ABSTRACT

Mönkölä, Sanna
Numerical Simulation of Fluid-Structure Interaction Between Acoustic and Elastic Waves
Jyväskylä: University of Jyväskylä, 2011, 136 p.
(Jyväskylä Studies in Computing
ISSN 1456-5390; 133)
ISBN 978-951-39-4438-4 (nid.)
ISBN 978-951-39-4439-1 (PDF)
Finnish summary
Diss.

This study considers developing numerical solution techniques for the computer simulations of the fluid-structure interaction. The focus is especially on the efficiency of the iterative methods based on exact controllability and spectral element methods. In particular, the thesis concentrates on the coupling between two linear wave equations: the scalar-valued equation concerning the propagation of acoustic waves and the vector-valued equation modeling the propagation of waves in an elastic medium. These fundamental equations occur in a number of physical applications, such as acoustics, medical ultrasonics, and geophysics.

We consider both transient and time-harmonic problems. Traditionally, the complex-valued time-harmonic equations and low-order finite elements are used for solving the time-harmonic problems. This leads to large-scale indefinite systems, for which it is challenging to develop efficient iterative solution methods. Taking account of these difficulties, we turn to time-dependent equations. It is known that time-dependent equations can be simulated with respect to time until a time-harmonic solution is reached, but the approach suffers from poor convergence. Thus, we accelerate the convergence rate by employing the exact 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.

The accuracy of the method is improved by utilizing higher-order spectral elements for spatial discretization. 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. This leads to high accuracy and diagonal mass matrices, thus providing computational efficiency when combined with an explicit time-stepping scheme.

The software implementation of the methods is done side by side with the method development. Problems related to the fluid-structure interaction between elastic materials and acoustic waves are analyzed and solved by computer simulations, which are efficient tools in the testing and optimizing of model parameters. For instance, in planning underwater structures, the design process can be improved and the development cycle shortened with computer aided modeling.

Keywords: exact controllability, spectral element method, coupled problem, numerical simulation, fluid-structure interaction, acoustic, elastic, wave equation

CONTENTS

PREFACE

The collapse of the Tacoma Narrows bridge in 1940 is a classic example of what may happen if the fluid-structure interaction is neglected in the design process. Traditionally, these kind of hazardous effects have been taken into account in building and manufacturing processes by using experimental tests. For instance, wind tunnel tests are comparatively expensive and require experimental efforts in contrast to computer simulations. This motivates using the computer simulations which are safe, fast, and cost-efficient when testing and optimizing the design with respect to certain properties of products or structures.

In general, the purpose of computer simulations is to use implementations of the numerical solution methods to imitate real-life phenomena. This can be accomplished by presenting a physical phenomenon as a mathematical model which is simulated by using the computer implementation of numerical solution methods (see Figure 1). Traditionally, mathematical models and computer simulations are used for solving the problems having a physical, biological or chemical background, but the application area is not restricted to these sciences. Computer simulations can be applied for both training and entertainment purposes. Surgeons are practicing with surgery simulators and pilots use flight simulators. Computer and video games, as well as special effects in movies, are nowadays based on computer simulations. Furthermore, computer simulations are useful in many industrial processes for testing, safety engineering, and performance optimization. Accordingly, using the computer simulations for industrial applications have effects also on national economy. Modeling natural or human systems can be realized by computer simulations to gain insight into the functioning of the systems. Weather forecasts and prognoses for climate change are actually results of computer simulations with certain input values. The related schemes can also be applied to show the effects of alternative conditions and courses of action in, for instance, financial and medical systems and social sciences. To get reliable simulation outputs, appropriate models, relevant methods, and valid input

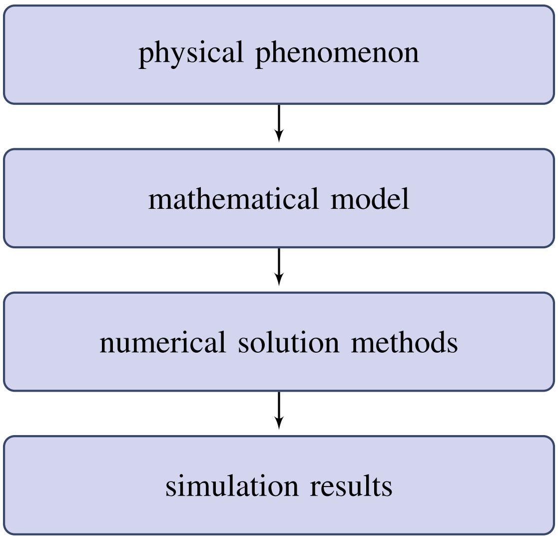


FIGURE 1 The physical phenomenon is presented as a mathematical model which is simulated by using the computer implementation of numerical solution methods.

information has to be defined carefully.

The development of numerical computing began actually before the advance of modern computers. First, research was concentrated on modeling the phenomena and the mathematical analysis of the models. Already at the beginning of this road, it was clear that closed-form analytical solutions, computed by ”pen and paper”, are possible for only a few simplifications of real-life problems. That is why numerical methods are needed. The development of numerical solution methods and the related algorithms enables us to consider new approaches and has the ability to offer more gain than the addition of simple raw computing power. The modern and efficient methods offer advantages over the earlier numerical approaches. Thus, larger and more demanding real-life problems, including more realistic models arising from coupled systems of several media, can be solved. Further, industrial applications are often considered in rather large and complex geometries, and solving the problem may demand a considerable amount of computing time and memory. Consequently, the method development is also a step towards better automatization in digital product development including design for manufacturing processes and numerical optimization for improving the product properties. On the other hand, the development of computing resources accelerates testing and verifying the robustness and accuracy of the new methods. This, in its turn, evolves the research of numerical mathematics. Hence, progress in the research of mathematical theory is feeding the method development, and vice versa.

Commercially available simulation software products are typically not based on most recent innovations in numerical solution or software engineering technology. Before applying to the commercial products, the methods are tested to be robust and their properties have to be well known. The algorithms are originally focused on finding solutions on the computers available at that time. Recently, the development of computing resources has been fast due to the needs of home entertainment and more realistic game environments. Public needs have driven the price down. While more resources have been available with the same price as earlier, it has been possible to improve the accuracy of the numerical simulations or solve more demanding problems. Thus, the problems which a few decades ago were computationally demanding are solved significantly faster with today’s computers. Nevertheless, increasing the number of available CPUs and memory is not enough as the size of the problem does not scale at the same pace as the available resources.

The design process – as well as research and development – can be improved and the development cycle shortened with computer aided modeling. Still, the experimental tests form a remarkable stage in today’s manufacturing processes. Simplifying approximations and assumptions are done within the numerical simulations, and also some experimental testing is needed to guarantee the relevant selection of key characteristics and behaviors applied in the mathematical models. Furthermore, defining material parameters is still a challenging task in the modeling process, since the properties of some materials are not well-known. Accurate models considering, for instance, phenomena with high intensity acoustic fields, like ultrasonic cleaning, are still under development. This indicates that further research related to accurate mathematical modeling, as well as efficient numerical simulation and optimization methods, is required.

1 INTRODUCTION

Physical phenomena, such as deformation of elastic bodies, fluid flow and heat conduction, can be modeled by partial differential equations (PDE). A system consisting of reciprocal interaction between physically heterogeneous components is a coupled problem. Fluid-structure interaction (FSI) defines a coupled problem of a physical system, in which the fluid flow causes the deformation of a solid structure, and the deformation of the structure, in turn, changes the fluid flow. This mutual interaction is modeled by combining certain parts of the boundaries of the fluid and solid regions. A fluid region may surround a solid region or be embedded into a solid region. Fluid-structure models can be found in many applications on different fields of natural sciences and engineering. Typical engineering applications are, for example, the response of bridges and large buildings to the wind, the loading dynamics of airfoils in aircrafts and wind power turbines, and the fluttering of the paper web during the paper making process. Even nuclear reactor safety issues have been analyzed as fluid-structure interaction problems [18]. Applications in, for example, biological systems include blood flow in elastic vessels [121, 154, 188] and the mechanical processing of acoustic signals by the cochlea in the inner ear [40, 86].

1.1 Interaction between acoustic and elastic waves

In a general case, handling the fluid-structure interface is a challenging task. That is why we restrict ourselves to small vibrations in solids and fluids, which can be modeled by elastic and acoustic wave equations. In this case, moving coordinate systems are not needed, and still, the coupled problems of elastic and acoustic wave equations cover a remarkable field of the fluid-structure interaction problems. A number of industrial and medical applications are based on vibrations created by ultrasonic transducers [102]. The vibrations cause intense effects, such as cavitation, streaming, and heating, which are essential, for example, in ultrasound imaging, sonochemistry, cleaning and welding. In medical sciences, ultrasound imaging data can be used to model computationally the structure of human tissue [189]. The medical use of ultrasound includes also pain relief and therapeutic purposes to destroy kidney stones and cancerous tissue [149]. There are

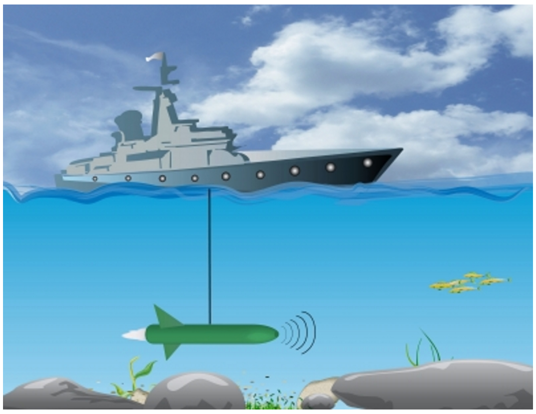


FIGURE 2 Ship for underwater research with a sonar submarine. Image: xedos4 / FreeDigitalPhotos.net.

numerous ultrasound related applications in audio technology and echo sounding (see, e.g., [138]). These kind of fluid-structure interaction simulations can be used also in applications of audio acoustics design, such as the modeling of concert halls and auditoriums, loudspeakers, noise barriers, mobile phones, and acoustic materials. There are also several noise reduction applications including car, elevator, and work machine cabins, airplanes as well as propagation of noise from engines or machines [1, 57, 131, 119]. Computational models, based on FSI, are utilized also for geophysical exploration and earthquake simulations [6, 126].

The mathematical models considering fluid-structure interaction between acoustic and elastic waves have been in focus for a long time (see, e.g., [70]). The development of the finite element models for this kind of fluid-structure interaction dates back to the 1960's, when Gladwell and Zimmermann (see [88, 90]) considered the problem in variational form. The work was further developed by Gladwell and Mason in [89], and it concentrated on applying the finite element method to the acoustic response of the bending of plates. In the early days of the development in the field of structural-acoustics, the research was concentrated on modeling the phenomena and the mathematical analysis of the models. Only simple finite element solutions were accomplished, since the computer hardware formed the bottleneck for computer simulations. For alleviating the problem, developing alternative methods and formulations came into existence.

Various formulations exist for fluid-structure interaction between acoustic and elastic waves. Typically, the displacement is solved in the elastic structure. The fluid can be modeled, for example, by using fluid pressure, displacement, velocity potential or displacement potential [67]. 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 the so-called non-symmetric formulation (see, e.g., [22, 151, 190]), while using the velocity potential results in a symmetric system of equations (see, e.g., [39, 84, 165, 186]). Most of the early presentations of structural-acoustics problems were based on the pressure presentation in the fluid domain and the displacement presentation in the solid domain. Olson and Bathe presented in [166] an alternative scheme to use both the velocity potential and pressure in the fluid region. It is also possible to utilize displacement in both domains as stated by [14]. Naturally, the models have been tailored corresponding to the applications. For instance, gravity effects are employed by Andrianarison and Ohayon in [7] and by Komatitsch and Tromp in [128].

Physical applications are often defined in unbounded, that is, exterior, domains. To solve the problems numerically, it is a standard procedure to truncate the original unbounded domain [157]. An artificial boundary condition, which is an approximation of the Sommerfeld-type radiation 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 (see, e.g., [117, 152]). Typically, the conventional first-order absorbing boundary condition is used. If more sophisticated boundary conditions and absorbing layers are needed, for instance, the second-order radiating boundary conditions introduced by Engquist and Majda [64] or a perfectly matched layer (PML) presented, e.g., by Berenger [21, 129, 108] are utilized. Other approaches for creating the artificial boundary are using infinite element methods (IFEM) [8, 24, 85] or the Dirichlet-to-Neumann mapping [157]. The non-reflecting boundaries are considered, for instance, in a review article by Givoli [87].

Computer simulation is an important component in testing and optimizing model parameters. Various FSI models and solution strategies, depending on the application area, have been developed for these purposes (see, e.g., [9, 34, 59]). The solution techniques vary with the formulation, and they are adapted to achieve efficient numerical simulations (see, e.g., [51]). Essentially, there are two strategies to solve the FSI problems. The equations governing the fluid domain and the structure domain can be solved simultaneously. This monolithic approach [15, 26, 110, 116], or direct coupling, needs implementation for the particular combination of physical problems. Another option is a segregated approach [37, 155], or staggered iterative coupling, which enables using distinct solvers for the fluid domain and the structure domain. The solution of one domain is used as a boundary condition for the other domain, and that is how the coupling information is transferred from the one domain to the other. Obviously, the segregated approach requires less memory than solving the problems simultaneously. However, coupling between the solvers has an important role in the efficiency of the method. When small oscillation amplitudes are considered, linear models and simplified solution techniques can be utilized.

1.2 Solution methods

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. Combinations of boundary and finite elements (see, e.g., [53, 198]) are typically used in three-dimensional simulations to reduce the size of the computational problem. We are especially interested in solving problems with varying material parameters. For such problems, boundary-based methods are not directly applicable, whereas domain-based methods, that is, field approaches are more flexible in this respect. Thus, we focus our attention to domain-based formulations which can be discretized, for instance, by finite difference (FDM) or Galerkin finite element methods (FEM) (see, e.g., [100, 117]). Especially the FEM approximation and the solution of time-harmonic wave equations have attracted a lot of attention during the past two decades (see, e.g., [185]).

Efficient solution methods, such as domain decomposition [19, 48, 69, 74, 75, 147], fictitious domain (domain embedding) [17, 72, 109, 112], and multigrid [28, 62, 96, 124, 187], have been developed for solving the wave equations. Various types of the domain decomposition methods (DDM) are applied for wave equations and higher-order discretizations. For instance, the Schwartz methods (see, e.g., [142, 143, 196]) are generalized to the spectral element discretization for the Helmholtz equation, for example, by Cai, Casarin, Elliott, and Widlund [35] and for elastic waves, for example, by Pavarino and Zampieri [171]. Mandel extended a modified finite element tearing and interconnecting (FETI) approach for solving the time-harmonic fluid-structure interaction problems [150]. Cummings and Feng proposed Jacobi and Gauss-Seidel domain decomposition algorithms using linear combinations of the interface conditions for the coupling between subdomains for the time-dependent fluid-structure interaction problem [52, 81]. The related methods for elastic and electromagnetic waves are presented by Bennethum and Feng [20] and Després, Roberts, and Joly [58], respectively. Recently, dual-primal FETI (FETI-DP) [73, 77, 76] and balancing domain decomposition by constraints (BDDC) preconditioners are used with spectral element space discretization by Klawonn, Pavarino and Rheinbach [125].

The early versions of iterative multilevel methods can be categorized as geometric multigrid (GMG) methods, in which the actual coarsening of the given mesh is needed for constructing the multilevel hierarchy (see, e.g., [29, 96, 187]). To attain independency of the problems, algebraic multigrid (AMG) methods, in which the geometrical background of the problem is not needed, were developed by Stüben and his associates (see, e.g., [178, 182]). Kickinger (see, e.g., [123]) developed a graph-based algebraic multigrid method, in which the efficient computation of coarse level systems is attained by using only the graph of the matrix for constructing the multilevel hierarchy. Recently, the multigrid methods have been widely used in many applications based on the wave equations. Brandt and Livshits presented a multigrid method for standing wave equations in [28]. Multigrid schemes used for solving the Helmholtz problem are considered, for instance, by Elman, Ernst, and O'Leary in [62], by Erlangga, Oosterlee, and Vuik in [65, 66], and by Kim and Kim in [124]. McCormick and Ruge applied algebraic multigrid methods to problems in computational structural mechanics in [156]. Han, Zhou, and Liu presented a multigrid method for numerical solutions of the elastic wave equation in [97]. Mandel and Popa extended the approach to the corresponding coupled fluid-structure interaction problem in [151], whereas Heil considered a multigrid approach with more general fluid-structure interaction problems in [110]. The spectral element method is considered in conjunction with the multigrid approach by Fischer and Lottes in [83] and by Heys, Manteuffel, McCormick, and Olson in [113].

A modification of Kickinger's method, presented by Martikainen, Pennanen, and Rossi in [153], stabilizes the method by eliminating the equations corresponding to Dirichlet boundary conditions. The method occupies the middle ground between geometric and algebraic multigrid. The selection of coarse grid variables has a geometric flavor, but the algebraic system is needed for constructing the multigrid cycle. With higher-order elements only the nearest neighboring nodes are considered when constructing the graph. This graph-based multigrid (GBMG) preconditioner for solving the problems arising from acoustic and elastic wave problems is considered by Airaksinen, Heikkola, Pennanen, and Toivanen in [3, 5]. The method was shown to be practical for approximating the inverse of a damped preconditioner with higher-order discretizations, especially when the frequency is not very high. The discretization mesh or matrix itself can be used for defining the graph. We have applied this approach at the preconditioning stage of the control algorithm in conjunction with acoustic and elastic waves in [107, 162], respectively. In this thesis, we extend the approach for solving a corresponding linear system of coupled equations.

When the time-harmonic problems are considered, the methods mentioned above are typically based on handling directly the complex-valued time-harmonic equations and low-order finite elements (see, e.g., [25, 95, 98]). They all lead to large-scale discrete problems with indefinite linear equations. Solving such problems with direct solution methods, like LU or Cholesky factorization, requires a significant amount of computing time and memory storage. Despite the robustness and the fixed number of computing steps, the direct methods are inadequate for solving large scale problems. In contrast to direct methods, iterative methods start from an initial guess and form successive approximations that converge to a solution within a particular tolerance. A comparatively low requirement in terms of computational resources and more practical implementation on parallel computers makes iterative solution methods an adequate choice over the direct solution methods. As the discretization of the time-harmonic wave problem leads to indefinite systems, also the road of using iterative methods, such as GMRES or Bi-CGSTAB, has its blocks. Efficient preconditioners are needed to guarantee a proper convergence speed for the iterative methods.

An alternative for solving the complex-valued time-harmonic equations is to simulate the time-dependent equation with respect to time, until the 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. The convergence rate can be accelerated by using control techniques [33].

1.3 Controllability

Control theory refers to the concepts and methods that can be used for optimal design, control and analysis of dynamical (i.e., time-dependent) systems. Systems having controllable inputs and outputs are considered. That is, a system and its state can be manipulated. The basic idea of controllability is to steer a dynamical system from an initial state to a particular state by using an appropriate control. Controllability cases can be divided into two categories, namely, exact and approximate controllability. Exact controllability enables forcing the system from a preassigned initial state to a specific final state by some control, whereas approximate controllability means that a final state belonging to an arbitrary small neighborhood of the final state can be achieved by controlling the system. Observability is closely related to controllability. It considers whether the initial state can be detected by using the measurements from the control and the final state. Propagation, observation, and control issues of wave equations are reviewed by Zuazua in [200].

Exact controllability is a well-known and extensively researched topic within classical wave equations [33]. Exact controllability can be boundary [61, 91, 164, 175], internal [101, 144, 145] 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 [68], both internal and pointwise controllability are also known as distributed controllability [141]. 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 time-harmonic problems. The main idea of this method is inspired by the Hilbert uniqueness method (HUM), which was introduced by Lions [140] as a systematic method to address controllability problems for partial differential equations.

Usually, the controlling procedure is possible to realize in several ways. Optimal control theory concerns finding the best (measured by a specific criterion) control, and it is used in several areas such as shape design, inverse problems and controllability of models described by the PDEs. A classical example of real-life optimal control problems is to find an optimal way to bring a satellite to a desired trajectory so that the minimum amount of fuel is consumed. In this thesis, we concentrate on a more academic example to find a periodic solution for a coupled wave equation. In this case, the difference between the initial condition and the terminal condition is minimized.

We follow the idea of Bristeau, Glowinski, and Périaux, presented in [30, 31, 32, 33, 92]. We avoid solving indefinite systems by returning to time-dependent equations and utilize the exact controllability approach for solving the time-harmonic problems. For this purpose, we use an algorithm which is based on control theory [140, 200]. The main idea of the algorithm is to find initial conditions such that after one time period, the solution and its time derivative coincide with the initial conditions. The controllability problem is reformulated as a least-squares optimization problem which is solved by a conjugate gradient (CG) algorithm. If an unpreconditioned CG algorithm is used, the number of iterations grows rapidly with the order of the spectral element [146]. That is why we use modern and computationally efficient methods, such as the graph-based multigrid method [153], for preconditioning the algorithm. Another essential stage of the optimization problem is computation of the gradient. We first discretize the wave equation and the function to be minimized. Then, we compute the gradient directly for the discretized problem by following the adjoint equation technique [172]. 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 [33]. Theoretical background of combining the exact controllability method with discrete exterior calculus for the generalized time-periodic wave equations is discussed in [169].

The exact controllability algorithm can be interpreted as a variation of the asymptotic approach with periodic constraints, in which the time-dependent equation is simulated in time until the time-harmonic solution is reached. In practice, the residual of the algorithm defines at each iteration how far the solution is from a periodic one. The residual is further used for control purposes to accelerate the convergence rate by giving an impulse to the system. On the other hand, the scheme can be seen as a shooting method for a transient two-point boundary value problem. This boundary value problem is replaced by a sequence of initial value problems. The initial value problem is integrated from the initial point to the other boundary. During the procedure, the initial guess is changed to hit the target, that is, the terminal condition more accurately. The iterations are continued until the desired accuracy is achieved.

1.4 Accuracy factors

The overall accuracy of the discrete solution given by the controllability method depends on many factors. In order to concentrate on the spatial discretization we choose the test problems in such a way that as many error factors as possible are eliminated. We try to isolate the effects of those error factors which we can not eliminate.

The accuracy depends, among other things, on the spatial and temporal discretization parameters, which are the mesh stepsize h, the order r of the spectral basis, and the timestep Δt. A large time step allows to compute the solution utilizing only a small amount of CPU time, but it may involve an error which deteriorates the accuracy of the method. Hence, time steps small enough are used to dissipate the temporal error.

The relative error of finite element solutions consists of two error sources, which are the approximation error and the pollution error [118]. To obtain accurate results, the discretization mesh needs to be adjusted to the wavelength. The approximation error of the hp-version [2] of finite element solutions in the H^1 -seminorm is of order $\left(\frac{\kappa h}{2r}\right)^r$, where κ is the wavenumber, h is the mesh stepsize, and r is the order of the basis functions [118]. Consequently, higher wavenumbers require finer meshes (i.e., smaller mesh step size h) to reach sufficient accuracy. Typically, the accuracy is maintained by keeping a fixed number of grid points in a wavelength $\ell(\mathbf{x}) = \frac{2\pi}{\kappa}$. Furthermore, the dimension of the system increases rapidly as the wavenumber (and frequency) increases, which makes the accurate solution even more challenging.

Another difficulty is the so-called pollution effect, that is, the wave number of the numerical solution differs from the wave number of the exact solution. The error increases as the wavenumber increases, which makes finding an accurate solution even more challenging. The pollution error is of order $\kappa\left(\frac{\kappa h}{2r}\right)^{2r}$, and it becomes the dominant source of the error at high wavenumbers. Hence, it deteriorates the accuracy when the wavenumber κ increases even if the quantity κh is kept constant (see, e.g., [117]). It is known that the pollution effect can not be avoided in two- and three-dimensional problems [12]. Thus, a fixed error level would require keeping the quantity $\kappa^2 h$ fixed, which leads to unacceptable computational costs for high frequency problems. That is why the pollution effect is an important consideration in the finite element solution of time-harmonic wave problems (see also [13, 117] and references there in).

