

**This is an electronic reprint of the original article.
This reprint *may differ* from the original in pagination and typographic detail.**

Author(s): Mönkölä, Sanna

Title: An optimization-based approach for solving a time-harmonic multiphysical wave problem with higher-order schemes

Year: 2013

Version:

Please cite the original version:

Mönkölä, S. (2013). An optimization-based approach for solving a time-harmonic multiphysical wave problem with higher-order schemes. *Journal of Computational Physics*, 242, 439-459. <https://doi.org/10.1016/j.jcp.2013.02.022>

All material supplied via JYX is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.

An optimization-based approach for solving a time-harmonic multiphysical wave problem with higher-order schemes

Sanna Mönkölä *

*Department of Mathematical Information Technology, University of Jyväskylä,
P.O. Box 35 (Agora), FI-40014 University of Jyväskylä, Finland*

Abstract

This study considers developing numerical solution techniques for the computer simulations of time-harmonic fluid-structure interaction between acoustic and elastic waves. The focus is on the efficiency of an iterative solution method based on a controllability approach and spectral elements. We concentrate on the model, in which the acoustic waves in the fluid domain are modeled by using the velocity potential and the elastic waves in the structure domain are modeled by using displacement.

Traditionally, the complex-valued time-harmonic equations are used for solving the time-harmonic problems. Instead of that, we focus on finding periodic solutions without solving the time-harmonic problems directly. The time-dependent equations can be simulated with respect to time until a time-harmonic solution is reached, but the approach suffers from poor convergence. To overcome this challenge, we follow the approach first suggested and developed for the acoustic wave equations by Bristeau, Glowinski, and Périaux. Thus, we accelerate the convergence rate by employing a controllability method. The problem is formulated as a least-squares optimization problem, which is solved with the conjugate gradient (CG) algorithm. Computation of the gradient of the functional is done directly for the discretized problem. A graph-based multigrid method is used for preconditioning the CG algorithm.

Key words: Controllability, Spectral element method, Coupled problem, Fluid-structure interaction, Fourth-order Runge–Kutta

PACS: 43.40

* Corresponding author.

Email address: `sanna.monkola@jyu.fi` (Sanna Mönkölä).

1 Introduction

The basic idea of the controllability approach is to steer a dynamical (i.e., time-dependent) system from an initial state to a particular state by using an appropriate control (see, e.g., [1,2]). The controlling approach is used in several areas such as shape design, inverse problems and controllability of models described by the PDEs. In this article, we use optimal control theory as a tool to find a periodic solution for a coupled wave equation. To be more precise, we consider the model, in which the acoustic waves in the fluid domain are modeled by using the velocity potential and the elastic waves in the structure domain are modeled by using displacement. We follow the idea of Bristeau, Glowinski, and Périaux, presented in [3,4,5,6,7], and avoid solving indefinite systems by returning to time-dependent equations. The difference between the initial condition and the terminal condition of the time-dependent system is minimized by an optimization algorithm.

Recently, the controllability approach is applied with finite element -based space discretizations for simulating time-dependent acoustic, elastic, and electromagnetic problems until the time-harmonic solution is reached in, e.g., [8,9,10,11]. The idea of using controllability approach was extended to solve a coupled acousto-elastic problems and first published in [12]. Coupling between the pressure and the displacement was discussed. Because of the choice of the least-squares functional, the method suffered from poor convergence rate. In this paper, we will present improvements for the mathematical formulation and computational algorithms.

The drawback of using the traditional formulation of the wave equations is that the energy norm is then of H^1 -type, and as such, the control problem needs preconditioning implying an additional computational cost. The preconditioning also makes it more complicated to implement the solver to utilize the parallel computing. In [13], a mixed finite element method is presented. Within that approach, the elliptic problem is not needed to be solved at the preconditioning stage and thus saving the computing time. For solving the generalized Maxwell equations, combining the controllability method with discrete exterior calculus provides even more promising alternative [14].

We consider minimizing the quadratic functional

$$J(\mathbf{e}) = \frac{1}{2} \mathbf{e}^T \mathbf{A} \mathbf{e} - \mathbf{b}^T \mathbf{e} + \mathbf{c}, \quad (1)$$

where \mathbf{A} is a symmetric and positive definite matrix of size $\hat{N} \times \hat{N}$ and the vectors \mathbf{e} , \mathbf{b} , and \mathbf{c} are of size \hat{N} . As we focus on large problems, the set of fea-

sible optimization algorithms is restricted to the methods with small memory requirements. That is why we neglect Newton-type methods and concentrate on the conjugate gradient (CG) method introduced by Hestenes and Stiefel in 1952 [15].

This article is organized as follows. First, the coupled model for acoustic and elastic waves is constructed in Section 2. Then, we discretize the coupled problem in a space domain with spectral elements in Section 3. For time discretization we use the fourth-order Runge–Kutta scheme in Section 4. In Section 5, we expand the control approach to the coupled problem. The objective functional is presented in Section 5.1. Further, we compute the gradient of the functional, an essential point of the method, using the adjoint state technique in Section 5.2. The optimization algorithm is considered in Section 6. To guarantee the smooth initial approximation for the CG algorithm we use a transition procedure, suggested by Mur [16]. The main principles of the CG method are presented in Section 6.1. For speeding up the convergence rate of the CG algorithm, we use the graph-based multigrid method, introduced in [17], in Section 6.2. Numerical experiments concerning the multiphysical propagation of time-harmonic waves show the efficiency of the algorithm in Section 7. The concluding remarks are presented in Section 8.

2 Coupled elastic-acoustic wave equations

Various formulations exist for the interaction between acoustic and elastic waves. Naturally, the models have been tailored corresponding to the applications. For instance, gravity effects are employed by Andrianarison and Ohayon in [18] and by Komatitsch and Tromp in [19].

Typically, the displacement is solved in the elastic structure. The fluid can be modeled using finite element formulations based on fluid pressure, displacement, velocity potential or displacement potential [20]. Two approaches, in which the displacement is solved in the elastic structure, predominate in modeling the interaction between acoustic and elastic waves. Expressing the acoustic wave equation by the pressure in the fluid domain leads to a non-symmetric formulation (see, e.g., [21,22,23,24]), while using the velocity potential results in a symmetric system of equations (see, e.g., [25,26,27,28]). We applied the exact controllability to the non-symmetric formulation in [12], whereas in this paper we consider the symmetric formulation.

The domain $\Omega \subset \mathbb{R}^2$ modeling the geometry of the phenomena is divided into a fluid part Ω_f and a structure part Ω_s (see Figure 1). The fluid domain is bounded by the boundary $\Gamma_f = \Gamma_{of} \cup \Gamma_{ef} \cup \Gamma_i$, and $\Gamma_s = \Gamma_{os} \cup \Gamma_{es} \cup \Gamma_i$ constitutes the boundary for the structure domain Ω_s . On the boundaries Γ_{of} and

Γ_{0s} we use the Dirichlet boundary conditions, whereas on the artificial boundaries Γ_{ef} and Γ_{es} we impose the conventional first order absorbing boundary conditions [29,30]. On the interface Γ_i between fluid and structure domains, we assume the continuity of normal components of displacements and traction forces. The outward normal vectors to domains Ω_f and Ω_s are presented as $\mathbf{n}_f = (n_{f1}, n_{f2})^T$ and $\mathbf{n}_s = (n_{s1}, n_{s2})^T$. From the controllability point of view, it is a convenient choice to model the acoustic waves in the fluid domain Ω_f by using the velocity potential and the elastic waves in the structure domain Ω_s by using the displacement variable. Thus, we present a symmetric formulation (see, e.g. [25,26,28])

$$\frac{1}{c(\mathbf{x})^2} \frac{\partial^2 \phi}{\partial t^2} - \nabla^2 \phi = f_\phi, \quad \text{in } Q_f = \Omega_f \times [0, T], \quad (2)$$

$$\phi = 0, \quad \text{on } \gamma_{0f} = \Gamma_{0f} \times [0, T], \quad (3)$$

$$\frac{1}{c(\mathbf{x})} \frac{\partial \phi}{\partial t} + \frac{\partial \phi}{\partial \mathbf{n}_f} = y_{\phi \text{ext}}, \quad \text{on } \gamma_{ef} = \Gamma_{ef} \times [0, T], \quad (4)$$

$$\frac{\partial \mathbf{u}_s}{\partial t} \cdot \mathbf{n}_s + \frac{\partial \phi}{\partial \mathbf{n}_f} = 0, \quad \text{on } \gamma_i = \Gamma_i \times [0, T], \quad (5)$$

$$\rho_s(\mathbf{x}) \frac{\partial^2 \mathbf{u}_s}{\partial t^2} - \nabla \cdot \sigma(\mathbf{u}_s) = \mathbf{f}, \quad \text{in } Q_s = \Omega_s \times [0, T], \quad (6)$$

$$\mathbf{u}_s = \mathbf{0}, \quad \text{on } \gamma_{0s} = \Gamma_{0s} \times [0, T], \quad (7)$$

$$\rho_s(\mathbf{x}) \mathbf{B} \frac{\partial \mathbf{u}_s}{\partial t} + \sigma(\mathbf{u}_s) \mathbf{n}_s = \mathbf{g}_{\text{ext}}, \quad \text{on } \gamma_{es} = \Gamma_{es} \times [0, T], \quad (8)$$

$$\sigma(\mathbf{u}_s) \mathbf{n}_s + \rho_f(\mathbf{x}) \frac{\partial \phi}{\partial t} \mathbf{n}_f = \mathbf{0}, \quad \text{on } \gamma_i = \Gamma_i \times [0, T]. \quad (9)$$

where f_ϕ , $y_{\phi \text{ext}}$, \mathbf{f} , and \mathbf{g}_{ext} are the source terms. Length of the time interval is marked as T , ϕ denotes the velocity potential, and $\mathbf{u}_s = (\mathbf{u}_{s1}, \mathbf{u}_{s2})^T$ is the displacement field. Coefficients $\rho_f(\mathbf{x})$ and $\rho_s(\mathbf{x})$ represent the densities of media in domains Ω_f and Ω_s , respectively, and $c(\mathbf{x})$ is the speed of sound in fluid domain. The stress tensor is expressed as $\sigma(\mathbf{u}_s) = \lambda(\nabla \cdot \mathbf{u}_s) \mathcal{I} + 2\mu \epsilon(\mathbf{u}_s)$

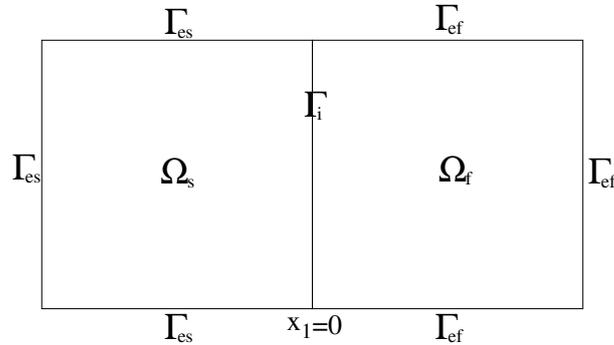


Figure 1. The domain Ω is divided into the structure part Ω_s and the fluid part Ω_f .

with the linearized strain tensor $\epsilon = \frac{1}{2}(\nabla \mathbf{u}_s + (\nabla \mathbf{u}_s)^T)$, the identity matrix \mathcal{I} , and the Lamé parameters μ and λ . The symmetric positive definite matrix \mathbf{B} is defined as [29,30]

$$\mathbf{B} = \begin{pmatrix} c_p(\mathbf{x})n_{s1}^2 + c_s(\mathbf{x})n_{s2}^2 & n_{s1}n_{s2}(c_p(\mathbf{x}) - c_s(\mathbf{x})) \\ n_{s1}n_{s2}(c_p(\mathbf{x}) - c_s(\mathbf{x})) & c_p(\mathbf{x})n_{s2}^2 + c_s(\mathbf{x})n_{s1}^2 \end{pmatrix},$$

with elements $c_p(\mathbf{x})$ and $c_s(\mathbf{x})$ representing the speed of the pressure waves and the speed of the shear waves, respectively.

