$. In addition, we chose the number of timesteps to be twice the number of timesteps needed to satisfy the stability condition (see Table 7 and Figure 43). As the wavenumber grows, the error increases for all orders of the elements (see Figure 44). In the case of FEM, this effect is most significant.

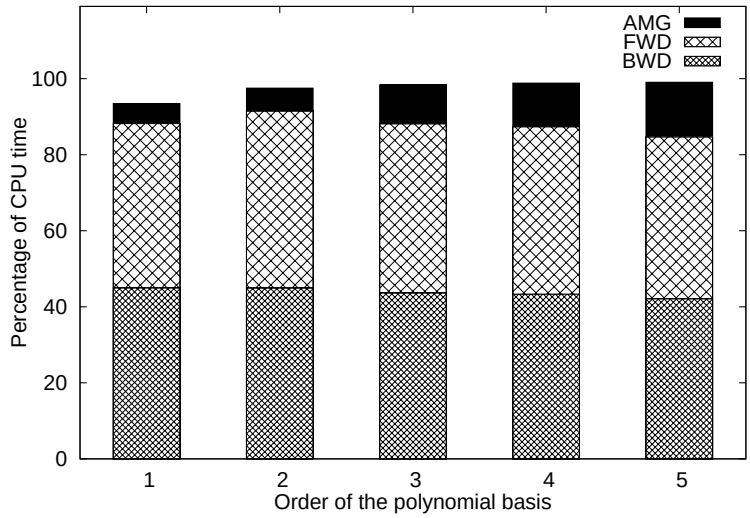


FIGURE 41 Proportion of CPU time (in percent) required by the AMG cycles in one CG iteration.

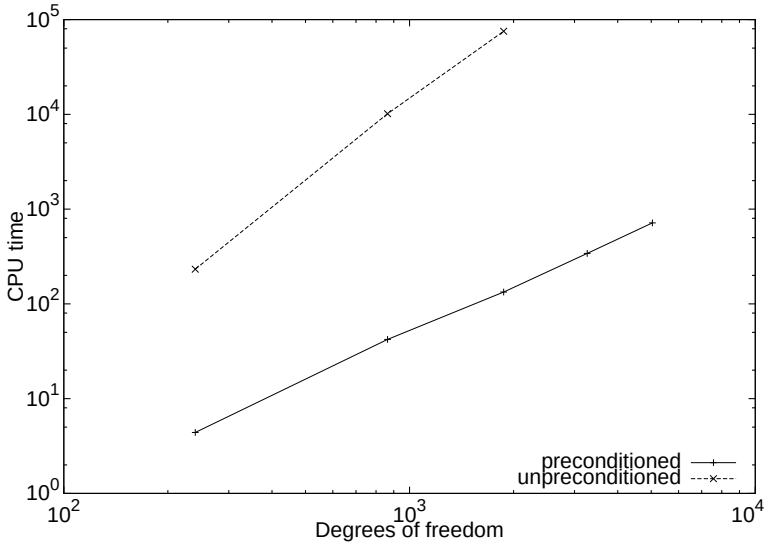
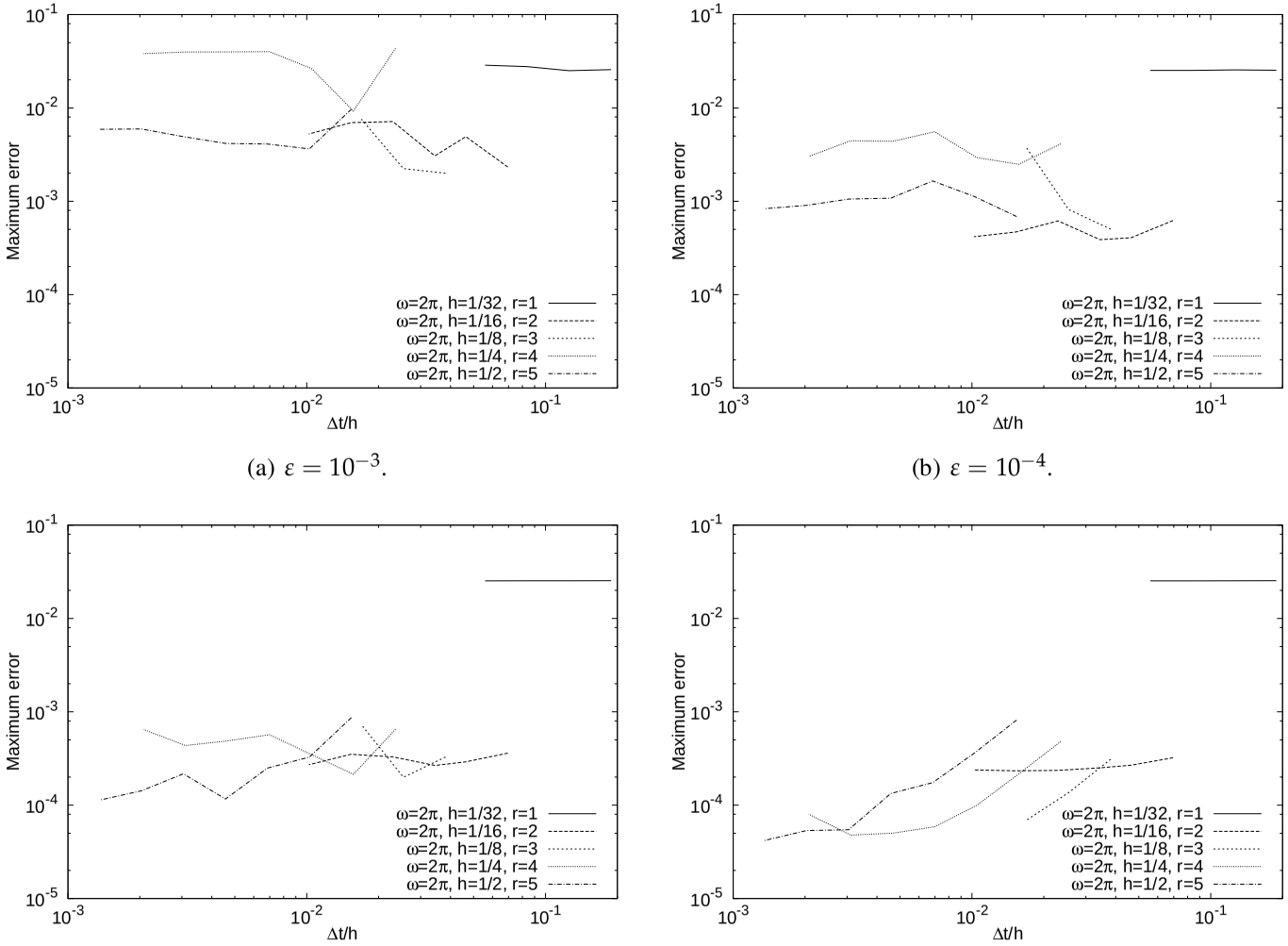


FIGURE 42 CPU time in seconds with respect to the number of degrees of freedom



(a) $\varepsilon = 10^{-3}$. (b) $\varepsilon = 10^{-4}$.

(c) $\varepsilon = 10^{-5}$. (d) $\varepsilon = 10^{-6}$.

FIGURE 43 Maximum errors obtained in the case of CD timestepping with four different stopping criteria ε.

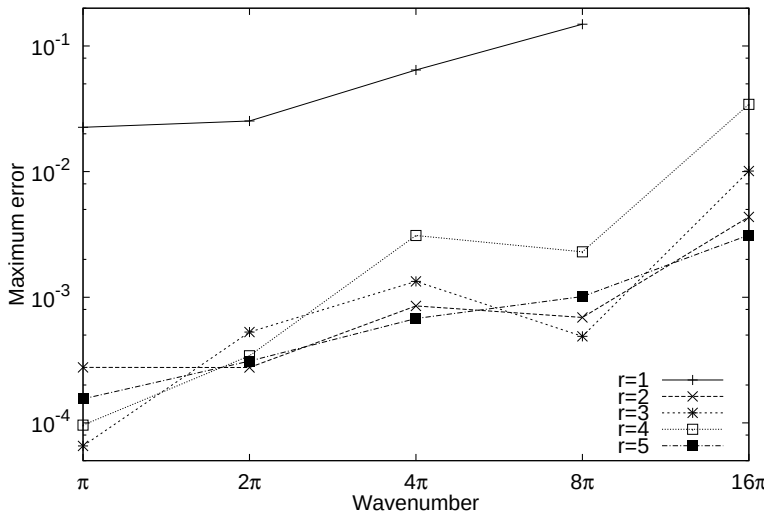


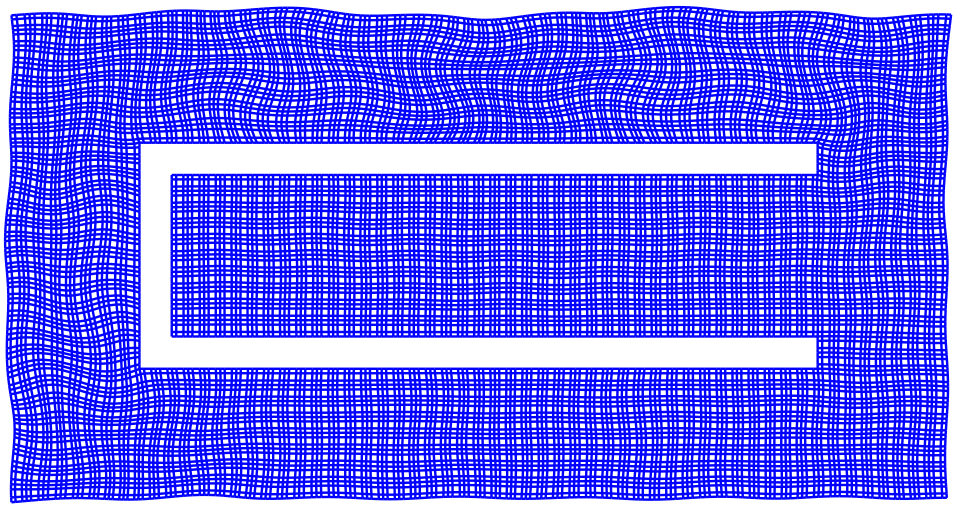
FIGURE 44 Behaviour of the error with respect to the wavenumber for different orders of the polynomial basis such that $\omega h = \pi 2^{r-6}$.

9.3.2 Elastic scattering

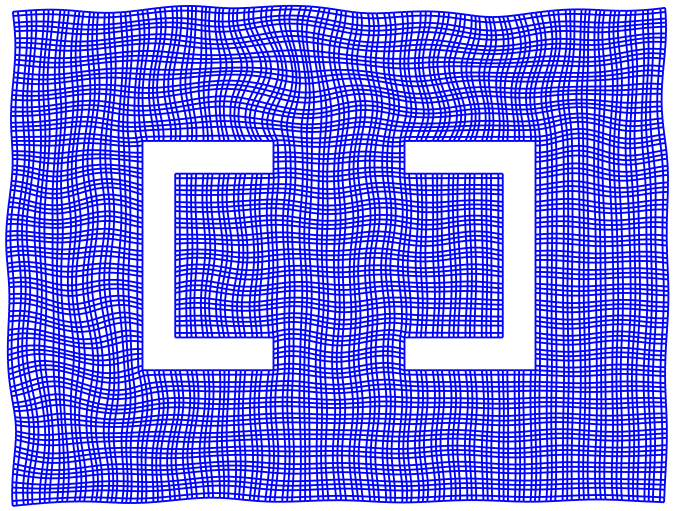
Number of iterations with three different kind of rigid obstacles (see Figures 5(a)-5(c)) with mesh stepsize $h = 1/8$ are discussed in Table 8. It is shown that the number of iterations needed to achieve the stopping criterion of the algorithm is not dependent on the number of degrees of freedom. Displacement fields of the cattering solutions are illustrated in Figures 45(a)-45(c). In these scattering experiments, CPU time required by the AMG preconditioner is less than 10% of the CPU time for the whole algorithm (see Figure 46).

TABLE 8 Number of iterations for scattering solutions with different geometries.

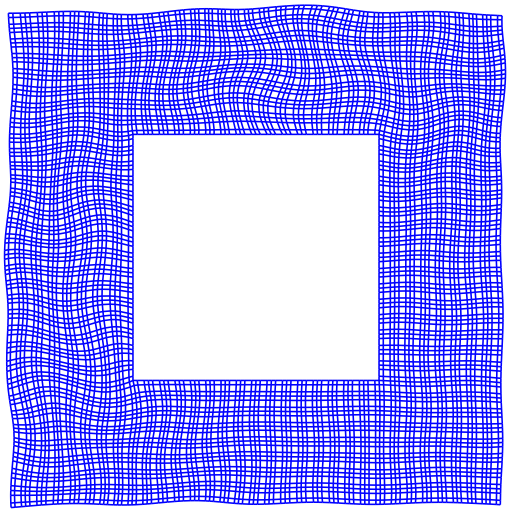
	convex obstacle (square)		two non-convex semi-open cavities		non-convex semi-open cavity	
r	DOF	iter	DOF	iter	DOF	iter
1	1728	61	2442	140	3472	115
2	6528	57	9270	167	13152	130
3	14400	93	20482	146	29040	132
4	25344	60	36078	129	51136	133
5	39360	72	56058	132	79440	138



(a) Non-convex semi-open cavity.



(b) Two non-convex semi-open cavities.



(c) Square obstacle.

FIGURE 45 Real parts of the elastic scattering solution $\mathbf{u_s}$ illustrated in spectral element mesh with $r = 3$.