Geometries with curved boundaries can not be represented exactly by a rectangular mesh, which also causes error. We avoid this error component by using only geometries with polygonal boundaries. Curvilinear geometries could be approximated accurately by using elements with curved edges.

The least-squares optimization problem is not solved exactly, since the CG algorithm is terminated after the given criterion is reached. This error component can be controlled by decreasing the stopping criterion of the iterative algorithm.

The approximation of the radiation condition leads to yet another error component. We eliminate this factor in the accuracy considerations by using test problems with a known analytical solution, which satisfies the absorbing boundary condition. The approximation of the radiation condition could be improved by using more sophisticated boundary conditions or absorbing layers.

1.5 More innovative discretization schemes

Several modifications of the classical finite element method (FEM) have been used to maintain the accuracy of the solution. These include such methods as the partition of unity (PUM) [10, 11], Galerkin generalized least-squares [99], discontinuous enrichment [199], discontinuous Galerkin [134, 43], and spectral collocation methods [42]. One way to decrease the pollution effect is to modify the polynomial basis of the standard FEM so that the local basis will consist of non-polynomial shape functions. This is done in the discontinuous Galerkin method [43, 71, 78, 79, 134]. Ultra weak variational formulation (UWVF) [38, 114, 115] uses standard finite element meshes and a new kind of variational formulation on the interfaces between the elements. It reduces the memory requirement compared with the standard FEM but might suffer from numerical instability. Also spectral [36, 170] and collocation methods [13] 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, by Šolín in [181]. Based on the theoretical findings, presented by Demkowicz in [54, 55], Demkowicz and Oden developed hp-adaptive methods for structural-acoustics simulations in [56]. The work was based on time-harmonic equations. Finite element discretization for displacements in the solid domain was coupled with boundary element presentation for the pressure field in the fluid region.

In both domains we specifically apply the spectral element method (SEM), which is considered by Cohen in [44], for spatial discretization. The method provides for a convenient treatment of complex geometries and varying material properties as well as high accuracy. The basis functions are higher-order Lagrange interpolation polynomials, and the nodes of these functions are placed at the 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 this choice, the spectral element discretization leads to diagonal mass matrices significantly improving the computational efficiency of the explicit time-integration used. Moreover, when using higher-order elements, the same accuracy is reached with fewer degrees of freedom than when using lower-order finite elements.

In [106], we used higher-order spectral elements for spatial discretization, derived a new way to compute the gradient of the least-squares functional, and used a multigrid method for preconditioning the conjugate gradient algorithm. The exact controllability approach was applied to acoustic scattering problems, and the central finite difference (CD) scheme was used for time discretization. This time discretization scheme is second-order accurate and with a diagonal mass matrix also fully explicit, which is an essential property for computational efficiency. Only matrix-vector products are needed in time-dependent simulation, but the scheme needs to satisfy the Courant–Friedrichs–Lewy (CFL) condition [50], which limits the length of the time step. When higher-order spectral elements are used with the second-order time discretization, the temporal error is larger than the spatial error, unless very small time steps are used. The accuracy of the method was improved in [107] by using the fourth-order accurate Runge–Kutta method. The explicitness of the method can be maintained with diagonal mass matrices, but still the method is only conditionally stable. Comparison between shifted-Laplacian preconditioning and the controllability method for computational acoustics is presented in [4]. According to the test results, the method based on exact controllability and spectral elements performs well. In [162], the method was applied to the vector-valued problem considering computational elasticity, and advances in preconditioning with the GBMG method were studied in detail within the numerical experiments. The preconditioning is shown to keeps the number of CG iterations independent of the order of spectral element basis. This confirms also that the GBMG method is suitable for higher-order discretizations. In this thesis, the methods developed for solving acoustic and elastic problems in separate domains, mainly presented in the papers [107, 162], are elaborated further and the coupling between acoustic and elastic media is applied to the model.

1.6 Contributions in the articles related to this thesis

The main ideas of this thesis are based on the earlier published methods developed for acoustic and elastic wave equations in separate domains without fluid-structure interaction. The author is the corresponding contributor in the peer-reviewed international journal articles [106, 107, 162] as well as in the peer-reviewed international conference papers [105, 159, 163] considering the topic. She has made the Fortran 90 implementation of the control method including assembling the related global level matrix and vector data structures, run the computational tests as well as analyzed the test results presented in those papers. The multigrid code realized in C programming language and provided by Janne Martikainen and Anssi Pennanen was combined to the Fortran 90 code for solving the linear system involved in preconditioning.

For the conference paper [103] considering the controllability method for the Helmholtz equation with spectral elements, the author carried out the numerical experiments. The author wrote the first version of the articles devoted to acoustic scattering [106, 107], for which the final preparation and modifications was done in co-operation with Erkki Heikkola, who also provided local level building blocks for the spectral elements used in the work. In [106], the central finite difference scheme was used for time discretization, whereas the accuracy of the method was improved by using the fourth-order accurate Runge–Kutta method in [107]. The author extended the method further to the Navier-Cauchy equation, which she described in [162]. The sections considering the multigrid preconditioner were written by Anssi Pennanen in the papers [104, 107, 161]. The other parts of the co-authored articles mentioned in this section were written by the author, except for minor modifications suggested by the co-authors at the finalizing stage. The author also prepared half of the manuscript, implemented the controllability algorithm and carried out the corresponding comparative numerical tests and conclusions presented in the article [4]. In that article, Tuomas Airaksinen performed the simulations related to the shifted-Laplacian preconditioner and wrote the related parts of the manuscript.

The research and its reporting published in the journal paper [160] was based on the ideas of the earlier research findings and carried out by the author herself. In that paper, the author considered an extension of the methods tailored for acoustic and elastic wave equations to the acousto-elastic interaction. Nevertheless, the reuse of the existing codes was not reasonable, since the only way of doing that would have been calling, at each time-step, both the solver for the scalar-valued Helmholtz equation concerning the propagation of acoustic waves and the solver for the vector-valued Navier-Cauchy equation modeling the propagation of waves in an elastic medium and updating the interface data. Therefore, a more efficient implementation, where the solution steps at each iteration are combined into one solver, was implemented. Moreover, the coupling between the earlier used paradigms, where the acoustic domain with pressure variables is coupled with the elastic domain whose variables are presented as displacements, leads to unsymmetric formulation, for which it is complicated to generate an efficient control model for two-dimensional simulations. This is what motivated searching for more meaningful approaches constituting this thesis.

In contrast to the author's earlier publications, this thesis mainly consists of exact controllability and spectral element methods that are applied to the efficient numerical simulation of fluid-structure interaction problems. The coupling involved in fluid-structure interaction problems makes applying the methods more complicated than when considering acoustic and elastic waves without coupling. Particularly, the computer implementations of acoustic and elastic waves are not straightforwardly extended to the coupled problem. Consequently, several issues have to be taken into account, which makes developing an efficient solution method for the fluid-structure interaction more challenging than in the uncoupled case. Thus, improvements are made compared with the earlier published results presented in conjunction with elastic materials and acoustic waves.

1.7 Outline

The thesis is organized as follows. First, the mathematical models for acoustic and elastic waves with some preliminaries are presented in Chapter 2. Then, the coupled problem is constructed in Chapter 3. We discretize the coupled problem in a space domain with spectral elements in Chapter 4. Since we are particularly interested in the time-dependent wave equations, we consider the time discretization in Chapter 5. The time discretization is accomplished using central finite differences in Section 5.1, whereas the fourth-order Runge–Kutta scheme is applied in Section 5.2. We also make an effort for decreasing the computing time by employing the fourth-order Adams–Bashforth (AB) method in Section 5.3. In Chapter 6, we expand the control approach to the coupled problem. The objective functional is presented in Section 6.1. In Sections 6.2 and 6.3, we compute the gradient of the functional, an essential point of the method, using the adjoint state technique. The optimization algorithm is considered in Chapter 7. Computation of the initial approximation for the optimization is described in Section 7.1, and the conjugate gradient algorithm is presented in Section 7.2. The graph-based multigrid method is used for preconditioning the conjugate gradient algorithm in Section 7.3. Numerical experiments concerning the acousto-elastic propagation of time-harmonic waves show the efficiency of the algorithm in Chapter 8. The concluding remarks are presented in Chapter 9.

2 ACOUSTIC AND ELASTIC WAVE EQUATIONS

Wave equations are partial differential equations which describe the propagation of various types of waves, such as acoustic, elastic and electromagnetic waves. This study concentrates on formulations suitable for simulating the coupling between acoustic and elastic waves. We focus in particular on two time-harmonic linear wave equations and the mutual interaction between them. The scalar-valued Helmholtz equation concerns the propagation of acoustic waves and the vector-valued Navier-Cauchy equation describes the propagation of waves in an elastic medium.

In this chapter, we consider the basic relations of acoustic and elastic waves, as well as the boundary conditions used to define well-posed and physically meaningful problems. Mathematical models of acoustic and elastic wave equations are based on the fundamental laws of continuum mechanics describing the conservation of mass, momentum, and energy. All of these conservation laws are based on the Reynolds transport theorem (see, e.g., [80]), and can be expressed for a vector or scalar-valued function u in a domain Ω as

$$\frac{d}{dt}\int\limits_{\Omega} u\,dx = \int\limits_{\Omega} \frac{\partial u}{\partial t}\,dx + \int\limits_{\partial\Omega} (\mathbf{v}\cdot\mathbf{n})u\,ds = 0, \tag{1}$$

where $\partial\Omega$ is the boundary of the domain Ω, $\mathbf{n}$ is the outward unit normal vector to the boundary, and $\mathbf{v}$ is the velocity field. The mass m is defined as an integral of the density ρ over the domain Ω. Thus, the mass conservation (continuity) is obtained by substituting $u = \rho$ into Equation (1). Since the linear momentum is the product of the mass m and velocity $\mathbf{v}$, the conservation of the linear momentum can be considered with $u = \rho\mathbf{v}$. The angular momentum can be derived respectively and used to show the symmetry of the stress tensor. If the momentum is not conserved, that is, the derivative of the momentum is non-zero, there is a force acting on the body. In the case of energy conservation, $u = \rho(u + \mathbf{v}^2/2 + \mathbf{g}z)$, where u is the thermodynamical internal energy of the system per unit mass, $\mathbf{v}^2/2$ presents the kinetic energy per unit mass, $\mathbf{g}$ is gravity and z states for the vertical distance from the zero level. Considering Ω as an infinitesimal volume results in the differential equation forms of the conservation laws.

In what follows, we apply, in a conventional way, the differential equations of the conservation laws to acoustic and elastic wave propagation in the two-dimensional isotropic media. Since the material is isotropic, material parameters are independent of the coordinate system and the material has identical properties in all directions at a point. We assume that the wave motions in these models consist of vibration-like movements. Hence, velocity $\mathbf{v}$, density ρ, and deformation have only comparatively small changes, that is, small oscillations around the equilibrium state characterized by the steady-state values. The steady-state values of the velocity and density are denoted by $\mathbf{v}_0$ and ρ_0, respectively. In addition, we assume that the domain is not moving, in other words, $\mathbf{v}_0 = 0$. Since the changes in velocity and density, indicated as $\hat{\mathbf{v}} = \mathbf{v} - \mathbf{v}_0$ and $\hat{\rho} = \rho - \rho_0$, are small, the presentation leads to linear wave equations. Such small deformations are considered that changes in the geometry of the domains can be neglected.

2.1 Acoustic wave equation

We consider acoustic wave propagation in fluid, which can be either a liquid or a gas medium. 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 particles in the fluid. That is why differential equations of acoustic pressures are used for modeling the fluid domain. The Helmholtz equation, which we use in the two-dimensional fluid media Ω_f with the spatial variable $\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$, can be derived by simplifying the basic equations of fluid dynamics (see, e.g., [176, 194]).

We concentrate on fluid flows that are assumed to be irrotational and inviscid. Irrotationality ensures that there are no circular movements causing crossing to the streamlines of the wave. From the mathematical point of view this is expressed as $\nabla \times \mathbf{v}_f = 0$. Since the fluid is inviscid, there is no friction between the fluid particles. The flow velocity is assumed to be small implying $\nabla \cdot \mathbf{v}_f = 0$, where $\mathbf{v}_f$ is the velocity of fluid particles. We also assume that no thermal diffusion is involved. Thus, the temperature varies only by the pressure variations.

The steady-state values of the velocity and density are denoted by $\mathbf{v}_{f0} = 0$ and ρ_{f0}. Taking these assumptions into account, we obtain the equation of continuity

$$\frac{\partial \rho_f}{\partial t} + \nabla \cdot (\rho_{f0} \mathbf{v}_f) = 0, \tag{2}$$

where t is time and ρ_f is the density of fluid. This equation is known also as the equation of mass conservation, describing that the mass is neither created nor destroyed within a volume element.

Neglecting the gravity and supposing that variations of the pressure are small compared with the steady-state pressure, the momentum conservation can be expressed as

$$\nabla p_f = -\rho_{f0} \frac{\partial \mathbf{v}_f}{\partial t}, \tag{3}$$

where p_f is the pressure. After dividing Equation (3) by ρ_f, approximating $\rho_f \approx \rho_{f0}$, and taking divergence, we get the form

$$\nabla \cdot \left(\frac{1}{\rho_f} \nabla p_f \right) = -\nabla \cdot \frac{\partial \mathbf{v}_f}{\partial t}. \tag{4}$$

We assume that the thermodynamical processes corresponding to the changes in pressure and volume are adiabatic. In other words, there is no heat transfer, and the changes in the internal energy of the system can be covered by considering the acoustic equation of state

$$\frac{\partial p_f}{\partial t} = \frac{1}{\varkappa \rho_{f0}} \frac{\partial \rho_f}{\partial t}, \tag{5}$$

where $\varkappa \left(\rho_f \frac{\partial p_f}{\rho_f \partial \rho_f} \right)^{-1} = \frac{1}{c^2 \rho_f}$ is the adiabatic compressibility depending on the speed of sound c. Then, we substitute the time derivative of density from Equation (2) to the equation of state (5), and get

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

By taking the time derivative, we get

$$\frac{1}{\rho_f c^2} \frac{\partial^2 p_f}{\partial t^2} = -\nabla \cdot \frac{\partial \mathbf{v}_f}{\partial t}. \tag{7}$$

From Equations (4) and (7) we form the wave equation. After adding a body force f, it reads

$$\frac{1}{\rho_f c^2} \frac{\partial^2 p_f}{\partial t^2} - \nabla \cdot \left(\frac{1}{\rho_f} \nabla p_f \right) = f. \tag{8}$$

We are going to solve the time-harmonic problem with a spatial variable $\mathbf{x} = (x_1, x_2)^T \in \mathbb{R}^2$. The time-dependence of the pressure is in the form $\exp(i\omega t)$, where $i = \sqrt{-1}$ is the imaginary unit. We substitute $p_f(\mathbf{x}, t)$ 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 (8). After assuming time-harmonic values of the frequency ω, we get the time-harmonic Helmholtz equation

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

The wavenumber κ describes how many waves there are for a 2π unit. It 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 wavenumber characterizes the oscillatory behavior of the solution and increases if the frequency $f(\mathbf{x}) = c(\mathbf{x}) / \ell(\mathbf{x})$ of the wave increases.

The Helmholtz equation (9) is a fundamental equation for time-harmonic wave propagation, and it can be solved for a broad band of frequencies. It occurs in a number of physical applications, such as underwater acoustics, medical ultrasonics, and geophysics. It can also be used to model the scattering of time-harmonic acoustic waves by an obstacle.

The complex-valued total acoustic pressure field P_f is the sum of the scattered wave $P_{f_{\text{scat}}}$ and the incident plane wave $P_{f_{\text{inc}}}$. The time-harmonic incident plane wave can be given by $P_{f_{\text{inc}}}(\mathbf{x}) = \exp(i\boldsymbol{\omega} \cdot \mathbf{x})$, where $\boldsymbol{\omega}$ gives the propagation direction. The angular frequency equals the euclidean norm of $\boldsymbol{\omega}$, that is, $\omega = \|\boldsymbol{\omega}\|_2$. Then, the function F in the equation above is of the form

$$F(\mathbf{x}) = -\frac{\kappa(\mathbf{x})^2}{\rho_f(\mathbf{x})} P_{f_{\text{inc}}}(\mathbf{x}) - \nabla \cdot \left(\frac{1}{\rho_f(\mathbf{x})} \nabla P_{f_{\text{inc}}}(\mathbf{x}) \right). \tag{10}$$

An irrotational vector field has a scalar potential, and the vector field is the gradient of its scalar potential. Since the velocity field is assumed to be irrotational, we can define the velocity potential ϕ such that $\mathbf{v}_f = \nabla\phi$. Substituting this to Equation (6) and eliminating the pressure in terms of the velocity potential ϕ such that $p_f = -\rho_f \frac{\partial \phi}{\partial t}$ result in another form of the acoustic wave equation

$$\frac{1}{c^2} \frac{\partial^2 \phi}{\partial t^2} - \nabla \cdot (\nabla \phi) = f_\phi. \tag{11}$$

The advantage of using this formulation is that the velocity field can be presented as functions of ϕ and pressure as a function of $\frac{\partial \phi}{\partial t}$. Consequently, the acoustic sound intensity expressed as the product of pressure and velocity, $I = p_f \mathbf{v}_f$, can be reformulated as $I = -\rho_f \frac{\partial \phi}{\partial t} \nabla \phi$.

By replacing $\phi(\mathbf{x}, t)$ by $\phi = \Phi(\mathbf{x}) \exp(i\omega t)$ and $f_\phi(\mathbf{x}, t)$ by $f_\phi = F_\Phi(\mathbf{x}) \exp(i\omega t)$ in Equation (11), that is, separating the scalar field to the space and the time-dependent components, we get for Equation (11) the following time-harmonic counterpart

$$-\kappa(\mathbf{x})^2 \Phi - \nabla^2 \Phi = F_\Phi. \tag{12}$$

2.2 Elastic wave equation

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 behavior. The relative geometric deformation of the solid is called a strain and forces that occur in the solid are described as stresses. If the 3D configuration possesses some symmetry, we can often simplify the model into a 2D one. In the case of a thin solid, we have a plane stress situation. If the solid is thick, a plane strain can be considered. The linear theory of elasticity [136] represented by the Navier-Cauchy equation models mechanical properties in a structure Ω_s assuming small deformations. With this theory several phenomena, such as seismic waves in the earth [130] 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 and derive the Navier-Cauchy equation, which governs the propagation of time-harmonic waves in elastic solids [94, 135].

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:

$$\sigma(\mathbf{u_s}) = C\epsilon(\mathbf{u_s}), \tag{13}$$

where the vector field $\mathbf{u_s} = (\mathbf{u_s}_1(\mathbf{x}), \mathbf{u_s}_2(\mathbf{x}))^T$ denotes the two-dimensional displacement depending on the spatial variable $\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$, $\sigma(\mathbf{u_s})$ is the stress tensor, C is the elastic moduli tensor and $\epsilon(\mathbf{u_s})$ is the 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 [120]. The strain tensor is defined through the derivatives of the displacement vector $\mathbf{u_s}_i$ such that

$$\epsilon(\mathbf{u_s}_{ik}) = \frac{1}{2}\left(\frac{\partial \mathbf{u_s}_i}{\partial x_k} + \frac{\partial \mathbf{u_s}_k}{\partial x_i} + \frac{\partial \mathbf{u_s}_l}{\partial x_i}\frac{\partial \mathbf{u_s}_l}{\partial x_k}\right). \tag{14}$$

For small strains, $|\frac{\partial \mathbf{u_s}_i}{\partial x_i}| \ll 1$, the strains are related to the displacements by the linearized strain tensor ϵ, which is defined by

$$\epsilon(\mathbf{u_s}) = \frac{1}{2}\left(\nabla \mathbf{u_s} + (\nabla \mathbf{u_s})^T\right) = \begin{pmatrix} \frac{\partial \mathbf{u_s}_1}{\partial x_1} & \frac{1}{2}(\frac{\partial \mathbf{u_s}_1}{\partial x_2} + \frac{\partial \mathbf{u_s}_2}{\partial x_1}) \\ \frac{1}{2}(\frac{\partial \mathbf{u_s}_1}{\partial x_2} + \frac{\partial \mathbf{u_s}_2}{\partial x_1}) & \frac{\partial \mathbf{u_s}_2}{\partial x_2} \end{pmatrix}, \tag{15}$$

where $\nabla \mathbf{u_s}$ is the Jacobian matrix of $\mathbf{u_s}$. A normal strain $\epsilon_{ii} = \frac{\partial \mathbf{u_s}_i}{\partial x_i}$ is the change of length in the x_i-direction divided by the length in the x_i-direction, and the shear strain $\epsilon_{ij} = \epsilon_{ji} = \frac{1}{2}(\frac{\partial \mathbf{u_s}_i}{\partial x_j} + \frac{\partial \mathbf{u_s}_j}{\partial x_i})$ tells that the angle between x_i and x_j axis would be diminished by $2\epsilon_{ij}$. This approximation does not apply in situations of deformations involving large angle shears.

The 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 the 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 Equation (13) can be written in the form

$$\begin{aligned} \sigma(\mathbf{u_s}) &= \lambda_s(\nabla \cdot \mathbf{u_s})\mathcal{I} + 2\mu_s\epsilon(\mathbf{u_s}) \\ &= \begin{pmatrix} (2\mu_s + \lambda_s)\frac{\partial \mathbf{u_s}_1}{\partial x_1} + \lambda_s\frac{\partial \mathbf{u_s}_2}{\partial x_2} & \mu_s(\frac{\partial \mathbf{u_s}_1}{\partial x_2} + \frac{\partial \mathbf{u_s}_2}{\partial x_1}) \\ \mu_s(\frac{\partial \mathbf{u_s}_1}{\partial x_2} + \frac{\partial \mathbf{u_s}_2}{\partial x_1}) & (2\mu_s + \lambda_s)\frac{\partial \mathbf{u_s}_2}{\partial x_2} + \lambda_s\frac{\partial \mathbf{u_s}_1}{\partial x_1} \end{pmatrix}, \end{aligned} \tag{16}$$

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

$$\mu_s = \frac{E}{2(1+\nu)}, \quad \lambda_s = \frac{E\nu}{(1+\nu)(1-2\nu)}, \tag{17}$$

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

The speed of pressure waves (P-waves) c_p and the speed of the shear waves (S-waves) c_s can be presented as functions of the Lamé parameters and pressure, such that

$$c_p = \sqrt{\frac{\lambda_s(\mathbf{x}) + 2\mu_s(\mathbf{x})}{\rho_s(\mathbf{x})}}, \quad c_s = \sqrt{\frac{\mu_s(\mathbf{x})}{\rho_s(\mathbf{x})}}. \tag{18}$$

The P-waves move in a compressional motion, while the motion of the S-waves is perpendicular to the direction of wave propagation [191].