Furthermore, we complete the time-dependent system (2)-(9) by the initial conditions $\mathbf{e} = (\mathbf{e}_0, \mathbf{e}_1)^T$ such that $\mathbf{e}_0 = (\mathbf{e}_{\phi f0}, \mathbf{e}_{s0})^T$ and $\mathbf{e}_1 = (\mathbf{e}_{\phi f1}, \mathbf{e}_{s1})^T$, and

$$\phi(\mathbf{x}, 0) = \mathbf{e}_{\phi f0}, \quad \text{in } \Omega_f, \quad (10)$$

$$\frac{\partial \phi}{\partial t}(\mathbf{x}, 0) = \mathbf{e}_{\phi f1}, \quad \text{in } \Omega_f, \quad (11)$$

$$\mathbf{u}_s(\mathbf{x}, 0) = \mathbf{e}_{s0}, \quad \text{in } \Omega_s, \quad (12)$$

$$\frac{\partial \mathbf{u}_s(\mathbf{x}, 0)}{\partial t} = \mathbf{e}_{s1}, \quad \text{in } \Omega_s. \quad (13)$$

For the weak formulation of the system (2)-(9) we introduce the function spaces V and \mathbf{V} by

$$V = \{v \in H^1(\Omega_f) \text{ such that } v = 0 \text{ on } \Gamma_{0f}\}, \quad (14)$$

$$\mathbf{V} = \{\mathbf{v} \in H^1(\Omega_s) \times H^1(\Omega_s) \text{ such that } \mathbf{v} = \mathbf{0} \text{ on } \Gamma_{0s}\}. \quad (15)$$

By multiplying Equation (2) with any test function v in the space V , and (6) with any test function \mathbf{v} in the space \mathbf{V} , using Green's formula, and substituting the boundary conditions results in the following weak formulation: Find (ϕ, \mathbf{u}_s) satisfying $(\phi(t), \mathbf{u}_s(t)) \in (V \times \mathbf{V})$ for any $t \in [0, T]$ and

$$a_{\phi f}(\phi, v) + \int_{\Gamma_i} \frac{\partial \mathbf{u}_s}{\partial t} \cdot \mathbf{n}_s v \, ds = f_{\phi f}(v), \quad (16)$$

$$a_s(\mathbf{u}_s, \mathbf{v}) + \int_{\Gamma_i} \rho_f(\mathbf{x}) \frac{\partial \phi}{\partial t} \mathbf{n}_f \cdot \mathbf{v} \, ds = \mathbf{f}_s(\mathbf{v}), \quad (17)$$

for any $(v, \mathbf{v}) \in (V \times \mathbf{V})$ and $t \in [0, T]$, where

$$a_{\phi f}(\phi, v) := \int_{\Omega_f} \frac{1}{c(\mathbf{x})^2} \frac{\partial^2 \phi}{\partial t^2} v \, dx + \int_{\Omega_f} \nabla \phi \cdot \nabla v \, dx + \int_{\Gamma_{ef}} \frac{1}{c(\mathbf{x})} \frac{\partial \phi}{\partial t} v \, ds, \quad (18)$$

$$f_{\phi f}(v) := \int_{\Omega_f} f_{\phi} v \, dx + \int_{\Gamma_{ef}} y_{\phi \text{ext}} v \, ds, \quad (19)$$

$$a_s(\mathbf{u}_s, \mathbf{v}) := \int_{\Omega_s} \rho_s(\mathbf{x}) \frac{\partial^2 \mathbf{u}_s}{\partial t^2} \cdot \mathbf{v} \, dx + \int_{\Omega_s} \sigma(\mathbf{u}_s) : \epsilon(\mathbf{v}) \, dx + \int_{\Gamma_{es}} \rho_s(\mathbf{x}) \mathbf{B} \frac{\partial \mathbf{u}_s}{\partial t} \cdot \mathbf{v} \, ds, \quad (20)$$

$$\mathbf{f}_s(\mathbf{v}) := \int_{\Omega_s} \mathbf{f} \cdot \mathbf{v} \, dx + \int_{\Gamma_{es}} \mathbf{g}_{\text{ext}} \cdot \mathbf{v} \, ds.$$

Further, we multiply (16) by $\rho_f(\mathbf{x})$, sum up (16) and (17), and make substitutions $v = \frac{\partial \phi}{\partial t}$ and $\mathbf{v} = \frac{\partial \mathbf{u}_s}{\partial t}$ to get the total energy

$$E_{\phi s f}(\phi, \mathbf{u}_s) = \frac{1}{2} \int_{\Omega_f} \left(\frac{\rho_f(\mathbf{x})}{c(\mathbf{x})^2} \left| \frac{\partial \phi}{\partial t} \right|^2 + \rho_f(\mathbf{x}) \left| \nabla \phi \right|^2 \right) dx + \frac{1}{2} \int_{\Omega_s} \left(\rho_s(\mathbf{x}) \left| \frac{\partial \mathbf{u}_s}{\partial t} \right|^2 + \sigma(\mathbf{u}_s) : \epsilon(\mathbf{u}_s) \right) dx. \quad (21)$$

The time-derivative of the total energy is

$$\begin{aligned} \frac{d}{dt} E_{\phi s f}(\phi, \mathbf{u}_s) &= - \int_{\Gamma_{ef}} \frac{\rho_f(\mathbf{x})}{c(\mathbf{x})} \left| \frac{\partial \phi}{\partial t} \right|^2 ds - \int_{\Gamma_{es}} \rho_s(\mathbf{x}) \frac{\partial \mathbf{u}_s^T}{\partial t} \mathbf{B} \frac{\partial \mathbf{u}_s}{\partial t} ds \\ &+ \int_{\Omega_f} \rho_f(\mathbf{x}) f_{\phi} \frac{\partial \phi}{\partial t} dx + \int_{\Gamma_{ef}} \rho_f(\mathbf{x}) y_{\phi \text{ext}} \frac{\partial \phi}{\partial t} ds + \int_{\Omega_s} \mathbf{f} \cdot \frac{\partial \mathbf{u}_s}{\partial t} dx + \int_{\Gamma_{es}} \mathbf{g}_{\text{ext}} \cdot \frac{\partial \mathbf{u}_s}{\partial t} ds. \end{aligned}$$

For $f_{\phi} = 0$, $y_{\phi \text{ext}} = 0$, $\mathbf{f} = \mathbf{0}$, and $\mathbf{g}_{\text{ext}} = \mathbf{0}$, the energy dissipates due to the absorbing boundary conditions on the boundaries Γ_{ef} and Γ_{es} . That is, $\frac{d}{dt} E_{\phi s f}(\phi, \mathbf{u}_s) \leq 0$ is fulfilled, and the problem at hand is stable. In the special case $\frac{d}{dt} E_{\phi s f}(\phi, \mathbf{u}_s) = 0$, the energy is conserved and for a given initial solution $(\mathbf{e}_{\phi f}, \mathbf{e}_s)^T$ holds $E_{\phi s f}(\phi, \mathbf{u}_s) = E_{\phi s f}(\mathbf{e}_{\phi f}, \mathbf{e}_s)$ for all t . For certain non-negative right-hand side terms f_{ϕ} , $y_{\phi \text{ext}}$, \mathbf{f} , and \mathbf{g}_{ext} the energy is non-dissipative, which may cause stability issues.

3 Spectral element discretization

The key factor in developing efficient solution methods is the use of high-order approximations to get high accuracy without computationally demanding ma-

trix inversions. We attempt to meet these requirements by using the spectral element [31] method (SEM) for space discretization.

In order to produce an approximate solution for the problem, the given domain Ω is discretized into a collection of N_e elements Ω_i , $i = 1, \dots, N_e$, such that $\Omega = \bigcup_{i=1}^{N_e} \Omega_i$. The elements are associated with a mesh, which defines the geometry of the domain. After the domain is decomposed into elements, a local polynomial basis is introduced in each element. The basis functions consist of sets of higher-order polynomials and are used to give the discrete values of the approximated solution. In particular, the geometry of the elements is described by invertible affine mappings $\mathcal{G}_i : \Omega_{\text{ref}} \rightarrow \Omega_i$ of a reference element $\Omega_{\text{ref}} = [0, 1]^2$ onto an element Ω_i in the physical coordinates.

In order to compute the elementwise integrals in the reference element, we introduce a set of Gauss–Lobatto points $\xi_i \in [0, 1]$, $i = 1, \dots, r + 1$ in each direction of the reference element. We use quadrilateral elements and Lagrangian method of interpolation for defining a polynomial of order r , the values of which can be determined at $r + 1$ space discretization points. The 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. The degrees of freedom corresponding to the basis functions are located at the Gauss–Lobatto integration points of the elements. With the Gauss–Lobatto integration rule, this makes the mass matrices diagonal without reducing the order of accuracy. Thus, the inversion of the mass matrix is a trivial and computationally efficient operation.

The spectral element method is obtained from the weak formulation of the model (16)-(17) by restricting the problem presented in the infinite-dimensional spaces V and \mathbf{V} into the finite-dimensional subspaces $V_h^r \subset V$ and $\mathbf{V}_h^r \subset \mathbf{V}$, respectively, such that

$$V_h^r = \{v_h \in V \text{ such that } v_h|_{\Omega_i} \circ \mathcal{G}_i \in Q^r\}, \quad (22)$$

$$\mathbf{V}_h^r = \{\mathbf{v}_h = (v_{h1}, v_{h2}) \in \mathbf{V} \text{ such that } v_{hk}|_{\Omega_i} \circ \mathcal{G}_i \in Q^r, k = 1, 2\}, \quad (23)$$

where

$$Q^r(\Omega_i) = \{v(\xi, \zeta) = \sum_{p=0}^r \sum_{q=0}^r a_{pq} \xi^p \zeta^q, \quad a_{pq} \in \mathbb{R}\} \quad (24)$$

is the set of polynomial basis functions of order r in each variable in space. The dimension of the space V_h^r is the number of space discretization points, whereas the dimension of the space \mathbf{V}_h^r is twice the number of space discretization points. The dimensions of the spaces V_h^r and \mathbf{V}_h^r are \hat{N}_f and \hat{N}_s , respectively.

By $\mathbf{u} \in \mathbb{R}^{\hat{N}}$ we denote the global block vector containing the values of the variables in both fluid and structure domains at time t at the Gauss–Lobatto points. The weak formulation for the problem (2)-(9) can now be rewritten in the matrix form

$$\mathcal{M} \frac{\partial^2 \mathbf{u}}{\partial t^2} + \mathcal{S} \frac{\partial \mathbf{u}}{\partial t} + \mathcal{K} \mathbf{u} = \mathcal{F}. \quad (25)$$

The entries of the $\hat{N} \times \hat{N}$ matrices \mathcal{M} , \mathcal{S} , and \mathcal{K} , and the right-hand side vector \mathcal{F} , are given by the formulas

$$\begin{aligned} \mathcal{M} &= \begin{pmatrix} \mathcal{M}_s & \mathbf{0} \\ \mathbf{0} & \mathcal{M}_f \end{pmatrix}, & \mathcal{S} &= \begin{pmatrix} \mathcal{S}_s & \mathcal{A}_{sf} \\ \mathcal{A}_{fs} & \mathcal{S}_f \end{pmatrix}, \\ \mathcal{K} &= \begin{pmatrix} \mathcal{K}_s & \mathbf{0} \\ \mathbf{0} & \mathcal{K}_f \end{pmatrix}, & \mathcal{F} &= \begin{pmatrix} \mathbf{f}_s \\ \mathbf{f}_{\phi f} \end{pmatrix}, \end{aligned}$$

where the $N_f \times N_f$ matrix blocks corresponding to the fluid domain are

$$\begin{aligned} (\mathcal{M}_f)_{ij} &= \int_{\Omega_f} \frac{\rho_f(\mathbf{x})}{c(\mathbf{x})^2} \varphi_i \varphi_j dx, \\ (\mathcal{S}_f)_{ij} &= \int_{\Gamma_{ef}} \frac{\rho_f(\mathbf{x})}{c(\mathbf{x})} \varphi_i \varphi_j ds, \\ (\mathcal{K}_f)_{ij} &= \int_{\Omega_f} \rho_f(\mathbf{x}) \nabla \varphi_i \cdot \nabla \varphi_j dx, \end{aligned}$$

where $i, j = 1, \dots, N_f$. The N_f -dimensional right-hand side vector corresponding to the fluid domain is

$$(\mathbf{f}_{\phi f})_i = \int_{\Omega_f} \rho_f(\mathbf{x}) f_\phi \varphi_i dx + \int_{\Gamma_{ef}} \rho_f(\mathbf{x}) y_{\phi \text{ext}} \varphi_i ds.$$

The $2N_s \times 2N_s$ block matrices and the $2N_s$ -dimensional vector representing the elastic waves have the components

$$\begin{aligned}
((\mathcal{M}_s)_{11})_{ij} &= \int_{\Omega_s} \rho_s(\mathbf{x}) \psi_j \psi_i dx, \\
((\mathcal{M}_s)_{12})_{ij} &= 0, \\
((\mathcal{M}_s)_{21})_{ij} &= 0, \\
((\mathcal{M}_s)_{22})_{ij} &= \int_{\Omega_s} \rho_s(\mathbf{x}) \psi_j \psi_i dx, \\
((\mathcal{S}_s)_{11})_{ij} &= \int_{\Gamma_{es}} \rho_s(\mathbf{x}) (c_p n_{s1}^2 + c_s n_{s2}^2) \psi_j \psi_i ds, \\
((\mathcal{S}_s)_{12})_{ij} &= \int_{\Gamma_{es}} \rho_s(\mathbf{x}) (c_p - c_s) n_{s1} n_{s2} \psi_j \psi_i ds, \\
((\mathcal{S}_s)_{21})_{ij} &= \int_{\Gamma_{es}} \rho_s(\mathbf{x}) (c_p - c_s) n_{s1} n_{s2} \psi_j \psi_i ds, \\
((\mathcal{S}_s)_{22})_{ij} &= \int_{\Gamma_{es}} \rho_s(\mathbf{x}) (c_p n_{s2}^2 + c_s n_{s1}^2) \psi_j \psi_i ds, \\
((\mathcal{K}_s)_{11})_{ij} &= \int_{\Omega_s} \left(\lambda \frac{\partial \psi_j}{\partial x_1} \frac{\partial \psi_i}{\partial x_1} + 2\mu \left(\frac{\partial \psi_j}{\partial x_1} \frac{\partial \psi_i}{\partial x_1} + \frac{1}{2} \frac{\partial \psi_j}{\partial x_2} \frac{\partial \psi_i}{\partial x_2} \right) \right) dx, \\
((\mathcal{K}_s)_{12})_{ij} &= \int_{\Omega_s} \left(\lambda \frac{\partial \psi_j}{\partial x_2} \frac{\partial \psi_i}{\partial x_1} + \mu \frac{\partial \psi_j}{\partial x_1} \frac{\partial \psi_i}{\partial x_2} \right) dx, \\
((\mathcal{K}_s)_{21})_{ij} &= \int_{\Omega_s} \left(\lambda \frac{\partial \psi_j}{\partial x_1} \frac{\partial \psi_i}{\partial x_2} + \mu \frac{\partial \psi_j}{\partial x_2} \frac{\partial \psi_i}{\partial x_1} \right) dx, \\
((\mathcal{K}_s)_{22})_{ij} &= \int_{\Omega_s} \left(\lambda \frac{\partial \psi_j}{\partial x_2} \frac{\partial \psi_i}{\partial x_2} + 2\mu \left(\frac{1}{2} \frac{\partial \psi_j}{\partial x_1} \frac{\partial \psi_i}{\partial x_1} + \frac{\partial \psi_j}{\partial x_2} \frac{\partial \psi_i}{\partial x_2} \right) \right) dx, \\
((\mathbf{f}_s)_1)_i &= \int_{\Omega_s} \mathbf{f}_1 \psi_i dx + \int_{\Gamma_{es}} \mathbf{g}_{ext1} \psi_i ds, \\
((\mathbf{f}_s)_2)_i &= \int_{\Omega_s} \mathbf{f}_2 \psi_i dx + \int_{\Gamma_{es}} \mathbf{g}_{ext2} \psi_i ds,
\end{aligned}$$

where $i, j = 1, \dots, N_s$. The matrices arising from the coupling between acoustic and elastic wave equations are $\mathcal{A}_{fs} = ((\mathcal{A}_{fs})_1, (\mathcal{A}_{fs})_2)$ and $\mathcal{A}_{sf} = ((\mathcal{A}_{sf})_1, (\mathcal{A}_{sf})_2)^T$, for which holds that

$$\begin{aligned}
((\mathcal{A}_{fs})_1)_{ij} &= \int_{\Gamma_i} \rho_f(\mathbf{x}) n_{s1} \psi_j \varphi_i ds, \\
((\mathcal{A}_{fs})_2)_{ij} &= \int_{\Gamma_i} \rho_f(\mathbf{x}) n_{s2} \psi_j \varphi_i ds, \\
((\mathcal{A}_{sf})_1)_{ij} &= \int_{\Gamma_i} \rho_f(\mathbf{x}) n_{f1} \varphi_j \psi_i ds, \\
((\mathcal{A}_{sf})_2)_{ij} &= \int_{\Gamma_i} \rho_f(\mathbf{x}) n_{f2} \varphi_j \psi_i ds.
\end{aligned}$$