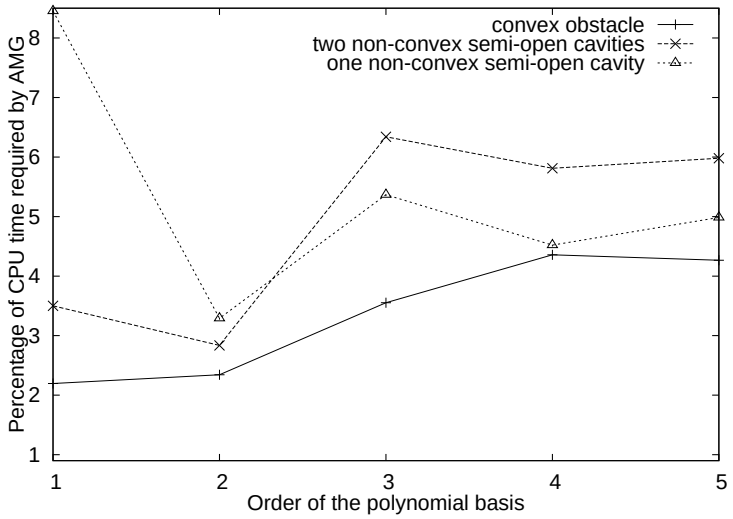


FIGURE 46 Percentage of CPU time required by AMG.

9.4 Comparison between central finite difference and Runge-Kutta time discretizations for the elasticity equation

We apply higher order time discretization scheme to the elasticity solver and consider the accuracy of spatial and temporal discretizations in Section 9.4.1 with several stopping criteria ε. Sections 9.4.2 and 9.4.3 are devoted to wave scattering results. For angular frequency we mainly use the value $\omega = 2\pi$.

9.4.1 Accuracy of space and time discretizations

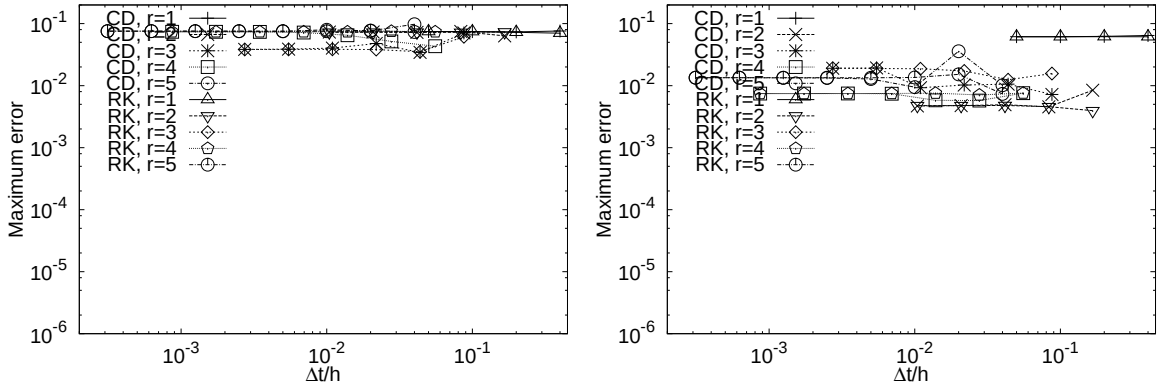
The accuracy and computational efficiency of the method is considered in an isotropic homogeneous elastic medium with $\rho = 2.7$, $c_p = 2$, and $c_s = 1$.

Approximation error with constant angular frequency

In the first experiment we have used constant spatial discretization such that the ratio between the order of elements r and the mesh stepsize h is $r/h \approx 20$. We have also compared the CD time discretization with the RK time discretization for element orders $r = 1, \ldots, 5$. In each cases, the number of timesteps needed for stability is first tested by

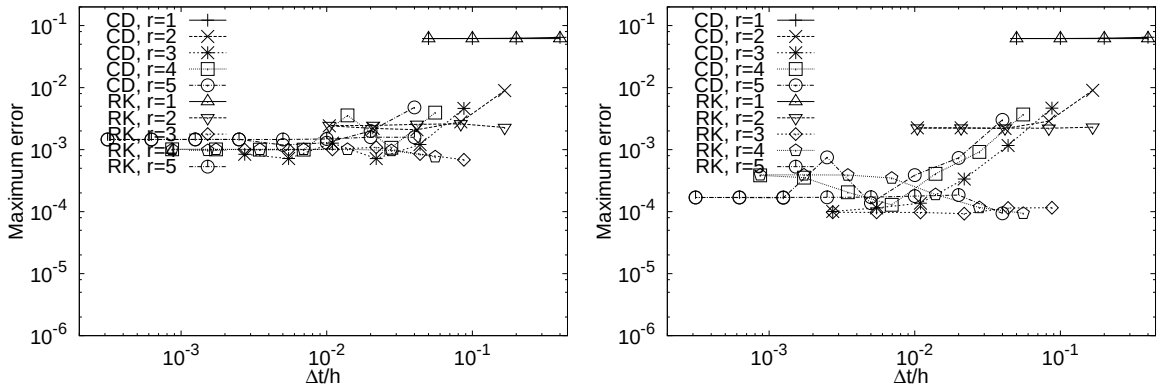
TABLE 9 Stability conditions.

	r	1	2	3	4	5
Number of timesteps	CD	50	60	80	90	100
	RK	50	60	80	90	100
$\Delta t/h$	CD	0.40	0.17	0.09	0.06	0.04
	RK	0.40	0.17	0.09	0.06	0.04



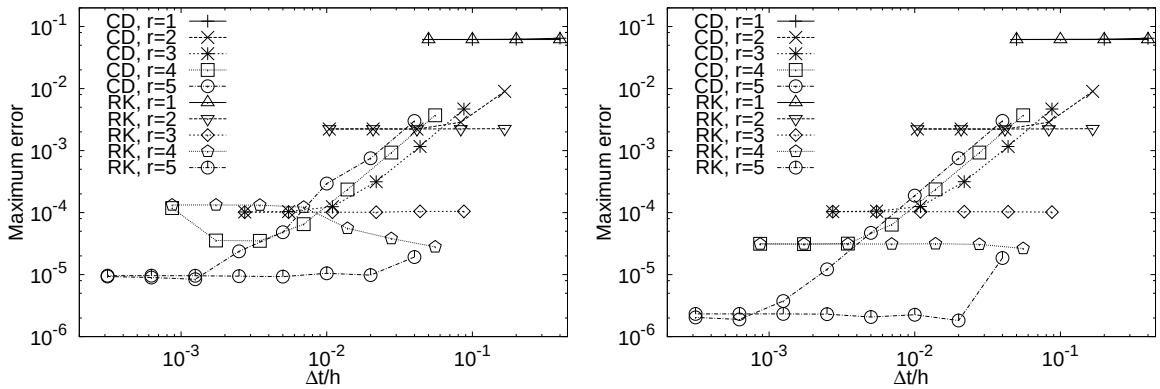
(a) $\varepsilon = 10^{-2}$

(b) $\varepsilon = 10^{-3}$



(c) $\varepsilon = 10^{-4}$

(d) $\varepsilon = 10^{-5}$



(e) $\varepsilon = 10^{-6}$

(f) $\varepsilon = 10^{-7}$

FIGURE 47 Maximum errors obtained in the case of CD and RK time stepping with six different stopping criteria ε.

using $10i$ timesteps per time period, for $i = 1, 2, 3, \ldots$, until a stable solution is achieved. Stability conditions corresponding to the largest stable timestep are given in Table 9. According to our numerical tests, these values are the same with both the CD and the RK time stepping. Moreover, Δt satifies

$$\frac{\Delta t}{h} = \frac{\alpha_r}{c_p\sqrt{2}}, \tag{157}$$

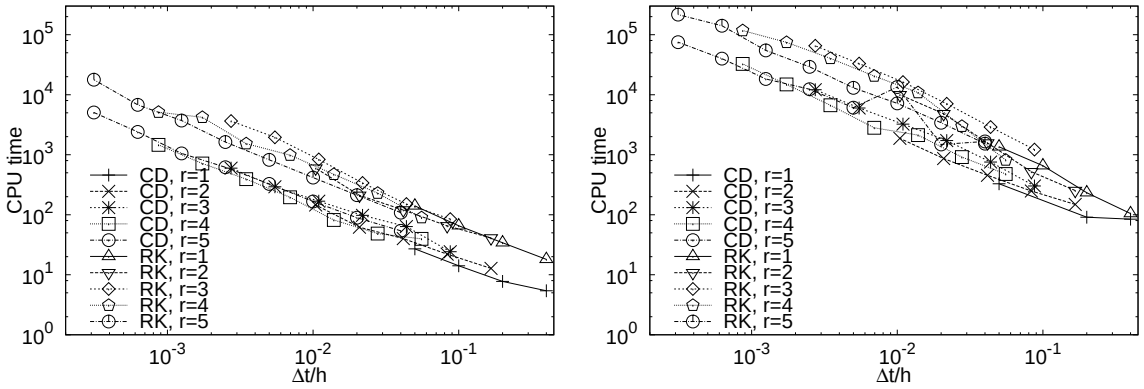
which is the well known CFL condition, where α_r is a stability constant for element order r [22].

We have started computations with the largest stable timestep and then multiplied the number of timesteps $N = T/\Delta t$ by two, until the problem for spectral order r is solved with $r + 3$ different number of timesteps. Proceeding this way a series of numerical results with various lengths of the timestep is achieved. Errors between the analytical solution and the experimental result are computed as L^∞-norms. Accuracy of the numerical solution is shown in Figure 47 as a function of the ratio between the time step Δt and the mesh step size h for both CD and RK time steppings with five element orders r and six different stopping criteria ε.

From results shown in Figures 47(a)-47(f) we can make observations regarding, for instance, the orders of temporal and spatial accuracy. As the order of the approximation in space increases, the solution becomes more accurate until the effect of stopping criterion or error of time or space discretization becomes dominant. In Figure 47(a), only accuracy of the results with $r = 1$ (i.e. the special case of spectral elements which corresponds to classical bilinear finite elements) is not limited by the stopping criterion. When the stopping criterion is tightened (see Figures 47(a)-47(f)), more accurate solutions are reached.

Depending on the accuracy of the time discretization, the error of temporal discretization might be dominating with large timesteps. This is shown especially in the case of CD time discretization. For example in Figure 47(f), timestep refinement decreases the error with CD time discretization for element orders higher than one until the level of space discretization is achieved. This is because CD time discretization is only of second order accuracy, and therefore the error of time discretization is larger than the other error sources unless very fine time steps are used. With time steps fine enough, the maximum error with respect to the length of timestep is not decreasing even if smaller timesteps are used. This means that the temporal error is eliminated. In Figure 47(f), the curve with a particular spectral order r converges to a characteristic error level when the temporal error is eliminated. This is something that we expected, since the error remaining, after the temporal error is eliminated, is the error of spatial discretization. Every curve represents computations with a particular spectral order which has a characteristic discretization error. Apparently, the order of the space discretization error decreases when higher order elements are used.

Naturally, for each element order r the solution with RK time stepping is at least as accurate as the one computed with CD time stepping. For the element order $r = 1$, the error of space discretization is dominant over the error of time discretization for all stable lengths of the timestep. Then, using second order and fourth order accurate



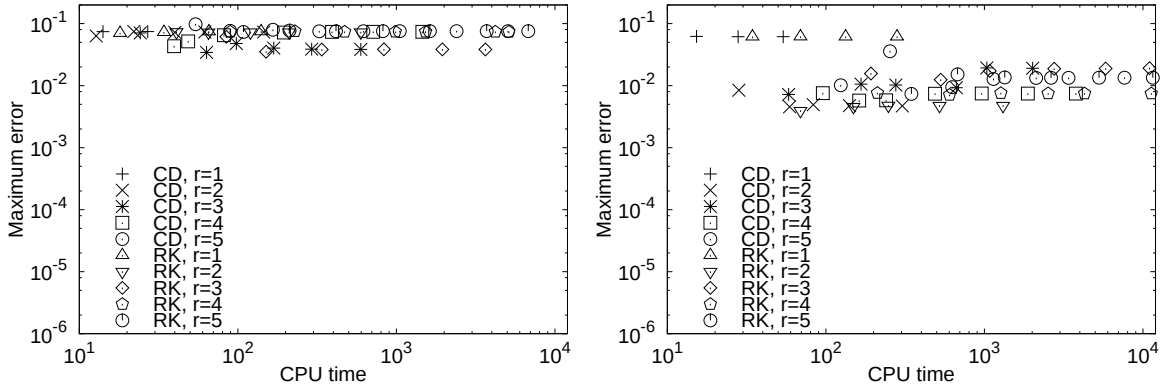
(a) $\varepsilon = 10^{-2}$ (b) $\varepsilon = 10^{-7}$

FIGURE 48 Comparisons of the computational efforts (CPU time in seconds) obtained in the case of CD and RK time stepping with two different stopping criteria ε.

time schemes lead to equally accurate results. The residual stayed slightly over 10^{-7} for some values of Δt in the algorithm with CD time discretization and $r = 1$. This is why there is a blank space in the corresponding curve in Figure 47(f). When higher order elements are used, we can see the benefit of RK time discretization. With RK time discretization, the error is constant for $r = 1,\ldots,4$. These horizontal curves in Figure 47(f) represent the situation where space discretization is dominant. These error levels are the ones which are typical for particular spectral elements. Only in the case where $r = 5$, the accuracy is improved with timestep refinement. The trifling oscillation in the left part of the curve presenting the error with RK time discretization and $r = 5$ is obviously caused by stopping criterion since we have eliminated the errors of spatial discretization and approximation of the radiation condition. These eliminations have been done by using polygonal geometries and the modified right-hand side function which satisfies the absorbing boundary condition. As we have seen, the reduction of the error is blocked by the stopping criterion of the CG method and the fixed time discretization, which makes the comparison of higher element orders difficult.