The conservation of the linear momentum can be presented in the form

$$\rho_s \left(\frac{\partial \mathbf{v}_s}{\partial t} + \mathbf{v}_s \cdot \nabla \mathbf{v}_s \right) - \nabla \cdot \sigma(\mathbf{u_s}) = \mathbf{f}, \tag{19}$$

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

$$\rho_s \frac{\partial^2 \mathbf{u_s}}{\partial t^2} - \nabla \cdot \sigma(\mathbf{u_s}) = \mathbf{f}, \tag{20}$$

where components of the stress tensor $\sigma(\mathbf{u_s})$ are

$$\sigma_{ij} = \frac{E}{1+\nu}\epsilon_{ij} + \frac{E\nu}{(1+\nu)(1-2\nu)}\epsilon_{kk}\delta_{ij} = 2\mu_s\epsilon_{ij} + \lambda_s\epsilon_{kk}\delta_{ij}, \quad i,j = 1,2.$$

In general, we assume the medium to be heterogeneous. Thus, the partial derivatives in $\nabla \cdot \sigma$ apply to λ_s and μ_s as well as to the displacement. In a homogeneous medium, the formula has a simpler form, and Equation (20) can be presented in the component form as follows:

$$\begin{cases} -(2\mu_s + \lambda_s)\frac{\partial^2 \mathbf{u_{s1}}}{\partial x_1^2} - \mu_s \frac{\partial^2 \mathbf{u_{s1}}}{\partial x_2^2} - (\mu_s + \lambda_s)\frac{\partial^2 \mathbf{u_{s2}}}{\partial x_1 \partial x_2} + \rho_s \frac{\partial^2 \mathbf{u_{s1}}}{\partial t^2} & = \mathbf{f}_1, \\ -(2\mu_s + \lambda_s)\frac{\partial^2 \mathbf{u_{s2}}}{\partial x_2^2} - \mu_s \frac{\partial^2 \mathbf{u_{s2}}}{\partial x_1^2} - (\mu_s + \lambda_s)\frac{\partial^2 \mathbf{u_{s1}}}{\partial x_1 \partial x_2} + \rho_s \frac{\partial^2 \mathbf{u_{s2}}}{\partial t^2} & = \mathbf{f}_2. \end{cases}$$

We consider the propagation of time-harmonic waves with angular frequency $\omega > 0$ such that the time-dependence is $\exp(i\omega t)$. Consequently, we get the time-harmonic version of Equation (20) 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

$$\sigma(\mathbf{u_s}) = (\lambda_s(\nabla \cdot \mathbf{U_s})\mathcal{I} + \mu_s(\nabla \mathbf{U_s} + (\nabla \mathbf{U_s})^T)) \exp(i\omega t) = \sigma(\mathbf{U_s}) \exp(i\omega t), \quad (21)$$

or, in the component form,

$$\sigma_{jk}(\mathbf{u_s}) = \left(\lambda_s \left(\frac{\partial \mathbf{u_{s}}_1}{\partial x_1} + \frac{\partial \mathbf{u_{s}}_2}{\partial x_2} \right) \delta_{ij} + \mu_s \left(\frac{\partial \mathbf{u_{s}}_j}{\partial x_k} + \frac{\partial \mathbf{u_{s}}_k}{\partial x_j} \right) \right) \exp(i\omega t),$$

where $j = 1, 2$. Now, the propagation of time-harmonic waves in an elastic and isotropic body can be governed by the Navier-Cauchy equation

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

having components

$$-\rho_s \omega^2 \mathbf{U_{s}}_j + \sum_{k=1}^{2} \frac{\partial \sigma_{jk}(\mathbf{U_s})}{\partial x_k} = \mathbf{F}_j, \; j = 1, 2. \quad (23)$$

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

$$\mathbf{F}(\mathbf{x}) = -\omega^2 \rho_s(\mathbf{x}) \mathbf{U_{s}}_{\text{inc}}(\mathbf{x}) - \nabla \cdot \sigma(\mathbf{U_{s}}_{\text{inc}}(\mathbf{x})). \quad (24)$$

Inserting the time-harmonic stress tensor as given by (21) into the time-harmonic equation of motion (22) and assuming constant material parameters, we yield to the following form of the Navier-Cauchy equation:

$$-\rho_s \omega^2 \mathbf{U_s} - \mu_s \nabla^2 \mathbf{U_s} - (\mu_s + \lambda_s) \nabla(\nabla \cdot \mathbf{U_s}) = \mathbf{F}, \quad (25)$$

which can be written in the component form

$$\begin{cases} -(2\mu_s + \lambda_s)\frac{\partial^2 \mathbf{u_{s}}_1}{\partial x_1^2} - \mu_s \frac{\partial^2 \mathbf{u_{s}}_1}{\partial x_2^2} - (\mu_s + \lambda_s)\frac{\partial^2 \mathbf{u_{s}}_2}{\partial x_1 \partial x_2} - \rho_s \omega^2 \mathbf{U_{s}}_1 & = \mathbf{F}_1, \\ -(2\mu_s + \lambda_s)\frac{\partial^2 \mathbf{u_{s}}_2}{\partial x_2^2} - \mu_s \frac{\partial^2 \mathbf{u_{s}}_2}{\partial x_1^2} - (\mu_s + \lambda_s)\frac{\partial^2 \mathbf{u_{s}}_1}{\partial x_1 \partial x_2} - \rho_s \omega^2 \mathbf{U_{s}}_2 & = \mathbf{F}_2. \end{cases} \quad (26)$$

The wave equation in fluid media can also be derived as a special case of the elasticity Equation (20). The displacement vector $\mathbf{u_f}$ in the fluid domain can be replaced by the corresponding pressure $p_f = -\lambda_f \nabla \cdot \mathbf{u_f}$. Then we have to take into account that in the solid media $\mu_s > 0$, whereas in the fluid media $\mu_f = 0$. In other words, we assume that there are no S-waves in fluids. That is because inviscid fluids have no internal friction and therefore can not support shear stresses. Taking the divergence of Equation (20) and substituting Equation (16) into Equation (20) yield to the wave equation in fluid media

$$\frac{1}{\rho_f c^2} \frac{\partial^2 p_f}{\partial t^2} - \nabla \cdot \left(\frac{1}{\rho_f} \nabla p_f \right) = f, \quad (27)$$

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

2.3 Physical boundary conditions

The equations need to be completed by the boundary conditions to get a well-posed and physically meaningful problem. The boundary conditions can be divided into the conditions on physical boundaries and artificial boundaries. In this section, we consider the physical boundary conditions, and the artificial boundary conditions are discussed in Section 2.5. Further, the coupling conditions on the interface boundary are derived in Chapter 3.

The Dirichlet and the 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. The Dirichlet boundary conditions give the value of the unknown at the given boundary. In the acoustic domain, the Neumann boundary conditions give the value of the gradient of the unknown at the boundary. In the elastic domain, the homogeneous Neumann boundary condition models a free boundary, where there are no external forces. The Robin boundary condition, also known as the impedance boundary condition, is a linear combination of the Dirichlet and Neumann boundary conditions.

When the scattering of acoustic pressure waves by a bounded obstacle Θ is considered, the boundary condition on the surface of the obstacle Γ_{0f} depends on its acoustic properties. The problem setting is illustrated in Figure 3, and the boundary condition can be stated as a general form

$$\mathcal{W}P_f = y_0, \text{ on } \Gamma_{0f}, \tag{28}$$

where the operator $\mathcal{W}$ sets the boundary condition and y_0 is the source function. With a sound-soft obstacle the total pressure is zero on the surface Γ_{0f}, which implies the homogeneous Dirichlet boundary condition with $y_0 = 0$ and $\mathcal{W}$ equal to the identity operator $\mathcal{I}$. The sound-hard obstacle leads to the Neumann boundary condition with $y_0 = 0$ and

$$\mathcal{W} = \frac{\partial}{\partial \mathbf{n}_f}, \tag{29}$$

where $\mathbf{n}_f$ is the outward pointing unit normal vector on the boundary Γ_{0f}. The third alternative is the Robin boundary condition presented with the acoustic admittance β, which is inversely proportional to the surface impedance. The time-harmonic version of the corresponding boundary condition is

$$\mathcal{W} = i\kappa(\mathbf{x})\beta + \frac{\partial}{\partial \mathbf{n}_f}, \tag{30}$$

whereas its time-dependent form, used in conjunction with the time-dependent pressure variable p_f, is

$$\mathcal{W} = \frac{\beta}{c(\mathbf{x})}\frac{\partial}{\partial t} + \frac{\partial}{\partial \mathbf{n}_f}. \tag{31}$$

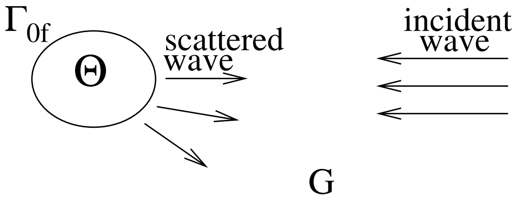


FIGURE 3 Scattering by the obstacle Θ in the exterior domain $G = \mathbb{R}^2 \setminus \left(\Theta \cup \Gamma_{0f}\right)$.

For the elastic wave equation the physical boundary conditions can be defined respectively. In the case of elastic scattering, we denote the boundary of the obstacle by Γ_{0s}, and the boundary condition is regarded as

$$\mathcal{W}\mathbf{U_s} = \mathbf{g}_0, \text{ on } \Gamma_{0s}. \tag{32}$$

Then, the Dirichlet boundary condition is considered with $\mathcal{W} = \mathcal{I}$, the Neumann boundary condition is set with $\mathcal{W} = \mathbf{A}_s$, where

$$\mathbf{A}_s = \begin{pmatrix} (2\mu_s + \lambda_s)n_{s1} & \mu_s n_{s2} \\ \lambda_s n_{s2} & \mu_s n_{s1} \end{pmatrix} \frac{\partial}{\partial x_1} + \begin{pmatrix} \mu_s n_{s2} & \lambda_s n_{s1} \\ \mu_s n_{s1} & (2\mu_s + \lambda_s)n_{s2} \end{pmatrix} \frac{\partial}{\partial x_2}, \tag{33}$$

and the Robin boundary condition is achieved for the time-harmonic problems by

$$\mathcal{W} = i\omega\rho_s(\mathbf{x})\beta\mathbf{B}_s + \mathbf{A}_s, \tag{34}$$

where $\mathbf{B}_s$ is a symmetric positive definite 2×2-matrix [64, 173], which can be computed by

$$\mathbf{B}_s = \begin{pmatrix} n_{s1} & n_{s2} \\ n_{s2} & -n_{s1} \end{pmatrix} \begin{pmatrix} c_p & 0 \\ 0 & c_s \end{pmatrix} \begin{pmatrix} n_{s1} & n_{s2} \\ n_{s2} & -n_{s1} \end{pmatrix}, \tag{35}$$

where $\mathbf{n}_s = (n_{s1}, n_{s2})^T$ is the outward pointing normal vector on Γ_{0s}. The corresponding time-dependent operator of the Robin boundary condition for the elastic wave equation is of the form

$$\mathcal{W} = \rho_s(\mathbf{x})\beta\mathbf{B}_s \frac{\partial}{\partial t} + \mathbf{A}_s. \tag{36}$$

For both the acoustic and elastic waves, we can see that $\beta = 0$ actually corresponds to the Neumann boundary condition. In the special case $\beta = 1$, the operator $\mathcal{W}$ approximates the waves propagating without reflections, which is further considered in Section 2.5.

2.4 Time-harmonic equations in exterior domains

Acoustic and elastic wave problems are formulated in unbounded, that is, 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 [49, 137, 174, 192, 193, 195]. In this chapter, we first describe this theory in the case of acoustic waves, which are modeled by the Helmholtz equation. Then, the elastic wave problem is treated in a similar manner.

We consider the scattering of a time-harmonic acoustic plane wave by a bounded, open obstacle $\Theta \subset \mathbb{R}^2$. We denote the boundary of the scattering obstacle by Γ_{0f}. Furthermore, we define the weighted function spaces in the exterior domain $G = \mathbb{R}^2 \setminus \left(\Theta \cup \Gamma_{0f}\right)$ for $a, b \in \mathbb{R}$ and $k \in \mathbb{N}$ as follows:

$$\begin{aligned}
L^2_{\text{loc}}(G) &= \{f \in L^2(D) \text{ for all compact sets } D \subset G\},\\
L^2_a(G) &= \{f \in L^2_{\text{loc}}(G) \text{ such that } \rho^a f \in L^2(G)\},\\
L^2_{<a}(G) &= \bigcap_{b<a} L^2_b(G), \qquad L^2_{>a}(G) = \bigcup_{b>a} L^2_b(G),\\
H^k_{\text{loc}}(G) &= \{f \in H^k(D) \text{ for all compact sets } D \subset G\},\\
H^k_a(G) &= \{f \in H^k_{\text{loc}}(G) \text{ such that } f, f', \ldots, f^{(k)} \in L^2_a(G)\},\\
\mathring{H}^k_a(G) &= \{\text{the closure of } C_0^\infty(G) \text{ in the norm of } H^k_a\},\\
H^k_{<a}(G) &= \bigcap_{b<a} H^k_b(G), \qquad \mathring{H}^k_{<a}(G) = \bigcap_{b<a} \mathring{H}^k_b(G),
\end{aligned}$$

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

$$-\kappa^2 P_f(\mathbf{x}) - \nabla^2 P_f(\mathbf{x}) = F, \qquad \text{in } G, \tag{37}$$

$$P_f(\mathbf{x}) = 0, \qquad \text{on } \Gamma_{0f}, \tag{38}$$

where $F \in L^2_{\text{loc}}(G)$ and $\kappa > 0$. In addition, we need the Sommerfeld radiation condition (see, e.g., [180])

$$-i\kappa P_f + \frac{\partial P_f}{\partial r} \in L^2_{>-1/2}(G) \tag{39}$$

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

$$\frac{\partial}{\partial r} = \frac{\mathbf{x} \cdot \nabla}{r}. \tag{40}$$

Definition 3.1 The outgoing solution to the exterior Dirichlet problem for Equation (37) with $F \in L^2_{\text{loc}}(G)$ is the function $P_f \in \mathring{H}^1_{<-1/2}(G)$ which satisfies the condition (39) and

$$\int_G \left(\nabla P_f \cdot \nabla \bar{\phi} - \kappa^2 P_f \bar{\phi}\right) dx = \int_G F \bar{\phi}\, dx, \quad \forall \phi \in C_0^\infty(G). \tag{41}$$

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 the 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 (37)-(38), 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

$$-\left(\kappa^2+i\tau\right) P_{f_\tau} - \nabla^2 P_{f_\tau} = F, \tag{42}$$

and use the limiting absorption principle [60] 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., [193]).

Remark 2.4.1 Proving the existence and uniqueness of the solution for the exterior problems with varying coefficients, such as

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

and

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

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

For computing purposes, the exterior problems defined in unbounded domains are truncated to a finite domain by boundary conditions, which are discussed in the next section.

2.5 Artificial boundary conditions

In wave propagation applications, the scattered waves are going out from the domain without reflections. To solve the scattering problem numerically, we need an artificial boundary condition that is an approximation of the Sommerfeld radiation condition. In the d-dimensional case, the Sommerfeld radiation condition can be stated, for instance, for the scalar-valued scattered pressure wave field $P_{f_{\text{scat}}}$ as

$$\lim_{r \to \infty} r^{\frac{d-1}{2}} \left(\frac{\partial P_{f_{\text{scat}}}}{\partial r} + i\kappa(\mathbf{x}) P_{f_{\text{scat}}} \right) = 0, \tag{43}$$

where r is the radial coordinate. The artificial boundary condition is set on the exterior artificial boundary, and it ensures that the solution either presents exactly or approximates the restriction of the solution in the original unbounded region. In practice, the choice of a suitable artificial boundary condition is a compromise between accuracy, computational efficiency, and the ease of implementation.

An exact (transparent) artificial boundary condition can be presented, for instance, as a Dirichlet-to-Neumann mapping. As the name of the method reflects, the relation between the unknown function and its normal derivative is considered on the artificial boundary. The conventional way of constructing the Dirichlet-to-Neumann mapping for computations is using a finite sum of terms in which the Hankel functions are involved (see, e.g., [122]). Since the resulting system of linear equations is not sparse, this approach demands sufficiently large computational effort and memory storage.

Perfectly matched layer [21] and absorbing boundary conditions (see, e.g., [8, 24, 25, 64, 85, 98, 129]) are examples of approximate artificial boundary conditions. In what follows, we concentrate on truncating the original unbounded domain with absorbing boundary conditions. The problem setting for acoustic scattering problems in bounded domains is illustrated in Figure 4, where $\Theta \subset \mathbb{R}^2$ denotes the obstacle and $\Omega_f \subset \mathbb{R}^2$ is the domain between the obstacle and the absorbing boundary Γ_{ef}. The boundary of the obstacle is denoted by Γ_{0f}. On Γ_{ef}, we impose the conventional first-order absorbing boundary condition [64],

$$i\kappa(\mathbf{x})P_f + \frac{\partial P_f}{\partial \mathbf{n}_f} = Y_{\text{ext}}, \text{ on } \Gamma_{ef}, \tag{44}$$

where P_f denotes the (complex-valued) total acoustic pressure field, $\mathbf{n}_f$ is the outward normal vector to Ω_f, and Y_{ext} is the source term due to the incident plane wave. If the time-harmonic incident plane wave is given by $P_{f_{\text{inc}}}(\mathbf{x}) = \exp(i\boldsymbol{\omega} \cdot \mathbf{x})$, the function Y_{ext} is of the form

$$Y_{\text{ext}}(\mathbf{x}) = i\kappa(\mathbf{x})P_{f_{\text{inc}}}(\mathbf{x}) + \frac{\partial P_{f_{\text{inc}}}(\mathbf{x})}{\partial \mathbf{n}_f}. \tag{45}$$

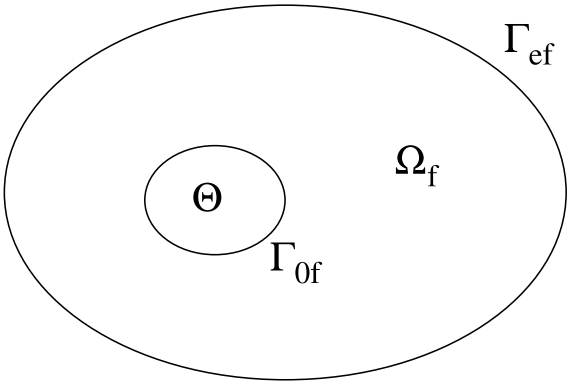


FIGURE 4 Obstacle Θ, domain Ω_f, and the two parts of the boundary $\partial\Omega_f = \Gamma_{0f} \cup \Gamma_{ef}$ of the domain Ω_f.

For the one-dimensional case, this boundary condition absorbs the waves exactly. In two dimensions, the formula gives an approximation of the phenomenon.

On the artificial boundary of the structure domain, Γ_{es}, we impose the corresponding first-order absorbing boundary condition

$$i\omega\rho_s(\mathbf{x})\mathbf{B}_s\mathbf{U_s} + \sigma(\mathbf{U_s})\mathbf{n}_s = \mathbf{G}_{\text{ext}}, \qquad \text{on } \Gamma_{es}, \tag{46}$$

where $\mathbf{B}_s$ is a matrix defined by the formula (35), and $\mathbf{n}_s = (n_{s1}, n_{s2})^T$ is the outward pointing normal vector on Γ_{es}. 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. The source function $\mathbf{G}_{\text{ext}}$ is of the form

$$\mathbf{G}_{\text{ext}}(\mathbf{x}) = i\omega\rho_s(\mathbf{x})\mathbf{B}_s\mathbf{U}_{\mathbf{s}\text{inc}}(\mathbf{x}) + \sigma(\mathbf{U}_{\mathbf{s}\text{inc}}(\mathbf{x}))\mathbf{n}_s. \tag{47}$$

When displacements are considered in both domains, we recall that in the fluid domain S-waves vanish, P-waves are represented as $c = \sqrt{\frac{\lambda_f}{\rho_f}}$, and the Lamé parameter corresponding to μ_s is $\mu_f = 0$. Thus, in the fluid domain we can use the respectively modified form of Equation (35) as follows:

$$\mathbf{B}_f = \begin{pmatrix} cn_{f1}^2 & cn_{f1}n_{f2} \\ cn_{f1}n_{f2} & cn_{f2}^2 \end{pmatrix}. \tag{48}$$

It is worth mentioning that the first-order absorbing boundary condition is the simplest alternative and not very accurate in approximating the Sommerfeld type radiation condition in a general two-dimensional case. To reduce reflections on the artificial boundary, higher-order absorbing boundary conditions could be utilized (see, e.g., [16, 63, 87]). However, the first-order absorbing boundary condition is an adequate choice for the presentation of the methods considered in this thesis.

3 COUPLED PROBLEM

The domain Ω modeling the geometry of the FSI phenomena is divided into the solid part Ω_s and the fluid part Ω_f (see Figure 5). When a wave coming from the fluid domain confronts the elastic domain, it is not totally reflected, but part of it passes to the elastic domain and turns to elastic vibrations. The analogous action is seen when the elastic wave propagates to the fluid, although usually the reflections from fluid to structure are minor when compared with the reflections from structure to fluid. To be more precise, the magnitude of the reflection depends on the difference between the densities and the wave speeds of the materials. Thus, the reflections are more significant from comparatively stiff structures than for more flexible obstacles.

Various formulations exist for the fluid-structure interaction between acoustic and elastic waves. 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 [67]. 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., [22, 126, 151, 190]), while using the velocity potential results in a symmetric system of equations (see, e.g., [39, 84, 166, 186]).

Solution methods are tailored, depending on the formulation, to achieve efficient numerical simulations. The symmetric formulation provides the opportunity to solve time-dependent equations governing the fluid domain and the displacement of the struc-

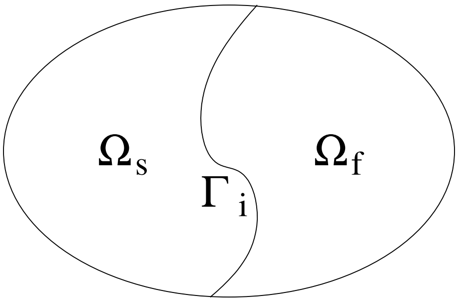


FIGURE 5 The domain Ω is divided into the solid part Ω_s and the fluid part Ω_f.

ture simultaneously (monolithic approach or direct coupling). In this case, the software implementation is required for the particular combination of physical problems. With non-symmetric formulation the time-dependent equations can be solved separately. This approach allows to utilize pre-existing solvers or black-box components which are developed for acoustic and elastic problems separately. This approach is suitable for very large problems since it requires less memory than solving the problems simultaneously. Still, an efficient transfer of the boundary conditions at the interface between solving acoustic and elastic problems plays a crucial role of the overall method's performance. In what follows, we first consider the boundary conditions corresponding to the coupling interface, and then present the mathematical models for transient coupled problems as three different formulations. Also the weak formulations and energy identities are presented. The equations for the corresponding time-harmonic models are briefly given in Appendix 1.

3.1 Coupling conditions

On the interface Γ_i between fluid and solid domains, normal components of displacements and forces are balanced:

$$\mathbf{u_s} \cdot \mathbf{n}_s + \mathbf{u_f} \cdot \mathbf{n}_f = 0, \tag{49}$$

$$\sigma(\mathbf{u_s})\mathbf{n}_s + \sigma(\mathbf{u_f})\mathbf{n}_f = 0. \tag{50}$$

Since $\mu_f = 0$, we get $\sigma(\mathbf{u_f}) = \lambda_f \nabla \cdot \mathbf{u_f} \mathcal{I} = -p_f \mathcal{I}$. After using Equation (50), the balance of normal forces is of the form

$$\sigma(\mathbf{u_s})\mathbf{n}_s = p_f \mathbf{n}_f. \tag{51}$$

To get the continuity across the interface, we differentiate (49) twice with respect to time and use Equation (20), where the body force is assumed to be zero on Γ_i, to get

$$\rho_f(\mathbf{x}) \frac{\partial^2 \mathbf{u_s}}{\partial t^2} \cdot \mathbf{n}_s = \frac{\partial p_f}{\partial \mathbf{n}_f}. \tag{52}$$

If the velocity potential ϕ is used in the fluid part, we set $p_f = -\rho_f(\mathbf{x}) \frac{\partial \phi}{\partial t}$ in Equation (51) and differentiate (49) with respect to time and use equation $\frac{\partial \mathbf{u_f}}{\partial t} = \mathbf{v}_f = \nabla \phi$ to get the following interface conditions:

$$\sigma(\mathbf{u_s})\mathbf{n}_s = -\rho_f(\mathbf{x}) \frac{\partial \phi}{\partial t} \mathbf{n}_f, \tag{53}$$

$$\frac{\partial \mathbf{u_s}}{\partial t} \cdot \mathbf{n}_s = -\frac{\partial \phi}{\partial \mathbf{n}_f}. \tag{54}$$

By combining equations of linear acoustics and elastodynamics, presented in the previous chapter, with the contact and boundary conditions, we get the coupled problem.