For \mathcal{A}_{fs} , $i = 1, \dots, N_f$ and $j = 1, \dots, N_s$, whereas for \mathcal{A}_{sf} , $i = 1, \dots, N_s$ and $j = 1, \dots, N_f$.

The computation of the elementwise matrices and vectors involves the integration over the elementwise subregions. In practice, we replace the 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, for $i \neq j$, $(\mathcal{M}_f)_{ij} = 0$ and $(\mathcal{M}_s)_{ij} = 0$ meaning that the matrix \mathcal{M} is diagonal. Thus, the inverse of the matrix \mathcal{M} can be easily computed. In practice, the stiffness matrix \mathcal{K} is assembled once at the beginning of the simulation. It is stored by using the compressed column storage including only the non-zero matrix elements. The other options would have been using a mixed spectral element formulation [32,33].

4 Time discretization

After space discretization, the time-harmonic equations can, in principle, be solved by either direct or iterative solvers. In practice, direct solvers are reasonable only for small problems. For large problems, iterative methods and efficient preconditioners are needed. Since developing efficient preconditioners is a challenging task, we return to another approach and use time-dependent equations for creating time-harmonic solutions. To continue towards that goal, we consider the time discretization. Previously, we used the central finite difference (CD) scheme for time discretization of acoustic and multiphysical problems in [34,12], and comparison with the Runge–Kutta (RK) method for disjoint acoustic and elastic domains was made in [10,11]. Although at each time step the computational effort of the RK method is approximately four times that of the CD scheme, the RK method seems to be more efficient in conjunction with the controllability approach.

The state equation (25) can be presented as a system of differential equations

$$\frac{\partial \mathbf{y}}{\partial t} = f(t, \mathbf{y}(t)), \quad (26)$$

where $\mathbf{y} = (\mathbf{u}, \mathbf{v})^T$ is a vector of time-stepping variables \mathbf{u} and $\mathbf{v} = \frac{\partial \mathbf{u}}{\partial t}$, 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

$$f_1(t, \mathbf{u}, \mathbf{v}) = \mathbf{v}, \quad (27)$$

$$f_2(t, \mathbf{u}, \mathbf{v}) = -\mathcal{M}^{-1}(\mathcal{S}\mathbf{v} + \mathcal{K}\mathbf{u} - \mathcal{F}). \quad (28)$$

To this modified form, we can apply the fourth-order Runge–Kutta method, which is a Taylor series method. In general, the Taylor series methods keep the errors small, but there is the disadvantage of requiring the evaluation of higher derivatives of the function $f(t, \mathbf{y}(t))$. The advantage of the 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 the values of $f(t, \mathbf{y}(t))$ are used to approximate $\mathbf{y}(t)$. In the fourth-order Runge–Kutta method, the approximate \mathbf{y} at the i th time step is defined as

$$\mathbf{y}^i = \mathbf{y}^{i-1} + \frac{\Delta t}{6} (k_1 + 2k_2 + 2k_3 + k_4), \quad (29)$$

where $\mathbf{y}^i = (\mathbf{u}^i, \frac{\partial \mathbf{u}^i}{\partial t})^T$ contains the global block vector \mathbf{u}^i , including the values of the variables in both the fluid and the structure domain at the i th time step, and its derivative $\mathbf{v}^i = \frac{\partial \mathbf{u}^i}{\partial t}$ at time $t = i\Delta t$, $i = 1, \dots, N$. The initial condition is given by $\mathbf{y}^0 = \mathbf{e} = (\mathbf{e}_0, \mathbf{e}_1)^T$, and $k_j = (k_{j1}, k_{j2})^T$, $j = 1, 2, 3, 4$, are

the differential estimates as follows:

$$\begin{pmatrix} k_{11} \\ k_{12} \end{pmatrix} = \begin{pmatrix} f_1(i\Delta t, \mathbf{u}^i, \mathbf{v}^i) \\ f_2(i\Delta t, \mathbf{u}^i, \mathbf{v}^i) \end{pmatrix}, \quad (30)$$

$$\begin{pmatrix} k_{21} \\ k_{22} \end{pmatrix} = \begin{pmatrix} f_1(i\Delta t + \frac{\Delta t}{2}, \mathbf{u}^i + \frac{k_{11}}{2}, \mathbf{v}^i + \frac{k_{12}}{2}) \\ f_2(i\Delta t + \frac{\Delta t}{2}, \mathbf{u}^i + \frac{k_{11}}{2}, \mathbf{v}^i + \frac{k_{12}}{2}) \end{pmatrix}, \quad (31)$$

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

$$\begin{pmatrix} k_{41} \\ k_{42} \end{pmatrix} = \begin{pmatrix} f_1(i\Delta t + \Delta t, \mathbf{u}^i + k_{31}, \mathbf{v}^i + k_{32}) \\ f_2(i\Delta t + \Delta t, \mathbf{u}^i + k_{31}, \mathbf{v}^i + k_{32}) \end{pmatrix}. \quad (33)$$

In other words, in order to get the differential estimates (30)-(33), the function f is evaluated at each time step four times by using the formulas (27)-(28), and then the successive approximation of \mathbf{y} is calculated by the formula (29). To make the application of the adjoint equation technique in Section 5 more convenient, we present the fully discrete state equation in the case of the Runge–Kutta time discretization as

$$s(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e})) = \begin{pmatrix} \mathcal{I} & & & & & \\ \mathcal{N} & \mathcal{I} & & & & \\ & \ddots & \ddots & & & \\ & & \mathcal{N} & \mathcal{I} & & \\ & & & \mathcal{N} & \mathcal{I} & \end{pmatrix} \begin{pmatrix} \mathbf{y}^0 \\ \mathbf{y}^1 \\ \vdots \\ \mathbf{y}^{N-1} \\ \mathbf{y}^N \end{pmatrix} - \begin{pmatrix} \mathcal{I} \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \mathbf{e} - \begin{pmatrix} 0 \\ \hat{\mathcal{F}}^1 \\ \vdots \\ \hat{\mathcal{F}}^{N-1} \\ \hat{\mathcal{F}}^N \end{pmatrix} = 0, \quad (34)$$

where $\hat{\mathbf{y}} = (\mathbf{y}^0, \mathbf{y}^1, \dots, \mathbf{y}^{N-1}, \mathbf{y}^N)^T$ includes the vectors $\mathbf{y}^i = (\mathbf{u}^i, \frac{\partial \mathbf{u}^i}{\partial t})^T$, $\mathbf{e} = (\mathbf{e}_0, \mathbf{e}_1)^T$ contains the initial values, and the matrix \mathcal{N} and the vector $\hat{\mathcal{F}}^i$ are defined by

$$\mathcal{N} = - \begin{pmatrix} \hat{\mathcal{C}} \\ 2\hat{\mathcal{C}} \\ 2\hat{\mathcal{C}} \\ \hat{\mathcal{C}} \end{pmatrix}^T \begin{pmatrix} \mathcal{I} & & & \\ & \hat{\mathcal{B}} \mathcal{I} & & \\ & & \hat{\mathcal{B}} \mathcal{I} & \\ & & & 2\hat{\mathcal{B}} \mathcal{I} \end{pmatrix}^{-1} \begin{pmatrix} 2\hat{\mathcal{B}} \\ 2\hat{\mathcal{B}} \\ 2\hat{\mathcal{B}} \\ 2\hat{\mathcal{B}} \end{pmatrix} - \mathcal{I}, \quad (35)$$

$$\hat{\mathcal{F}}^i = - \begin{pmatrix} \hat{\mathcal{C}} \\ 2\hat{\mathcal{C}} \\ 2\hat{\mathcal{C}} \\ \hat{\mathcal{C}} \end{pmatrix}^T \begin{pmatrix} \mathcal{I} & & & \\ & \hat{\mathcal{B}} \mathcal{I} & & \\ & & \hat{\mathcal{B}} \mathcal{I} & \\ & & & 2\hat{\mathcal{B}} \mathcal{I} \end{pmatrix}^{-1} \begin{pmatrix} \hat{\mathcal{D}}^{i-1} \\ \hat{\mathcal{D}}^{i-\frac{1}{2}} \\ \hat{\mathcal{D}}^{i-\frac{1}{2}} \\ \hat{\mathcal{D}}^i \end{pmatrix}. \quad (36)$$

The matrix blocks $\hat{\mathcal{C}}$ and $\hat{\mathcal{B}}$ and the vector blocks $\hat{\mathcal{D}}^i$ are given by the formulas

$$\begin{aligned} \hat{\mathcal{C}} &= \begin{pmatrix} -\frac{1}{6}\mathcal{I} & 0 \\ 0 & -\frac{1}{6}\mathcal{I} \end{pmatrix}, \\ \hat{\mathcal{B}} &= \begin{pmatrix} 0 & -\frac{\Delta t}{2}\mathcal{I} \\ \frac{\Delta t}{2}\mathcal{M}^{-1}\mathcal{K} & \frac{\Delta t}{2}\mathcal{M}^{-1}\mathcal{S} \end{pmatrix}, \\ \hat{\mathcal{D}}^i &= \begin{pmatrix} 0 \\ \Delta t \mathcal{M}^{-1} \mathcal{F}^i \end{pmatrix}, \end{aligned}$$

where \mathcal{I} is the identity matrix, \mathcal{F}^i is the vector \mathcal{F} at time $t = i\Delta t$. 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

$$\begin{pmatrix} \mathcal{I} \\ \hat{\mathcal{B}} \mathcal{I} \\ \hat{\mathcal{B}} \mathcal{I} \\ 2\hat{\mathcal{B}} \mathcal{I} \end{pmatrix} \begin{pmatrix} k_1 \\ k_2 \\ k_3 \\ k_4 \end{pmatrix} + \begin{pmatrix} 2\hat{\mathcal{B}} \\ 2\hat{\mathcal{B}} \\ 2\hat{\mathcal{B}} \\ 2\hat{\mathcal{B}} \end{pmatrix} \mathbf{y}^{i-1} - \begin{pmatrix} \hat{\mathcal{D}}^{i-1} \\ \hat{\mathcal{D}}^{i-\frac{1}{2}} \\ \hat{\mathcal{D}}^{i-\frac{1}{2}} \\ \hat{\mathcal{D}}^i \end{pmatrix} = 0. \quad (37)$$

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

$$\mathbf{y}^i = \mathbf{y}^{i-1} - \begin{pmatrix} \hat{c} & 2\hat{c} & 2\hat{c} & \hat{c} \end{pmatrix} \begin{pmatrix} k_1 \\ k_2 \\ k_3 \\ k_4 \end{pmatrix}. \quad (38)$$

Since the matrix \mathcal{M} is diagonal, the only matrix inversion needed in time-stepping (e.g., \mathcal{M}^{-1} in Equation (28)) 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.

5 Exact controllability approach

The time-harmonic solution of the acoustic-elastic interaction problem is needed in many applications. Our objective is to return to the time-dependent wave equation and achieve the time-harmonic solution by minimizing the difference between initial conditions and the corresponding variables after one time period. Thus, the basic idea is to have preassigned initial and final states such that beginning from the initial state, the final state can be achieved by controlling the initial conditions. Proceeding in 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 (CG) algorithm.

Solving the time-harmonic equation is equivalent to finding a time-periodic solution for the corresponding time-dependent wave equation with the initial conditions

$$\mathbf{u}(\mathbf{x}, 0) = \mathbf{e}_0, \quad \frac{\partial \mathbf{u}(\mathbf{x}, 0)}{\partial t} = \mathbf{e}_1. \quad (39)$$

The time period corresponding to the angular frequency ω is given by $T = \frac{2\pi}{\omega}$. 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.