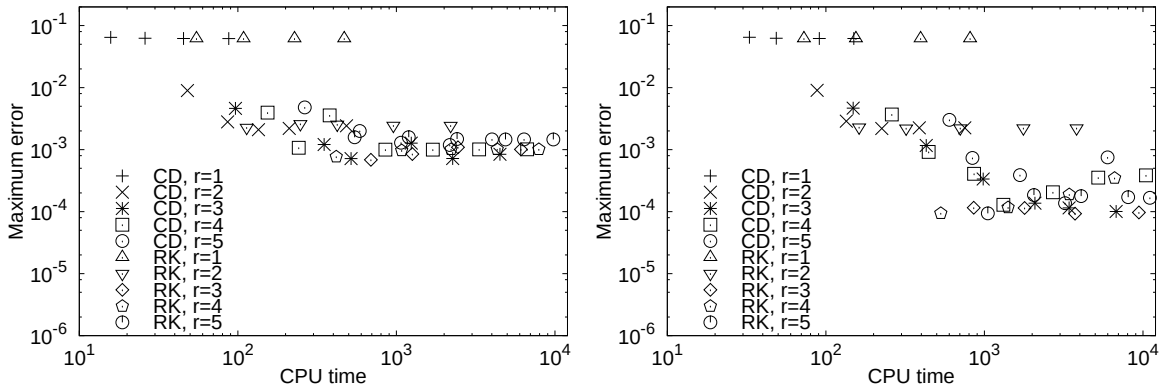
When stopping criterion ε is tightened, also the number of CG iterations needed to attain the stopping criterion grows. This implies a larger computational effort (see Figure 48). As discussed in Section 7, the computational cost of the CG algorithm depends on the number of timesteps $N = T/\Delta t$, the number of degrees of freedom, the order of element r, and the number of CG iterations. At each curve, presenting the computational cost with respect to $\Delta t/h$ in Figure 48, the number of degrees of freedom and the element order are constants. The linear dependence between CPU time and the product of number of timesteps and number of iterations tells that the order of number of iterations remains constant, when timestep refinement is done.

According to Figures 47, 48, and 49 the spatial accuracy with FEM ($r = 1$) is so poor that the most efficient solution strategy with FEM is to use a combination of CD time discretization, the largest stable timestep, and rough stopping criterion. With higher order elements, the best choise for time discretization depends strongly on the stopping criterion and the length of timestep. With $\varepsilon \gtrsim 10^{-4}$ the CD time discretization and sufficiently large time steps might be recommended, since these allow to compute



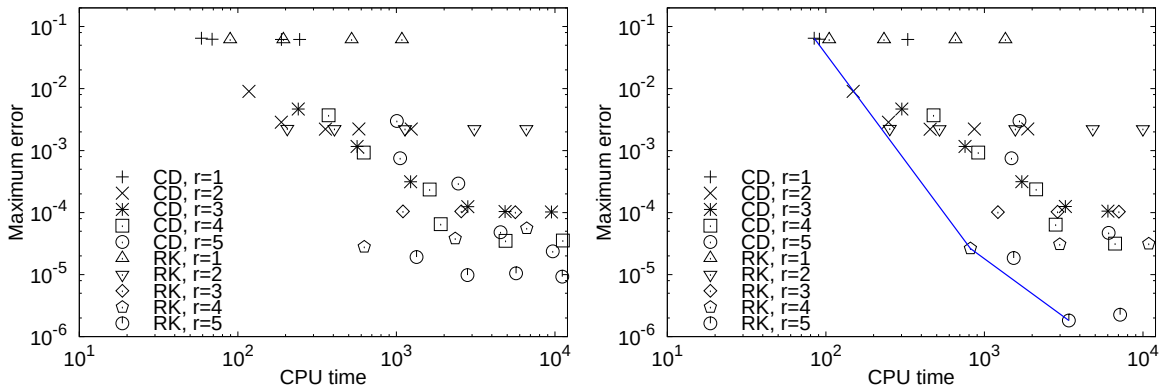
(a) $\varepsilon = 10^{-2}$

(b) $\varepsilon = 10^{-3}$



(c) $\varepsilon = 10^{-4}$

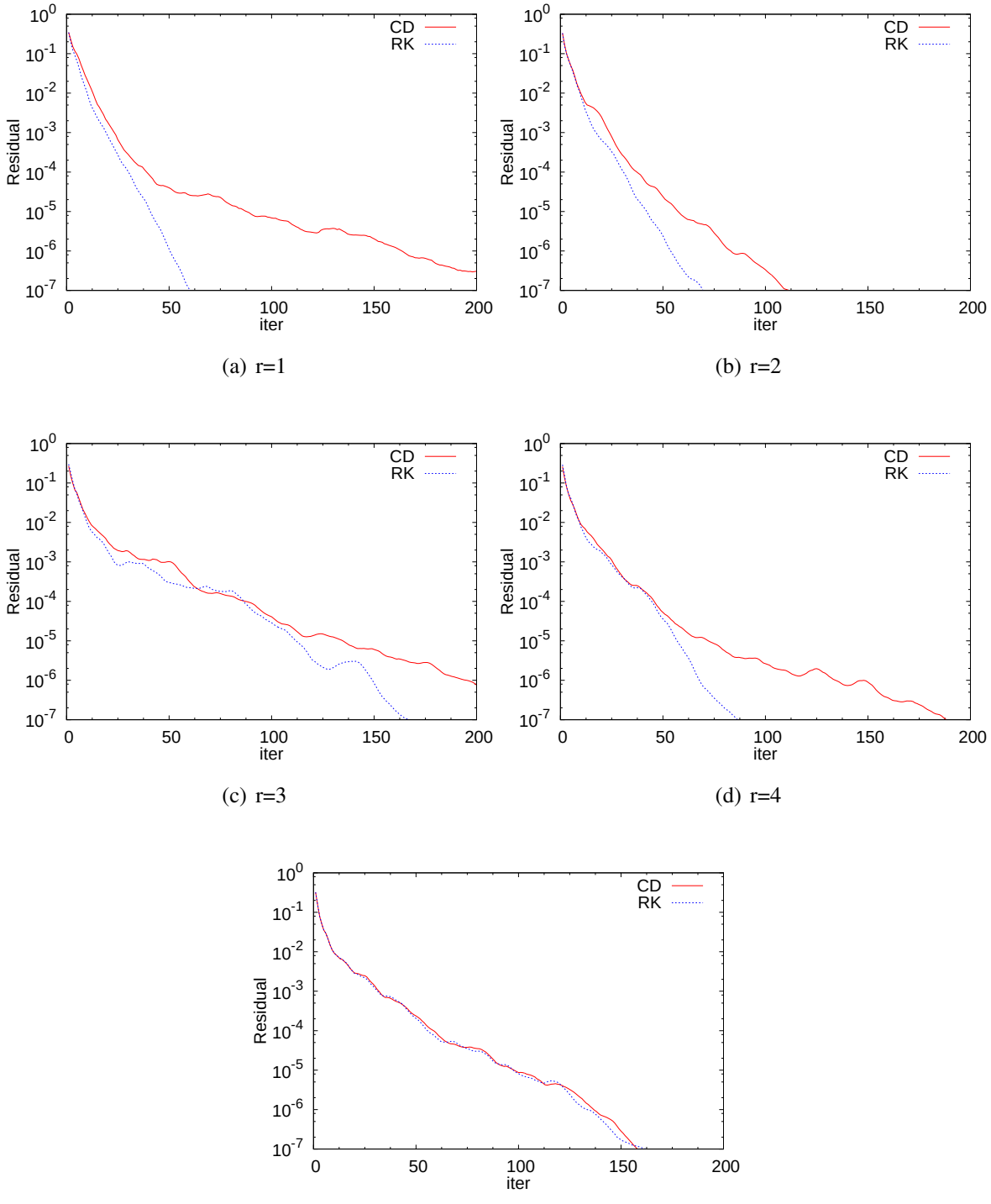
(d) $\varepsilon = 10^{-5}$



(e) $\varepsilon = 10^{-6}$

(f) $\varepsilon = 10^{-7}$

FIGURE 49 Maximum errors with respect to CPU time (in seconds) obtained in the case of CD and RK time stepping with six different stopping criteria ε.



(a) r=1 (b) r=2

(c) r=3 (d) r=4

(e) r=5

FIGURE 50 Convergence history.

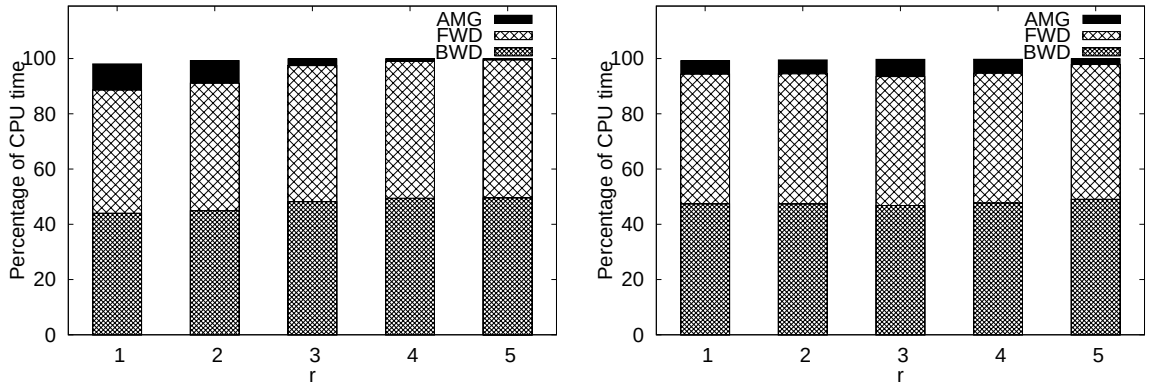
the solution utilizing only small amount of CPU time. It should be taken into account that using such large stopping criteria involves an error which deteriorates the accuracy of the method when higher order elements are used. To take full advantage of the high accuracy provided by spectral elements, smaller stopping criteria should be used. Then, the RK time discretization becomes more efficient than the CD time discretization. This can be seen in Figure 49, where errors are presented as functions of CPU time. In the case of RK time discretization with $r = 4$ and $\varepsilon \leq 10^{-5}$ the error of stopping critetion has only a small influence (or no influence at all) on the solution. Moreover, the error of time discretization is not affecting at this point, which makes possible to use the largest stable timestep accurately. These properties induce the remarkable efficiency for $r = 4$ in connection with RK time discretization and the largest stable timestep, which is seen in Figures 49(d)-49(f).

Choosing the combination of time discretization scheme and order of space discretization can be considered as a multiobjective optimization task (see [80]). That is, we have two criteria (error and computational cost), both of which are to be minimized. Pareto optimal front, defined by the points offering the minimum computation time with maximum accuracy, is drawn with blue color in Figure 49(f). As we can see, the Pareto optimal set includes the points of the CD time discretization with $r = 1$ and $r = 2$ and the points of the RK time discretization with $r = 4$ and $r = 5$. Also a point with the RK time discretization and $r = 2$ is near the Pareto optimal front. We can decide which of the points laying on the Pareto front suits best for our purposes. When solutions are needed quickly, we compromize with accuracy, and the optimal choice is, for instance, the CD time discretization with conventional finite elements ($r = 1$). When high accuracy is needed and sufficient computational capacity is available, it is best to use the RK time discretization with $r = 4$ or $r = 5$. It is worth mentioning that the Pareto point with the RK time discretization and $r = 5$ is not achieved with the largest stable timestep.

Since we are interested in partular in the accuracy of spatial discretization, we hereafter concentrate on the case where the error of time discretion is negligible. By using the results shown in Figure 47(f), we have chosen for each order of element the number of time steps which eliminates the temporal error. These numbers of time steps are shown in Table 10. It is noteworthy that with the CD time discretization the number of timesteps needed to attain the given accuracy is at least ten times that of the RK time discretization for $r \geq 3$. This confirms the better efficiency of the RK time discretization with higher order elements. The secret behind this behaviour, which is not in line with proportions of arithmetic operations needed at each CG iteration, lies in the number of CG iterations. This is seen in Figure 50, where the values of residual are presented with respect to number of iterations. Since the algorithm with RK time discretization solves the problem at

TABLE 10 Number of timesteps to attain the error level of spatial discretization for different spectral orders.