3.2 Non-symmetric formulation

With pressure-displacement formulation the coupled problem is

$$
\begin{aligned}
\frac{1}{\rho_f(\mathbf{x})c(\mathbf{x})^2}\frac{\partial^2 p_f}{\partial t^2} - \nabla\cdot\left(\frac{1}{\rho_f(\mathbf{x})}\nabla p_f\right) &= f, && \text{in } Q_f = \Omega_f \times [0,T], && (55)\\
p_f &= 0, && \text{on } \gamma_{0\mathrm{f}} = \Gamma_{0\mathrm{f}} \times [0,T], && (56)\\
\frac{1}{c(\mathbf{x})}\frac{\partial p_f}{\partial t} + \frac{\partial p_f}{\partial \mathbf{n}_f} &= y_{\mathrm{ext}}, && \text{on } \gamma_{ef} = \Gamma_{ef} \times [0,T], && (57)\\
\rho_f(\mathbf{x})\frac{\partial^2 \mathbf{u_s}}{\partial t^2}\cdot \mathbf{n}_s - \frac{\partial p_f}{\partial \mathbf{n}_f} &= 0, && \text{on } \gamma_i = \Gamma_i \times [0,T], && (58)\\
\rho_s(\mathbf{x})\frac{\partial^2 \mathbf{u_s}}{\partial t^2} - \nabla\cdot\sigma(\mathbf{u_s}) &= \mathbf{f}, && \text{in } Q_s = \Omega_s \times [0,T], && (59)\\
\mathbf{u_s} &= \mathbf{0}, && \text{on } \gamma_{0\mathrm{s}} = \Gamma_{0\mathrm{s}} \times [0,T], && (60)\\
\rho_s(\mathbf{x})\mathbf{B}_s\frac{\partial \mathbf{u_s}}{\partial t} + \sigma(\mathbf{u_s})\mathbf{n}_s &= \mathbf{g}_{\mathrm{ext}}, && \text{on } \gamma_{es} = \Gamma_{es} \times [0,T], && (61)\\
\sigma(\mathbf{u_s})\mathbf{n}_s - p_f\mathbf{n}_f &= \mathbf{0}, && \text{on } \gamma_i = \Gamma_i \times [0,T]. && (62)
\end{aligned}
$$

In addition to the system (55)-(62), we take into account the initial conditions

$$
\begin{aligned}
p_f(\mathbf{x},0) &= \mathbf{e}_{f0}, && \text{in } \Omega_f, && (63)\\
\frac{\partial p_f}{\partial t}(\mathbf{x},0) &= \mathbf{e}_{f1}, && \text{in } \Omega_f, && (64)\\
\mathbf{u_s}(\mathbf{x},0) &= \mathbf{e}_{s0}, && \text{in } \Omega_s, && (65)\\
\frac{\partial \mathbf{u_s}(\mathbf{x},0)}{\partial t} &= \mathbf{e}_{s1}, && \text{in } \Omega_s. && (66)
\end{aligned}
$$

A coupled model like this is used, for instance, in [126]. For the existence and uniqueness of the solution for the problem (55)-(62), we refer to [82], and for the corresponding time-harmonic problem to [52] and [148].

For the weak formulation of the system (55)-(62) we introduce the function spaces V and $\mathbf{V}$ by

$$
\begin{aligned}
V &= \{v \in H^1(\Omega_f) \text{ such that } v = 0 \text{ on } \Gamma_{0f}\}, && (67)\\
\mathbf{V} &= \{\mathbf{v} \in H^1(\Omega_s) \times H^1(\Omega_s) \text{ such that } \mathbf{v} = \mathbf{0} \text{ on } \Gamma_{0s}\}. && (68)
\end{aligned}
$$

By multiplying Equation (55) with any test function v in the space V, and (59) with any test function $\mathbf{v}$ in the space $\mathbf{V}$, using Green's formula results in

$$\int\limits_{\Omega_f} \frac{1}{\rho_f(\mathbf{x})c(\mathbf{x})^2}\frac{\partial^2 p_f}{\partial t^2} v\,dx + \int\limits_{\Omega_f} \frac{1}{\rho_f(\mathbf{x})}\nabla p_f \cdot \nabla v\,dx - \int\limits_{\Gamma_f} \frac{1}{\rho_f(\mathbf{x})}\frac{\partial p_f}{\partial n_f} v\,ds = \int\limits_{\Omega_f} f v\,dx,$$

$$\int\limits_{\Omega_s} \rho_s(\mathbf{x})\frac{\partial^2 \mathbf{u_s}}{\partial t^2}\cdot \mathbf{v}\,dx + \int\limits_{\Omega_s} C\epsilon(\mathbf{u_s}) : \epsilon(\mathbf{v})\,dx - \int\limits_{\Gamma_s} \sigma(\mathbf{u_s})\mathbf{n}_s \cdot \mathbf{v}\,ds = \int\limits_{\Omega_s} \mathbf{f}\cdot\mathbf{v}\,dx.$$

After substituting the boundary conditions, we get the following weak formulation: Find $(p_f, \mathbf{u_s})$ satisfying $(p_f(t), \mathbf{u_s}(t)) \in (V \times \mathbf{V})$ for any $t \in [0, T]$ and

$$a_f(p_f, v) - \int\limits_{\Gamma_i} \frac{\partial^2 \mathbf{u_s}}{\partial t^2}\cdot \mathbf{n}_s v\,ds = f_f(v), \tag{69}$$

$$a_s(\mathbf{u_s}, \mathbf{v}) - \int\limits_{\Gamma_i} p_f \mathbf{n}_f \cdot \mathbf{v}\,ds = \mathbf{f}_s(\mathbf{v}), \tag{70}$$

where

$$\begin{aligned} a_f(p_f, v) :=& \int\limits_{\Omega_f} \frac{1}{\rho_f(\mathbf{x})c(\mathbf{x})^2}\frac{\partial^2 p_f}{\partial t^2} v\,dx + \int\limits_{\Omega_f} \frac{1}{\rho_f(\mathbf{x})}\nabla p_f \cdot \nabla v\,dx \\ &+ \int\limits_{\Gamma_{ef}} \frac{1}{c(\mathbf{x})\rho_f(\mathbf{x})}\frac{\partial p_f}{\partial t} v\,ds, \end{aligned} \tag{71}$$

$$f_f(v) := \int\limits_{\Omega_f} f v\,dx + \int\limits_{\Gamma_{ef}} \frac{1}{\rho_f(\mathbf{x})} y_{\text{ext}} v\,ds, \tag{72}$$

$$\begin{aligned} a_s(\mathbf{u_s}, \mathbf{v}) :=& \int\limits_{\Omega_s} \rho_s(\mathbf{x})\frac{\partial^2 \mathbf{u_s}}{\partial t^2}\cdot\mathbf{v}\,dx + \int\limits_{\Omega_s} C\epsilon(\mathbf{u_s}) : \epsilon(\mathbf{v})\,dx \\ &+ \int\limits_{\Gamma_{es}} \rho_s(\mathbf{x})\mathbf{B}_s \frac{\partial \mathbf{u_s}}{\partial t}\cdot\mathbf{v}\,ds, \end{aligned} \tag{73}$$

$$\mathbf{f}_s(\mathbf{v}) := \int\limits_{\Omega_s} \mathbf{f}\cdot\mathbf{v}\,dx + \int\limits_{\Gamma_{es}} \mathbf{g}_{\text{ext}}\cdot\mathbf{v}\,ds. \tag{74}$$

The energy identities can be derived from the weak formulations. By summing up (69) and (70) and making substitutions $v = \frac{\partial p_f}{\partial t}$, $\mathbf{u_s} = \nabla \mathbf{u_s}$, and $\mathbf{v} = \frac{\partial \nabla \mathbf{u_s}}{\partial t}$ we can derive, in the one-dimensional case, the energy equation

$$\begin{aligned} &E_{sf}(p_f, \mathbf{u_s}) = E_f(p_f) + E_s(\mathbf{u_s}) \\ &= \frac{1}{2}\int\limits_{\Omega_f}\left(\frac{1}{\rho_f c}\left|\frac{\partial p_f}{\partial t}\right|^2 + \frac{1}{\rho_f}\left|\frac{\partial p_f}{\partial x_1}\right|^2\right) dx + \frac{1}{2}\int\limits_{\Omega_s}\left(\rho_s\left|\frac{\partial}{\partial x}\frac{\partial \mathbf{u_s}}{\partial t}\right|^2 + (\lambda_s + 2\mu_s)\left|\frac{\partial^2 \mathbf{u_s}}{\partial x_1^2}\right|^2\right) dx. \end{aligned} \tag{75}$$

Since the formulation is non-symmetric, deriving the energy equation in the two-dimensional case is more complicated.

3.3 Symmetric formulation - velocity potential in the fluid domain

The energy formulation is further used in Chapter 6, where we present the exact controllability approach for solving the time-harmonic wave problem. Thus, we turn to a more convenient choice from the controllability point of view and present a symmetric formulation, in which we use the velocity potential ϕ in the fluid domain instead of the pressure p_f. By proceeding in this way, the coupled problem can be presented as

$$\frac{1}{c(\mathbf{x})^2}\frac{\partial^2\phi}{\partial t^2} - \nabla\cdot\left(\nabla\phi\right) = f_\phi, \qquad \text{in } Q_f = \Omega_f \times [0,T], \tag{76}$$

$$\phi = 0, \qquad \text{on } \gamma_{0\mathrm{f}} = \Gamma_{0\mathrm{f}} \times [0,T], \tag{77}$$

$$\frac{1}{c(\mathbf{x})}\frac{\partial\phi}{\partial t} + \frac{\partial\phi}{\partial\mathbf{n}_f} = y_{\phi\mathrm{ext}}, \qquad \text{on } \gamma_{ef} = \Gamma_{ef} \times [0,T], \tag{78}$$

$$\frac{\partial\mathbf{u_s}}{\partial t}\cdot\mathbf{n}_s + \frac{\partial\phi}{\partial\mathbf{n}_f} = 0, \qquad \text{on } \gamma_i = \Gamma_i \times [0,T], \tag{79}$$

$$\rho_s(\mathbf{x})\frac{\partial^2\mathbf{u_s}}{\partial t^2} - \nabla\cdot\sigma(\mathbf{u_s}) = \mathbf{f}, \qquad \text{in } Q_s = \Omega_s \times [0,T], \tag{80}$$

$$\mathbf{u_s} = \mathbf{0}, \qquad \text{on } \gamma_{0\mathrm{s}} = \Gamma_{0\mathrm{s}} \times [0,T], \tag{81}$$

$$\rho_s(\mathbf{x})\mathbf{B}_s\frac{\partial\mathbf{u_s}}{\partial t} + \sigma(\mathbf{u_s})\mathbf{n}_s = \mathbf{g}_{\mathrm{ext}}, \qquad \text{on } \gamma_{es} = \Gamma_{es} \times [0,T], \tag{82}$$

$$\sigma(\mathbf{u_s})\mathbf{n}_s + \rho_f(\mathbf{x})\frac{\partial\phi}{\partial t}\mathbf{n}_f = \mathbf{0}, \qquad \text{on } \gamma_i = \Gamma_i \times [0,T]. \tag{83}$$

Furthermore, we complete the time-dependent system (76)-(83) by the initial conditions

$$\phi(\mathbf{x},0) = \mathbf{e}_{\phi f0}, \qquad \text{in } \Omega_f, \tag{84}$$

$$\frac{\partial\phi}{\partial t}(\mathbf{x},0) = \mathbf{e}_{\phi f1}, \qquad \text{in } \Omega_f, \tag{85}$$

$$\mathbf{u_s}(\mathbf{x},0) = \mathbf{e}_{s0}, \qquad \text{in } \Omega_s, \tag{86}$$

$$\frac{\partial\mathbf{u_s}(\mathbf{x},0)}{\partial t} = \mathbf{e}_{s1}, \qquad \text{in } \Omega_s. \tag{87}$$

This kind of coupled model is used, for instance, in [39, 186, 84].

When the formulation with velocity potential and displacement is used, the weak form is: Find $(\phi, \mathbf{u_s})$ satisfying $(\phi(t), \mathbf{u_s}(t)) \in (V \times \mathbf{V})$ for any $t \in [0,T]$ and

$$a_{\phi f}(\phi, v) + \int\limits_{\Gamma_i}\frac{\partial\mathbf{u_s}}{\partial t}\cdot\mathbf{n}_s v\,ds = f_{\phi f}(v), \tag{88}$$

$$a_s(\mathbf{u_s},\mathbf{v}) + \int\limits_{\Gamma_i}\rho_f(\mathbf{x})\frac{\partial\phi}{\partial t}\mathbf{n}_f\cdot\mathbf{v}\,ds = \mathbf{f}_s(\mathbf{v}), \tag{89}$$

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

$$a_{\phi f}(\phi, v) := \int\limits_{\Omega_f} \frac{1}{c(\mathbf{x})^2} \frac{\partial^2 \phi}{\partial t^2} v \, dx + \int\limits_{\Omega_f} \nabla \phi \cdot \nabla v \, dx + \int\limits_{\Gamma_{ef}} \frac{1}{c(\mathbf{x})} \frac{\partial \phi}{\partial t} v \, ds, \tag{90}$$

$$f_{\phi f}(v) := \int\limits_{\Omega_f} f_\phi v \, dx + \int\limits_{\Gamma_{ef}} y_{\phi \text{ext}} v \, ds. \tag{91}$$

Further, we multiply (88) by $\rho_f(\mathbf{x})$, sum up (88) and (89), and make substitutions $v = \frac{\partial \phi}{\partial t}$ and $\mathbf{v} = \frac{\partial \mathbf{u_s}}{\partial t}$ to get for the coupled two-dimensional domain $\Omega_f \bigcup \Omega_s$ the total energy

$$\begin{aligned} E_{\phi s f}(\phi, \mathbf{u_s}) &= E_{\phi f}(\phi) + E_s(\mathbf{u_s}) \\ &= \frac{1}{2} \int\limits_{\Omega_f} \left(\frac{\rho_f(\mathbf{x})}{c(\mathbf{x})^2} \left| \frac{\partial \phi}{\partial t} \right|^2 + \rho_f(\mathbf{x}) \left| \nabla \phi \right|^2 \right) dx + \frac{1}{2} \int\limits_{\Omega_s} \left(\rho_s(\mathbf{x}) \left| \frac{\partial \mathbf{u_s}}{\partial t} \right|^2 + C\epsilon(\mathbf{u_s}) : \epsilon(\mathbf{u_s}) \right) dx. \end{aligned} \tag{92}$$

The derivative of the total energy is

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

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 $\mathbf{g}_{\text{ext}}$ the energy is non-dissipative, which may cause stability issues.

3.4 Symmetric formulation - displacement in both domains

Another option is to use displacements in both domains. This is considered, for instance, in [177], and the coupled form is

$$\rho_f(\mathbf{x})\frac{\partial^2\mathbf{u_f}}{\partial t^2} - c^2\rho_f(\mathbf{x})\nabla\cdot(\nabla\cdot\mathbf{u_f}) = \mathbf{f}_f, \qquad \text{in } Q_f = \Omega_f \times [0,T], \tag{93}$$

$$\mathbf{u_f} = \mathbf{0}, \qquad \text{on } \gamma_{0\mathrm{f}} = \Gamma_{0\mathrm{f}} \times [0,T], \tag{94}$$

$$\rho_f(\mathbf{x})\mathbf{B}_f\frac{\partial\mathbf{u_f}}{\partial t} + c^2\rho_f(\mathbf{x})(\nabla\cdot\mathbf{u_f})\mathbf{n}_f = \mathbf{y}_{f\mathrm{ext}}, \qquad \text{on } \gamma_{ef} = \Gamma_{ef} \times [0,T], \tag{95}$$

$$\mathbf{u_s}\cdot\mathbf{n}_s + \mathbf{u_f}\cdot\mathbf{n}_f = 0, \qquad \text{on } \gamma_i = \Gamma_i \times [0,T], \tag{96}$$

$$\rho_s(\mathbf{x})\frac{\partial^2\mathbf{u_s}}{\partial t^2} - \nabla\cdot\sigma(\mathbf{u_s}) = \mathbf{f}, \qquad \text{in } Q_s = \Omega_s \times [0,T], \tag{97}$$

$$\mathbf{u_s} = \mathbf{0}, \qquad \text{on } \gamma_{0\mathrm{s}} = \Gamma_{0\mathrm{s}} \times [0,T], \tag{98}$$

$$\rho_s(\mathbf{x})\mathbf{B}_s\frac{\partial\mathbf{u_s}}{\partial t} + \sigma(\mathbf{u_s})\mathbf{n}_s = \mathbf{g}_{\mathrm{ext}}, \qquad \text{on } \gamma_{es} = \Gamma_{es} \times [0,T], \tag{99}$$

$$\sigma(\mathbf{u_s})\mathbf{n}_s + c^2\rho_f(\mathbf{x})(\nabla\cdot\mathbf{u_f})\mathbf{n}_f = \mathbf{0}, \qquad \text{on } \gamma_i = \Gamma_i \times [0,T], \tag{100}$$

with the initial conditions

$$\mathbf{u_f}(\mathbf{x},0) = \mathbf{e}_{uf0}, \qquad \text{in } \Omega_f, \tag{101}$$

$$\frac{\partial\mathbf{u_f}}{\partial t}(\mathbf{x},0) = \mathbf{e}_{uf1}, \qquad \text{in } \Omega_f, \tag{102}$$

$$\mathbf{u_s}(\mathbf{x},0) = \mathbf{e}_{s0}, \qquad \text{in } \Omega_s, \tag{103}$$

$$\frac{\partial\mathbf{u_s}(\mathbf{x},0)}{\partial t} = \mathbf{e}_{s1}, \qquad \text{in } \Omega_s. \tag{104}$$

The weak form is: Find $(\mathbf{u_f},\mathbf{u_s})$ satisfying $(\mathbf{u_f}(t),\mathbf{u_s}(t)) \in (\mathbf{V}_f \times \mathbf{V})$ for any $t \in [0,T]$ and

$$a_{uf}(\mathbf{u_f},\mathbf{v}_f) + \int_{\Gamma_i} \sigma(\mathbf{u_s})\mathbf{n}_s\cdot\mathbf{v}_f\,ds = \mathbf{f}_{uf}(\mathbf{v}_f), \tag{105}$$

$$a_s(\mathbf{u_s},\mathbf{v}) + \int_{\Gamma_i} c^2\rho_f(\mathbf{x})(\nabla\cdot\mathbf{u_f})\mathbf{n}_f\cdot\mathbf{v}\,ds = \mathbf{f}_s(\mathbf{v}), \tag{106}$$

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

$$\begin{aligned} a_{uf}(\mathbf{u_f},\mathbf{v}_f) :=& \int_{\Omega_f} \rho_f(\mathbf{x})\frac{\partial^2\mathbf{u_f}}{\partial t^2}\cdot\mathbf{v}_f\,dx + \int_{\Omega_f} c^2\rho_f(\mathbf{x})(\nabla\cdot\mathbf{u_f}))(\nabla\cdot\mathbf{v}_f)\,dx \\ &+ \int_{\Gamma_{ef}} \rho_f(\mathbf{x})\mathbf{B}_f\frac{\partial\mathbf{u_f}}{\partial t}\cdot\mathbf{v}_f\,ds, \end{aligned} \tag{107}$$

$$\mathbf{f}_{uf}(\mathbf{v}_f) := \int_{\Omega_f} \mathbf{f}_f\mathbf{v}_f\,dx + \int_{\Gamma_{ef}} \mathbf{y}_{f\mathrm{ext}}\mathbf{v}_f\,ds. \tag{108}$$

The divergence of the variable in the fluid domain is involved in the weak formulation. That is why it needs to be square-integrable. Thus, we define the function space $\mathbf{V}_f$ by

$$\mathbf{V}_f = \{\mathbf{v} \in H(div, \Omega_f) \times H(div, \Omega_f) \text{ such that } \mathbf{v} = \mathbf{0} \text{ on } \Gamma_{0f}\}. \tag{109}$$

Again, we sum the variational formulations (105) and (106) and make substitutions $\mathbf{v}_f = \frac{\partial \mathbf{u_f}}{\partial t}$ and $\mathbf{v} = \frac{\partial \mathbf{u_s}}{\partial t}$ to get the corresponding energy formulation. The total energy for the coupled domain $\Omega_f \bigcup \Omega_s$ in the case of the pure displacement formulation is

$$\begin{aligned} E_{usf}(\mathbf{u_f}, \mathbf{u_s}) &= E_{uf}(\mathbf{u_f}) + E_s(\mathbf{u_s}) \\ &= \frac{1}{2} \int\limits_{\Omega_f} \left(\rho_f(\mathbf{x}) \left| \frac{\partial \mathbf{u_f}}{\partial t} \right|^2 + c^2 \rho_f(\mathbf{x}) \left| \nabla \cdot \mathbf{u_f} \right|^2 \right) dx \\ &+ \frac{1}{2} \int\limits_{\Omega_s} \left(\rho_s(\mathbf{x}) \left| \frac{\partial \mathbf{u_s}}{\partial t} \right|^2 + C\epsilon(\mathbf{u_s}) : \epsilon\left(\mathbf{u_s}\right) \right) dx. \end{aligned} \tag{110}$$

Consequently, the derivative of the total energy is

$$\begin{aligned} \frac{d}{dt} E_{usf}(\mathbf{u_f}, \mathbf{u_s}) = &- \int\limits_{\Gamma_{ef}} \rho_f(\mathbf{x}) \frac{\partial \mathbf{u_f}}{\partial t}^T \mathbf{B}_f \frac{\partial \mathbf{u_f}}{\partial t} \, ds - \int\limits_{\Gamma_{es}} \rho_s(\mathbf{x}) \frac{\partial \mathbf{u_s}}{\partial t}^T \mathbf{B}_s \frac{\partial \mathbf{u_s}}{\partial t} \, ds \\ &+ \int\limits_{\Omega_f} \mathbf{f}_f \frac{\partial \mathbf{u_f}}{\partial t} \, dx + \int\limits_{\Gamma_{ef}} \mathbf{y}_{f\text{ext}} \frac{\partial \mathbf{u_f}}{\partial t} \, ds + \int\limits_{\Omega_s} \mathbf{f} \cdot \frac{\partial \mathbf{u_s}}{\partial t} \, dx + \int\limits_{\Gamma_{es}} \mathbf{g}_{\text{ext}} \cdot \frac{\partial \mathbf{u_s}}{\partial t} \, ds. \end{aligned}$$

4 SPATIAL DISCRETIZATION

Discretization methods play a crucial role in the 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 attempt to meet these requirements by using the spectral element [44] method (SEM) for space discretization.

The SEM was pioneered in the mid 1980's by Patera [168] and Maday [146], and it combines the geometric flexibility of finite elements [120, 132] with the high accuracy of spectral methods [36]. When using the SEM, the computational domain is typically divided into non-overlapping quadrilateral elements, but also triangular elements can be used [27, 127]. Contrary to the quadrilateral spectral elements, mass matrices are not generally diagonal with triangular elements [183]. Spectral triangles and tetrahedrons providing mass lumping are constructed and applied for a seismic application of the time-dependent acoustic wave problem by Mulder, Chin-Joe-Kong, and van Veldhuizen in [41].