Thus, we formulate the exact controllability problem as follows: Find initial

conditions $\mathbf{e} = (\mathbf{e}_0, \mathbf{e}_1)^T$ such that the weak formulation holds with the terminal conditions

$$\mathbf{u}(\mathbf{x}, T) = \mathbf{e}_0, \quad \frac{\partial \mathbf{u}(\mathbf{x}, T)}{\partial t} = \mathbf{e}_1. \quad (40)$$

The purpose of optimal control problems is to minimize an objective functional (cost function) J defined in a control space $Z = (\mathbf{V} \times V) \times L^2(\Omega)$.

5.1 Objective functional

The algorithm involves computation of the gradient of a least-squares functional J , which is an essential stage of the algorithm. We have chosen to minimize the functional based on the natural energy norm associated with the energy formulation (21).

In order to define the optimal control, an objective functional corresponding to the energy formulation (21) is defined as

$$\begin{aligned} J(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e})) &= \frac{1}{2} \int_{\Omega_f} \left(\rho_f(\mathbf{x}) \left| \nabla(\phi(\mathbf{x}, T) - \mathbf{e}_{f0}) \right|^2 + \frac{\rho_f(\mathbf{x})}{c(\mathbf{x})^2} \left| \frac{\partial \phi(\mathbf{x}, T)}{\partial t} - \mathbf{e}_{f1} \right|^2 \right) dx \\ &+ \frac{1}{2} \int_{\Omega_s} \left(2\mu_s \left| \epsilon(\mathbf{u}_s(\mathbf{x}, T) - \mathbf{e}_{s0}) \right|^2 + \lambda_s \left| \nabla \cdot (\mathbf{u}_s(\mathbf{x}, T) - \mathbf{e}_{s0}) \right|^2 + \rho_s(\mathbf{x}) \left| \frac{\partial \mathbf{u}_s(\mathbf{x}, T)}{\partial t} - \mathbf{e}_{s1} \right|^2 \right) dx, \end{aligned} \quad (41)$$

where $\mathbf{e} = (\mathbf{e}_0, \mathbf{e}_1) \in Z$ and $\hat{\mathbf{y}}$ is the solution of the state equation. The optimal control problem can then be reformulated as seeking the control that minimizes the objective functional. The discrete counterpart of the objective functional (41) is

$$J(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e})) = \frac{1}{2} (\mathbf{y}^N - \mathbf{e})^T \begin{pmatrix} \mathcal{K} & \mathbf{0} \\ \mathbf{0} & \mathcal{M} \end{pmatrix} (\mathbf{y}^N - \mathbf{e}), \quad (42)$$

where \mathbf{y}^i are given by Equation (34).

In order to solve the exact controllability problem, we use the least-squares formulation

$$\min_{\mathbf{e} \in Z} J(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e})), \quad (43)$$

where $\hat{\mathbf{y}}(\mathbf{e})$ solves the transient initial value problem (state equation (34)) and $J(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e}))$ is the discretized objective functional. The result of the minimization problem (43) is the optimal control \mathbf{e}^* and the corresponding state $\hat{\mathbf{y}}(\mathbf{e}^*)$ is the optimal state. The purpose is to minimize the functional J , which depends on the initial conditions both directly and indirectly through the solution of the wave equation. Since the vector $\hat{\mathbf{y}}$ depends linearly on the initial conditions \mathbf{e}_0 and \mathbf{e}_1 , J is a quadratic functional. Thus, solving the minimization problem (43) is equivalent to finding initial conditions $\mathbf{e}^* \in Z$ such that the gradient of J is zero, that is,

$$\nabla J(\mathbf{e}^*, \hat{\mathbf{y}}(\mathbf{e}^*)) = 0. \quad (44)$$

Since J is a quadratic functional, (44) defines a linear system, and the minimization problem can be solved by a conjugate gradient (CG) algorithm. Each iteration step of the algorithm requires the gradient of J with respect to the control variables $\mathbf{e}_0 = (\mathbf{e}_{0_1}, \mathbf{e}_{0_2}, \dots, \mathbf{e}_{0_{(2N_s+N_f)}})^T$ and $\mathbf{e}_1 = (\mathbf{e}_{1_1}, \mathbf{e}_{1_2}, \dots, \mathbf{e}_{1_{(2N_s+N_f)}})^T$. One option for computing the gradient would be using the central finite difference approximation of the gradient of the objective functional. In order to implement an efficient algorithm, we proceed in a more practical way and compute the derivative of J by the adjoint equation technique. For condensing the formulation, we represent the state equation (34) in the generic form

$$s(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e})) = 0, \quad (45)$$

where $\mathbf{e} = (\mathbf{e}_0, \mathbf{e}_1)^T$ contains the initial values and $\hat{\mathbf{y}}$ contains the vectors $\mathbf{y}^i = \left(\mathbf{u}^i, \frac{\partial \mathbf{u}^i}{\partial t}\right)^T$. By $s_0(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e})) = 0$ we denote the state equation in the special case with $\mathcal{F}^i = 0$ for all i .

Using the standard adjoint equation technique of the optimal control theory (see, e.g., [35]), we see that

$$\frac{dJ(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e}))}{d\mathbf{e}_k} = \frac{\partial J(\mathbf{e}, \hat{\mathbf{y}})}{\partial \mathbf{e}_k} - \hat{\mathbf{z}}^T \frac{\partial s(\mathbf{e}, \hat{\mathbf{y}})}{\partial \mathbf{e}_k}, \quad k = 0, 1, \quad (46)$$

where $\hat{\mathbf{z}}$ is the adjoint state vector containing the vectors \mathbf{p}^i presenting the adjoint state variable at time $i\Delta t$. The vector $\hat{\mathbf{z}}$ is the solution of the adjoint equation

$$\left(\frac{\partial s(\mathbf{e}, \hat{\mathbf{y}})}{\partial \hat{\mathbf{y}}}\right)^T \hat{\mathbf{z}} = (\nabla_{\hat{\mathbf{y}}} J(\mathbf{e}, \hat{\mathbf{y}}))^T. \quad (47)$$

Computing the gradient in the whole domain by the adjoint equation technique requires computing the state and the corresponding adjoint state equation and some additional matrix-vector multiplications. The CPU time for computing the state equation is approximately the same as for computing the corresponding adjoint state equation. This computational cost is much larger than that needed for the additional matrix-vector multiplications. On the other hand, computing the gradient component with respect to the variable \mathbf{e}_k with the finite difference approximation requires solving the state equation twice per each degree of freedom. Hence, computing the gradient by the adjoint equation technique takes only half of the time that is consumed for computing the gradient for each degree of freedom with the finite difference formula.

5.2 Gradient for the fourth-order Runge–Kutta time discretization

In the case of the fourth-order Runge–Kutta time discretization, the adjoint equation corresponding to the state equation (34) is

$$\begin{pmatrix} \mathcal{I} & \mathcal{N}^T & & & & \\ & \mathcal{I} & \mathcal{N}^T & & & \\ & & \ddots & \ddots & & \\ & & & \mathcal{I} & \mathcal{N}^T & \\ & & & & \mathcal{I} & \end{pmatrix} \begin{pmatrix} \mathbf{z}^0 \\ \mathbf{z}^1 \\ \vdots \\ \mathbf{z}^{N-1} \\ \mathbf{z}^N \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \frac{\partial J}{\partial \mathbf{y}^N} \end{pmatrix}, \quad (48)$$

where $\mathbf{z}^i = (\mathbf{p}^i, \frac{\partial \mathbf{p}^i}{\partial t})^T$ contains the solution of the adjoint equation and its time derivative at $t = i\Delta t$, $i = N, \dots, 0$. The non-zero right-hand side terms are defined as

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

Thus, the evolution in time with the adjoint state equation starts with the value \mathbf{z}^N , and after one time period we get the solution \mathbf{z}^0 . Then, we can compute the gradient components for the fourth-order Runge–Kutta scheme, which are

$$\frac{dJ(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e}))}{d\mathbf{e}_0} = \mathcal{K}(\mathbf{e}_0 - \mathbf{u}^N) + \mathbf{p}^0, \quad (49)$$

$$\frac{dJ(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e}))}{d\mathbf{e}_1} = \mathcal{M} \left(\mathbf{e}_1 - \frac{\partial \mathbf{u}^N}{\partial t} \right) + \frac{\partial \mathbf{p}^0}{\partial t}. \quad (50)$$

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

$$\begin{pmatrix} \mathcal{I} & \hat{\mathcal{H}}^T & & \\ & \mathcal{I} & \hat{\mathcal{H}}^T & \\ & & \mathcal{I} & 2\hat{\mathcal{H}}^T \\ & & & \mathcal{I} \end{pmatrix} \begin{pmatrix} k_1 \\ k_2 \\ k_3 \\ k_4 \end{pmatrix} = \begin{pmatrix} -\hat{R} \\ -2\hat{R} \\ -2\hat{R} \\ -\hat{R} \end{pmatrix} \mathbf{z}^{i+1}, \quad (51)$$

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

$$\mathbf{z}^i = \mathbf{z}^{i+1} - \begin{pmatrix} 2\hat{\mathcal{H}}^T & 2\hat{\mathcal{H}}^T & 2\hat{\mathcal{H}}^T & 2\hat{\mathcal{H}}^T \end{pmatrix} \begin{pmatrix} k_1 \\ k_2 \\ k_3 \\ k_4 \end{pmatrix}. \quad (52)$$

6 Optimization algorithm

Although the CG method has been shown to be robust with respect to the initial values in conjunction with the exact controllability approach (see, e.g., [6]), it is important to have smooth initial approximations for \mathbf{e}_0 and \mathbf{e}_1 , which satisfy the boundary conditions. In [6], a special procedure suggested by Mur in [16] was used leading to faster convergence to the time-harmonic solution of scattering problems for harmonic planar waves by purely reflecting non-convex obstacles. That is, they focused on acoustic scattering with sound-soft obstacles and electromagnetic applications with perfectly conducting obstacles. Now, we generalize the same procedure to the coupled problem, and first define a smooth transition function $\theta(t)$, which increases from zero to one in the time interval $[0, T_{\text{tr}}]$ as follows:

$$\theta(t) = \begin{cases} \left(2 - \sin\left(\frac{\pi t}{2T_{\text{tr}}}\right)\right) \sin\left(\frac{\pi t}{2T_{\text{tr}}}\right), & \text{if } 0 \leq t \leq T_{\text{tr}}, \\ 1, & \text{if } t \geq T_{\text{tr}}. \end{cases} \quad (53)$$

The length of the time interval should be chosen as a multiple of the period T , that is, $T_{\text{tr}} = nT$ with n a positive integer. we solve the following initial value problem:

$$\frac{1}{c(\mathbf{x})^2} \frac{\partial^2 \phi}{\partial t^2} - \nabla^2 = \theta(t) f_\phi, \quad \text{in } Q_f = \Omega_f \times [0, T_{\text{tr}}], \quad (54)$$

$$\phi = 0, \quad \text{on } \gamma_{\text{of}} = \Gamma_{\text{of}} \times [0, T_{\text{tr}}], \quad (55)$$

$$\frac{1}{c(\mathbf{x})} \frac{\partial \phi}{\partial t} + \frac{\partial \phi}{\partial \mathbf{n}_f} = \theta(t) y_{\phi \text{ext}}, \quad \text{on } \gamma_{ef} = \Gamma_{ef} \times [0, T_{\text{tr}}], \quad (56)$$

$$\frac{\partial \mathbf{u}_s}{\partial t} \cdot \mathbf{n}_s + \frac{\partial \phi}{\partial \mathbf{n}_f} = 0, \quad \text{on } \gamma_i = \Gamma_i \times [0, T_{\text{tr}}], \quad (57)$$

$$\rho_s(\mathbf{x}) \frac{\partial^2 \mathbf{u}_s}{\partial t^2} - \nabla \cdot \sigma(\mathbf{u}_s) = \theta(t) \mathbf{f}, \quad \text{in } Q_s = \Omega_s \times [0, T_{\text{tr}}], \quad (58)$$

$$\mathbf{u}_s = \mathbf{0}, \quad \text{on } \gamma_{\text{os}} = \Gamma_{\text{os}} \times [0, T_{\text{tr}}], \quad (59)$$

$$\rho_s(\mathbf{x}) \mathbf{B} \frac{\partial \mathbf{u}_s}{\partial t} + \sigma(\mathbf{u}_s) \mathbf{n}_s = \theta(t) \mathbf{g}_{\text{ext}}, \quad \text{on } \gamma_{es} = \Gamma_{es} \times [0, T_{\text{tr}}], \quad (60)$$

$$\sigma(\mathbf{u}_s) \mathbf{n}_s + \rho_f(\mathbf{x}) \frac{\partial \phi}{\partial t} \mathbf{n}_f = \mathbf{0}, \quad \text{on } \gamma_i = \Gamma_i \times [0, T_{\text{tr}}], \quad (61)$$

$$\phi(\mathbf{x}, 0) = \mathbf{0}, \quad \text{in } \Omega_f, \quad (62)$$

$$\frac{\partial \phi}{\partial t}(\mathbf{x}, 0) = \mathbf{0}, \quad \text{in } \Omega_f, \quad (63)$$

$$\mathbf{u}_s(\mathbf{x}, 0) = \mathbf{0}, \quad \text{in } \Omega_s, \quad (64)$$

$$\frac{\partial \mathbf{u}_s}{\partial t}(\mathbf{x}, 0) = \mathbf{0}, \quad \text{in } \Omega_s. \quad (65)$$

After solving the problem (54)-(65), the initial approximations for the control variables \mathbf{e}_0 and \mathbf{e}_1 are constructed to consist of the solutions ϕ and \mathbf{u}_s and their time derivatives at time T_{tr} such that

$$\begin{aligned} \mathbf{e}_0^0 &= (\phi(\mathbf{x}, T_{\text{tr}}), \mathbf{u}_s(\mathbf{x}, T_{\text{tr}}))^T, \\ \mathbf{e}_1^0 &= \left(\frac{\partial \phi}{\partial t}(\mathbf{x}, T_{\text{tr}}), \frac{\partial \mathbf{u}_s}{\partial t}(\mathbf{x}, T_{\text{tr}}) \right)^T. \end{aligned}$$

If there are no interacting reflections, already this initial procedure may con-

verge rapidly to the time-harmonic solution. However, in general the convergence is slow and we need to continue with the control algorithm.