	r	1	2	3	4	5
Number of	CD	50	140	800	1300	3600
timesteps	RK	50	60	80	90	210



(a) CD (b) RK

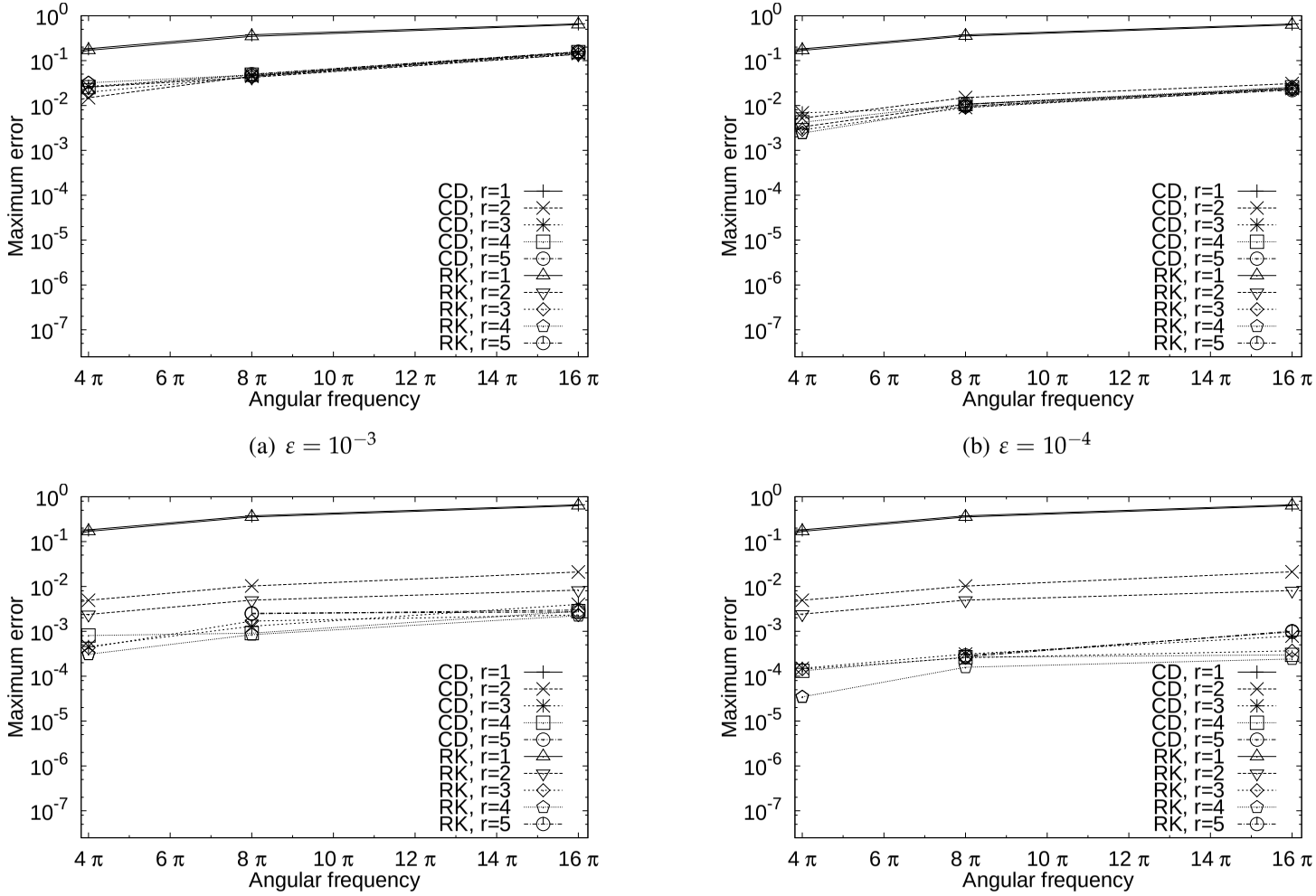
FIGURE 51 Comparisons of the proportions of CPU time (in percent) required by AMG cycles in the case of the CD and the RK time steppings with $\varepsilon = 10^{-7}$.

each iteration in a more accurate manner than the algorithm with CD time discretization, the RK version needs smaller number of iterations. This phenomenon is even emphasized, when the stopping criterion is tightened. We can see one exception of this rule in Figure 50(e). The reason for this might be that the error of the RK time discretization is eliminated by using considerably large number of time steps. In some cases of the CD time discretization (see Figures 50(a) and 50(c)), the convergence rate becomes slow (or convergence stops) when the value of residual is small enough. Of course, this can happen also in the case of the RK time discretization, but with the CD time discretization this kind of limit is detected with rather small number of iterations and large value of the residual. The proportion of CPU time required for different parts of the CG algorithm, with number of timesteps shown in Table 10, is seen in Figure 51.

Pollution effect

Even though we have eliminated the main error sources, numerical dispersion deteriorates the accuracy of solutions with small waves. The computed wavenumber differs from the wavenumber of the exact solution, and with high angular frequencies this part of approximation error becomes dominant. To show that using higher order elements alleviates this inaccuracy, we have performed another set of experiments by varying both the angular frequency and the resolution of the mesh such that $\omega h = r\pi/10$. Since the efficiency of the method is not getting significantly better with $\varepsilon = 10^{-7}$ than with $\varepsilon = 10^{-6}$, and the method with the CD time discretization had problems to converge to the stopping criterion $\varepsilon = 10^{-7}$ with certain lengths of the timestep, we, from now on, concentrate especially on the case with $\varepsilon = 10^{-6}$. Then, the stopping criterion is not significantly limiting the accuracy of the numerical solutions computed with the CG algorithm.

The accuracy of the solution with respect to the angular frequency is presented in Figure 52. As the wavenumber grows, the error increases for all orders of the elements. In the case of classical finite element discretization, i.e. $r = 1$, the error becomes considerably large as the wavenumber increases. This happens even if ωh is kept constant. With higher orders, the pollution effect is not eliminated but the accuracy is significantly better



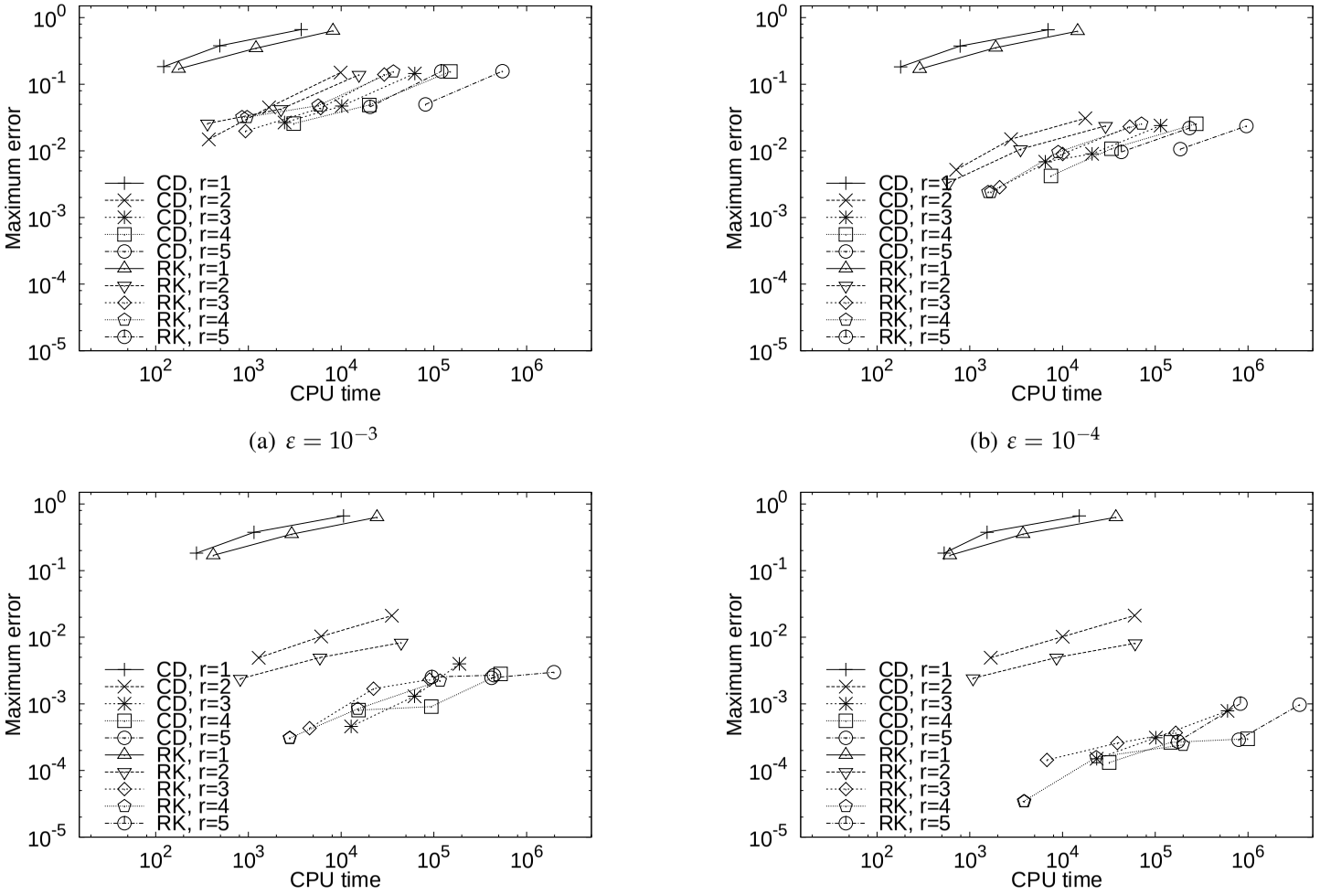
(a) $\varepsilon = 10^{-3}$

(b) $\varepsilon = 10^{-4}$

(c) $\varepsilon = 10^{-5}$

(d) $\varepsilon = 10^{-6}$

FIGURE 52 Maximum errors obtained in the case of the CD and the RK time stepping with four different stopping criteria ε.



(a) $\varepsilon = 10^{-3}$

(b) $\varepsilon = 10^{-4}$

(c) $\varepsilon = 10^{-5}$

(d) $\varepsilon = 10^{-6}$

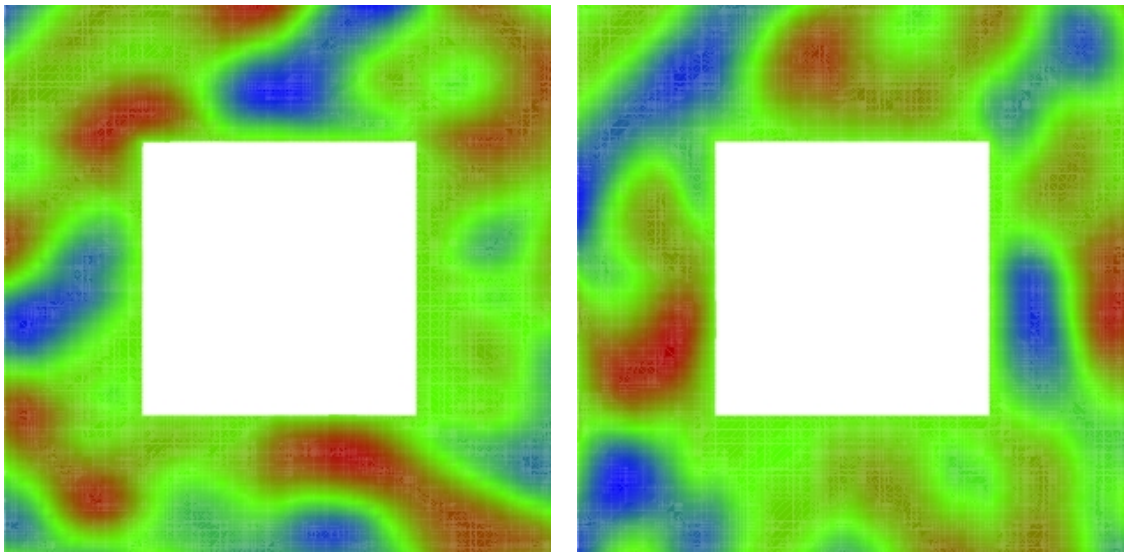
FIGURE 53 Maximum errors with respect to CPU time in the case of the CD and the RK time stepping with four different stopping criteria ε.

also for high angular frequencies. As seen in Figure 52, the same level of accuracy is attained with the CD and the RK time discretizations also for higher wavenumbers when error of time discretization is eliminated. With the RK time discretization this level of accuracy is achieved with lower computational cost than with the CD time discretization (see Figure 53). From Figures 52 and 53 we further notice that CPU time for algorithm grows with wavenumber. The reason for this is the increase in number of CG iterations.

9.4.2 Elastic scattering by a rigid obstacle

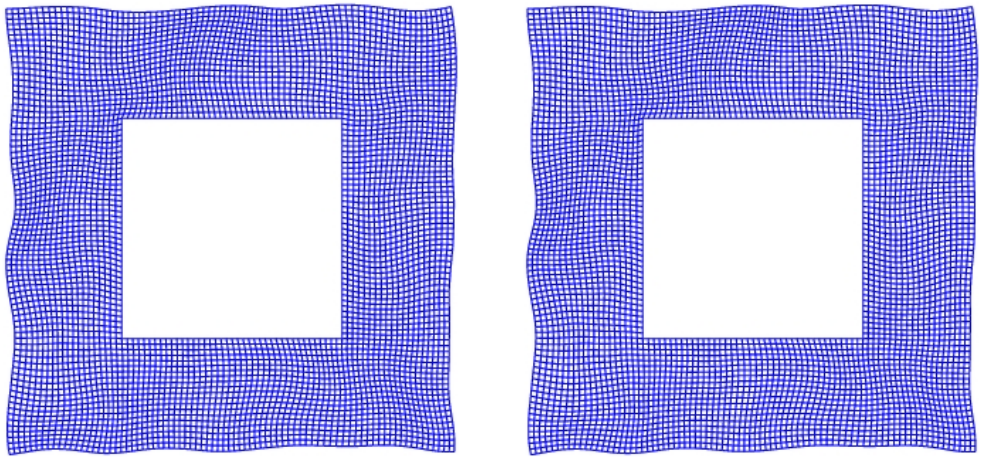
We now discuss elastic scattering by a rigid square obstacle of side length 2 in an isotropic homogeneous elastic medium. The material parameters are the same as in the previous test case, and the absorbing boundary Γ_{ext} is located around the obstacle at a distance of 1 perpendicular to each side of the obstacle. Contours of the real parts of the solution components $\mathbf{u}_{\mathbf{s}1}(\mathbf{x})$ and $\mathbf{u}_{\mathbf{s}2}(\mathbf{x})$ are presented with $r = 3$ in Figure 54 with $\omega = 2\pi$, $\varepsilon = 10^{-7}$, $\rho = 2.7$, $c_p = 2$, and $c_s = 1$. Solution of the elastic displacement $\mathbf{e}_0$ is equal to the real part of the time-harmonic wave $\mathbf{u}_\mathbf{s} = (\mathbf{u}_{\mathbf{s}1}(\mathbf{x}), \mathbf{u}_{\mathbf{s}2}(\mathbf{x}))$, which is illustrated in Figure 55 with five different element orders. In this picture, also the structure of spectral elements is shown.