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, the accuracy has been observed to be slightly better on quadrilaterals than on triangles, and the condition number of the stiffness matrices grows faster for triangles than quadrilaterals [167]. At present, it seems that when polygonal elements are considered, the triangle-based SEM is competitive with the quadrilateral one only if the domain Ω has a curved shape. Curvilinear geometries could be approximated accurately by using elements with curved edges. However, the elements with curved edges are left for future work, and we concentrate on polygonal elements in this thesis. These are the reasons why we have chosen to use quadrilateral elements and the associated polynomial spectral basis. A detailed comparison of the SEM on quadrilaterals and triangles is made in [167], and quadrature formulas needed for quadrilateral and triangle-based methods are recently presented, for instance, in [44] and [184], respectively. The mixed form of the spectral element method is utilized by Cohen and Fauqueux for acoustics in [46] and for linear elasticity in [47].

For stability and convergence analysis see [197] in the case of acoustic and [42] in the case of elastic waves.

In order to produce an approximate solution for the problem, 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 principle, the error of the approximation decreases as the size of the 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.

After the domain is decomposed into elements, a local polynomial basis is introduced in each element. The 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. As, for instance, in [127], we locate the degrees of freedom corresponding to the basis functions at the Gauss–Lobatto integration points of the elements (see Figures 6-7). 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.

4.1 Discrete weak formulation

The computational domain Ω is divided into N_e quadrilateral elements $\Omega_i, i = 1, \dots, N_e$, such that $\Omega = \bigcup_{i=1}^{N_e} \Omega_i$, that is, 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. For the discrete formulation, we define the reference element $\Omega_{\text{ref}} = [0, 1]^2$ and invertible affine mappings $\mathcal{G}_i : \Omega_{\text{ref}} \to \Omega_i$ such that $\mathcal{G}_i(\Omega_{\text{ref}}) = \Omega_i$. Each of N_e elements is individually mapped to the reference element, and we make use

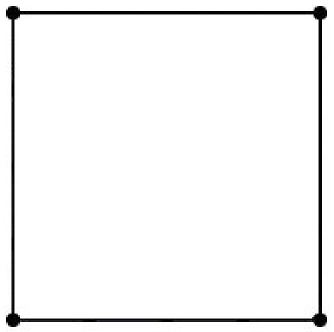


FIGURE 6 Spectral element of order $r = 1$.

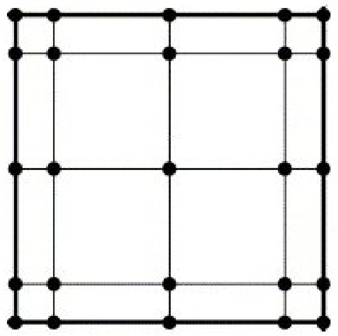


FIGURE 7 Spectral element of order $r = 4$.

of the affine mapping to make transformations from the physical domain to the reference domain, and on the contrary. The mapping between the reference element and the ith element is defined such that $\mathcal{G}_i(\xi,\zeta) = \mathbf{x} = (x_1, x_2) \in \Omega_i$.

The variable which we use in the solid domain is the displacement $\mathbf{u_s}(\mathbf{x},t)$. Thus, the number of degrees of freedom (DOF) in the solid domain, $\hat{N}_s$, is in the two-dimensional domain twice the number of discretization points in the solid domain N_s. The total number of degrees of freedom, that is, $\hat{N}$, depends on the variable of the fluid domain. If the fluid domain is modeled by using pressure $p_f(\mathbf{x},t)$ or velocity potential $\phi(\mathbf{x},t)$, we have scalar values at each spatial discretization point, and the number of degrees of freedom in the fluid domain, expressed as $\hat{N}_f$, is equal to the number of discretization points in the fluid domain N_f. Using the vector-valued displacement $\mathbf{u_f}(\mathbf{x},t)$ in the fluid domain doubles the number of degrees of freedom in the fluid domain. In other words, $\hat{N}_f = 2N_f$. Therefore also the memory consumption for computing the pure displacement-displacement interaction is higher than in the case of the other couplings considered.

Furthermore, if the formulation with displacement in both domains is considered, the divergence of the variable in the fluid domain is involved in the weak formulation (105)-(106). Since $\mathbf{u_f} \in \mathbf{V}_f = \{\mathbf{v} \in H(div,\Omega_f) \times H(div,\Omega_f)$ such that $\mathbf{v} = \mathbf{0}$ on $\Gamma_{0f}\}$, we would need a scheme that approximates the functions in $H(div,\Omega_f) \times H(div,\Omega_f)$ better than the spectral element method does. For instance, Raviart-Thomas finite elements, which are used for acoustic wave equations in [133], could be used for this purpose. Furthermore, the Raviart-Thomas elements are not a good choice for discretizating the solid domain. Nevertheless, the coupling conditions should be satisfied at the interface of the two domains. That is, we would need to change the discretization approach in the solid domain as well, to make the degrees of freedom coincide or fulfill the coupling conditions, for example, by using Lagrange multipliers as it is done in [23]. The approaches needed for that kind of coupling are out of the scope of this thesis, and that is why we have left these observations for future work. In what follows, we concentrate on the two remaining formulations, and model the fluid domain in terms of pressure or

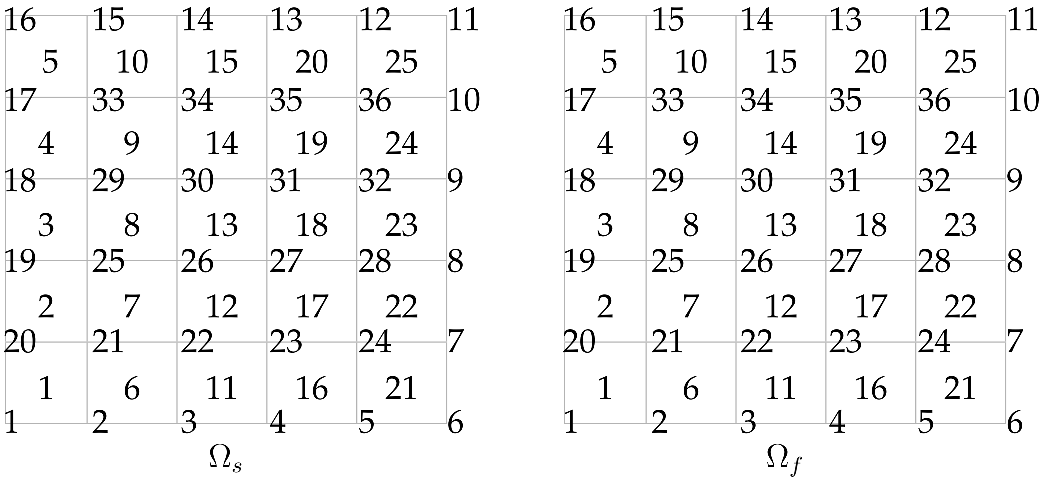


FIGURE 8 An example of the global numbering of nodes and elements in structure and fluid domains divided into quadrilateral elements.

velocity potential.

The spectral element method is obtained from the weak formulation of the pressure-based model (69)-(70) or the velocity potential -based model (88)-(89) 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

$$V_h^r = \{v_h \in V \text{ such that } v_h|_{\Omega_i} \circ \mathcal{G}_i \in Q^r\}, \tag{111}$$

$$\mathbf{V}_h^r = \{\mathbf{v}_h = (v_{h1}, v_{h2}) \in \mathbf{V} \text{ such that } v_{hk}|_{\Omega_i} \circ \mathcal{G}_i \in Q^r, k = 1, 2\}, \tag{112}$$

where

$$Q^r(\Omega_i) = \{v(\xi, \zeta) = \sum_{p=0}^{r} \sum_{q=0}^{r} a_{pq} \xi^p \zeta^q, \quad a_{pq} \in \mathbb{R}\} \tag{113}$$

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. It is also worth mentioning that the special case $r = 1$ corresponds to the bilinear finite elements.

Based on the definitions presented above, we can write the semidiscrete weak formulation, for instance, for the problem (55)-(62), as follows: Find $(p_{f_h}, \mathbf{u_s}_h)$ satisfying $(p_f(t)_h, \mathbf{u_s}(t)_h) \in (V_h^r \times \mathbf{V}_h^r)$ for any $t \in [0, T]$ and

$$a_f(p_{f_h}, v_h) - \int\limits_{\Gamma_i} \frac{\partial^2 \mathbf{u_s}_h}{\partial t^2} \cdot \mathbf{n}_s v_h \, ds = f_f(v_h), \tag{114}$$

$$a_s(\mathbf{u_s}_h, \mathbf{v}_h) - \int\limits_{\Gamma_i} p_{f_h} \mathbf{n}_f \cdot \mathbf{v}_h \, ds = \mathbf{f}_s(\mathbf{v}_h), \tag{115}$$

for all $(v_h, \mathbf{v}_h) \in (V_h^r \times \mathbf{V}_h^r)$ and $t \in [0, T]$. The discrete variables p_{f_h} and $\mathbf{u_s}_h$ are approximated as linear combinations of the corresponding nodal values and the basis functions φ_i, $i = 1, \ldots, N_f$ and ψ_i, $i = 1, \ldots, N_s$. The basis functions φ_n (and, respectively, ψ_n) are constructed with the help of the basis functions $\hat{\varphi}_{jk}$, $j, k = 1, \ldots, r+1$, defined 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 $\varphi_n|_{\Omega_i} \circ \mathcal{G}_i = \hat{\varphi}_{jk}$. In the next section, we take a closer look for constructing these basis functions by following the presentation by Cohen in [44].

4.2 Polynomial basis

In order to compute the elementwise integrals in the reference element, we introduce a set of Gauss–Lobatto (GL) points $\xi_i \in [0, 1], i = 1, \ldots, r+1$ in each direction of the

TABLE 1 GL points ξ_i and weights w_i of the interval $[0,1]$.

					i		
r			1	2	3	4	5
1	ξ_i	0	1				
	w_i	$\frac{1}{2}$	$\frac{1}{2}$				
2	ξ_i	0	$\frac{1}{2}$	1			
	w_i	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$			
3	ξ_i	0	$\frac{5-\sqrt{5}}{10}$	$\frac{5+\sqrt{5}}{10}$	1		
	w_i	$\frac{1}{12}$	$\frac{5}{12}$	$\frac{5}{12}$	$\frac{1}{12}$		
4	ξ_i	0	$\frac{7-\sqrt{21}}{14}$	$\frac{1}{2}$	$\frac{7+\sqrt{21}}{14}$	1	
	w_i	$\frac{1}{20}$	$\frac{49}{180}$	$\frac{16}{45}$	$\frac{49}{180}$	$\frac{1}{20}$	
5	ξ_i	0	$\frac{21-\sqrt{21(7+2\sqrt{7})}}{42}$	$\frac{1}{2}-\frac{\sqrt{147-42\sqrt{7}}}{42}$	$\frac{1}{2}+\frac{\sqrt{147-42\sqrt{7}}}{42}$	$\frac{21+\sqrt{21(7+2\sqrt{7})}}{42}$	1
	w_i	$\frac{1}{30}$	$\frac{63}{20(14+\sqrt{7})}$	$\frac{63}{280-20\sqrt{7}}$	$\frac{63}{280-20\sqrt{7}}$	$\frac{63}{20(14+\sqrt{7})}$	$\frac{1}{30}$

reference element. The rth order GL quadrature points in the one-dimensional reference element $[0,1]$ are the zeroes of $x(1-x)L_r'(2x-1)$, where $x\in[0,1]$ and

$$L_r'(x) = \frac{2r-1}{r}\left(L_{r-1}(x) + xL_{r-1}'(x)\right) - \frac{r-1}{r}L_{r-2}'(x) \tag{116}$$

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

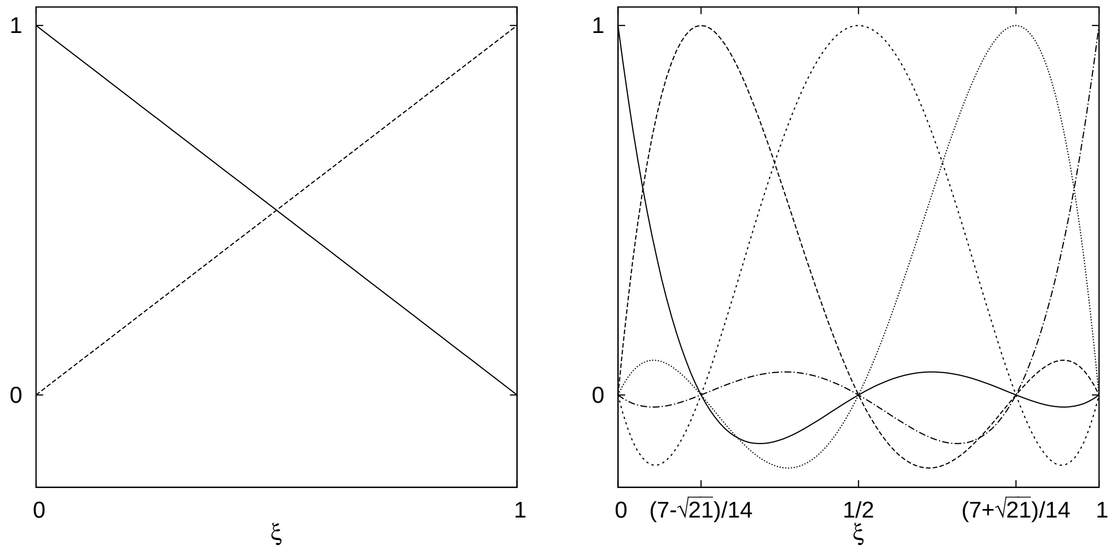
$$\begin{cases} L_0(x) &= 1, \\ L_1(x) &= x, \\ L_r(x) &= \frac{2r-1}{r}xL_{r-1}(x) - \frac{r-1}{r}L_{r-2}(x),\ r>1. \end{cases} \tag{117}$$

In practice, the GL points ξ_i and the corresponding weights,

$$w_i = \frac{1}{r(r+1)(L_r(2\xi_i-1))^2},\ i=1,\dots,r+1, \tag{118}$$

are computed beforehand for the particular values of r used in the computations. Then, the values are entered into the computer implementation. The exact values for $r=1,\dots,5$ are presented in Table 1.

We use the 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 one-dimensional rth order basis functions on the interval $[0,1]$ are constructed as a set of Lagrange interpolants (see Figure 9) associated to the point ξ_i such that



(a) $r = 1$ (b) $r = 4$

FIGURE 9 One-dimensional Lagrangian interpolants of degree r. Each of the $r+1$ functions has the value one at one GL quadrature point and zero at the other GL quadrature points.

$$\hat{\varphi}_j(\xi_i) = \prod_{p=1,p\neq j}^{r+1} \left(\frac{\xi_i - \xi_p}{\xi_j - \xi_p} \right), \quad j = 1, \ldots, r+1. \tag{119}$$

The basis functions are uniquely determined by the requirement that each function has the value one at one GL quadrature point and zero at the other GL quadrature points

$$\hat{\varphi}_j(\xi_i) = \delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j. \end{cases} \tag{120}$$

Then, the piecewise polynomial approximation p_{f_h} of the pressure variable on the reference element can be approximated as

$$p_{f_h}(\xi) = \sum_{j=1}^{r+1} \hat{\varphi}_j(\xi) p_f(\xi_j). \tag{121}$$

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 $\hat{\varphi}_j(\xi_i), j = 1, \ldots, r+1$. 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. In the two-dimensional case, p_{f_h} is approximated by the interpolation formula

$$p_{f_h}(\xi,\zeta) = \sum_{i=1}^{r+1}\sum_{j=1}^{r+1} \hat{\varphi}_{ij}(\xi,\zeta) p_f(\xi_i,\zeta_j). \tag{122}$$

4.3 Semidiscretized equation

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 of the quadrilateral mesh. The weak formulation for both the problem (55)-(62) and the problem (76)-(83) can now be rewritten in the matrix form

$$\mathcal{M}\frac{\partial^2 \mathbf{u}}{\partial t^2} + \mathcal{S}\frac{\partial \mathbf{u}}{\partial t} + \mathcal{K}\mathbf{u} = \mathcal{F}. \tag{123}$$

If the time-harmonic equations are considered, the corresponding matrix form after the space discretization is

$$\left(-\omega^2\mathcal{M} + i\omega\mathcal{S} + \mathcal{K}\right)\mathbf{U} = F, \tag{124}$$

such that $F\exp(i\omega t) = \mathcal{F}$.

In the case of the formulation with pressure and displacement, 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

$$\mathcal{M} = \begin{pmatrix} (\mathcal{M}_s)_{11} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (\mathcal{M}_s)_{22} & \mathbf{0} \\ (\mathcal{A}_{\text{fs}})_1 & (\mathcal{A}_{\text{fs}})_2 & \mathcal{M}_f \end{pmatrix}, \qquad \mathcal{S} = \begin{pmatrix} (\mathcal{S}_s)_{11} & (\mathcal{S}_s)_{12} & \mathbf{0} \\ (\mathcal{S}_s)_{21} & (\mathcal{S}_s)_{22} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{S}_f \end{pmatrix},$$

$$\mathcal{K} = \begin{pmatrix} (\mathcal{K}_s)_{11} & (\mathcal{K}_s)_{12} & (\mathcal{A}_{\text{sf}})_1 \\ (\mathcal{K}_s)_{21} & (\mathcal{K}_s)_{22} & (\mathcal{A}_{\text{sf}})_2 \\ \mathbf{0} & \mathbf{0} & \mathcal{K}_f, \end{pmatrix}, \qquad \mathcal{F} = \begin{pmatrix} \mathbf{f}_s \\ \mathbf{f}_f \end{pmatrix},$$

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

$$(\mathcal{M}_f)_{ij} = \int\limits_{\Omega_f} \frac{1}{\rho_f(\mathbf{x})c(\mathbf{x})^2}\varphi_i\varphi_j dx,$$

$$(\mathcal{S}_f)_{ij} = \int\limits_{\Gamma_{ef}} \frac{1}{\rho_f(\mathbf{x})c(\mathbf{x})}\varphi_i\varphi_j ds,$$

$$(\mathcal{K}_f)_{ij} = \int\limits_{\Omega_f} \frac{1}{\rho_f(\mathbf{x})}\nabla\varphi_i \cdot \nabla\varphi_j dx,$$

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

$$(\mathbf{f}_f)_i = \int\limits_{\Omega_f} f \varphi_i dx + \int\limits_{\Gamma_{ef}} y_{\mathrm{ef}} \varphi_i ds.$$

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

$$\mathcal{M}_s = \begin{pmatrix} (\mathcal{M}_s)_{11} & \mathbf{0} \\ \mathbf{0} & (\mathcal{M}_s)_{22} \end{pmatrix}, \qquad \mathcal{S}_s = \begin{pmatrix} (\mathcal{S}_s)_{11} & (\mathcal{S}_s)_{12} \\ (\mathcal{S}_s)_{21} & (\mathcal{S}_s)_{22} \end{pmatrix},$$

$$\mathcal{K}_s = \begin{pmatrix} (\mathcal{K}_s)_{11} & (\mathcal{K}_s)_{12} \\ (\mathcal{K}_s)_{21} & (\mathcal{K}_s)_{22} \end{pmatrix}, \qquad \mathbf{f}_s = \begin{pmatrix} (\mathbf{f}_s)_1 \\ (\mathbf{f}_s)_2 \end{pmatrix},$$

which have the components

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

$$((\mathbf{f}_s)_1)_i = \int\limits_{\Omega_s} \mathbf{f}_1 \psi_i \, dx + \int\limits_{\Gamma_{es}} \mathbf{g}_{ext1} \psi_i \, ds,$$
$$((\mathbf{f}_s)_2)_i = \int\limits_{\Omega_s} \mathbf{f}_2 \psi_i \, dx \int\limits_{\Gamma_{es}} \mathbf{g}_{ext2} \psi_i \, ds,$$

where $i, j = 1, \ldots, N_s$. The matrices arising from the coupling between acoustic and elastic wave equations are $\mathcal{A}_{\mathrm{fs}}$ and $\mathcal{A}_{\mathrm{sf}}$, for which holds that

$$((\mathcal{A}_{\mathrm{fs}})_1)_{ij} = -\int\limits_{\Gamma_i} n_{s1} \psi_j \varphi_i \, ds,$$
$$((\mathcal{A}_{\mathrm{fs}})_2)_{ij} = -\int\limits_{\Gamma_i} n_{s2} \psi_j \varphi_i \, ds,$$

$$((\mathcal{A}_{\mathrm{sf}})_1)_{ij} = -\int\limits_{\Gamma_i} n_{f1} \varphi_j \psi_i \, ds,$$
$$((\mathcal{A}_{\mathrm{sf}})_2)_{ij} = -\int\limits_{\Gamma_i} n_{f2} \varphi_j \psi_i \, ds.$$

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

The computation of the elementwise matrices and vectors involves the integration over the elementwise subregions. Evaluating these integrals analytically is usually complicated, even impossible. That is why a numerical integration procedure is used. 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 matrices $\mathcal{M}_f$ and $\mathcal{M}_s$ are diagonal. Furthermore, the matrix $\mathcal{M}$ is a lower triangular block matrix with diagonal blocks. Thus, the inverse of the matrix $\mathcal{M}$ is also a lower triangular block matrix with diagonal blocks,

$$\mathcal{M}^{-1} = \begin{pmatrix} (\mathcal{M}_s)_{11}^{-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (\mathcal{M}_s)_{22}^{-1} & \mathbf{0} \\ -\mathcal{M}_f^{-1}(\mathcal{A}_{\mathrm{fs}})_1(\mathcal{M}_s)_{11}^{-1} & -\mathcal{M}_f^{-1}(\mathcal{A}_{\mathrm{fs}})_2(\mathcal{M}_s)_{22}^{-1} & \mathcal{M}_f^{-1} \end{pmatrix},$$

and explicit time stepping with central finite differences requires only matrix-vector multiplications.

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 [45, 47].

In the case of the formulation with velocity potential and displacement, 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

$$\mathcal{M} = \begin{pmatrix} (\mathcal{M}_s)_{11} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (\mathcal{M}_s)_{22} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{M}_f \end{pmatrix}, \qquad \mathcal{S} = \begin{pmatrix} (\mathcal{S}_s)_{11} & (\mathcal{S}_s)_{12} & (\mathcal{A}_{\mathrm{sf}})_1 \\ (\mathcal{S}_s)_{21} & (\mathcal{S}_s)_{22} & (\mathcal{A}_{\mathrm{sf}})_2 \\ (\mathcal{A}_{\mathrm{fs}})_1 & (\mathcal{A}_{\mathrm{fs}})_2 & \mathcal{S}_f \end{pmatrix},$$

$$\mathcal{K} = \begin{pmatrix} (\mathcal{K}_s)_{11} & (\mathcal{K}_s)_{12} & \mathbf{0} \\ (\mathcal{K}_s)_{21} & (\mathcal{K}_s)_{22} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{K}_f, \end{pmatrix}, \qquad \mathcal{F} = \begin{pmatrix} \mathbf{f}_s \\ \mathbf{f}_{\phi f} \end{pmatrix},$$

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

$$(\mathcal{M}_f)_{ij} = \int\limits_{\Omega_f} \frac{\rho_f(\mathbf{x})}{c(\mathbf{x})^2} \varphi_i \varphi_j dx,$$

$$(\mathcal{S}_f)_{ij} = \int\limits_{\Gamma_{ef}} \frac{\rho_f(\mathbf{x})}{c(\mathbf{x})} \varphi_i \varphi_j ds,$$

$$(\mathcal{K}_f)_{ij} = \int\limits_{\Omega_f} \rho_f(\mathbf{x}) \nabla \varphi_i \cdot \nabla \varphi_j dx,$$

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

$$\left(\mathbf{f}_{\phi f}\right)_i = \int\limits_{\Omega_f} \rho_f(\mathbf{x}) f_\phi \varphi_i dx + \int\limits_{\Gamma_{ef}} \rho_f(\mathbf{x}) y_{\phi \mathrm{ext}} \varphi_i ds.$$

The $2N_s \times 2N_s$ block matrices and the $2N_s$-dimensional vector representing the elastic waves are exactly the same as in the non-symmetric formulation. The matrices arising from the coupling between acoustic and elastic wave equations are $\mathcal{A}_{\mathrm{fs}}$ and $\mathcal{A}_{\mathrm{sf}}$, for which holds that

$$((\mathcal{A}_{\mathrm{fs}})_1)_{ij} = \int\limits_{\Gamma_i} \rho_f(\mathbf{x}) n_{s1} \psi_j \varphi_i \, ds,$$

$$((\mathcal{A}_{\mathrm{fs}})_2)_{ij} = \int\limits_{\Gamma_i} \rho_f(\mathbf{x}) n_{s2} \psi_j \varphi_i \, ds,$$

$$((\mathcal{A}_{\mathrm{sf}})_1)_{ij} = \int\limits_{\Gamma_i} \rho_f(\mathbf{x}) n_{f1} \varphi_j \psi_i \, ds,$$

$$((\mathcal{A}_{\mathrm{sf}})_2)_{ij} = \int\limits_{\Gamma_i} \rho_f(\mathbf{x}) n_{f2} \varphi_j \psi_i \, ds.$$

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

5 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 in this chapter.