6.1 Conjugate gradient method

In the conjugate gradient (CG) method, the first search direction is chosen to be the direction of the steepest descent, that is, $\mathbf{w}^0 = -\nabla J(\mathbf{e}^0)$. The successive directions \mathbf{w}^i , $i = 1, 2, \dots$ are generated to be conjugant with the matrix \mathbf{A} . Thus, the inner product of \mathbf{w}^i and $\mathbf{A}\mathbf{w}^j$ is zero, that is, $(\mathbf{w}^i, \mathbf{A}\mathbf{w}^j) = 0$, $i \neq j$ meaning that the vectors $\mathbf{w}^0, \mathbf{w}^1, \dots, \mathbf{w}^{\hat{N}-1}$ are said to be \mathbf{A} -conjugate. A set of non-zero \mathbf{A} -conjugate vectors are linearly independent and form a basis which spans the vector space of \mathbf{e} . Assume we are given a starting point \mathbf{e}^0 and a \mathbf{A} -conjugate set $\{\mathbf{w}^0, \mathbf{w}^1, \dots, \mathbf{w}^{\hat{N}-1}\}$. Since the vectors $\mathbf{w}^0, \mathbf{w}^1, \dots, \mathbf{w}^{\hat{N}-1}$ form a basis, we can write the vector representing the move from \mathbf{e}^0 to the minimum point \mathbf{e}^* as a linear combination of these vectors, in other words, we have

$$\mathbf{e}^* = \mathbf{e}^0 + \sum_{i=0}^{\hat{N}-1} \eta^i \mathbf{w}^i = \mathbf{e}^k + \sum_{i=k}^{\hat{N}-1} \eta^i \mathbf{w}^i, \quad (66)$$

where η^i , $i = 0, \dots, \hat{N} - 1$ are scalars. Multiplying the previous equation by $(\mathbf{w}^j)^T \mathbf{A}$ and substituting \mathbf{b} for $\mathbf{A}\mathbf{e}^*$ gives

$$(\mathbf{w}^j)^T (\mathbf{b} - \mathbf{A}\mathbf{e}^k) = \sum_{i=k}^{\hat{N}-1} \eta^i (\mathbf{w}^j)^T \mathbf{A}\mathbf{w}^i. \quad (67)$$

If $\mathbf{w}^0, \mathbf{w}^1, \dots, \mathbf{w}^{\hat{N}-1}$ were not \mathbf{A} -conjugate, determining $\eta^0, \eta^1, \dots, \eta^{\hat{N}-1}$ would involve solving \hat{N} linear equations in \hat{N} unknowns. \mathbf{A} -conjugacy eliminates the cross terms and gives a closed form equation for η^i , $i = 0, \dots, \hat{N} - 1$ such that the line search parameter is

$$\eta^i = \frac{-(\mathbf{w}^i)^T \mathbf{g}^i}{(\mathbf{w}^i)^T \mathbf{A}\mathbf{w}^i}, \quad (68)$$

where $\mathbf{g}^i = \mathbf{A}\mathbf{e}^i - \mathbf{b}$ is the gradient of J at point \mathbf{e}^i . Since $\mathbf{g}^i - \mathbf{g}^{i-1} = \mathbf{A}(\mathbf{e}^i - \mathbf{e}^{i-1})$, we can, in practice, compute the gradient by $\mathbf{g}^i = \mathbf{g}^{i-1} + \eta^{i-1} \mathbf{A}\mathbf{w}^{i-1}$.

Each conjugate gradient iteration step requires computation of the gradient of the least-squares functional, ∇J by the formulas (49)-(50), which involves the solution of the (34) and the corresponding adjoint equation (48), the solution of a linear system with the preconditioner, and some other matrix-vector operations.

The convergence of the CG method is dependent on the spectrum of the eigenvalues of the matrix \mathbf{A} . For problems with a large condition number, we accelerate the convergence rate by preconditioning. That is, we decrease the condition number by multiplying the linear problem $\mathbf{A}\mathbf{e} = \mathbf{b}$ by a matrix \mathcal{L}^{-1} implying that instead of seeking a solution from the space $\mathbf{e}^0 + K_j(\mathbf{g}^0, \mathbf{A})$, where K_j is the j :th Krylov subspace defined as $K_j(\mathbf{g}^0, \mathbf{A}) = \text{span}(\mathbf{g}^0, \mathbf{A}\mathbf{g}^0, \dots, \mathbf{A}^{j-1}\mathbf{g}^0)$, we are seeking the solution from the space $\mathbf{e}^0 + K_j(\mathcal{L}^{-1}\mathbf{g}^0, \mathcal{L}^{-1}\mathbf{A})$. The geometrical interpretation of preconditioning is that we minimize a functional with contourlines more spherical than in the case of the original functional (for more information, see, e.g., [36]).

The new direction is determined as a linear combination of the previous direction and the steepest descent direction by using the scaling factor γ^i . By choosing a Fletcher–Reeves type formula for computing γ^i and the stopping criterion measuring the relative norm of the residual such that the iteration stops as $\sqrt{\frac{c}{c_0}} < \varepsilon$, we get the following algorithm:

Algorithm 1 *Preconditioned CG algorithm*

Compute the initial value $\mathbf{e}^0 = (\mathbf{e}_0^0, \mathbf{e}_1^0)^T$.

Solve the state equation $s(\mathbf{e}^0, \hat{\mathbf{y}}(\mathbf{e}^0)) = 0$ by the formula (34).

Solve the adjoint state equation $\left(\frac{\partial s(\mathbf{e}^0, \hat{\mathbf{y}}(\mathbf{e}^0))}{\partial \hat{\mathbf{y}}(\mathbf{e}^0)}\right)^T \hat{\mathbf{z}} = \left(\frac{\partial J(\mathbf{e}^0, \hat{\mathbf{y}}(\mathbf{e}^0))}{\partial \hat{\mathbf{y}}(\mathbf{e}^0)}\right)^T$ by the formula (47).

Compute the gradient $\mathbf{g}^0 = (g_0^0, g_1^0)^T$ by the formulas (49)-(50).

Solve linear system with the preconditioner $\mathcal{L}\mathbf{w}^0 = -\mathbf{g}^0$.

Set $c_0 = -(\mathbf{w}^0, \mathbf{g}^0)$, $c = c_0$ and $\ell = 1$.

Repeat until $\sqrt{\frac{c}{c_0}} < \varepsilon$

Solve the state equation $s_0(\mathbf{w}^{\ell-1}, \hat{\mathbf{y}}(\mathbf{w}^{\ell-1})) = 0$ by the formula (34) with $\mathcal{F}^i = 0$ for all i .

Solve the adjoint state equation $\left(\frac{\partial s(\mathbf{w}^{\ell-1}, \hat{\mathbf{y}}(\mathbf{w}^{\ell-1}))}{\partial \hat{\mathbf{y}}(\mathbf{w}^{\ell-1})}\right)^T \hat{\mathbf{z}} = \left(\frac{\partial J(\mathbf{w}^{\ell-1}, \hat{\mathbf{y}}(\mathbf{w}^{\ell-1}))}{\partial \hat{\mathbf{y}}(\mathbf{w}^{\ell-1})}\right)^T$ by the formula (47).

Compute the gradient update $\mathbf{v}^{\ell-1} = (v_0^{\ell-1}, v_1^{\ell-1})^T$ by the formulas (49)-(50) .

Compute $\eta^{\ell-1} = \frac{c}{(\mathbf{w}^{\ell-1}, \mathbf{v}^{\ell-1})}$.

Update the control vector $\mathbf{e}^\ell = \mathbf{e}^{\ell-1} + \eta^{\ell-1}\mathbf{w}^{\ell-1}$.

Update the residual vector $\mathbf{g}^\ell = \mathbf{g}^{\ell-1} + \eta^{\ell-1}\mathbf{v}^{\ell-1}$.

Solve linear system with the preconditioner $\mathcal{L}\mathbf{v}^\ell = -\mathbf{g}^\ell$.

Compute $\gamma^{\ell-1} = \frac{1}{c}$, $c = -(\mathbf{v}^\ell, \mathbf{g}^\ell)$, $\gamma^{\ell-1} = c\gamma^{\ell-1}$.

Update minimizing direction $\mathbf{w}^\ell = \mathbf{v}^\ell + \gamma^{\ell-1}\mathbf{w}^{\ell-1}$.

Set $\ell = \ell + 1$,

Compute the time-harmonic solution $\hat{\mathbf{e}}^$,*

where ℓ refers to the number of iterations. Values of the control variables \mathbf{e} at the ℓ th iteration are denoted by \mathbf{e}_0^ℓ and \mathbf{e}_1^ℓ , $\hat{\mathbf{y}}$ is the solution of the state

equation (34), $\hat{\mathbf{z}} = (\mathbf{p}^0, \frac{\partial \mathbf{p}^0}{\partial t})$ is the solution of the adjoint state equation (48), and the gradient variable $\mathbf{g} = (g_0, g_1)$ is computed by the formulas (49)-(50). By $s_0(\mathbf{w}, \hat{\mathbf{y}}(\mathbf{w})) = 0$ we denote the state equation (34), where $\mathcal{F}^i = 0$ for all i . The algorithm provides the time-harmonic computational solution, that is, $\hat{\mathbf{e}}^*$. The time-dependence of the model (2)-(9) is supposed to be of the form $\exp(i\omega t)$, where i is the imaginary unit and ω is the angular frequency, implying that $\hat{\mathbf{e}}^* = \mathbf{e}_0^{\ell-1} - i \mathbf{e}_1^{\ell-1} / \omega$.

6.2 Preconditioning using the graph-based multigrid

From the formula (42), we see that the functional depends on the initial conditions both directly and indirectly through the solution of the linear wave equation. By substituting $\hat{\mathbf{y}} = \mathbf{B}\mathbf{e} + \mathbf{d}$ into (42) and comparing with the formula (1), we get

$$\mathbf{A} = (\mathbf{B} - \mathcal{I})^T \begin{pmatrix} \mathcal{K} & \mathbf{0} \\ \mathbf{0} & \mathcal{M} \end{pmatrix} (\mathbf{B} - \mathcal{I}), \quad (69)$$

$$\mathbf{b} = -\mathbf{d}^T \begin{pmatrix} \mathcal{K} & \mathbf{0} \\ \mathbf{0} & \mathcal{M} \end{pmatrix} (\mathbf{B} - \mathcal{I}), \quad \mathbf{c} = \frac{1}{2} \mathbf{d}^T \begin{pmatrix} \mathcal{K} & \mathbf{0} \\ \mathbf{0} & \mathcal{M} \end{pmatrix} (\mathbf{B} - \mathcal{I}) \mathbf{d}. \quad (70)$$

Since the block-matrix $\text{diag}\{\mathcal{K}, \mathcal{M}\}$ is symmetric, also the matrix \mathbf{A} , as presented above, is symmetric.

We use a block-diagonal preconditioner

$$\mathcal{L} = \begin{pmatrix} \mathcal{K} & \mathbf{0} \\ \mathbf{0} & \mathcal{M} \end{pmatrix}, \quad (71)$$

which corresponds to the energy formulation (21). The solution of the linear system with the block-diagonal 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 graph-based multigrid (GBMG) method [17] that we have applied to acoustic and elastic problems in separate domains, e.g., in the papers [10,11]. As a smoother of the GBMG we have used successive over-relaxation (SOR), with over-relaxation parameter 1.2, unless mentioned otherwise. One iteration of the SOR is used for pre- and post-smoothing. Additionally, at the beginning of every multigrid iteration, four iterations of the SOR are used to smooth the solution initially. The so called

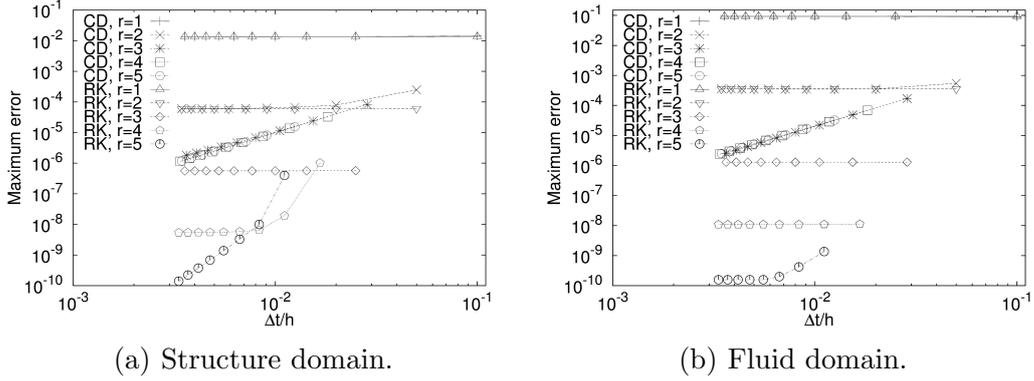


Figure 2. Accuracy of the numerical solution of the state equation obtained by the second-order central finite difference (CD) and the fourth-order Runge–Kutta (RK) time discretizations.

W-cycle [37,38] is utilized as a multigrid iteration until the residual norm of the solution is smaller than 10^{-6} .