To take closer look at the iteration numbers, computational effort, and accomplishment of the preconditioner, we tested the above mentioned properties with respect to the relaxation parameter for AMG (see Figure 56). The best performance of the AMG solver is achieved when the value of the relaxation parameter, depending on the order of elements, is between 1.2 and 1.6 (see Figures 56(c)-56(d)). From Figure 56(b), we see that the computational effort is considerably larger for $r = 5$ than for the other element orders. This is because such a fine timestep is used with $r = 5$ to eliminate the temporal error.

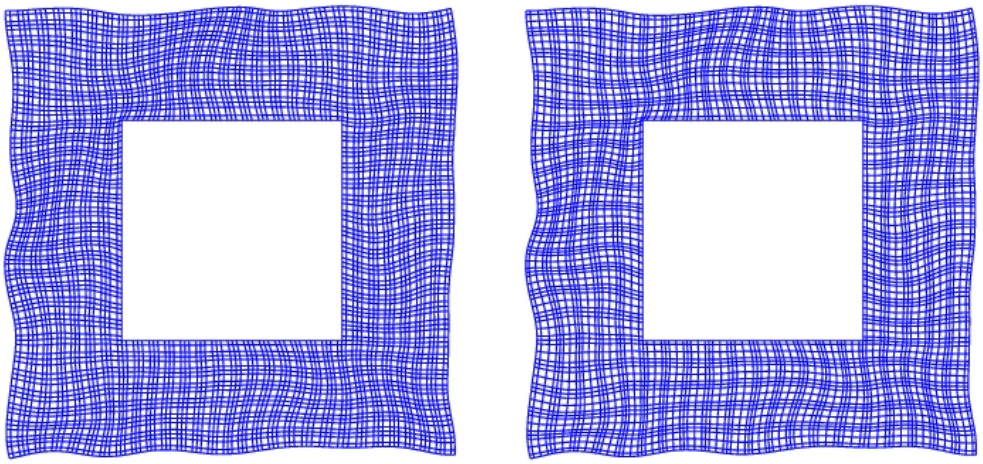


(a) Re($\mathbf{u}_{\mathbf{s}1}$) (b) Re($\mathbf{u}_{\mathbf{s}2}$)

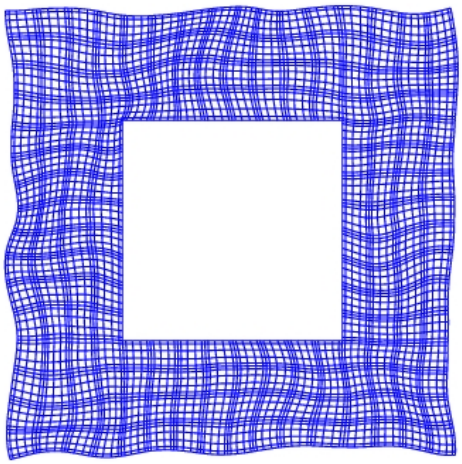
FIGURE 54 Components of the real part of the elastic scattering solution $\mathbf{u}_\mathbf{s}$ in homogeneous domain, where $\rho = 2.7$, $c_p = 2$, and $c_s = 1$.



(a) $r = 1$ (b) $r = 2$

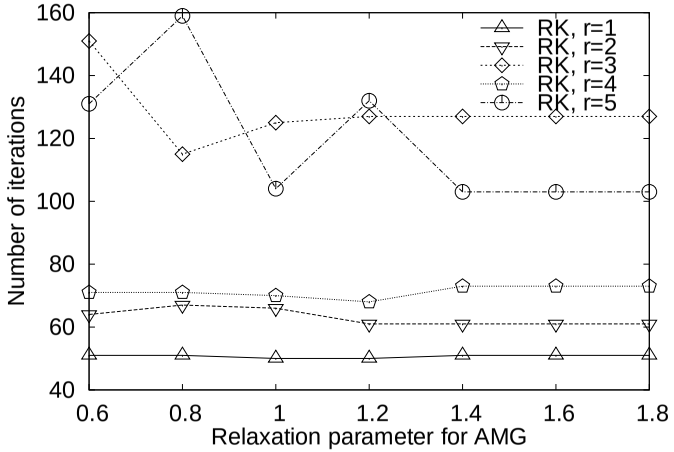


(c) $r = 3$ (d) $r = 4$

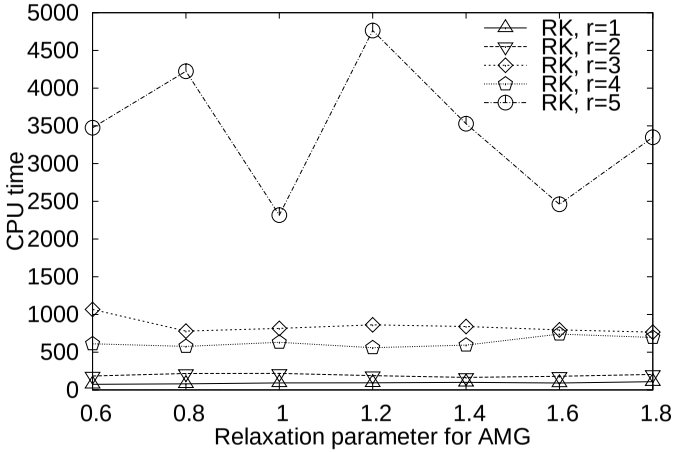


(e) $r = 5$

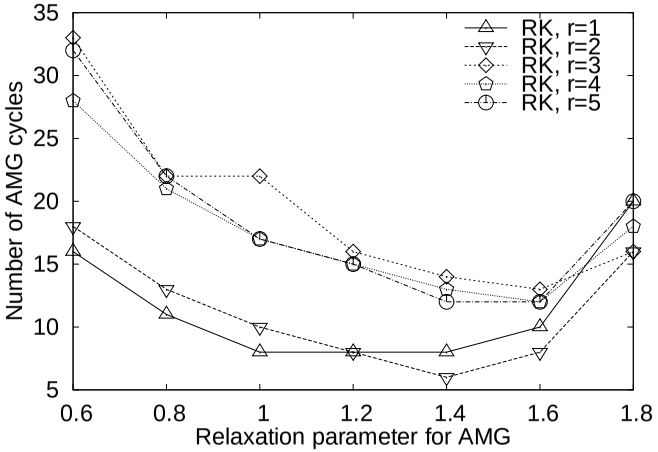
FIGURE 55 Real parts of the elastic scattering solution $\mathbf{u_s}$ in the case of RK time stepping with $\omega = 2\pi$ and $\varepsilon = 10^{-7}$ in homogeneous domain, where $\rho = 2.7$, $c_p = 2$, and $c_s = 1$.



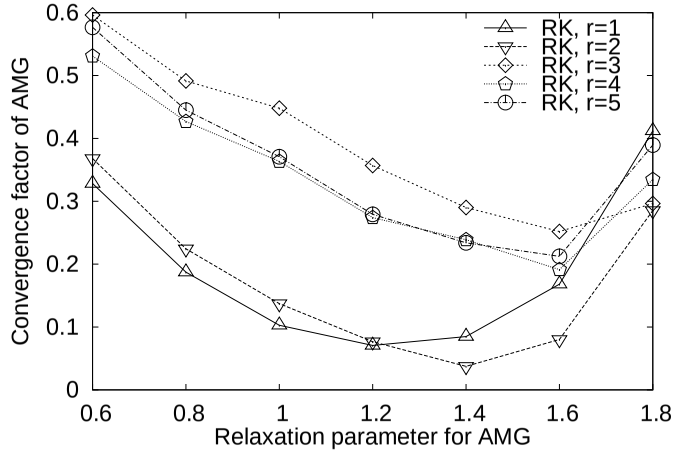
(a) The number of iterations with respect to the relaxation parameter for AMG.



(b) CPU time with respect to the relaxation parameter for AMG.



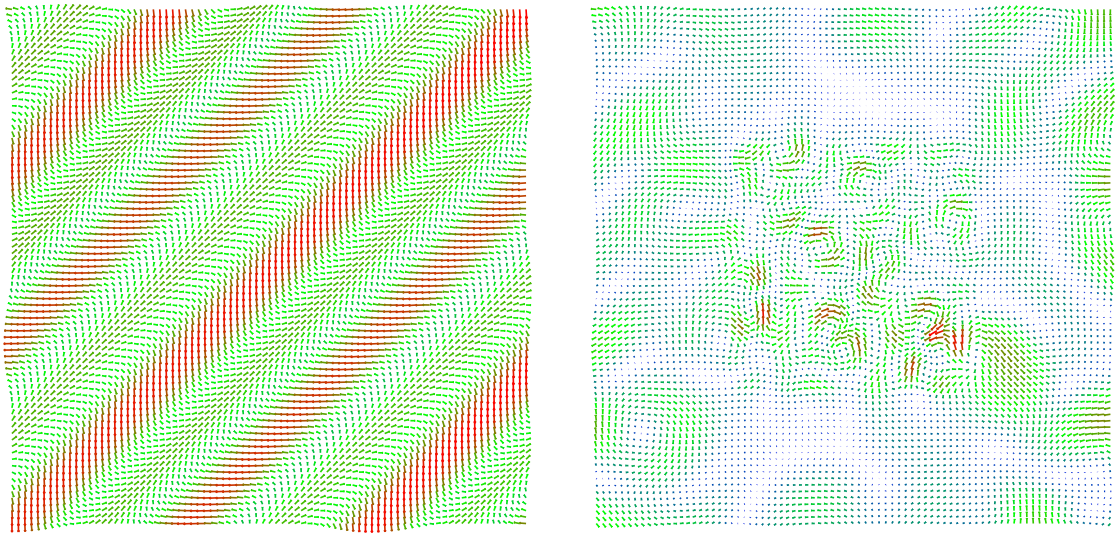
(c) Number of AMG cycles at the first CG iteration with respect to the relaxation parameter for AMG.



(d) Convergence factor at the first CG iteration with respect to the relaxation parameter for AMG.

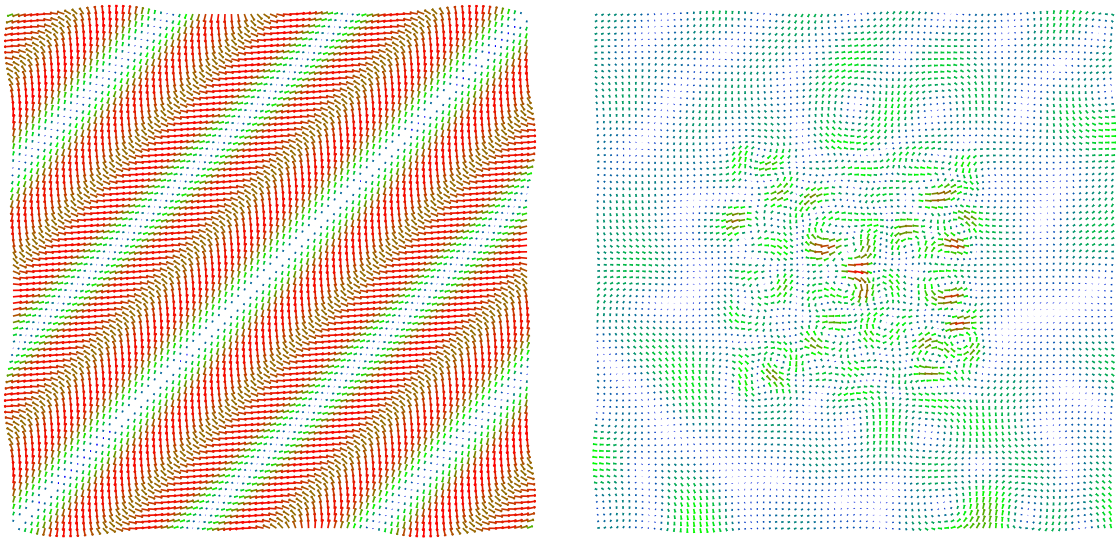
FIGURE 56 Elastic scattering by a square obstacle in the case of RK time stepping with $\omega = 2\pi$ and $\varepsilon = 10^{-6}$ in homogeneous domain.

The number of iterations is varying between different element orders more than expected (see Figure 56(a)). The reason for this is probably the tightness of the stopping criterion ε. Also our previous experiments show that when a very tight stopping criterion is used, the convergence rate might become lower or the residual might converge to a level which is higher than the stopping criterion. That is why the number of iterations is almost constant for all element orders when $\varepsilon \gtrsim 10^{-4}$, but differs to a certain extent between the element orders for stricter stopping criteria.



(a) Homogeneous domain. (b) Heterogeneous domain.

FIGURE 57 Displacement vector fields for the real part of the solution of elastic scattering in homogeneous and heterogeneous domain.