Previously, we used the central finite difference (CD) scheme for time discretization of acoustic problems in [106], and comparison with the Runge–Kutta (RK) method for disjoint acoustic and elastic domains was made in [107, 162]. With respect to the time step Δt, the CD method is second-order accurate, while the RK method is fourth-order accurate. Both methods lead to an explicit time-stepping scheme, and only matrix-vector products are needed in time-dependent non-coupled simulations. These properties are essential for computational efficiency. The drawback is that the schemes need to satisfy the stability condition, which limits the length of the time step. In addition, the computational effort of the RK method is approximately four times that of the the CD scheme at each time step.

In this chapter, the time discretization of the semi-discrete equation is performed with the central finite differences (CD) in Section 5.1 and with the fourth-order Runge–Kutta (RK) method in Section 5.2. In Section 5.3, we make an effort for decreasing the computing time with the fourth-order Adams–Bashforth (AB) method. After dividing the time interval $[0, T]$ into N time steps, each of size $\Delta t = T/N$, applying the appropriate time discretization into the semidiscretized form (123), and taking into account the initial conditions, we obtain the matrix form of the fully discrete state equation.

5.1 Central finite difference method

The spectral element approximation in space is combined with the standard second-order central finite difference scheme in time by replacing the time derivatives in the semidiscretized form (123) at time $i\Delta t$ by the following approximations

$$\frac{\partial \mathbf{u}}{\partial t} \approx \frac{\mathbf{u}^{i+1} - \mathbf{u}^{i-1}}{2\Delta t} \tag{125}$$

$$\frac{\partial^2 \mathbf{u}}{\partial t^2} \approx \frac{\mathbf{u}^{i+1} - 2\mathbf{u}^{i} + \mathbf{u}^{i-1}}{\Delta t^2}, \quad i = 0, \ldots, N, \tag{126}$$

where $\mathbf{u}^i$ is the vector $\mathbf{u}$ at time $i\Delta t$. Taking into account the initial conditions, we obtain the fully discrete state equation. It can be represented in the general matrix form for both symmetric and non-symmetric formulations as

$$s^{\mathrm{CD}}(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e})) =$$

$$\begin{pmatrix} \mathcal{I} & & & & & \\ \frac{1}{2}\mathcal{C} & \mathcal{M} & & & & \\ \mathcal{B} & \mathcal{C} & \mathcal{D} & & & \\ & \ddots & \ddots & \ddots & & \\ & & \mathcal{B} & \mathcal{C} & \mathcal{D} & \\ & & & \mathcal{B} & \mathcal{C} & \mathcal{D} \end{pmatrix} \begin{pmatrix} \mathbf{u}^0 \\ \mathbf{u}^1 \\ \vdots \\ \vdots \\ \mathbf{u}^N \\ \mathbf{u}^{N+1} \end{pmatrix} - \begin{pmatrix} \mathcal{I} & 0 \\ 0 & \Delta t\, \mathcal{B} \\ 0 & 0 \\ \vdots & \vdots \\ \vdots & \vdots \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{e}_0 \\ \mathbf{e}_1 \end{pmatrix} - \Delta t^2 \begin{pmatrix} 0 \\ \frac{1}{2}\mathcal{F}^0 \\ \mathcal{F}^1 \\ \vdots \\ \vdots \\ \mathcal{F}^N \end{pmatrix} = 0, \tag{127}$$

where $\hat{\mathbf{y}} = (\mathbf{u}^0, \mathbf{u}^1, \ldots, \mathbf{u}^N, \mathbf{u}^{N+1})^T$ contains the vectors $\mathbf{u}^i$, the initial condition is $\mathbf{e} = (\mathbf{e}_0, \mathbf{e}_1)^T$, and $\mathcal{F}^i$ is the vector $\mathcal{F}$ at time $t = i\Delta t$. The matrix blocks $\mathcal{B}$, $\mathcal{C}$ and $\mathcal{D}$ are given by the formulas

$$\mathcal{D} = \mathcal{M} + \frac{\Delta t}{2}\mathcal{S}, \tag{128}$$

$$\mathcal{C} = \Delta t^2 \mathcal{K} - 2\mathcal{M}, \tag{129}$$

$$\mathcal{B} = \mathcal{M} - \frac{\Delta t}{2}\mathcal{S}, \tag{130}$$

while $\mathcal{I}$ is the identity matrix. The form (127) is further used to derive the adjoint state equation in Section 6.

Example 5.1.1 As a simplified example we present the efficiency of higher-order space discretization with the central finite difference time discretization on the accuracy of a two-dimensional problem

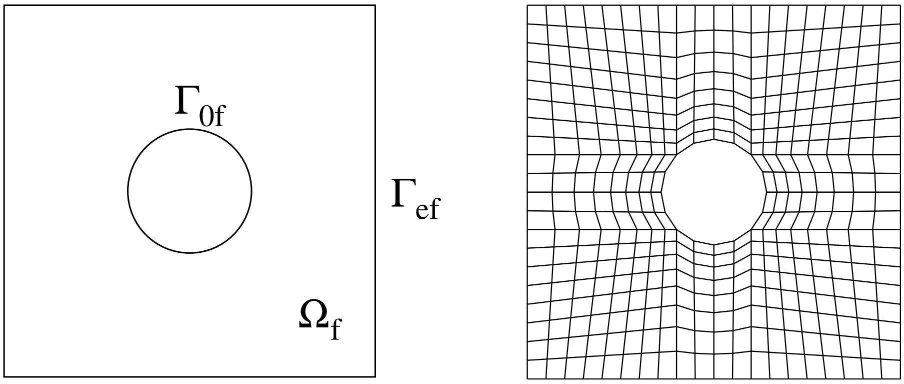
$$\frac{\partial^2 p_f}{\partial t^2} - \nabla^2 p_f = f, \qquad \text{in } Q_f = \Omega_f \times (0, T), \tag{131}$$

$$p_f = \sin(\omega(-2\,t + x_1^2 + x_2^2)), \qquad \text{on } \gamma_{0\mathrm{f}} = \Gamma_{0\mathrm{f}} \times (0, T), \tag{132}$$

$$\frac{\partial p_f}{\partial t} + \frac{\partial p_f}{\partial \mathbf{n}_f} = 0, \qquad \text{on } \gamma_{ef} = \Gamma_{ef} \times (0, T), \tag{133}$$

with the body force

$$f = 4\omega\left(-\cos(\omega(-2t + x_1^2 + x_2^2)) + \omega(-1 + x_1^2 + x_2^2)\sin(\omega(-2\,t + x_1^2 + x_2^2))\right),$$



(a) Computational domain.	(b) The coarsest mesh.

FIGURE 10 Geometry of the domain and the coarsest mesh used.

and the initial conditions

$$\mathbf{e}_0 = p_f(\mathbf{x}, 0) = \sin(\omega(x_1^2 + x_2^2)), \tag{134}$$

$$\mathbf{e}_1 = \frac{\partial p_f(\mathbf{x}, 0)}{\partial t} = -2\omega \cos(\omega(x_1^2 + x_2^2)). \tag{135}$$

The formulation implies that the matrix and vector blocks, as well as the initial values, corresponding to the structure domain vanish in (127). We have set the total time $T = 0.001$ and the number of time steps, each of size $\Delta t = T/N$, is $N = 100$. For the wave number we use $\omega = 4\pi/5$. In the center of the computational domain we have a bounded circular sound-soft obstacle with radius $1/5 \sin(\pi/4)$. The boundary of the scatterer is denoted by Γ_{0f} and the surrounding domain is bounded by an artificial boundary Γ_{ef} as seen in Figure 10(a). The lower left corner of the domain is at the point $(-1.0, -1.0)$ and the upper right corner is at the point $(1.0, 1.0)$. Thus, the analytical solution of the problem is $p_f(\mathbf{x}, t) = \sin(\omega(-2\,t + x_1^2 + x_2^2))$.

We carry out spectral basis order refinement (r-refinement) corresponding to the SEM discretization by using the mesh shown in Figure 10(b) 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 coarsest mesh shown in Figure 10(b) to 2, 3, 4 and 5 mesh stepsizes of equal length. Thus, the numbers of degrees of freedom in these meshes are set to be the same as in the test with r-refinement. Since the time-stepping scheme mainly involves matrix-vector multiplications, the number of degrees of freedom reflecting the mesh density is essential for computational efficiency. That is why the comparison between the FEM and the SEM discretizations is presented, in Figure 11, in terms of the number of degrees of freedom. As the order of the polynomial basis increases, the maximum error between the numerical solution and the analytical solution decreases. The error becomes smaller also with mesh step refinement, but the convergence rate is higher for r-refinement than h-refinement. 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 accuracy.

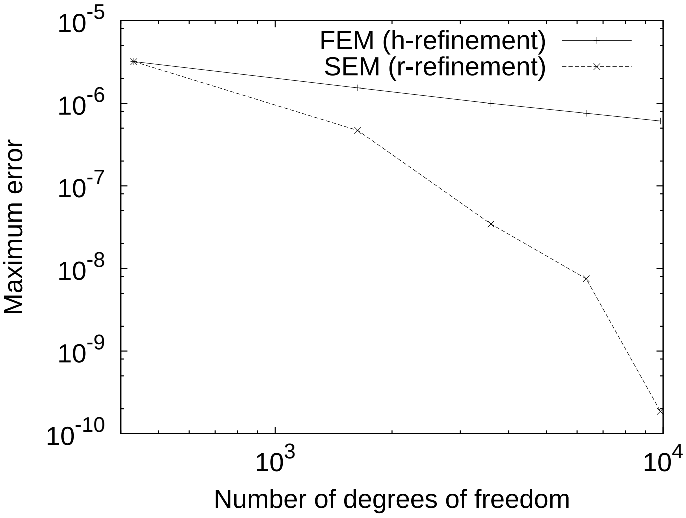


FIGURE 11 Maximum error computed as L^∞-norms versus number of degrees of freedom. The number of timesteps is fixed to be 100.

Remark 5.1.1 Since we use different definitions for the matrices $\mathcal{M}$, $\mathcal{S}$, and $\mathcal{K}$ with the symmetric and the non-symmetric formulation, we arrive to the same state equation (127) for both formulations. Despite that, efficient implementations of these formulations are based on different solution procedures. In the case of the non-symmetric formulation, we at each time step i compute first the displacement $\mathbf{u_s}^i$ and then the pressure $p_f{}^i$ from the equations

$$\begin{aligned}
&\mathcal{M}_s \frac{\mathbf{u_s}^{i+1} - 2\mathbf{u_s}^i + \mathbf{u_s}^{i-1}}{\Delta t^2} + \mathcal{S}_s \frac{\mathbf{u_s}^{i+1} - \mathbf{u_s}^{i-1}}{2\Delta t} + \mathcal{K}_s \mathbf{u_s}^i + \mathcal{A}_{\mathrm{sf}} p_f{}^i = \mathbf{f}_s^i\\
&\mathcal{M}_f \frac{p_f{}^{i+1} - 2p_f{}^i + p_f{}^{i-1}}{\Delta t^2} + \mathcal{S}_f \frac{p_f{}^{i+1} - p_f{}^{i-1}}{2\Delta t} + \mathcal{K}_f p_f{}^i + \mathcal{A}_{\mathrm{fs}} \frac{\mathbf{u_s}^{i+1} - 2\mathbf{u_s}^i + \mathbf{u_s}^{i-1}}{\Delta t^2} = \mathbf{f}_f^i,
\end{aligned}$$

with the initial conditions

$$\begin{aligned}
\mathbf{u_s}^0 &= \mathbf{e}_{s0}, & \frac{\mathbf{u_s}^1 - \mathbf{u_s}^{-1}}{2\Delta t} &= \mathbf{e}_{s1},\\
p_f{}^0 &= \mathbf{e}_{f0}, & \frac{p_f{}^1 - p_f{}^{-1}}{2\Delta t} &= \mathbf{e}_{f1},
\end{aligned}$$

where $\mathbf{u_s}^i$, $p_f{}^i$, $\mathbf{f}_s^i$, and $\mathbf{f}_f^i$ are the vectors $\mathbf{u_s}$, p_f, $\mathbf{f}_s$, and $\mathbf{f}_f$ at $t = i\Delta t$. Because the matrice sums $(\mathcal{M}_s + \frac{\Delta t}{2}\mathcal{S}_s)$ and $(\mathcal{M}_f + \frac{\Delta t}{2}\mathcal{S}_f)$ are diagonal, their inverses are obtained simply by inverting each diagonal element.

The state equation for the formulation with velocity potential and displacement is

$$\begin{aligned}
&\mathcal{M}_s \frac{\mathbf{u_s}^{i+1} - 2\mathbf{u_s}^i + \mathbf{u_s}^{i-1}}{\Delta t^2} + \mathcal{S}_s \frac{\mathbf{u_s}^{i+1} - \mathbf{u_s}^{i-1}}{2\Delta t} + \mathcal{K}_s \mathbf{u_s}^i + \mathcal{A}_{\mathrm{sf}} \frac{\phi^{i+1} - \phi^{i-1}}{2\Delta t} = \mathbf{f}_s^i\\
&\mathcal{M}_f \frac{\phi^{i+1} - 2\phi^i + \phi^{i-1}}{\Delta t^2} + \mathcal{S}_f \frac{\phi^{i+1} - \phi^{i-1}}{2\Delta t} + \mathcal{K}_f \phi^i + \mathcal{A}_{\mathrm{fs}} \frac{\mathbf{u_s}^{i+1} - \mathbf{u_s}^{i-1}}{2\Delta t} = \mathbf{f}_{\phi f}^i,
\end{aligned}$$

with the initial conditions

$$\mathbf{u_s}^0 = \mathbf{e}_{s0}, \qquad \frac{\mathbf{u_s}^1 - \mathbf{u_s}^{-1}}{2\Delta t} = \mathbf{e}_{s1},$$

$$\phi^0 = \mathbf{e}_{\phi f0}, \qquad \frac{\phi^1 - \phi^{-1}}{2\Delta t} = \mathbf{e}_{\phi f1},$$

where $\mathbf{u_s}^i$, ϕ^i, $\mathbf{f}_s^i$, and $\mathbf{f}_{\phi f}^i$ are the vectors $\mathbf{u_s}$, ϕ, $\mathbf{f}_s$, and $\mathbf{f}_{\phi f}$ at $t = i\Delta t$. In this case, $\mathbf{u_s}^{i+1}$ and ϕ^{i+1} need to be solved simultaneously, thus involving the inversion of $(\mathcal{M} + \frac{\Delta t}{2}\mathcal{S})$ at each timestep. If the boundaries of the computational domain consist only of horizontal and vertical lines, the coefficient matrix $(\mathcal{M} + \frac{\Delta t}{2}\mathcal{S})$ needed to invert at each timestep can be implemented as a block matrix consisting of diagonal blocks. Then, $\mathbf{u_s}^i$ and ϕ^i can be solved simply by using matrix-vector multiplications from the formulas

$$\mathbf{u_s}^i = \left(\mathcal{I} - \mathcal{D}_s^{-1}\frac{\Delta t}{2}\mathcal{A}_{\mathrm{sf}}\mathcal{D}_f^{-1}\frac{\Delta t}{2}\mathcal{A}_{\mathrm{fs}}\right)^{-1}\mathcal{D}_s^{-1}\left(y_1 - \frac{\Delta t}{2}\mathcal{A}_{\mathrm{sf}}\mathcal{D}_f^{-1}y_2\right),$$

$$\phi^i = \mathcal{D}_f^{-1}\left(y_2 - \frac{\Delta t}{2}\mathcal{A}_{\mathrm{fs}}\mathbf{u_s}^i\right), i = 1, \ldots, N-1,$$

where $\mathcal{D}_s = \mathcal{M}_s + \frac{\Delta t}{2}\mathcal{S}_s$ and $\mathcal{D}_f = \mathcal{M}_f + \frac{\Delta t}{2}\mathcal{S}_f$ are diagonal matrices, and

$$y_1 = \left(2\mathcal{M}_s - \Delta t^2\mathcal{K}_S\right)\mathbf{u_s}^i + \left(\frac{\Delta t}{2}\mathcal{S}_S - \mathcal{M}_s\right)\mathbf{u_s}^{i-1} + \Delta t^2\mathbf{f}_s^i + \frac{\Delta t}{2}\mathcal{A}_{\mathrm{sf}}\phi^{i-1},$$

$$y_2 = \left(2\mathcal{M}_f - \Delta t^2\mathcal{K}_f\right)\phi^i + \left(\frac{\Delta t}{2}\mathcal{S}_f - \mathcal{M}_f\right)\phi^{i-1} + \Delta t^2\mathbf{f}_{\phi f}^i + \frac{\Delta t}{2}\mathcal{A}_{\mathrm{fs}}\mathbf{u_s}^{i-1}.$$

It would also be possible to uncenter one of the two first-order derivatives in time to uncouple the problem. That approach would involve first-order difference approximations and introduce some dissipation. That is why we neglect deeper considerations of the uncentered schemes.

Example 5.1.2 We illustrate the computational cost of both symmetric and non-symmetric formulation by solving a time-dependent fluid-structure interaction problem both expressing the acoustic wave equation by the pressure and by using the velocity potential in the fluid domain. The right-hand sides and initial conditions in Equations (55)-(66) and (76)-(87) are defined to satisfy the analytical solution $p_f = \omega\rho_f(\mathbf{x})\sin(\boldsymbol{\omega}\cdot\mathbf{x})\cos(\omega t)$, $\phi = -\sin(\boldsymbol{\omega}\cdot\mathbf{x})\sin(\omega t)$, 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 problem is solved in a domain, which consists of the solid part $\Omega_s = [-1, 0] \times [0, 1]$ and the fluid part $\Omega_f = [0, 1] \times [0, 1]$ (see Figure 12). We use square-element meshes with mesh stepsize $h = 0.1$, and the element order is increased in both parts of the domain from 1 to 5. The meshes are matching on the coupling interface Γ_i set at $x_1 = 0$ for $x_2 \in [0, 1]$. On the other boundaries we have 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 solid domain, we use the values $c_p(\mathbf{x}) = 6.20$, $c_s(\mathbf{x}) = 3.12$, and $\rho_s(\mathbf{x}) = 2.7$. The angular frequency $\omega = 4\pi$ is the same for both media, and we set the propagation direction $(1,0)$ by the vector $\boldsymbol{\omega} = (\omega_1, \omega_2) = (1,0)\omega$. The time interval $[0.0, 0.5]$ is divided into 400 steps, each of size $\Delta t = 0.00125$, to guarantee the stability condition also for the higher element orders.

When the pressure formulation is considered, the time marching involves only matrix-vector multiplications, and actual matrix inversions are not needed. In the implementation of the velocity potential formulation, a ”one-shot” method is required for solving the linear system including $\mathbf{u_s}^{i+1}$ and ϕ^{i+1} at each timestep $i = 1, \ldots, N$. In that case, the matrix which is needed to be inverted is stored either as a band matrix or by using the compressed column storage including only the non-zero matrix elements.

The Lapack LU decomposition routines `dgbtrf` and `dgbtrs` use the band matrix storage mode. With these routines the memory and CPU time consumption increases rapidly when the element order is increased (see Figure 13). The SuperLU library routines perform an LU decomposition with partial pivoting, and the triangular system solves through forward and backward substitution. At this stage, we also utilize the sparsity of the matrix by using the compressed column storage mode. Since all the non-zero elements are not near the diagonal of the matrix, the compressed column storage mode requires less storage and computing operations than the band storage mode. That is why the SuperLU library gives a less demanding procedure for solving the linear system than the Lapack library. This is because the matrices arising from the space discretization and including the coupling terms have, in general, a sufficiently large band width. Thus, a remarkably larger amount of memory is needed for storing the sparse coefficient matrix in the band matrix form used in conjunction with Lapack routines than in the compressed column storage utilized with the SuperLU. Consequently, less time is needed when fewer elements are employed in the solution procedure with the SuperLU. The performance of the Lapack routines could be improved, for instance, by using an appropriate node numbering of the mesh.

We conclude that the computational efforts are of the same order of magnitude whether the problem in the fluid domain is solved with respect to pressure or whether we use velocity potential formulation in conjunction with the linear solver provided by the SuperLU library. The results of these experiments are carried out on an AMD Opteron 885 processor at 2.6 GHz.

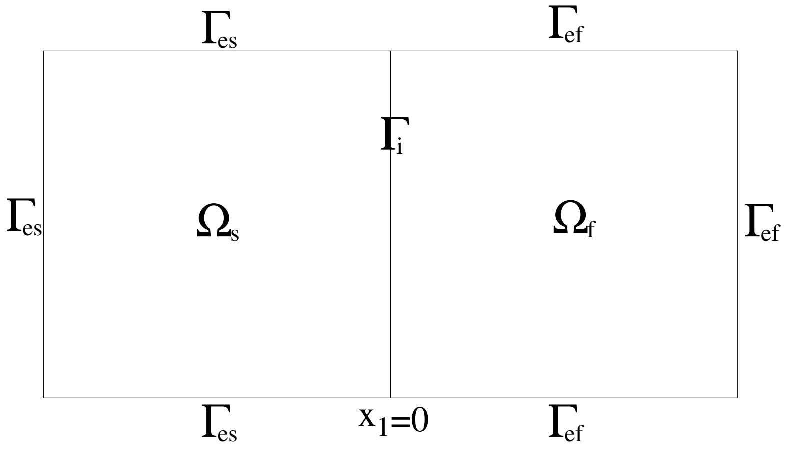
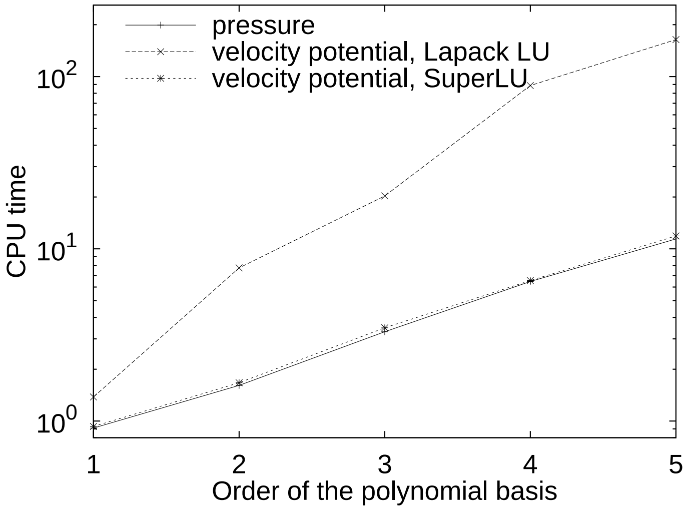
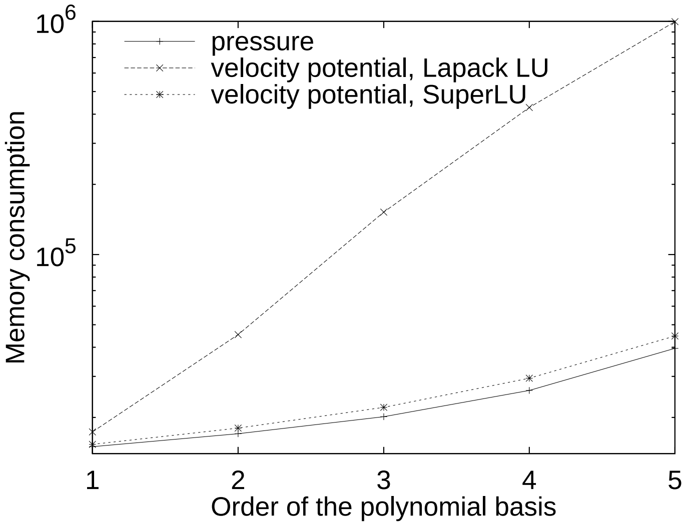


FIGURE 12 The domain Ω is divided into the solid part Ω_s and the fluid part Ω_f.