7 Numerical examples

In what follows, we show numerical results in order to validate the controllability method discussed in previous chapters and to demonstrate some properties of the proposed algorithm. The simulation results are presented for several geometries. All of these geometries are truncated by the absorbing boundary, and the computational domain is divided into square-elements, each having a side length h . For each element order r , we construct meshes, which are matching on the interface Γ_i between the domains. The iterative process set by Algorithm 1 is continued until the convergence requirement with $\varepsilon = 10^{-5}$ is reached. The practical realization of the algorithm is implemented in Fortran 95/2003, and numerical experiments are carried out on an Intel Xeon E7-8837 processor at 2.67 GHz.

7.1 Accuracy of the time discretization

In the first example, we show how the accuracy of the numerical solution of the state equation, as a part of the control algorithm, is improved as higher-order Runge–Kutta (RK) time discretization is applied. That is, we solve state equation (34) and compare the accuracy with the results obtained by solving the corresponding equation discretized in time domain by the second-order central finite difference (CD) method (see, [10,11]).

The spatial domain consists of two parts, Ω_s and Ω_f , such that $\Omega_s = [-1, 0] \times$

Table 1

Mesh stepsizes, number of time steps, and number of degrees of freedom for different spectral orders with $\omega = 4\pi$.

	r	1	2	3
Mesh stepsize		1/20	1/10	1/7
Number of time steps		100	100	130
Number of degrees of freedom		1323	1323	1452

$[0, 1]$ and $\Omega_f = [0, 1] \times [0, 1]$ (see Figure 1). We use square-element meshes with mesh stepsize $h = 1/20$. The coupling interface Γ_i is set at $x_1 = 0$ for $x_2 \in [0, 1]$, and on the other boundaries we set the absorbing boundary conditions. The material parameters in the fluid domain are $\rho_f(\mathbf{x}) = 1.0$ and $c(\mathbf{x}) = 1.0$. In the structure domain, we use the values $\lambda = 51.1$, $\mu = 26.3$, and $\rho_s(\mathbf{x}) = 2.7$. The angular frequency ω , satisfying $\omega T = 2\pi$, is $\omega = 4\pi$ for both media. Further, we set the propagation direction $(1, 0)$ by the vector such that $\boldsymbol{\omega} = (\omega_1, \omega_2) = (1, 0)\omega$.

The right-hand sides and initial conditions in Equations (2)-(13) are defined to satisfy the analytical solution $\phi = -\sin(\boldsymbol{\omega} \cdot \mathbf{x}) \sin(\omega t)$, and $\mathbf{u}_s = (\cos(\boldsymbol{\omega} \cdot \mathbf{x}/c_p(\mathbf{x})) \cos(\omega t), \cos(\boldsymbol{\omega} \cdot \mathbf{x}/c_s(\mathbf{x})) \cos(\omega t))^T$. The time domain $[0, T]$ is divided into N time steps, each of size $\Delta t = T/N$. Several numbers of time steps between $N = 100$ and $N = 3000$ are tested. This time step refinement gives a series of numerical results with various lengths of the time step for each element order r .

The maximum errors compared to analytical formulas, computed as L^∞ -norms after one time period, are reported with respect to $\Delta t/h$ in Figure 2. The solution for each element order r with the RK time-stepping is at least as accurate as the one computed with the CD time-stepping. It is also seen that for higher-order space discretization combined by sufficiently sparse time discretization the RK time-stepping provides better accuracy than the CD approach.

7.2 Comparison between symmetric and non-symmetric approaches

In this section, we consider the convergence rate of the iterative process set by Algorithm 1. The right-hand side source functions, the material parameters, and the meshes are the same as the ones used in the previous example. The number of time steps N is chosen for each element order as reported in Table 1.

Further, we show the key difference of the performance between the symmetric approach presented in this paper and the non-symmetric approach presented

in [12]. The right-hand side source functions in the fluid domain for the non-symmetric approach are defined to satisfy $p_f = \omega \rho_f(\mathbf{x}) \sin(\boldsymbol{\omega} \cdot \mathbf{x}) \cos(\omega t)$ for the pressure variable (see, [12]), while the other variables and parameters remain the same as for the symmetric formulation. From Figure 3, we can see that the number of iterations is two orders of magnitude smaller with the symmetric approach presented in this paper than with the non-symmetric approach presented in [12].

In Figure 4, we present the convergence histories for solving the symmetric problem and the corresponding non-symmetric problem with $r = 1$. In the case of the non-symmetric problem, the convergence rate is remarkably slow implying computational inefficiency. From Figure 5 we can further see that also the value of the objective functional is several orders of magnitude smaller in the case of the symmetric formulation. The objective functional, that we have used for the both formulations, can be derived as a natural energy functional for the symmetric formulation. That accounts for the improvement made by replacing the non-symmetric formulation with the symmetric one. Thus, we can conclude that choosing a proper functional for minimization has a crucial role in the efficiency of the method.

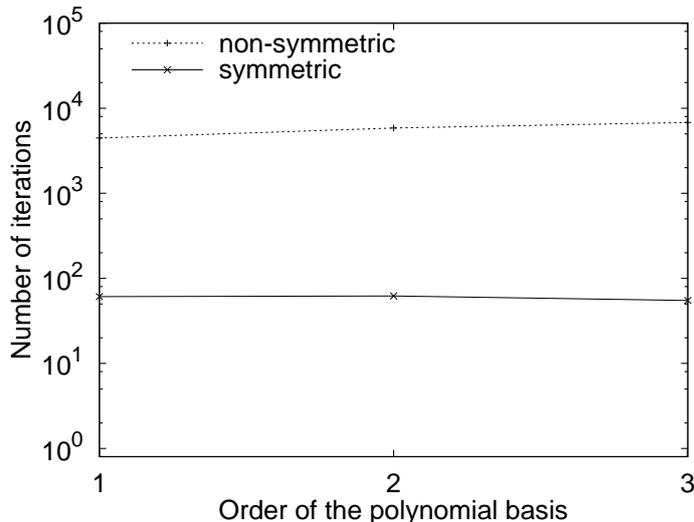


Figure 3. Number of iterations with respect to element order.

7.3 Accuracy of the control approach

We test the accuracy of the control method in a domain consisting of two parts, Ω_s and Ω_f , such that $\Omega_s = [-1, 0] \times [0, 1]$ and $\Omega_f = [0, 1] \times [0, 1]$ (see Figure 1). We use square-element meshes with mesh stepsize h . The coupling interface Γ_i is set at $x_1 = 0$ for $x_2 \in [0, 1]$, and on the other boundaries we set the absorbing boundary conditions. We solve a time-harmonic problem, satisfying the analytical solution

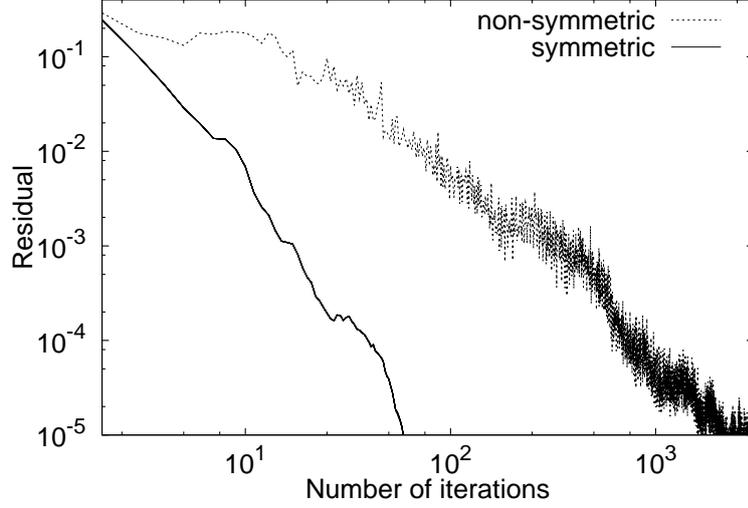


Figure 4. Convergence histories of the relative euclidean norm of the residual with respect to the number of iterations for the non-symmetric and the symmetric approach.

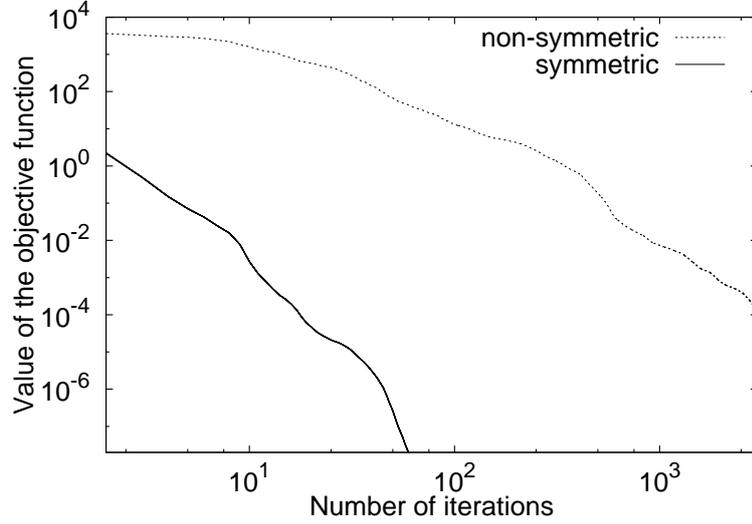


Figure 5. Values of the functional with respect to the number of iterations for the non-symmetric and the symmetric approach.

$$\mathbf{e}^* = \begin{pmatrix} (1+i)(\sin(\boldsymbol{\omega} \cdot \mathbf{x}) - \cos(\boldsymbol{\omega} \cdot \mathbf{x})) \\ (1-i)\left(\sin\left(\frac{\boldsymbol{\omega} \cdot \mathbf{x}}{c_p}\right) + \cos\left(\frac{\boldsymbol{\omega} \cdot \mathbf{x}}{\lambda+2\mu}\right)\right) \\ 0 \end{pmatrix}, \quad (72)$$

comparable to the numerical solution $\hat{\mathbf{e}}^*$ solved by Algorithm 1. Accordingly, for the systems (2)-(9) and (54)-(61), we use the source functions $f_\phi = 0$, $y_{\phi_{\text{ext}}} = ((w + c\boldsymbol{\omega} \cdot \mathbf{n}_f) \cos(\omega t + \boldsymbol{\omega} \cdot \mathbf{x}) + (w - c\boldsymbol{\omega} \cdot \mathbf{n}_f) \sin(\omega t - \boldsymbol{\omega} \cdot \mathbf{x})) / c$, $\mathbf{f} =$

$\mathbf{0}$, and $\mathbf{g}_{\text{ext}} = (\omega (n_{s1}(n_{s1} + 1) \cos(\omega t + \boldsymbol{\omega} \cdot \mathbf{x}) - (n_{s1} - 1)n_{s1} \sin(\omega t - \boldsymbol{\omega} \cdot \mathbf{x})), 0)^T$, and the material parameters, $\rho_f(\mathbf{x}) = 1$, $c(\mathbf{x}) = 1$, $c_p(\mathbf{x}) = 1$, $c_s(\mathbf{x}) = 1/2$, $\rho_s(\mathbf{x}) = 1$, $\mu = 1/4$, $\lambda = 1/2$. Furthermore, we set the propagation direction $(1, 0)$ by the vector $\boldsymbol{\omega} = (\omega_1, \omega_2) = (1, 0)\omega$.

We set the time period $T = 2\pi/\omega$ by using angular frequency $\omega = 4\pi$ in both domains. The number of time steps, each of size $\Delta t = T/N$, is chosen to be $N = 180$. We carry out spectral basis order refinement (r -refinement) corresponding to the SEM discretization by using the mesh with $h = 1/10$, and increasing the order of the spectral basis r from 1 to 5. For comparison, we perform the mesh step refinement (h -refinement) corresponding to the classical FEM discretization with linear elements. For this purpose, we construct a hierarchy of quadrilateral element meshes with smaller and smaller element sizes by dividing the mesh stepsizes of each element of the mesh with $h = 1/10$ to 2, 3, 4 and 5 mesh stepsizes of equal length. Thus, we use the meshes with $h = \{1/10, 1/20, 1/30, 1/40, 1/50\}$ to get results with the same numbers of degrees of freedom as in the test with r -refinement. Since the number of optimization variables is twice the number of degrees of freedom, the density of the spatial discretization is essential for the computational efficiency. The difference between the control algorithm solution $\hat{\mathbf{e}}^*$ and the analytical solution \mathbf{e}^* is measured in domains Ω_f and Ω_s using the L^∞ -norm. The results are presented, with respect to the number of optimization variables, in Figure 6.

The comparison between numerical and analytical solution shows that in both media the accuracy improves when the element order grows until a certain error level is reached. This error level represents the level of the dominating error source, which is caused by some factor other than the spatial discretization.

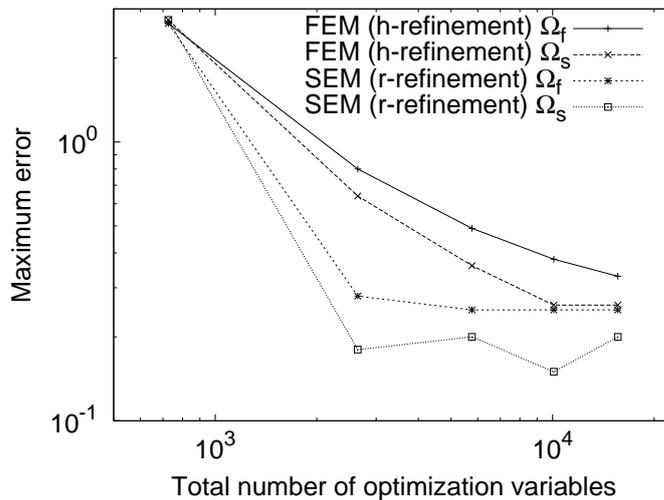


Figure 6. Maximum error for h -refinement and r -refinement in domains Ω_f and Ω_s computed as L^∞ -norms versus number of optimization variables. The number of time steps is fixed to be $N = 180$.