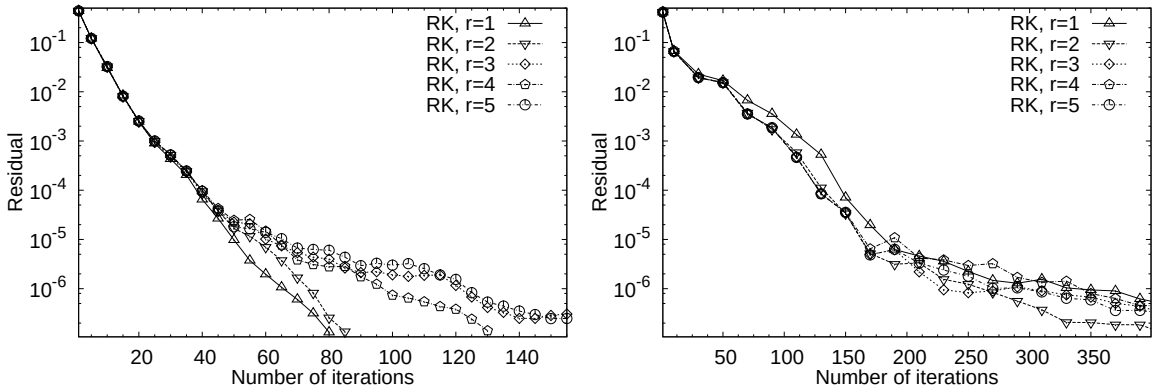
(a) Homogeneous domain. (b) Heterogeneous domain.

FIGURE 58 Velocity vector fields for the real part of the solution of elastic scattering in homogeneous and heterogeneous domain.

9.4.3 Wave propagation in homogeneous and heterogeneous materials

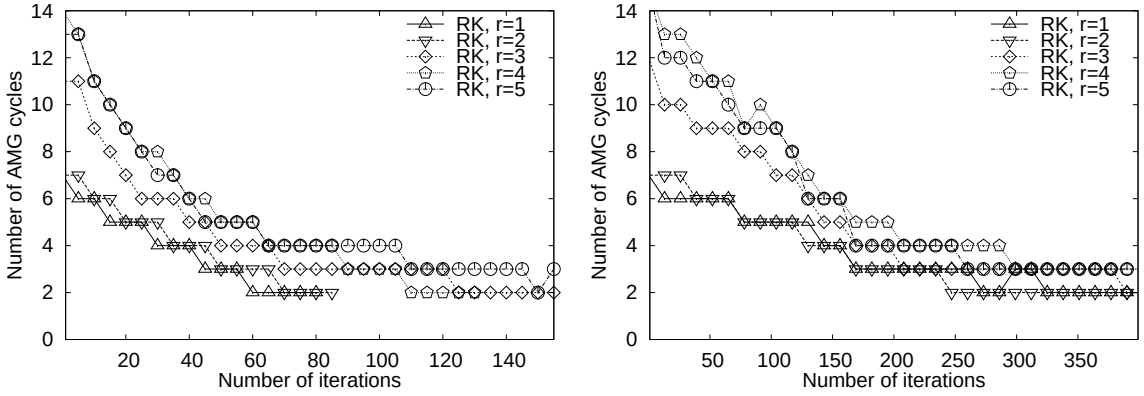
In this Section, we consider the wave propagation in homogeneous and piecewise heterogeneous media without rigid obstacles. Now, the boundary Γ_0 is not involved, and to avoid problems arising from the singularity of the stiffness matrix, we have used $\mathcal{K} + \frac{\mathcal{M}}{10^6}$ instead of $\mathcal{K}$ in preconditioning. In the homogeneous case, the domain is a square $\Omega = [0,4] \times [0,4]$ surrounded by the boundary Γ_{ext} and provided with material parameters $c_p = 2$, $c_s = 1$, and $\rho = 2.7$. The piecewise heterogeneous material is constructed by fixing the above described material parameters in $\Omega \setminus ([1,3] \times [1,3])$ and the material parameters $c_p = 0.6$, $c_s = 0.4$, and $\rho = 7.19$, into a square $[1,3] \times [1,3]$. The simulation results with homogeneous and heterogeneous material are shown in Figures 57 and 58 as displacement and velocity fields.

Convergence histories of simulations in homogeneous and heterogeneous media, in which relaxation parameter 1.2 is used for AMG, are shown in Figure 59. In both of these cases, where residuals are plotted with respect to the number of iterations, the convergence



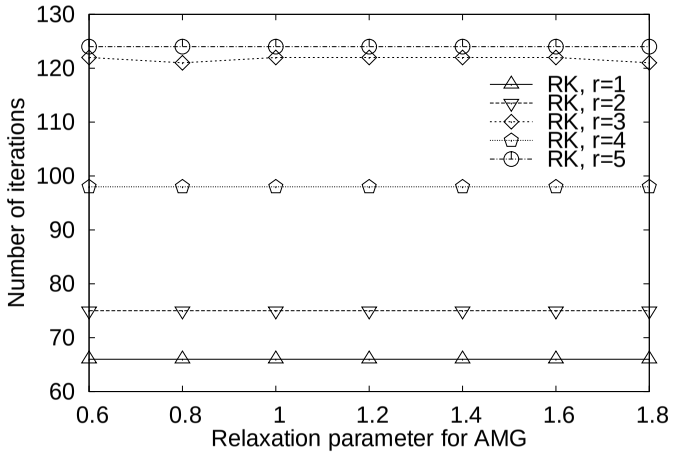
(a) Homogeneous domain. (b) Heterogeneous domain.

FIGURE 59 Convergence histories of elastic scattering in homogeneous and heterogeneous domain.

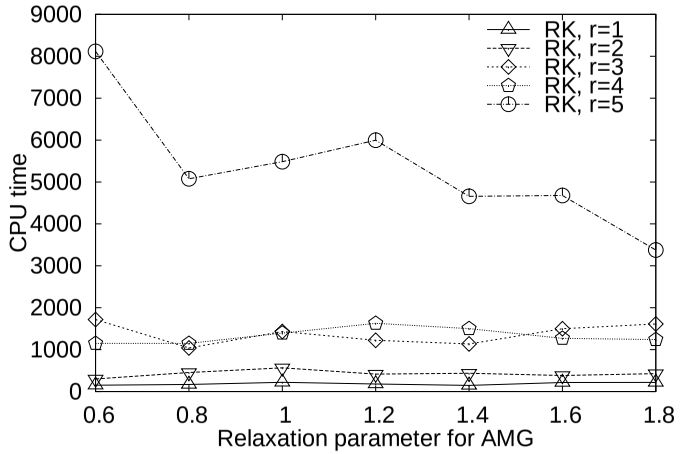


(a) Homogeneous domain. (b) Heterogeneous domain.

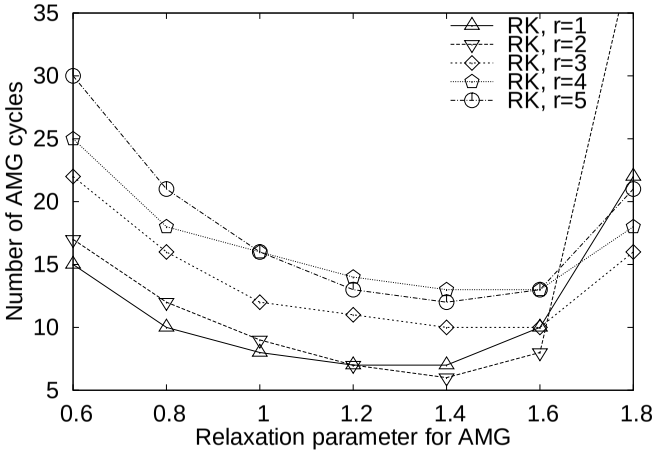
FIGURE 60 Number of AMG cycles with respect to the number of CG iterations in homogeneous and heterogeneous domain.



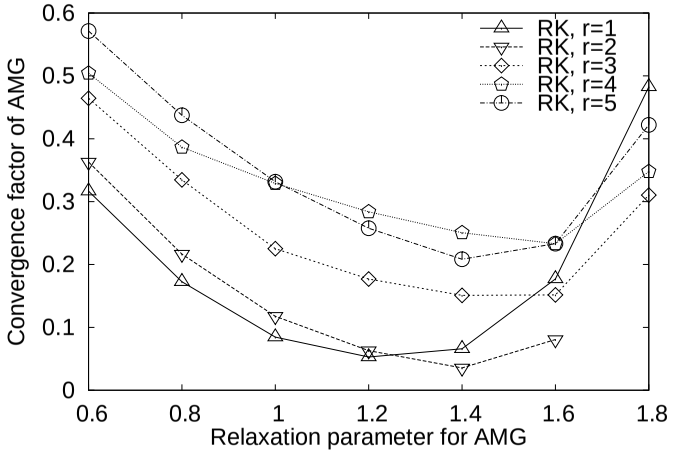
(a) The number of iterations with respect to the relaxation parameter for AMG.



(b) CPU time with respect to the relaxation parameter for AMG.

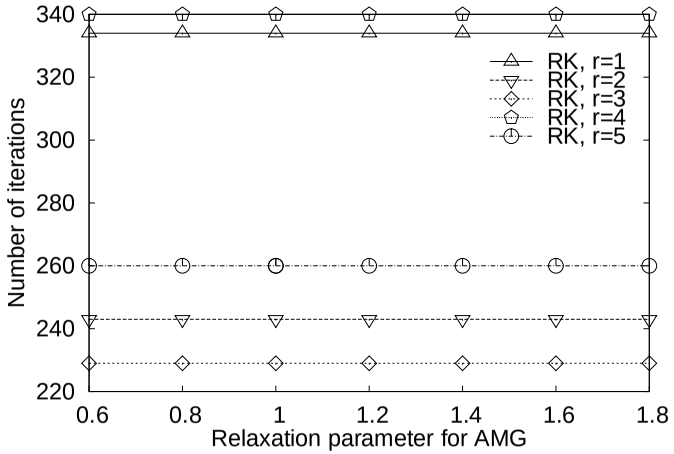


(c) Number of AMG cycles at the first CG iteration with respect to the relaxation parameter for AMG.

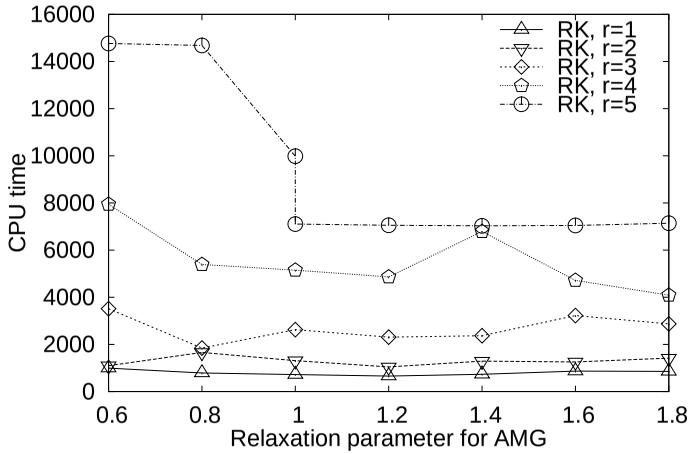


(d) Convergence factor at the first CG iteration with respect to the relaxation parameter for AMG.

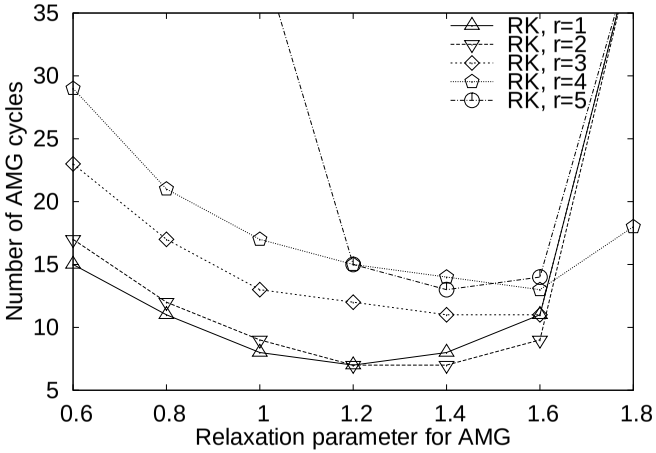
FIGURE 61 Wave propagation in the case of the RK time stepping with $\omega = 2\pi$ and $\varepsilon = 10^{-6}$ in homogeneous domain.



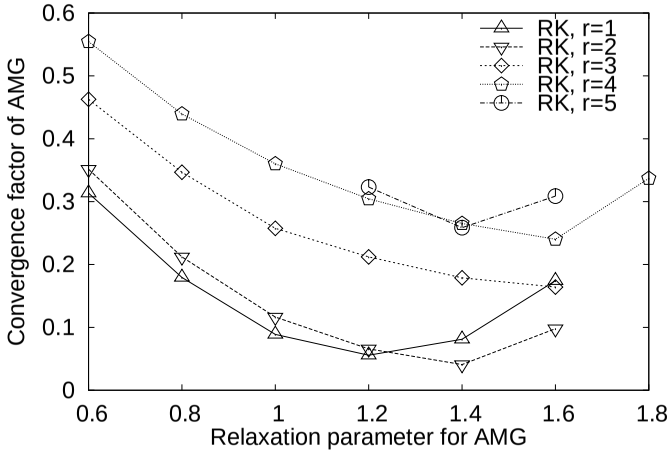
(a) The number of iterations with respect to the relaxation parameter for AMG.



(b) CPU time with respect to the relaxation parameter for AMG.



(c) Number of AMG cycles at the first CG iteration with respect to the relaxation parameter for AMG.



(d) Convergence factor at the first CG iteration with respect to the relaxation parameter for AMG.

FIGURE 62 Wave propagation in the case of the RK time stepping with $\omega = 2\pi$ and $\varepsilon = 10^{-6}$ in heterogeneous domain.

rate is equal for all element orders during the first iterations. As the residual becomes smaller, the convergence rate might become lower for some element orders. Still, the highest element order does not necessarily mean the poorest convergence rate. Insufficient preconditioning is the most feasible explanation for this observation. As the residual becomes smaller, the values in the right hand side vector $\mathbf{g}$, appearing in preconditioning (see Algorithm 1), become closer to zero. With small residuals, the stopping criterion of the AMG solver is fullfilled with small number of AMG cycles (or without any AMG cycles) as is seen in Figure 60.