(a) CPU time with respect to the element order.



(b) Memory consumption with respect to the element order.

FIGURE 13 CPU time (in seconds) and memory (in kilobytes) consumed for solving a time-dependent fluid-structure interaction problem with different formulations. The number of timesteps is fixed to be 400, and square-element meshes with mesh stepsize $h = 0.1$ are used in both media.

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 (see Figure 14). This error level, shown as a horizontal line in Figure 14, reflects the error level of time discretization. Since we use a fixed number of timesteps at all element orders, the error of time discretization becomes dominant for higher-order elements. Hence, finer timesteps or higher-order time discretizations are needed in conjunction with higher-order elements.

In principle, both symmetric and non-symmetric formulation should lead to the same order of accuracy at each element order. The results obtained in the solid domain and depicted in Figure 14(a) are perfectly in balance with this hypothesis. However, in Figure 14(b) we see that in the fluid domain higher accuracy is obtained with the velocity potential than with the pressure formulation at each element order. One reason for that might be the derivation of the interface conditions. In the case of pressure formulation, the displacement components in the formula (49) are differentiated twice with respect to time. From the physical point of view, some information is lost in that procedure, and linear growth of the displacement with respect to time is not eliminated at the interface. However, the solutions of the pressure formulation seem to converge towards the analytical solution. Although the pressure formulation gives less accurate results than the velocity potential formulation, it is still a plausible choice at some point. For instance, the implementation process can be hastened when the pre-existing solvers of acoustic and elastic problems can be harnessed in the implementation, and no additional linear solvers are needed.

In what follows, we concentrate on further developing the symmetric formulation expressing the acoustic wave equation by the velocity potential in the fluid domain. In the forthcoming examples with the central finite difference time-stepping, we use the implementation in which the linear solver provided by the SuperLU library is used. The non-symmetric pressure formulation is utilized for comparison purposes in Chapter 8.

5.2 Fourth-order Runge–Kutta method

The state equation (123) can be presented as a system of differential equations

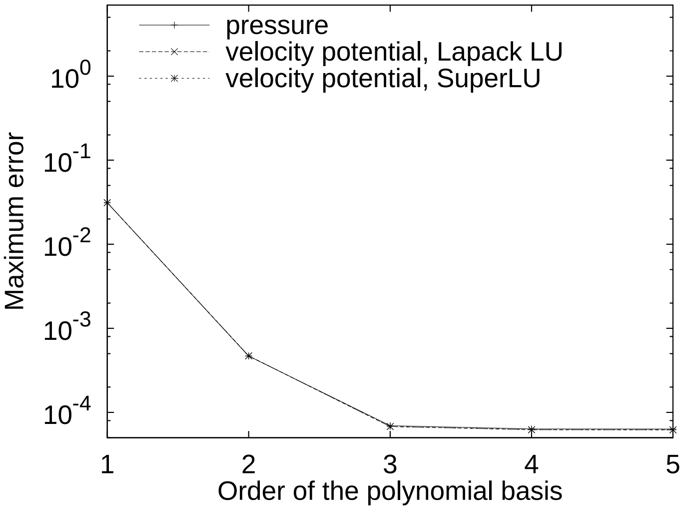
$$\frac{\partial \mathbf{y}}{\partial t} = f(t, \mathbf{y}(t)), \tag{136}$$

where $\mathbf{y} = (\mathbf{u}, v)^T$ is a vector of time-stepping variables $\mathbf{u}$ and $v = \frac{\partial \mathbf{u}}{\partial t}$, and the function $f(t, y(t)) = (f_1(t, \mathbf{u}, v), f_2(t, \mathbf{u}, v))^T$ has components

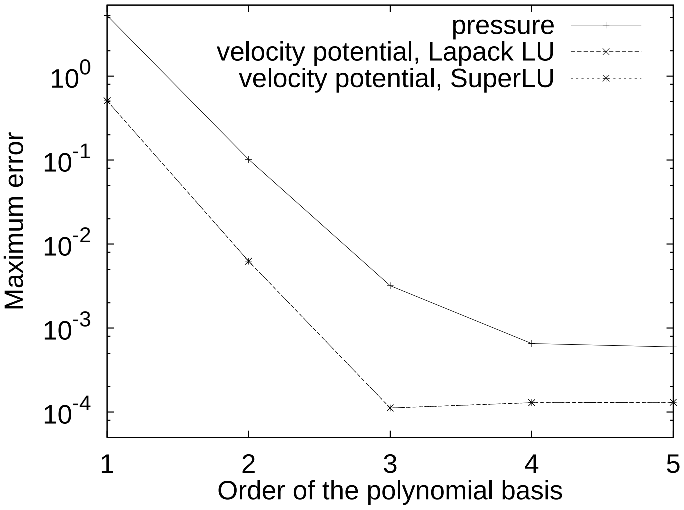
$$f_1(t, \mathbf{u}, v) = v, \tag{137}$$

$$f_2(t, \mathbf{u}, v) = -\mathcal{M}^{-1} \left(\mathcal{S} v + \mathcal{K} \mathbf{u} - \mathcal{F} \right). \tag{138}$$

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



(a) Solid domain.



(b) Fluid domain.

FIGURE 14 Maximum errors, computed as L^{∞}-norms, with respect to the element order. The number of timesteps is fixed to be 400, and square-element meshes with mesh stepsize $h = 0.1$ are used in both media.

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 ith time step is defined as

$$\mathbf{y}^i = \mathbf{y}^{i-1} + \frac{\Delta t}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right), \tag{139}$$

where $\mathbf{y}^i = \left(\mathbf{u}^i, \frac{\partial \mathbf{u}^i}{\partial t}\right)^T$ contains the global block vector $\mathbf{u}^i$, including the values of the variables in both the fluid and the structure domain at the ith timestep, and its derivative $v^i = \frac{\partial \mathbf{u}^i}{\partial t}$ 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:

$$\begin{pmatrix} k_{11} \\ k_{12} \end{pmatrix} = \begin{pmatrix} f_1(i\Delta t, \mathbf{u}^i, v^i) \\ f_2(i\Delta t, \mathbf{u}^i, v^i) \end{pmatrix}, \tag{140}$$

$$\begin{pmatrix} k_{21} \\ k_{22} \end{pmatrix} = \begin{pmatrix} f_1(i\Delta t + \frac{\Delta t}{2}, \mathbf{u}^i + \frac{k_{11}}{2}, v^i + \frac{k_{12}}{2}) \\ f_2(i\Delta t + \frac{\Delta t}{2}, \mathbf{u}^i + \frac{k_{11}}{2}, v^i + \frac{k_{12}}{2}) \end{pmatrix}, \tag{141}$$

$$\begin{pmatrix} k_{31} \\ k_{32} \end{pmatrix} = \begin{pmatrix} f_1(i\Delta t + \frac{\Delta t}{2}, \mathbf{u}^i + \frac{k_{21}}{2}\, v^i + \frac{k_{22}}{2}) \\ f_2(i\Delta t + \frac{\Delta t}{2}, \mathbf{u}^i + \frac{k_{21}}{2}\, v^i + \frac{k_{22}}{2}) \end{pmatrix}, \tag{142}$$

$$\begin{pmatrix} k_{41} \\ k_{42} \end{pmatrix} = \begin{pmatrix} f_1(i\Delta t + \Delta t, \mathbf{u}^i + k_{31}, v^i + k_{32}) \\ f_2(i\Delta t + \Delta t, \mathbf{u}^i + k_{31}, v^i + k_{32}) \end{pmatrix}. \tag{143}$$

In other words, in order to get the differential estimates (140)-(143), the function f is evaluated at each time step four times by using the formulas (137)-(138), and then the successive approximation of $\mathbf{y}$ is calculated by the formula (139). To make the application of the adjoint equation technique in Section 6 more convenient, we present the fully discrete state equation in the case of the Runge–Kutta time discretization as

$$s^{\mathrm{RK}}(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e})) = \begin{pmatrix} \mathcal{I} & & & & \\ \mathcal{N} & \mathcal{I} & & & \\ & \ddots & \ddots & & \\ & & \mathcal{N} & \mathcal{I} & \\ & & & \mathcal{N} & \mathcal{I} \end{pmatrix} \begin{pmatrix} \mathbf{y}^0 \\ \mathbf{y}^1 \\ \vdots \\ \mathbf{y}^{N-1} \\ \mathbf{y}^N \end{pmatrix} - \begin{pmatrix} \mathcal{I} \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \mathbf{e} - \begin{pmatrix} 0 \\ \hat{\mathcal{F}}^1 \\ \vdots \\ \hat{\mathcal{F}}^{N-1} \\ \hat{\mathcal{F}}^N \end{pmatrix} = 0, \tag{144}$$

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

$$\mathcal{N} = -\begin{pmatrix} \hat{\mathcal{C}} \\ 2\hat{\mathcal{C}} \\ 2\hat{\mathcal{C}} \\ \hat{\mathcal{C}} \end{pmatrix}^T \begin{pmatrix} \mathcal{I} & & & \\ \hat{\mathcal{B}} & \mathcal{I} & & \\ & \hat{\mathcal{B}} & \mathcal{I} & \\ & & 2\hat{\mathcal{B}} & \mathcal{I} \end{pmatrix}^{-1} \begin{pmatrix} 2\hat{\mathcal{B}} \\ 2\hat{\mathcal{B}} \\ 2\hat{\mathcal{B}} \\ 2\hat{\mathcal{B}} \end{pmatrix} - \mathcal{I}, \tag{145}$$

$$\hat{\mathcal{F}}^i = -\begin{pmatrix} \hat{\mathcal{C}} \\ 2\hat{\mathcal{C}} \\ 2\hat{\mathcal{C}} \\ \hat{\mathcal{C}} \end{pmatrix}^T \begin{pmatrix} \mathcal{I} & & & \\ \hat{\mathcal{B}} & \mathcal{I} & & \\ & \hat{\mathcal{B}} & \mathcal{I} & \\ & & 2\hat{\mathcal{B}} & \mathcal{I} \end{pmatrix}^{-1} \begin{pmatrix} \hat{\mathcal{D}}^{i-1} \\ \hat{\mathcal{D}}^{i-\frac{1}{2}} \\ \hat{\mathcal{D}}^{i-\frac{1}{2}} \\ \hat{\mathcal{D}}^{i} \end{pmatrix}. \tag{146}$$

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 $t = i\Delta t$. The block-matrix form (144) of the fully discrete state equation with the RK time-stepping is analogous to the state equation (127). 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

$$\begin{pmatrix} \mathcal{I} & & & \\ \hat{\mathcal{H}} & \mathcal{I} & & \\ & \hat{\mathcal{H}} & \mathcal{I} & \\ & & 2\hat{\mathcal{H}} & \mathcal{I} \end{pmatrix} \begin{pmatrix} k_1 \\ k_2 \\ k_3 \\ k_4 \end{pmatrix} + \begin{pmatrix} 2\hat{\mathcal{H}} \\ 2\hat{\mathcal{H}} \\ 2\hat{\mathcal{H}} \\ 2\hat{\mathcal{H}} \end{pmatrix} \mathbf{y}^{i-1} - \begin{pmatrix} \hat{\mathcal{D}}^{i-1} \\ \hat{\mathcal{D}}^{i-\frac{1}{2}} \\ \hat{\mathcal{D}}^{i-\frac{1}{2}} \\ \hat{\mathcal{D}}^{i} \end{pmatrix} = 0, \tag{147}$$

where the matrix blocks $\hat{\mathcal{H}}$ are given by the formula

$$\hat{\mathcal{H}} = \begin{pmatrix} 0 & -\frac{\Delta t}{2}\mathcal{I} \\ \frac{\Delta t}{2}\mathcal{M}^{-1}\mathcal{K} & \frac{\Delta t}{2}\mathcal{M}^{-1}\mathcal{S} \end{pmatrix}. \tag{148}$$

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

$$\mathbf{y}^i = \mathbf{y}^{i-1} - \begin{pmatrix} \hat{R} & 2\hat{R} & 2\hat{R} & \hat{R} \end{pmatrix} \begin{pmatrix} k_1 \\ k_2 \\ k_3 \\ k_4 \end{pmatrix}, \qquad \hat{R} = \begin{pmatrix} -\frac{1}{6}\mathcal{I} & 0 \\ 0 & -\frac{1}{6}\mathcal{I} \end{pmatrix}. \tag{149}$$

If the matrix $\mathcal{M}$ is diagonal, as it is in the formulation with the velocity potential, the only matrix inversion needed in time-stepping (e.g., $\mathcal{M}^{-1}$ in Equation (138)) 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 and coupling terms, 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 the 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 (139), the 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 the RK time-stepping. Although the computational cost is of the same order for both the CD and the RK time-steppings, the number of floating point operators needed for the RK is nearly four times that of the CD.

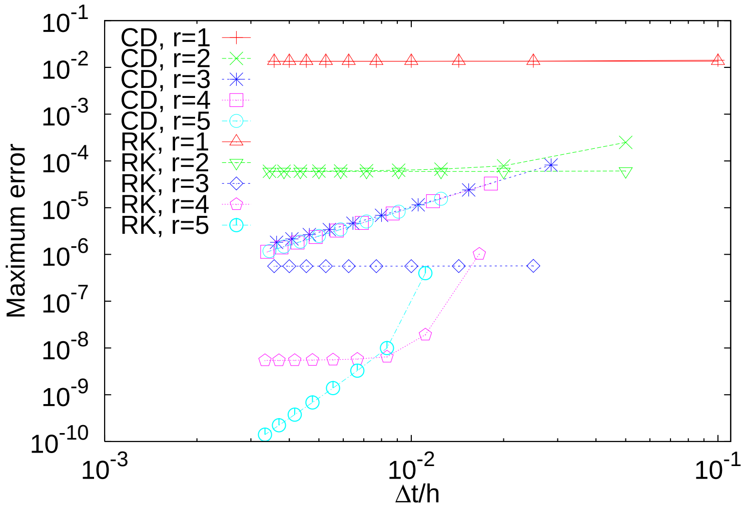
Example 5.2.1 We demonstrate how the efficiency of the method can be improved by using the fourth-order Runge–Kutta scheme instead of the central finite difference time discretization. In principle, the error of time discretization in the central finite difference scheme is of order $O(\Delta t^3)$, whereas in the fourth-order Runge–Kutta method it is of order $O(\Delta t^5)$. However, the error of space discretization limits the accuracy of the overall time-stepping scheme as well, and the stability issue restricts choosing a feasible length of the timestep. To better illustrate the error of time and space discretization, we continue with Example 5.1.2, except that we change the mesh stepsize to $h = 1/20$ and use various timestep lengths to observe stability and accuracy issues. The number of timesteps needed for stability is first determined numerically by using $50i$ timesteps per time period, for $i = 1, 2, 3, \ldots$, until a stable solution is achieved. From these results, we can define stability constant α_r for each element order r such that

$$\frac{\Delta t}{h} = \frac{\alpha_r}{\max\{c, c_p, c_s\}\sqrt{2}}. \tag{150}$$

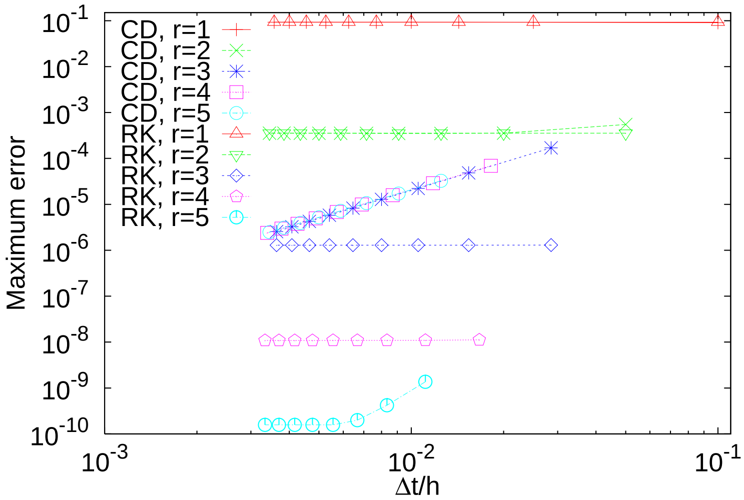
The stability conditions corresponding to the largest stable timestep are given in Table 2. The stability region seems to be exactly the same with both the CD and the RK time-stepping for the element orders $r = 1, 2, 3$ and differs only slightly for the higher-order elements. The results reflect the well known CFL condition, and are in good agreement with the experiments presented for acoustic waves with the CD time discretization by Cohen in [44].

TABLE 2 Stability conditions for the CD and the RK time discretization schemes.

	r	1	2	3	4	5
Number of timesteps	CD	100	200	350	550	800
	RK	100	200	350	600	900
α_r	CD	0.8765	0.4383	0.2504	0.1594	0.1096
	RK	0.8765	0.4383	0.2504	0.1461	0.0974



(a) Solid domain.



(b) Fluid domain.

FIGURE 15 Comparison between maximum errors with the CD and the RK time discretizations. The mesh stepsize is fixed to be $h = 1/20$, and the timestep refinement gives a series of numerical results with various lengths of the timestep for each element order.

We start the computations with the largest stable timestep and then repeatedly add the number of timesteps $N = T/\Delta t$ by 300, until their number is larger than 3000. Proceeding in this way, for each element order we achieve a series of numerical results with various lengths of the timestep. The maximum errors between the numerical and the analytical solution with respect to $\Delta t/h$ are computed as L^∞-norms. Accuracy of the numerical solution is shown in Figure 15 as a function of the ratio between the time step Δt and the mesh step size h for both the CD and the RK time-steppings with five element orders r. 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. It is worth mentioning that errors are somewhat smaller in the solid domain than in the fluid domain.

It is seen that with the element orders $r = 1$ and $r = 2$ both time-stepping schemes give the same accuracy even when sufficiently large timesteps are used. Moreover, this is the accuracy of the space discretization since the error of spatial discretization dominates with low-order elements. In other words, the temporal error is eliminated, and the maximum error with respect to the length of the timestep is not decreasing significantly even if smaller timesteps are used.

Naturally, for each element order r the solution with the RK time-stepping is at least as accurate as the one computed with the CD time-stepping. When higher-order elements are used, results computed with the RK time discretization are more accurate than the ones computed with the CD time discretization. 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 the CD time discretization with basis orders $r \geq 3$. In principle, also with these higher-order elements, the error curves turn to horizontal lines, reflecting the accuracy of the space discretization, when the length of the timestep is refined enough. In the case of the RK time discretization, the error of space discretization is dominant even when long timesteps are used. With the RK time discretization, very fine timesteps are needed only for $r = 5$ in the solid domain to achieve the error level of spatial discretization.

With the help of the results depicted in Figure 15 we can extrapolate that very fine timesteps are needed with the CD time discretization to eliminate the temporal error when higher-order elements are involved. For $r = 3$, we would already need thousands of timesteps with the CD time discretization to get the same accuracy as with the RK time discretization with 350 timesteps. Tens of thousands timesteps are required to eliminate the temporal error with the CD time discretization with $r = 4$. Respectively, for $r = 5$ we need hundreds of thousands timesteps to achieve the error level of space discretization. The values of k_r, satisfying

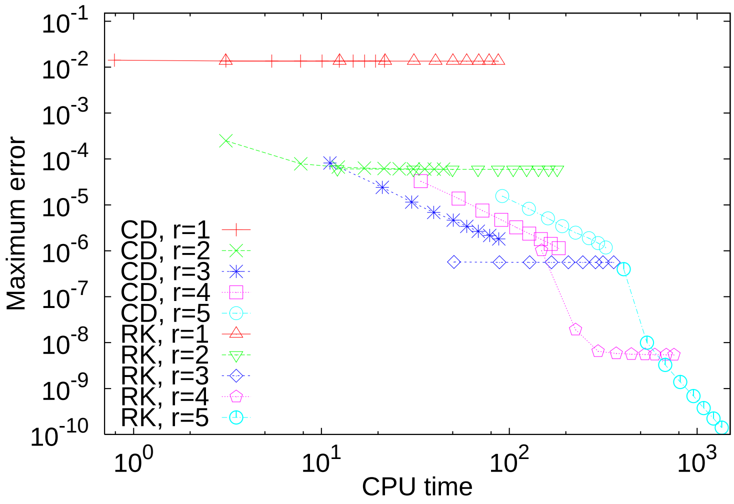
$$\frac{\Delta t}{h} = \frac{k_r}{\max\{c, c_p, c_s\}\sqrt{2}}, \tag{151}$$

and eliminating the temporal error, are reported for different element orders in Table 3.

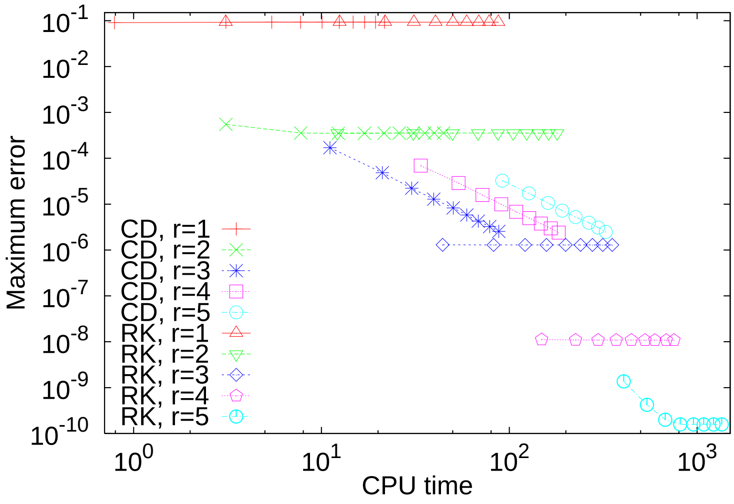
Since refining the timesteps means more evaluations, we are also interested in measuring the CPU time needed in these simulations. The tests about CPU time consumption are carried out on an AMD Opteron 885 processor at 2.6 GHz, and the results are seen in Figure 16. To eliminate the temporal error for $r = 1$ and $r = 2$, less computation time is needed with the CD time discretization than with the RK time discretization. With

TABLE 3 Constants k_r eliminating the temporal error with the CD and the RK time discretizations.

	r	1	2	3	4	5
k_r	CD	0.8765	0.1096	0.0167	0.0017	0.0003
	RK	0.8765	0.4383	0.2191	0.0730	0.0266



(a) Solid domain.



(b) Fluid domain.