The error becomes smaller also with mesh step refinement, but the convergence rate is higher for r -refinement than h -refinement. Slightly larger errors are observed in domain Ω_f than in domain Ω_s for both r -refinement and h -refinement. With $r = 2$, we obtain in the both domains a level of accuracy that is better than with the densest h -refinement in this test. On the other hand, the computational cost seems to depend linearly on the number of optimization variables for the both refinements (see Figure 7). Based on these results, it seems clear that, instead of refining the mesh with bilinear elements, it is better to increase the order of the basis to improve the computational efficiency. As the order of element increases in the SEM space discretization, also the memory consumption increases (see Figure 8). This, along with the fact that the fourth order Runge-Kutta method is not very efficient in temporal discretization for the element orders higher than $r = 4$, makes the element orders $r = 2, 3, 4$ the most feasible choices for this kind of simulations.

In the last computational experiment, we test if the accuracy is maintained when higher frequencies are involved. We solve the problem with angular frequencies $\omega = \{4\pi, 8\pi, 16\pi, 32\pi, 64\pi, 128\pi\}$ by increasing the element order r in both parts of the domain from 1 to 5. In these experiments, the resolution of the spatial discretization per acoustic wavelength, that is, the number of degrees of freedom, is $r2\pi/h\omega \approx 10$. The time interval $T = 2\pi/\omega$ is divided into $N = 5(r+2)$ steps, each of equal size, to guarantee the stability condition for each element order r . The errors between the control algorithm solution and the analytical solution in domain Ω_f is presented, as L^∞ -norm, with respect to angular frequency in Figure 9. It is shown that for high frequencies better accuracy is achieved when higher-order elements are used.

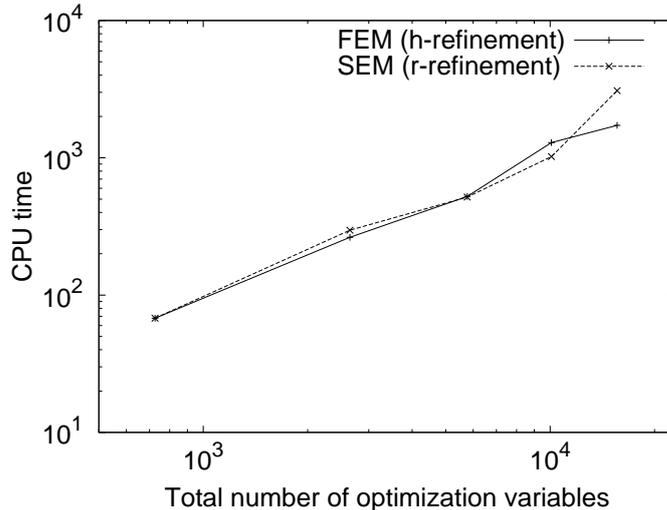


Figure 7. CPU time in seconds consumed for solving the problem formulated by the FEM space discretization with different mesh densities and by the SEM space discretization with different element orders. The number of time steps is fixed to be $N = 180$.

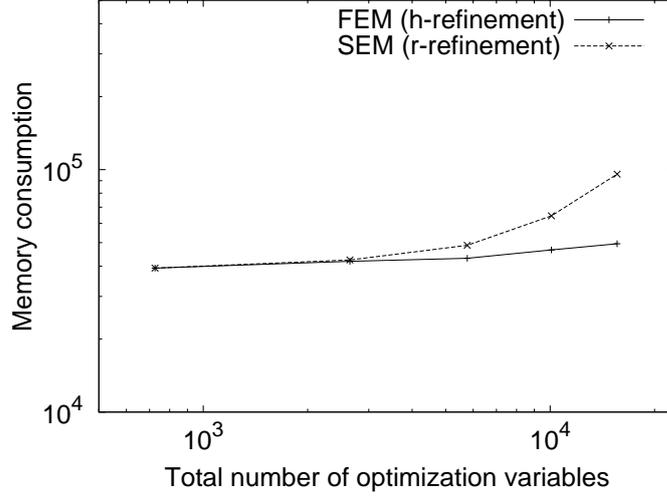


Figure 8. Memory requirement, in kilobytes, for solving the problem formulated by the FEM space discretization with different mesh densities and by the SEM space discretization with different element orders. The number of time steps is fixed to be $N = 180$.

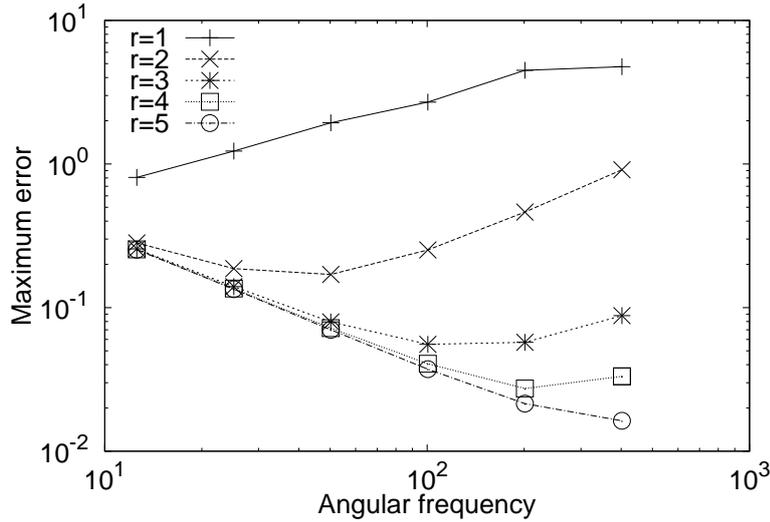


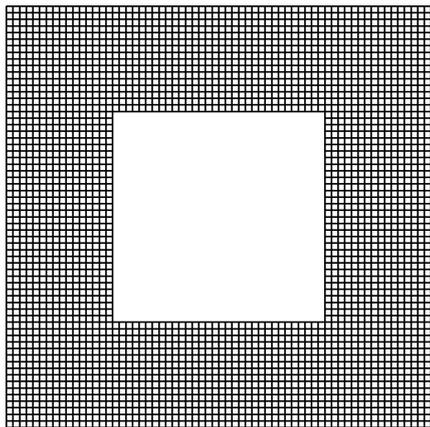
Figure 9. Errors, observed in domain Ω_f , with respect to angular frequency.

7.4 Scattering experiments

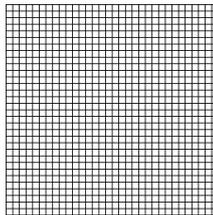
In this section, we compare the simulation results with acoustic scattering by a sound-soft obstacle with scattering by an elastic obstacle. If the scatterer is assumed to be sound-soft, the obstacles are surrounded by the boundary Γ_{of} . When an elastic obstacle is involved, the obstacles defining the scatterer Ω_s are surrounded by the boundary Γ_i . The obstacles are centered in the computational domain Ω and located at the perpendicular distance of 1 from the boundary Γ_{ef} truncating the domain.

In these experiments, we use the angular frequency $\omega = 4\pi$, which implies that the artificial boundary is located at distance of two wavelengths from the scatterer. We consider the acoustic wave equation with an incident plane wave implying $y_{\phi_{\text{ext}}} = (\omega - \boldsymbol{\omega} \cdot \mathbf{n}_s) \sin(\boldsymbol{\omega} \cdot \mathbf{x} - \omega t)$. Furthermore, there are no other source terms in the systems (2)-(9) and (54)-(61), that is, $f_\phi = 0$, $\mathbf{f} = \mathbf{0}$, and $\mathbf{g}_{\text{ext}} = \mathbf{0}$. Since the velocity is higher in the structure medium than in the fluid medium, we need to use more time steps to satisfy the stability conditions when the elastic obstacle is considered (see Table 2). The propagation direction is chosen to be $\boldsymbol{\omega} = \omega \left(-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right)$. We have set densities $\rho_f(\mathbf{x}) = 1$ and $\rho_s(\mathbf{x}) = 2.7$ and the propagation speeds $c(\mathbf{x}) = 1$, $c_p(\mathbf{x}) = 5.95$ and $c_s(\mathbf{x}) = 3.12$.

The domain Ω with a square obstacle with side length 2 was defined such that the surrounding boundary Γ_{ef} coincided with the border of the square $[0, 4] \times [0, 4]$. In Figure 10, we present an example of the meshes for the fluid domain and for the elastic obstacle with element order $r = 1$ and mesh step-size $h = 1/16$. For these geometries, the number of degrees of freedom is $3 \left(\left(\frac{2r}{h} + 1 \right)^2 - 1 \right)$ in the fluid domain and $2 \left(\frac{2r}{h} + 1 \right)^2$ in the structure domain. The simulation results with $r = 3$ and $h = 1/28$ are seen in Figure 11. In this test, the number of degrees of freedom is 85680 in the fluid domain and 57122 in the structure domain.

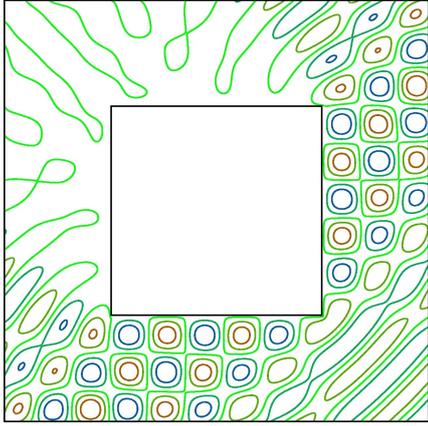


(a) Mesh for the fluid domain.

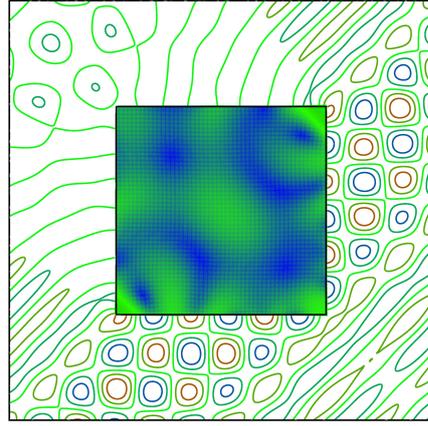


(b) Mesh for the elastic obstacle.

Figure 10. Matching meshes for the computational domains with $r = 1$ and $h = 1/16$.

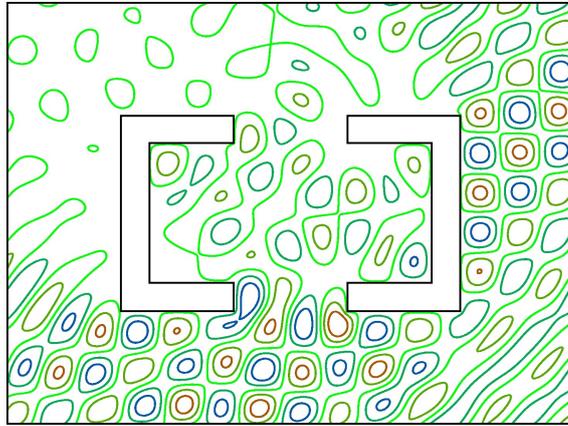


(a) Sound-soft convex obstacle.

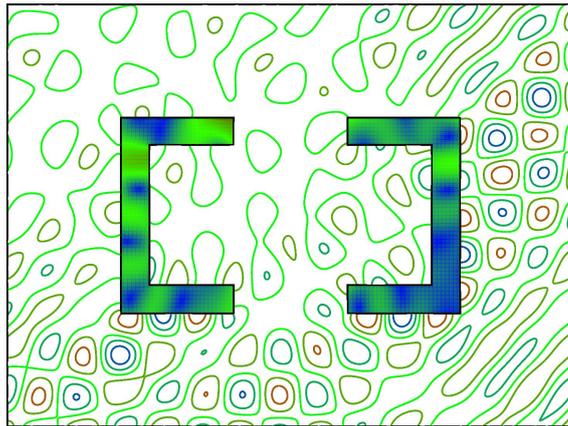


(b) Elastic convex obstacle.

Figure 11. Scattering by a convex obstacle with $r = 3$ and $h = 1/28$.



(a) Sound-soft non-convex semi-open obstacles.



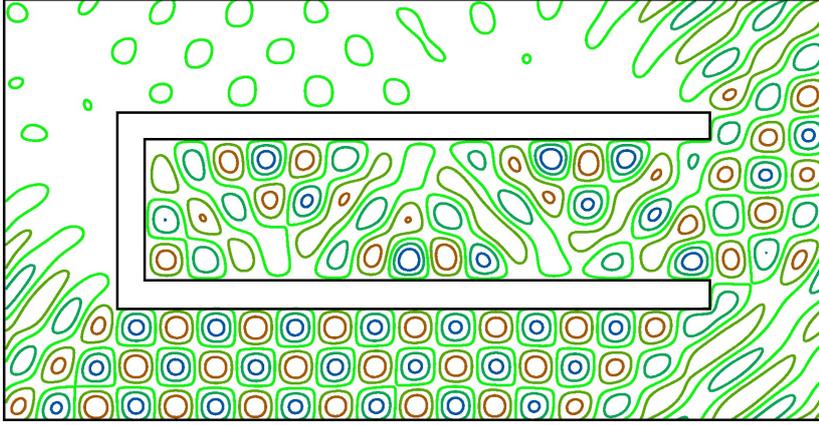
(b) Elastic non-convex semi-open obstacles.

Figure 12. Scattering by a system of two non-convex semi-open obstacles with $r = 3$ and $h = 1/28$.