It is seen that the best performance of the AMG solver is achieved, in both cases, with the value of relaxation parameter $1.2 - 1.6$ (see Figures 61(c)-61(d) and 62(c)-62(d)). The number of iterations is not dependent on the value of relaxation parameter (see Figures 61(a) and 62(a)), but too small value of relaxation parameter causes an increase in the number of AMG cycles and decrease in the convergence factor of AMG. This induces remarkably large CPU time requirements, which are seen in the left part of Figures 61(b) and 62(b). Since smaller wavelengths are involved in the center of the domain in the heterogeneous case than in the homogeneous case, the larger number of iterations and the larger amount of CPU time is needed for computations in the heterogeneous domain.

10 CONCLUSIONS

We considered the use of controllability techniques to solve the time-harmonic wave equations with spectral elements. The spectral element formulation used in this thesis results in a global mass matrix that is diagonal by construction. No inversion of a mass matrix is needed, which leads to a very efficient implementation. This is an advantage compared to classical finite element method.

Spatial discretization based on spectral elements is very accurate since it is based on high degree polynomials. To achieve the same accuracy, spectral element method requires fewer grid points per wavelength than finite element method. Consequently, accurate results are reached by solving smaller systems, i.e. fewer computational operations, which saves CPU time. More precise results concerning expenditure of CPU time seems to show linear dependence on the number of degrees of freedom. The number of preconditioned CG iterations is independent of the order of the spectral element basis, which confirms the efficiency of the AMG preconditioner, and makes the solver feasible for higher orders.

To make good use of higher order elements also the time discretization should be done with a higher order scheme. As a rule of thumb we can say that the efficiency of the overall method suffers from the error of time discretization if order of element is greater than the order of the time discretization method used. The second order CD time discretization method is efficient with finite elements, but when high accuracy is needed, it is best to use the RK time discretization.

YHTEENVETO (FINNISH SUMMARY)

Tämä lisensiaatintutkimus käsittelee tehokkaiden ratkaisumenetelmien kehittämistä aikaharmonisille aaltoyhtälöille. Tutkimus keskittyy akustisten ja elastisten aaltojen etenemistä kuvaaviin Helmholtzin ja Navierin yhtälöihin. Näiden akustisia painekenttiä ja elastisia materiaaleja kuvaavien osittaisdifferentiaaliyhtälöiden avulla voidaan mallintaa ja simuloida useita käytännön ilmiöitä, kuten vedenalaisten rakenteiden kestävyyttä, maanjäristysaaltojen etenemistä sekä konserttisalien tai auton sisätilojen akustisia ominaisuuksia. Monimutkaisissa käytännön sovelluksissa tarvitaan usein näiden kahden yhtälön muodostamaa kytkettyä systeemiä, joka on kuitenkin rajattu tämän tutkielman ulkopuolelle.

Perinteisesti aikaharmonisten aalto-ongelmien ratkaisemisessa on käytetty kompleksiarvoisia aikaharmonisia yhtälöitä, joiden diskretisointi on tehty äärellisten elementtien menetelmällä. Tämä johtaa suuriin indefiniitteihin yhtälöryhmiin, joille on hankala kehittää tehokkaita iteratiivisia ratkaisumenetelmiä. Erityisen haastavaksi aaltoyhtälöiden ratkaiseminen tulee aallon pituuden lyhentyessä, jolloin numeerisen ratkaisun ja todellisen ratkaisun välinen virhe kasvaa. Ratkaisun tarkkuuden takaaminen tässä tilanteessa edellyttää erityisen tiheää diskretisointia, mikä puolestaan vaatii runsaasti laskentaresursseja.

Edellä mainituista syistä johtuen tässä työssä käytetään säätöteoriaan perustuvaa tekniikkaa sekä korkeamman asteen elementtimenetelmää. Tehtävän ratkaisemisessa ei käytetä aikaharmonisia aaltoyhtälöitä, vaan aikaharmoninen tehtävä esitetään tarkan säädettävyyden tehtävänä ajasta riippuvan yhtälön avulla. Tämän jälkeen tehtävä muotoillaan pienimmän neliön optimointiongelmaksi, joka ratkaistaan liittogradienttimenetelmällä. Gradientin laskenta, joka on menetelmän keskeinen osa, suoritetaan liittotilatekniikalla. Liittogradienttialgoritmin pohjustuksessa käytetään lohkodiagonaalista pohjustinta, jonka yhteydessä muodostuvat jäykkyysmatriisin sisältävät yhtälöt ratkaistaan algebrallisella monihilamenetelmällä.

Paikkadiskretisoinnissa käytettävän spektraalielementtimenetelmän kantafunktiot ovat korkeamman asteen Lagrangen polynomeja, joiden nollakohdat sijoittuvat Gauss-Lobatto -pisteisiin. Numeerisessa integroinnissa sovelletaan Gauss-Lobatto -integrointisääntöä, joka tässä yhteydessä mahdollistaa diagonaalisten massamatriisien muodostamisen. Paikkadiskretisoinnin jälkeen saatava semidiskreetti yhtälöryhmä diskretisoidaan ajan suhteen joko keskeisdifferensseillä tai neljännen kertaluvun Runge-Kutta -menetelmällä. Käänteismatriisien muodostaminen diagonaalisille massamatriiseille on helppoa, mikä yhdessä edellä mainittujen eksplisiittisten aikadiskretisointimenetelmien kanssa mahdollistaa tehokkaan aikasimuloinnin. Koska tässä työssä aikaharmonisten yhtälöiden ratkaisemiseksi kehitetty menetelmä sisältää ajasta riippuvan aaltoyhtälön ratkaisemisen, edellä mainittuja diskretisointimenetelmiä voidaan käyttää myös ajasta riippuvien aaltojen laskennallisesti tehokkaaseen simulointiin.

Esitettyjen menetelmien tehokkuutta vertaillaan numeeristen esimerkkien avulla. Numeeriset tulokset osoittavat korkeamman asteen elementtimenetelmän käytöstä aiheutuvan tehokkuuden. Spektraalielementtimenetelmän yhteydessä sama tarkkuus saavutetaan vähemmällä määrällä laskentaoperaatioita kuin perinteisellä äärellisten elementtien menetelmällä. Parhaan tehokkuuden saavuttamiseksi korkeamman asteen elementtimenetelmän yhteydessä on syytä käyttää myös korkeamman asteen aikadiskretisointia.

REFERENCES

[1] M. Ainsworth. Discrete dispersion relation for hp-version finite element approximation at high wave number. *SIAM Journal on Numerical Analysis*, 42(2):553–575, 2004.

[2] T. Airaksinen, E. Heikkola, A. Pennanen, and J. Toivanen. An algebraic multigrid based shifted-Laplacian preconditioner for the Helmholtz equation. *Journal of Computational Physics*, 226:1196–1210, 2007.

[3] R. J. Astley and J.-P. Coyette. The performance of spheroidal infinite elements. *International Journal for Numerical Methods in Engineering*, 52(12):1379–1396, 2001.

[4] I. M. Babuska and S. A. Sauter. Is the pollution effect of the FEM avoidable for the Helmholtz equation considering high wave numbers? *SIAM Journal on Numerical Analysis*, 34(6):2392–2423, 1997.

[5] G. Bao, G. W. Wei, and S. Zhao. Numerical solution of the Helmholtz equation with high wavenumbers. *International Journal for Numerical Methods in Engineering*, 59(3):389–408, 2004.

[6] E. Bécache, P. Joly, and C. Tsogka. Application of the fictitious domain method to 2D linear elastodynamic problems. *Journal of Computational Acoustics*, 9(3):1175–1202, 2001.

[7] J.-D. Benamou and B. Desprès. A domain decomposition method for the Helmholtz equation and related optimal control problems. *Journal of Computational Physics*, 136(1):68–82, 1997.

[8] L. S. Bennethum and X. Feng. A domain decomposition method for solving a Helmholtz-like problem in elasticity based on the Wilson nonconforming element. *RAIRO Modélisation Mathématique et Analyse Numérique*, 31:1–25, 1997.

[9] P. Bettess. *Infinite Elements*. Penshaw Press, Sunderland, U.K., 1992.

[10] D. S. Bindel and S. Govindjee. Elastic PMLs for resonator anchor loss simulation. *International Journal for Numerical Methods in Engineering*, 64(6):789–818, 2005.

[11] L. Bos, M. A. Taylor, and B.A. Wingate. Tensor product Gauss-Lobatto points are Fekete points for the cube. *Mathematics of Computation*, 70:1543–1547, 2001.

[12] A. Brandt and I. Livshits. Wave-Ray multigrid method for standing wave equations. *Electronic Transactions on Numerical Analysis*, 6:162–181, 1997.

[13] M. O. Bristeau, R. Glowinski, and J. Périaux. Using exact controllability to solve the Helmholtz equation at high wave numbers. In R. Kleinman, T. Angell, D. Colton, F. Santosa, and I. Stakgold, editors, *Mathematical and Numerical*

Aspects of Wave Propagation, pages 113–127, Philadelphia, Pennsylvania, 1993. SIAM.

[14] M. O. Bristeau, R. Glowinski, and J. Périaux. Exact controllability to solve the Helmholtz equation with absorbing boundary conditions. In M. Křížek, P. Neittaanmäki, and R. Stenberg, editors, *Finite Element Methods: Fifty Years of the Courant Elements*, pages 75–93, New York, N.Y., 1994. Marcel Dekker.

[15] M. O. Bristeau, R. Glowinski, and J. Périaux. On the numerical solution of the Helmholtz equation at large wave numbers using exact controllability methods. Application to scattering. *Contemporary Mathematics*, 157:399–419, 1994.

[16] M. O. Bristeau, R. Glowinski, and J. Périaux. Controllability methods for the computation of time-periodic solutions; application to scattering. *Journal of Computational Physics*, 147(2):265–292, 1998.

[17] X.-C. Cai, M. Casarin, F. Elliott, and O. Widlund. Overlapping Schwarz algorithms for solving Helmholtz's equation. *Contemporary Mathematics*, 218:391–399, 1998.

[18] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang. *Spectral Methods in Fluid Dynamics*. Springer-Verlag, Berlin, 1988.

[19] O. Cessenat and B. Després. Using plane waves as base functions for solving time harmonic equations with the ultra weak variational formulation. *Journal of Computational Acoustics*, 11(2):227–238, 2003.

[20] A. Cividini, A. Quarteroni, and E. Zampieri. Numerical solution of linear elastic problems by spectral collocation methods. *Computer Methods in Applied Mechanics and Engineering*, 104(1):49–76, 1993.

[21] B. Cockburn, G. E. Karniadakis, and C.-W. Shu, editors. *Discontinuous Galerkin Methods: Theory, Computation and Applications*. Springer-Verlag, Berlin, 2000.

[22] G. Cohen. *Higher-Order Numerical Methods for Transient Wave Equations*. Springer-Verlag, Berlin, 2001.

[23] S. Collino, F. Ghanemi and P. Joly. Domain decomposition method for harmonic wave propagation: a general presentation. *Computer Methods in Applied Mechanics and Engineering*, 184(2-4):171–211, 2000.

[24] D. Colton and R. Kress. *Inverse Acoustic and Electromagnetic Scattering Theory*. Springer-Verlag, Berlin, 1992.

[25] D. M. Eidus. The principle of limiting absorption. *Matematicheskii Sbornik*, 57(99):13–44, 1962. English translation in *American Mathematical Society Translations*, Series 2, 47:157–191, 1965.

[26] A. El Badia and T. Ha-Duong. Determination of point wave sources by boundary measurements. *Inverse Problems*, 17:1127–1139, 2001.

[27] H. C. Elman, O. G. Ernst, and D. P. O'Leary. A multigrid method enhanced by Krylov subspace iteration for discrete Helmholtz equations. *SIAM Journal on Scientific Computing*, 23(4):1291–1315, 2001.

[28] B. Engquist and A. Majda. Radiation boundary conditions for acoustic and elastic wave calculations. *Communications on Pure and Applied Mathematics*, 32:313–357, 1979.

[29] C. Fabre and J.-P. Puel. Pointwise controllability as limit of internal controllability for the wave equation in one space dimension. *Portugaliae Mathematica*, 51(3):335–350, 1994.

[30] E. Faccioli, F. Maggio, A. Quarteroni, and A. Tagliani. Spectral domain decomposition methods for the solution of acoustic and elastic wave propagation. *Geophysics*, 61:1160–1174, 1996.

[31] C. Farhat, I. Harari, and U. Hetmaniuk. A discontinuous Galerkin method with Lagrange multipliers for the solution of Helmholtz problems in the mid-frequency regime. *Computer Methods in Applied Mechanics and Engineering*, 192(11-12):1389–1419, 2003.

[32] C. Farhat and U. Hetmaniuk. A fictitious domain decomposition method for the solution of partially axisymmetric acoustic scattering problems. I. Dirichlet boundary conditions. *International Journal for Numerical Methods in Engineering*, 54(9):1309–1332, 2002.

[33] C. Farhat and J. Li. An iterative domain decomposition method for the solution of a class of indefinite problems in computational structural dynamics. *Applied Numerical Mathematics*, 54:150–166, 2005.

[34] C. Farhat, A. Macedo, and M. Lesoinne. A two-level domain decomposition method for the iterative solution of high frequency exterior Helmholtz problems. *Numerische Mathematik*, 85(2):283–308, 2000.

[35] C. Farhat, R. Tezaur, and P. Wiedemann-Goiran. Higher-order extensions of a discontinuous Galerkin method for mid-frequency Helmholtz problems. *International Journal for Numerical Methods in Engineering*, 61(11):1938–1956, 2004.