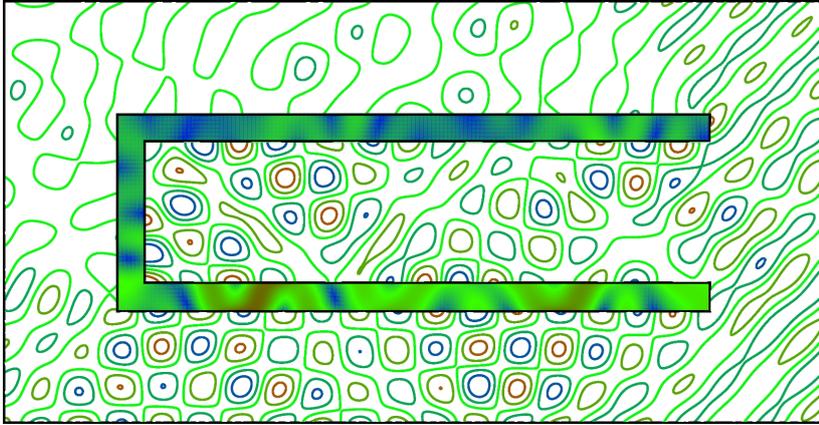
Table 2

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

r		1	2	3	4	5
Mesh stepsize h		1/80	1/40	1/28	1/20	1/16
Number of time steps N	sound-soft obstacle	60	100	140	150	150
	elastic obstacle	300	360	480	540	600



(a) Sound-soft non-convex semi-open obstacle.



(b) Elastic non-convex semi-open obstacle.

Figure 13. Scattering by a non-convex semi-open obstacle with $r = 3$ and $h = 1/28$.

Scattering by two semi-open obstacles (see Figure 12) is solved in a domain with the artificial boundary Γ_{ef} coinciding with the perimeter of the rectangle $[0, 5] \times [0, 4]$. The internal width and the height of each obstacle is $3/4$ and $5/4$. The thickness of the wall is $1/4$, and the distance between the obstacles is 1. The number of degrees of freedom is $\frac{137}{8} \frac{r^2}{h^2} + \frac{63}{4} \frac{r}{h} - 1$ in the fluid domain and $\frac{13}{4} \frac{r^2}{h^2} + 14 \frac{r}{h} + 4$ in the structure domain. The simulation results with $r = 3$

Table 3

The number of iterations of the preconditioned CG algorithm with different scatterers.

Geometry and type of the obstacle		Element order r				
		1	2	3	4	5
convex obstacle (square)	sound-soft	59	75	74	76	75
	elastic	111	115	107	107	109
non-convex semi-open obstacle	sound-soft	211	300	301	300	299
	elastic	189	188	188	188	188
two non-convex semi-open obstacles	sound-soft	123	146	145	145	145
	elastic	133	165	163	144	164

and $h = 1/28$ are presented in Figure 12. The number of degrees of freedom is 122156 in the fluid domain and 24112 in the structure domain.

In simulations with one non-convex semi-open obstacle (see Figure 13), the shape of the object resembles a tuning fork. 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 $(7.25, 3.75)$. The internal width and the height of the obstacle are 5 and $5/4$, and the thickness of the wall is $1/4$. The number of degrees of freedom is $\frac{97}{4} \frac{r^2}{h^2} + 23 \frac{r}{h}$ in the fluid domain and $\frac{47}{8} \frac{r^2}{h^2} + 24 \frac{r}{h} + 2$ in the structure domain. The simulation results with $r = 3$ and $h = 1/28$ presented in Figure 13. The number of degrees of freedom is 173040 in the fluid domain and 43472 in the structure domain.

As we can see from the results presented in Table 3, the number of iterations is substantially smaller in the case of convex square scatterer than in the cases of non-convex scatterers. Further, in all the experiments it appears that preconditioning keeps the number of CG iterations bounded with respect to r . Numerical solutions with $r = 3$ are illustrated in Figures 11-13. The acoustic part of the solution is presented as the contour lines of the real part of the velocity potential in the fluid domain, whereas the elastic part of the time-harmonic solution is demonstrated as a combination of the real part of the time-harmonic displacement vectors and the displacement amplitudes, expressing the euclidean norm of the real part of the displacement in the structure domain. As we can see, there are differences in the wave motion, especially on the backside of the scatterer, depending on whether the scatterer is modeled as a sound-soft or an elastic obstacle.

8 Conclusions

Solving the time-harmonic problems was accomplished by using the time-dependent equation and utilizing the exact controllability method by following the idea of Bristeau, Glowinski, and P eriaux. That is, complex-valued indefinite linear systems were not involved. The main idea of the method was to find such initial conditions that after one time period the solution and its time derivative coincide with the initial conditions.

We reformulated the controllability problem as a least-squares optimization problem and used a preconditioned conjugate gradient algorithm for solving the time-harmonic problem via transient equations. First, we discretized the wave equation and the objective function. Then, we computed the gradient directly for the discretized problem by following the adjoint equation technique.

We concentrated on the symmetric system of equations coupling the displacement in the structure domain with the velocity potential in the fluid domain, and applied the corresponding natural energy formulation for an objective function to be minimized. Significant improvements are obtained by using the symmetric approach compared to the non-symmetric one. Thus, the objective function is shown to play a key role in the efficiency of the method.

The simulation results show that the number of iterations required to attain the stopping criterion is independent of the element order. A certain difference between the scattering simulations of sound-soft and elastic obstacles was discovered. Hence, accurate results can not be achieved if elastic obstacles are approximated as sound-soft scatterers in the fluid-structure simulations.

The validation of the accuracy of the control approach is done by comparing the results with a known analytical solutions. The accuracy of the spatial discretization is shown to increase with the element order until an error factor, such as time discretization or a stopping criterion, disturbs the approach. The computational effort of the method seems to have linear dependence on the number of optimization variables. For high-frequency problems, better accuracy is achieved when higher-order elements are used.

Acknowledgements

The author is thankful to Dr. Jari Toivanen and Prof. Tuomo Rossi for useful comments. The research was supported by the Academy of Finland, grant 252549.

References

- [1] J. L. Lions, Exact controllability, stabilization and perturbations for distributed systems, *SIAM Review* 30 (1) (1988) 1–68.
- [2] E. Zuazua, Propagation, observation, and control of waves approximated by finite difference methods, *SIAM Review* 47 (2) (2005) 197–243.
- [3] M. O. Bristeau, R. Glowinski, J. Périaux, Using exact controllability to solve the Helmholtz equation at high wave numbers, in: R. Kleinman, T. Angell, D. Colton, F. Santosa, I. Stakgold (Eds.), *Mathematical and Numerical Aspects of Wave Propagation*, SIAM, Philadelphia, Pennsylvania, 1993, pp. 113–127.
- [4] M. O. Bristeau, R. Glowinski, J. Périaux, Exact controllability to solve the Helmholtz equation with absorbing boundary conditions, in: M. Křížek, P. Neittaanmäki, R. Stenberg (Eds.), *Finite Element Methods: Fifty Years of the Courant Elements*, Marcel Dekker, New York, N.Y., 1994, pp. 75–93.
- [5] M. O. Bristeau, R. Glowinski, J. Périaux, On the numerical solution of the Helmholtz equation at large wave numbers using exact controllability methods. Application to scattering, *Contemporary Mathematics* 157 (1994) 399–419.
- [6] M. O. Bristeau, R. Glowinski, J. Périaux, Controllability methods for the computation of time-periodic solutions; application to scattering, *Journal of Computational Physics* 147 (2) (1998) 265–292.
- [7] R. Glowinski, J. L. Lions, Exact and approximate controllability for distributed parameter systems (II), *Acta Numerica* (1995) 159–333.
- [8] M. O. Bristeau, R. Glowinski, J. Périaux, T. Rossi, 3D harmonic Maxwell solutions on vector and parallel computers using controllability and finite element methods, *Tech. Rep. RR-3607*, INRIA (1999).
- [9] R. Glowinski, T. Rossi, A mixed formulation and exact controllability approach for the computation of the periodic solutions of the scalar wave equation. (i) Controllability problem formulation and related iterative solution, *C. R. Math. Acad. Sci. Paris* 343 (7) (2006) 493–498.
- [10] E. Heikkola, S. Mönkölä, A. Pennanen, T. Rossi, Controllability method for the Helmholtz equation with higher-order discretizations, *Journal of Computational Physics* 225 (2) (2007) 1553–1576.
- [11] S. Mönkölä, E. Heikkola, A. Pennanen, T. Rossi, Time-harmonic elasticity with controllability and higher order discretization methods, *Journal of Computational Physics* 227 (11) (2008) 5513–5534.
- [12] S. Mönkölä, Time-harmonic solution for acousto-elastic interaction with controllability and spectral elements, *Journal of Computational and Applied Mathematics* 234 (6) (2010) 1904–1911.

- [13] S. Kähkönen, R. Glowinski, T. Rossi, R. Mäkinen, Solution of time-periodic wave equation using mixed finite-elements and controllability techniques, *Journal of Computational Acoustics* 19 (4) (2011) 335–352.
- [14] D. Pauly, T. Rossi, Theoretical considerations on the computation of generalized time-periodic waves, *Advances in Mathematical Sciences and Applications* 21 (1) (2011) 105–131.
- [15] M. R. Hestenes, E. Stiefel, Methods of conjugate gradients for solving linear systems, *Journal of Research of the National Bureau of Standards* 49 (6) (1952) 409–436.
- [16] G. Mur, The finite-element modeling of three-dimensional electromagnetic fields using edge and nodal elements, *IEEE Transactions on Antennas and Propagation* 41 (7) (1993) 948–953.
- [17] J. Martikainen, A. Pennanen, 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).
- [18] O. Andrianarison, R. Ohayon, Compressibility and gravity effects in internal fluid-structure vibrations: Basic equations and appropriate variational formulations, *Computer Methods in Applied Mechanics and Engineering* 195 (17-18) (2006) 1958–1972, *fluid-Structure Interaction*.
- [19] D. Komatitsch, J. Tromp, Spectral-element simulations of global seismic wave propagation II: 3-d models, oceans, rotation, and gravity, *Geophysical Journal International* 150 (2002) 303–318.
- [20] G. C. Everstine, Finite element formulations of structural acoustics problems, *Computers & Structures* 65 (3) (1997) 307–321.
- [21] A. Bermúdez, P. Gamallo, L. Hervella-Nieto, E. Rodríguez, Finite element analysis of pressure formulation of the elastoacoustic problem, *Numerische Mathematik* 95 (1) (2003) 29–51.
- [22] D. Komatitsch, C. Barnes, J. Tromp, Wave propagation near a fluid-solid interface: A spectral-element approach, *Geophysics* 65 (2) (2000) 623–631.
- [23] J. Mandel, M. O. Popa, Iterative solvers for coupled fluid-solid scattering, *Applied Numerical Mathematics* 54 (2) (2005) 194–207.
- [24] X. Wang, K.-J. Bathe, Displacement/pressure based mixed finite element formulations for acoustic fluid-structure interaction problems, *International Journal for Numerical Methods in Engineering* 40 (11) (1997) 2001–2017.
- [25] E. Chaljub, Y. Capdeville, J.-P. Vilotte, Solving elastodynamics in a fluid-solid heterogeneous sphere: a parallel spectral element approximation on non-conforming grids, *Journal of Computational Physics* 187 (2) (2003) 457–491.

- [26] B. Flemisch, M. Kaltenbacher, B. I. Wohlmuth, Elasto-acoustic and acoustic-acoustic coupling on non-matching grids, *International Journal for Numerical Methods in Engineering* 67 (13) (2006) 1791–1810.
- [27] L. G. Olson, K.-J. Bathe, Analysis of fluid-structure interactions. A direct symmetric coupled formulation based on the fluid velocity potential, *Computers & Structures* 21 (1985) 21–32.
- [28] L. L. Thompson, P. M. Pinsky, A space-time finite element method for structural acoustics in infinite domains, part I: Formulation, stability and convergence, *Computer Methods in Applied Mechanics and Engineering* 132 (1996) 195–227.
- [29] B. Engquist, A. Majda, Radiation boundary conditions for acoustic and elastic wave calculations, *Communications on Pure and Applied Mathematics* 32 (1979) 313–357.
- [30] A. Quarteroni, T. Tagliani, E. Zampieri, Generalized Galerkin approximations of elastic waves with absorbing boundary conditions, *Computer Methods in Applied Mechanics and Engineering* 163 (1998) 323–341.
- [31] G. Cohen, *Higher-Order Numerical Methods for Transient Wave Equations*, Springer-Verlag, Berlin, 2001.
- [32] G. Cohen, M. Duruffé, Mixed spectral elements for the Helmholtz equation, in: G. Cohen, E. Heikkola, P. Joly, P. Neittaanmäki (Eds.), *Mathematical and Numerical Aspects of Wave Propagation, Proceedings of WAVES 2003*, Springer-Verlag, Jyväskylä, Finland, 2003, pp. 743–748.
- [33] G. Cohen, S. Fauqueux, Mixed spectral finite elements for the linear elasticity system in unbounded domains, *SIAM Journal on Scientific Computing* 26 (3) (2005) 844–863.
- [34] E. Heikkola, S. Mönkölä, A. Pennanen, T. Rossi, Controllability method for acoustic scattering with spectral elements, *Journal of Computational and Applied Mathematics* 204 (2) (2007) 344–355.
- [35] J. L. Lions, *Optimal Control of Systems Governed by Partial Differential Equations*, Springer-Verlag, Berlin, 1971.
- [36] G. H. Golub, C. F. van Loan, *Matrix computations*, John Hopkins University Press, 1996.
- [37] W. Hackbusch, *Multigrid Methods and Applications*, Springer-Verlag, Berlin, Germany, 1985.
- [38] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd edition, SIAM, Philadelphia, 2003.