[36] C. Farhat, P. Wiedemann-Goiran, and R. Tezaur. A discontinuous Galerkin method with plane waves and Lagrange multipliers for the solution of short wave exterior Helmholtz problems on unstructured meshes. *Wave Motion*, 39(4):307–317, 2004.

[37] G. K. Gächter and M. J. Grote. Dirichlet-to-Neumann map for three-dimensional elastic waves. *Wave Motion*, 37(3):293–311, 2003.

[38] P. L. George. *Automatic Mesh Generation, Application to Finite Element Methods*. John Wiley & Sons, New York, 1991.

[39] K. Gerdes. A review of infinite element methods for exterior Helmholtz problems. *Journal of Computational Acoustics*, 8(1):43–62, 2000.

[40] R. Glowinski. Ensuring well-posedness by analogy; Stokes problem and boundary control for the wave equation. *Journal of Computational Physics*, 103(2):189–221, 1992.

[41] R. Glowinski and J. L. Lions. Exact and approximate controllability for distributed parameter systems (I). *Acta Numerica*, pages 269–378, 1994.

[42] R. Glowinski and J. L. Lions. Exact and approximate controllability for distributed parameter systems (II). *Acta Numerica*, pages 159–333, 1995.

[43] R. Glowinski, J. Périaux, and J. Toivanen. Time-periodic solutions of wave equation via controllability and fictitious domain methods. In G. Cohen, E. Heikkola, P. Joly, and P. Neittaanmäki, editors, *Mathematical and Numerical Aspects of Wave Propagation, Proceedings of WAVES 2003*, pages 805–810, Jyväskylä, Finland, 2003. Springer-Verlag.

[44] A. E. Green and W. Zerna. *Theoretical Elasticity*. Oxford University Press, London, 1960.

[45] W. Hackbusch. *Multigrid Methods and Applications*. Springer-Verlag, Berlin, Germany, 1985.

[46] I. Harari and U. Albocher. Studies of FE/PML for exterior problems of time-harmonic elastic waves. *Computer Methods in Applied Mechanics and Engineering*, 195(29-32):3854–3879, 2006.

[47] I. Harari and S. Haham. Improved finite element methods for elastic waves. *Computer Methods in Applied Mechanics and Engineering*, 166(1-2):143–164, 1998.

[48] I. Harari and E. Turkel. Accurate finite difference methods for time-harmonic wave propagation. *Journal of Computational Physics*, 119:252–270, 1995.

[49] T. Havarneanu, C. Popa, and S.S. Sritharan. Exact controllability for the three dimensional Navier Stokes equations with the Navier slip boundary conditions. *Indiana University Mathematics Journal*, 54(5):1303–1350, 2005.

[50] E. Heikkola, S. Mönkölä, A. Pennanen, and T. Rossi. Controllability method for acoustic scattering with spectral elements. *Journal of Computational and Applied Mathematics*, 204(2):344–355, 2007.

[51] E. Heikkola, S. Mönkölä, A Pennanen, and T. Rossi. Controllability method for the Helmholtz equation with higher-order discretizations. *Journal of Computational Physics*, 225(2):1553–1576, 2007.

[52] E. Heikkola, T. Rossi, and J. Toivanen. A parallel fictitious domain method for the three-dimensional Helmholtz equation. *SIAM Journal on Scientific Computing*, 24(5):1567–1588, 2003.

[53] U. Hetmaniuk and C. Farhat. A fictitious domain decomposition method for the solution of partially axisymmetric acoustic scattering problems. II. Neumann boundary conditions. *International Journal for Numerical Methods in Engineering*, 58(1):63–81, 2003.

[54] J.J. Heys, T.A. Manteuffel, S.F. McCormick, and L.N. Olson. Algebraic multigrid for higher-order finite elements. *Journal of Computational Physics*, 204(2):520–532, 2005.

[55] T. Huttunen, J. P. Kaipio, and P. Monk. The perfectly matched layer for the ultra weak variational formulation of the 3D Helmholtz equation. *International Journal for Numerical Methods in Engineering*, 61(7):1072–1092, 2004.

[56] T. Huttunen, P. Monk, F. Collino, and J. Kaipio. The ultra-weak variational formulation for elastic wave problems. *SIAM Journal on Scientific Computing*, 25(5):1717–1742, 2004.

[57] F. Ihlenburg. *Finite Element Analysis of Acoustic Scattering*. Springer-Verlag, Berlin, 1998.

[58] F. Ihlenburg and I. Babuska. Finite element solution of the Helmholtz equation with high wave number part II: The h-p-version of the FEM. *SIAM Journal on Numerical Analysis*, 34(1):315–358, 1997.

[59] C. Johnson. *Numerical Solution of Partial Differential Equations by the Finite Element Method*. Cambridge University Press, Cambridge, 1987.

[60] M. Käser and M. Dumbser. An arbitrary high order discontinuous Galerkin method for elastic waves on unstructured meshes I: The two-dimensional isotropic case with external source terms. *Geophysical Journal International*, 166(2):855–877, 2006.

[61] F. Kickinger. Algebraic multigrid for discrete elliptic second-order problems. In *Multigrid Methods V (Stuttgart, 1996)*, pages 157–172. Springer-Verlag, Berlin, 1998.

[62] S. Kim and S. Kim. Multigrid simulation for high-frequency solutions of the Helmholtz problem in heterogeneous media. *SIAM Journal on Scientific Computing*, 24(2):684–701, 2002.

[63] D. Komatitsch, R. Martin, J. Tromp, M. A. Taylor, and B. A. Wingate. Wave propagation in 2-D elastic media using a spectral element method with triangles and quadrangles. *Journal of Computational Acoustics*, 9(2):703–718, 2001.

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

[65] D. Komatitsch, J. P. Vilotte, R. Vai, J. M. Castillo-Covarrubias, and F. J. Sánchez-Sesma. The Spectral Element method for elastic wave equations: application to 2D and 3D seismic problems. *International Journal for Numerical Methods in Engineering*, 45:1139–1164, 1999.

[66] M. Křížek and P. Neittaanmäki. *Finite Element Approximation of Variational Problems and Applications*. Longman Scientific & Technical, Harlow, UK, 1990.

[67] W. M. Lai, D. Rubin, and E. Krempl. *Introduction to Continuum Mechanics*. Number 17 in Pergamon Unified Engineering Series. Pergamon Press, New York, 1974.

[68] L. D. Landau and E. M. Lifshitz. *Theory of Elasticity*. Pergamon Press, Oxford, 1975.

[69] R. Leis. *Initial Boundary Value Problems in Mathematical Physics*. John Wiley, Chichester, 1986.

[70] K.H. Lim, J.H. Lee, G. Ye, and Q.H Liu. An efficient forward solver in electrical impedance tomography by spectral element method. *IEEE Transactions on Medical Imaging*, 25(8):1044–1051, 2006.

[71] J. L. Lions. Exact controllability, stabilization and perturbations for distributed systems. *SIAM Review*, 30(1):1–68, March 1988.

[72] J. L. Lions. *Applied and industrial mathematics*, chapter Exact controllability for distributed systems. Some trends and some problems, pages 59–84. Mathematics and Its Applications. Kluwer Academic Publishers, Netherlands, 1991.

[73] J. L. Lions. On the controllability of distributed systems. *Proceedings of the National Academy of Sciences of The United States of America*, 94(10):4828–4835, 1997.

[74] W. Liu. Exact distributed controllability for the semilinear wave equation. *Portugaliae Mathematica*, 57(4):493–508, 2000.

[75] W. Liu and G. H. Williams. Exact internal controllability for the semilinear heat equation. *Journal of Mathematical Analysis and Applications*, 211:258–272, 1997.

[76] Y. Maday and A. T. Patera. Spectral element methods for the incompressible Navier-Stokes equations. In A. K. Noor and J. T. Oden, editors, *State-of-the-Art Surveys on Computational Mechanics*, pages 71–143, New York, 1989. American Society of Mechanical Engineering.

[77] F. Magoulès, F.-X. Roux, and S. Salmon. Optimal discrete transmission conditions for a nonoverlapping domain decomposition method for the Helmholtz equation. *SIAM Journal on Scientific Computing*, 25(5):1497–1515, 2004.

[78] J. Martikainen, A. Pennanen, and T. Rossi. Application of an algebraic multigrid method to incompressible flow problems. Reports of the Department of Mathematical Information Technology, Series B. Scientific Computing, B 2/2006, Department of Mathematical Information Technology, University of Jyväskylä, 2006.

[79] Y. McCormick and J. Ruge. Algebraic multigrid methods applied to problems in computational structural mechanics. In A. K. Noor and J. T. Oden, editors, *State-of-the-Art Surveys on Computational Mechanics*, pages 237–270, New York, 1989. American Society of Mechanical Engineering.

[80] K. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, Boston, 1999.

[81] S. Mönkölä, E. Heikkola, A Pennanen, and T. Rossi. Time-harmonic elasticity with controllability and higher order discretization methods. *Journal of Computational Physics*, 2008, to appear.

[82] G. Mur. The finite-element modeling of three-dimensional electromagnetic fields using edge and nodal elements. *IEEE Transactions on Antennas and Propagation*, 41(7):948–953, 1993.

[83] A. K. Nandakumaran. Convergence of the boundary control for the wave equation in domains with holes of critical size. *Electronic Journal of Differential Equations*, 35:1–10, 2002.

[84] R. Pasquetti and F. Rapetti. Spectral element methods on triangles and quadrilaterals: comparisons and applications. *Journal of Computational Physics*, 198(1):349–362, 2004.

[85] A. T. Patera. Spectral element method for fluid dynamics: Laminar flow in a channel epansion. *Journal of Computational Physics*, 15:468–488, 1984.

[86] L. F. Pavarino and E. Zampieri. Preconditioners for spectral discretizations of Helmholtz's equation with Sommerfeld boundary conditions. *Computer Methods in Applied Mechanics and Engineering*, 190(40-41):5341–5356, 2001.

[87] A. Quarteroni, T. Tagliani, and E. Zampieri. Generalized Galerkin approximations of elastic waves with absorbing boundary conditions. *Computer Methods in Applied Mechanics and Engineering*, 163:323–341, 1998.

[88] A. G. Ramm. *Scattering by Obstacles*. Kluwer Academic Publishers, 1986.

[89] B. Rao. Exact boundary controllability elasticity of a hybrid system by the HUM method. http://citeseer.nj.nec.com/rao00exact.html.

[90] S. Reitzinger. *Algebraic Multigrid Methods for Large Scale Finite Element Equations*. PhD thesis, Institute of Computational Mathematics, Johannes Kepler University Linz, Linz, 2001.

[91] J. W. Ruge and K. Stüben. Algebraic multigrid (AMG). In Stephen F. McCormick, editor, *Multigrid Methods*, Frontiers in Applied Mathematics, pages 73–130. SIAM, Philadelphia, Pennsylvania, 1987.

[92] Y. Saad. *Iterative Methods for Sparse Linear Systems, 2nd edition*. SIAM, Philadelphia, 2003.

[93] P. Šolín, K. Segeth, and I. Doležel. *Higher-Order Finite Element Methods*. Chapman & Hall/ CRC Press, Boca Raton, 2004.

[94] K. Stüben. A review of algebraic multigrid. *Journal of Computational and Applied Mathematics,*, 128(1-2):281–309, 2001.

[95] M. A. Taylor and B. A. Wingate. A generalized diagonal mass matrix spectral element method for non-quadrilateral elements. *Applied Numerical Mathematics*, 33:259–265, 2000.

[96] M. A. Taylor, B. A. Wingate, and L. P. Bos. A cardinal function algorithm for computing multivariate quadrature points. *SIAM Journal on Numerical Analysis*, 45(1):193–205, 2007.

[97] L. L. Thompson. A review of finite-element methods for time-harmonic acoustics. *Journal of the Acoustical Society of America*, 119(3):1315–1330, 2006.

[98] U. Trottenberg, C. W. Oosterlee, and A. Schüller. *Multigrid*. Academic Press, London, 2001.

[99] C. P. A. Wapenaar and A. J. Berkhout. *Elastic Wave Field Extrapolation: Redatuming of Single and Multi-component Seismic Data*. Elsevier, Amsterdam, 1989.

[100] N. Weck and K. J. Witsch. Complete low frequency analysis for the reduced wave equation with variable coefficients in three dimensions. *Communications in Partial Differential Equations*, 17:1619–1663, 1992.

[101] N. Weck and K. J. Witsch. Generalized linear elasticity in exterior domains–i: Radiation problems. *Mathematical Methods in the Applied Sciences*, 20:1469–1500, 1997.

[102] F. M. White. *Fluid Mechanics*. McGraw-Hill, New York, 1994.

[103] C. H. Wilcox. *Scattering Theory for the d'Alembert Equation in Exterior Domains*, volume 442 of *Lecture Notes in Mathematics*. Springer–Verlag, Berlin Heidelberg, 1975.

[104] J. Y. Yuan and A. N. Iusem. Preconditioned conjugate gradient method for generalized least squares problems. *Journal of Computational and Applied Mathematics*, 71:287–297, 1996.

[105] L. Zhang, R. Tezaur, and C. Farhat. The discontinuous enrichment method for elastic wave propagation in the medium-frequency regime. *International Journal for Numerical Methods in Engineering*, 66(13):2086–2114, 2006.

[106] O. C. Zienkiewicz, R. L. Taylor, and J. Z. Zhu. *The finite element method: Its Basis & Fundamentals*. Elsevier, Oxford, 2005.

[107] E. Zuazua. Propagation, observation, and control of waves approximated by finite difference methods. *SIAM Review*, 47(2):197–243, 2005.