FIGURE 16 Accuracy with respect to CPU time consumptions (in seconds) with the CD and the RK time discretizations. The mesh stepsize is fixed to be $h = 1/20$, and the timestep refinement gives a series of numerical results with various lengths of the timestep for each element order.

higher-order elements, more remarkable time saving occurs by using the RK time discretization to attain the accuracy of the space discretization. Although the difference in time consumption is not significant with sufficiently small number of degrees of freedom, it can play an important role in large-scale real-life applications.

5.3 Fourth-order Adams–Bashforth method

Next, we make an effort for decreasing the computing time and still maintaining the high accuracy provided by higher-order time discretizations. For this purpose, we present the fourth-order Adams–Bashforth method based on approximating the functions $f(t, \mathbf{y}(t))$ by interpolating polynomials. It gives the solution $\mathbf{y}$ at the ith time step as

$$\begin{aligned}\mathbf{y}^i = \mathbf{y}^{i-1} + \frac{h}{24}\Big(55\, f\left((i-1)\Delta t, \mathbf{y}^{i-1}\right) - 59\, f\left((i-2)\Delta t, \mathbf{y}^{i-2}\right) \\ + 37\, f\left((i-3)\Delta t, \mathbf{y}^{i-3}\right) - 9\, f\left((i-4)\Delta t, \mathbf{y}^{i-4}\right)\Big),\end{aligned} \tag{152}$$

where $\mathbf{y}^i = \left(\mathbf{u}^i, \frac{\partial \mathbf{u}^i}{\partial t}\right)^T$ contains the vector $\mathbf{u}^i$ and its derivative $v^i = \frac{\partial \mathbf{u}^i}{\partial t}$ at time $t = i\Delta t$, $i = 4, \ldots, N$. Hence, it is a multistep method that requires information at four previous time steps implying that another method is needed for starting the time marching. At this stage, we utilize the fourth-order Runge–Kutta method for computing the start-up values $\mathbf{y}^0$, $\mathbf{y}^1$, $\mathbf{y}^2$, and $\mathbf{y}^3$ to initialize the multistep method. The state equation for the scheme is

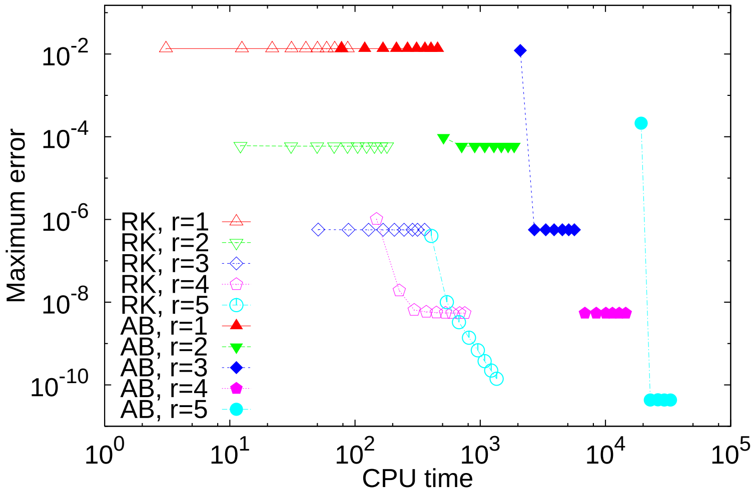
$$\begin{aligned} s^{\mathrm{AB}}(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e})) = \begin{pmatrix} \mathcal{I} & & & & & & \\ \mathcal{N} & \mathcal{I} & & & & & \\ & \mathcal{N} & \mathcal{I} & & & & \\ & & \mathcal{N} & \mathcal{I} & & & \\ -\frac{9}{12}\hat{\mathcal{B}} & \frac{37}{12}\hat{\mathcal{B}} & -\frac{59}{12}\hat{\mathcal{B}} & \frac{55}{12}\hat{\mathcal{B}} - \mathcal{I} & \mathcal{I} & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & -\frac{9}{12}\hat{\mathcal{B}} & \frac{37}{12}\hat{\mathcal{B}} & -\frac{59}{12}\hat{\mathcal{B}} & \frac{55}{12}\hat{\mathcal{B}} - \mathcal{I} & \mathcal{I} \end{pmatrix} \begin{pmatrix} \mathbf{y}^0 \\ \mathbf{y}^1 \\ \mathbf{y}^2 \\ \mathbf{y}^3 \\ \mathbf{y}^4 \\ \vdots \\ \mathbf{y}^N \end{pmatrix} \\ - \begin{pmatrix} \mathcal{I} \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \mathbf{e} - \begin{pmatrix} 0 \\ \hat{\mathcal{F}}^1 \\ \hat{\mathcal{F}}^2 \\ \hat{\mathcal{F}}^3 \\ 0 \\ \vdots \\ 0 \end{pmatrix} - \begin{pmatrix} 0 & & & & & & \\ & 0 & & & & & \\ & & 0 & & & & \\ & & & 0 & & & \\ \frac{9}{4}\hat{\mathcal{C}} & -\frac{37}{4}\hat{\mathcal{C}} & \frac{59}{4}\hat{\mathcal{C}} & -\frac{55}{4}\hat{\mathcal{C}} & 0 & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & \frac{9}{4}\hat{\mathcal{C}} & -\frac{37}{4}\hat{\mathcal{C}} & \frac{59}{4}\hat{\mathcal{C}} & -\frac{55}{4}\hat{\mathcal{C}} & 0 \end{pmatrix} \begin{pmatrix} 0 \\ \hat{\mathcal{D}}^1 \\ \hat{\mathcal{D}}^2 \\ \hat{\mathcal{D}}^3 \\ \hat{\mathcal{D}}^4 \\ \vdots \\ \hat{\mathcal{D}}^N \end{pmatrix} = 0. \end{aligned} \tag{153}$$

Example 5.3.1 Only one evaluation of the function $f(t, \mathbf{y}(t))$ is needed at each timestep. The values are stored to be used in the four following timesteps meaning that the memory consumption is of the same order of magnitude than when the fourth-order Runge–Kutta method is utilized. In theory, employing the fourth-order Adams–Bashforth time-stepping scheme instead of the fourth-order Runge–Kutta method saves time. Nevertheless, the

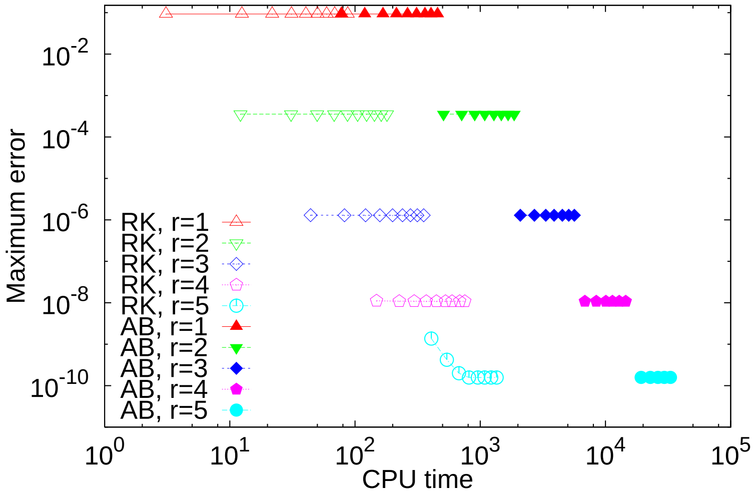
TABLE 4 Stability conditions for the AB time discretization scheme.

	r	1	2	3	4	5
Number of timesteps	AB	500	1550	3250	5450	8250
α_r	AB	0.1753	0.0565	0.0270	0.0161	0.0106

stability region is smaller for the fourth-order Adams–Bashforth than the fourth-order Runge–Kutta method. The largest stable timesteps with the Adams–Bashforth method, determined numerically in the same way as it is done in the previous example, are reported in Table 4. This practical realization shows that the length of the timestep that guarantees the stability conditions is much smaller with the Adams–Bashforth method than with the Runge–Kutta method. The computations are continued by carrying out the simulations with the Adams–Bashforth time discretization with smaller timesteps. That is, for the element order r the addition of $300r$ timesteps is repeated until the number of timesteps is larger than $3000r$. The results are presented in Figure 17 as accuracy with respect to CPU time consumption. For comparison, the results of the RK time discretization



(a) Solid domain.



(b) Fluid domain.

FIGURE 17 Accuracy with respect to CPU time consumptions (in seconds) with the RK and the AB time discretizations. The mesh stepsize is fixed to be $h = 1/20$, and the timestep refinement gives a series of numerical results with various lengths of the timestep for each element order.

from the previous example are also depicted there. Clearly, stability issues deteriorate the efficiency of the AB time-stepping scheme. The CPU time used for the computation is in favor of the RK time discretization, namely more than an order of magnitude longer time is consumed to get the same accuracy with the AB time discretization than with the RK time discretization. We can conclude that no improvement is achieved by utilizing the AB time discretization.

6 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

$$\mathbf{u}(\mathbf{x},0) = \mathbf{e}_0, \quad \frac{\partial \mathbf{u}(\mathbf{x},0)}{\partial t} = \mathbf{e}_1. \tag{154}$$

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

$$\mathbf{u}(\mathbf{x},T) = \mathbf{e}_0, \quad \frac{\partial \mathbf{u}(\mathbf{x},T)}{\partial t} = \mathbf{e}_1. \tag{155}$$

The purpose of optimal control problems is to minimize an objective functional (cost function) J defined in a control space Z. In practice, Z is a function space containing real or vector-valued functions.

6.1 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 formulations given in Chapter 3. In this section we present, as an example, how the exact controllability approach is applied to the symmetric fluid-structure interaction between the velocity potential and the displacement. In principle, the same stages can be used in conjunction with the other formulations. However, since the pressure-displacement formulation is non-symmetric, J can be derived straightforwardly for that formulation only for the 1D model.

In order to define the optimal control, an objective functional corresponding to the energy formulation (92) is defined as

$$\begin{aligned} J(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e})) =& \frac{1}{2} \int\limits_{\Omega_f} \left(\rho_f(\mathbf{x}) \left| \nabla(\phi(\mathbf{x}, T) - \mathbf{e}_{f0}) \right|^2 + \frac{\rho_f(\mathbf{x})}{c(\mathbf{x})^2} \left| \frac{\partial \phi(\mathbf{x}, T)}{\partial t} - \mathbf{e}_{f1} \right|^2 \right) dx \\ &+ \frac{1}{2} \int\limits_{\Omega_s} \left(2\mu_s \left| \epsilon(\mathbf{u_s}(\mathbf{x}, T) - \mathbf{e}_{s0}) \right|^2 + \lambda_s \left| \nabla \cdot (\mathbf{u_s}(\mathbf{x}, T) - \mathbf{e}_{s0}) \right|^2 \right) dx \\ &+ \frac{1}{2} \int\limits_{\Omega_s} \rho_s(\mathbf{x}) \left| \frac{\partial \mathbf{u_s}(\mathbf{x}, T)}{\partial t} - \mathbf{e}_{s1} \right|^2 dx, \end{aligned} \tag{156}$$

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 (156) is

$$J(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e})) = \frac{1}{2} \left(\mathbf{y}^N - \mathbf{e} \right)^T \begin{pmatrix} \mathcal{K}_s & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathcal{K}_f & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{M}_s & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{M}_f \end{pmatrix} \left(\mathbf{y}^N - \mathbf{e} \right), \tag{157}$$

where $\mathbf{y}^i$ are given by Equation (127) or (144).

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

$$\min_{\mathbf{e} \in Z} J(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e})), \tag{158}$$

where $\hat{\mathbf{y}}(\mathbf{e})$ solves the transient initial value problem (state equation (127) or (144)) and $J(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e}))$ is the discretized objective functional. The result of the minimization problem (158) 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 (158) is equivalent to finding initial conditions $\mathbf{e}^* \in Z$

TABLE 5 Number of degrees of freedom, number of timesteps for the time period $T = 1/2$, and CPU time consumption (in seconds) used for computing the gradient by the central finite difference approximation for different spectral orders r with the RK time discretization scheme and angular frequency $\omega = 4\pi$ and mesh stepsize $h = 1/20$.

r	1	2	3
Number of degrees of freedom	1322	5043	11163
Number of timesteps	100	200	400
CPU time	$1.80 \cdot 10^4$	$2.47 \cdot 10^5$	$2.24 \cdot 10^6$

such that the gradient of J is zero, that is,

$$\nabla J(\mathbf{e}^*, \hat{\mathbf{y}}(\mathbf{e}^*)) = 0. \tag{159}$$

Since J is a quadratic functional, (159) 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 $\mathbf{e}_0 = (\mathbf{e}_{0_1}, \mathbf{e}_{0_2}, \ldots \mathbf{e}_{0_{(2N_s+N_f)}})^T$ and $\mathbf{e}_1 = (\mathbf{e}_{1_1}, \mathbf{e}_{1_2}, \ldots \mathbf{e}_{1_{(2N_s+N_f)}})^T$. One option for computing the gradient would be using the central finite difference approximation of the gradient of the objective functional, that is,

$$\frac{dJ(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e}))}{d\mathbf{e}_{k_i}} \approx \frac{J(\mathbf{e} + \eta \hat{e}^{\hat{N}k+i}, \hat{\mathbf{y}}(\mathbf{e} + \eta \hat{e}^{\hat{N}k+i})) - J(\mathbf{e} - \eta \hat{e}^{\hat{N}k+i}, \hat{\mathbf{y}}(\mathbf{e} - \eta \hat{e}^{\hat{N}k+i}))}{2\eta}, \tag{160}$$

where $k = 0, 1$, $\hat{N} = 2N_s + N_f$, $i = 1, \ldots, \hat{N}$, and η is the step length to the direction of the $(\hat{N}k + i)$th elementary vector $\hat{e}^{\hat{N}k+i}$, and J is given by Equation (157). In that case, computing the gradient of the objective functional (157) requires solving the state equation four times per each degree of freedom.

Example 6.1.1 As the number of degrees of freedom grows, the computing time becomes insufficiently long. To demonstrate this, we apply the central finite difference gradient computation, defined by Equation (160), to the problem presented in Example 5.2.1 with the RK time discretization (i.e., the variable $\mathbf{y}^N$ shown in Equation (157) is computed by Equation (144)). The CPU time consumption is presented in Table 5 for the element orders $r = 1, 2, 3$. The lengths of the timestep are chosen to eliminate the temporal error according to Table 3. For the spatial difference step we use $\eta = 0.01$, which affects the accuracy of the finite difference approximation but not the CPU time consumption.

In order to implement an efficient algorithm, we proceed in more practical way and compute the derivative of J by the adjoint equation technique. For condensing the formulation, we represent the state equations (127) and (144) in the generic form

$$s(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e})) = 0, \tag{161}$$

where $\mathbf{e} = (\mathbf{e}_0, \mathbf{e}_1)^T$ contains the initial values and $\hat{\mathbf{y}}$ contains the vectors $\mathbf{u}^i$ in the case of Equation (127) or the vectors $\mathbf{y}^i = \left(\mathbf{u}^i, \frac{\partial \mathbf{u}^i}{\partial t}\right)^T$ in the case of Equation (144). The state equation (161) is also called the forward equation because it is solved by advancing forward in time. 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., [139]), we see that

$$\frac{dJ(\mathbf{e}, \hat{\mathbf{y}}(\mathbf{e}))}{d\mathbf{e}_k} = \frac{\partial J(\mathbf{e}, \hat{\mathbf{y}})}{\partial \mathbf{e}_k} - \hat{\mathbf{z}}^T \frac{\partial s(\mathbf{e}, \hat{\mathbf{y}})}{\partial \mathbf{e}_k}, \quad k = 0, 1, \tag{162}$$

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

$$\left(\frac{\partial s(\mathbf{e}, \hat{\mathbf{y}})}{\partial \hat{\mathbf{y}}}\right)^T \hat{\mathbf{z}} = \nabla_{\hat{\mathbf{y}}} J(\mathbf{e}, \hat{\mathbf{y}}). \tag{163}$$

The adjoint equation (163) requires advancing backward in time, so it is called the backward equation [33].

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.

6.2 Gradient for the central finite difference time discretization

In the case of the central finite difference time discretization corresponding to the matrix form (127) the adjoint state equation is given by

$$\begin{pmatrix} \mathcal{I} & \frac{1}{2}\mathcal{C} & \mathcal{B} & & & \\ & \mathcal{M} & \mathcal{C} & \mathcal{B} & & \\ & & \mathcal{D} & \ddots & \ddots & \\ & & & \ddots & \ddots & \mathcal{B} \\ & & & & \mathcal{D} & \mathcal{C} \\ & & & & & \mathcal{D} \end{pmatrix} \begin{pmatrix} \mathbf{p}^0 \\ \mathbf{p}^1 \\ \vdots \\ \vdots \\ \mathbf{p}^N \\ \mathbf{p}^{N+1} \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \frac{\partial J}{\partial \mathbf{u}^{N-1}} \\ \frac{\partial J}{\partial \mathbf{u}^{N}} \\ \frac{\partial J}{\partial \mathbf{u}^{N+1}} \end{pmatrix}, \tag{164}$$

where

$$\frac{\partial J}{\partial \mathbf{u}^{N-1}} = \frac{1}{2\Delta t}\mathcal{M}\left(\mathbf{e}_1 - \frac{\partial \mathbf{u}^N}{\partial t}\right), \tag{165}$$

$$\frac{\partial J}{\partial \mathbf{u}^{N+1}} = \frac{1}{2\Delta t}\mathcal{M}\left(\frac{\partial \mathbf{u}^N}{\partial t} - \mathbf{e}_1\right), \tag{166}$$

$$\frac{\partial J}{\partial \mathbf{u}^{N}} = \mathcal{K}\left(\mathbf{u}^N - \mathbf{e}_0\right). \tag{167}$$

The gradient components for the central finite difference time discretization are then the following:

$$\frac{dJ(\mathbf{e},\hat{\mathbf{y}}(\mathbf{e}))}{d\mathbf{e}_0} = \mathcal{K}(\mathbf{e}_0 - \mathbf{u}^N) + \mathbf{p}^0, \tag{168}$$

$$\frac{dJ(\mathbf{e},\hat{\mathbf{y}}(\mathbf{e}))}{d\mathbf{e}_1} = \mathcal{M}\left(\mathbf{e}_1 - \frac{\partial \mathbf{u}^N}{\partial t}\right) + \Delta t \mathcal{B}^T \frac{\partial \mathbf{p}^0}{\partial t}. \tag{169}$$

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

In case of the fourth-order Runge–Kutta time discretization, the adjoint equation corresponding to the state equation (144) is

$$\begin{pmatrix} \mathcal{I} & \mathcal{N}^T & & & \\ & \mathcal{I} & \mathcal{N}^T & & \\ & & \ddots & \ddots & \\ & & & \mathcal{I} & \mathcal{N}^T \\ & & & & \mathcal{I} \end{pmatrix} \begin{pmatrix} \mathbf{z}^0 \\ \mathbf{z}^1 \\ \vdots \\ \mathbf{z}^{N-1} \\ \mathbf{z}^N \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \frac{\partial J}{\partial \mathbf{y}^N} \end{pmatrix}, \tag{170}$$

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 $t = i\Delta t, i = N, \ldots, 0$. 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

$$\frac{dJ(\mathbf{e},\hat{\mathbf{y}}(\mathbf{e}))}{d\mathbf{e}_0} = \mathcal{K}(\mathbf{e}_0 - \mathbf{u}^N) + \mathbf{p}^0, \tag{171}$$

$$\frac{dJ(\mathbf{e},\hat{\mathbf{y}}(\mathbf{e}))}{d\mathbf{e}_1} = \mathcal{M}\left(\mathbf{e}_1 - \frac{\partial \mathbf{u}^N}{\partial t}\right) + \frac{\partial \mathbf{p}^0}{\partial t}. \tag{172}$$

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

$$\begin{pmatrix} \mathcal{I} & \hat{\mathcal{H}}^T & & \\ & \mathcal{I} & \hat{\mathcal{H}}^T & \\ & & \mathcal{I} & 2\hat{\mathcal{H}}^T \\ & & & \mathcal{I} \end{pmatrix} \begin{pmatrix} k_1 \\ k_2 \\ k_3 \\ k_4 \end{pmatrix} = \begin{pmatrix} -\hat{R} \\ -2\hat{R} \\ -2\hat{R} \\ -\hat{R} \end{pmatrix} \mathbf{z}^{i+1}, \tag{173}$$

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

$$\mathbf{z}^i = \mathbf{z}^{i+1} - \begin{pmatrix} 2\hat{\mathcal{H}}^T & 2\hat{\mathcal{H}}^T & 2\hat{\mathcal{H}}^T & 2\hat{\mathcal{H}}^T \end{pmatrix} \begin{pmatrix} k_1 \\ k_2 \\ k_3 \\ k_4 \end{pmatrix}. \tag{174}$$

Example 6.3.1 To show the benefit of the adjoint equation technique, we repeat the gradient computations presented in Example 6.1.1 by using the adjoint equation technique and compare the CPU time requirements. The results, carried out on an AMD Opteron 885 processor at 2.6 GHz and depicted in Figure 18, show that, for computing the gradient, the adjoint equation technique is several orders of magnitude faster than the central finite difference approximation.

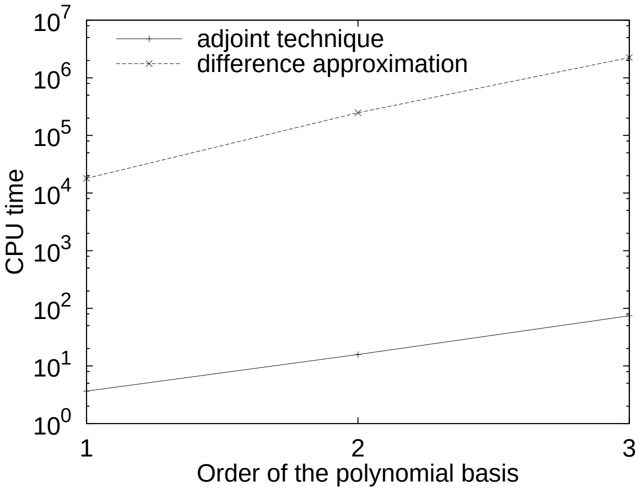


FIGURE 18 CPU time (in seconds) for computing the gradient by the adjoint equation technique and by the central finite difference approximation for the element orders $r = 1, 2, 3$ with the RK time discretization scheme and angular frequency $\omega = 4\pi$. The number of timesteps is chosen to eliminate the temporal error, and square-element meshes with mesh stepsize $h = 1/20$ are used in both media.

7 OPTIMIZATION ALGORITHM

The optimization algorithms can be classified in several ways. The two major categories are stochastic and deterministic optimization algorithms. Stochastic methods are suitable especially for the problems involving probability or uncertainty. They are typically used in the application areas where the model depends on unavailable or uncertain quantities, such logistics or financial planning. We concentrate on deterministic algorithms in which the model is fully specified. With these algorithms the solution procedure is reproducible, that is, the same solution is obtained with a particular input information. These methods can roughly be divided to direct and iterative ones. Further, the methods can be classified depending on the need of first or second derivatives (gradients or Hessian matrices), which is related to the efficiency of the method. Usually, the methods utilizing derivative information converge faster, but computing, for instance, Hessian matrices consumes a considerable amount of computing time and memory. Another important point of view along with the memory requirement and computational cost is the quality of the optimization solution. Although global solutions are necessary in some applications, many optimization methods attain only the local minimum. Convex optimization is a special case in which the local solution is also the global solution.

In what follows, we briefly describe some methods suitable for unconstrained optimization and give reasons for using the conjugate gradient method for the problem at hand. We consider minimizing the quadratic functional

$$J(\mathbf{e}) = \frac{1}{2}\mathbf{e}^T \mathsf{A}\mathbf{e} - \mathsf{b}^T\mathbf{e} + \mathsf{c}, \tag{175}$$

where A is a symmetric and positive definite matrix of size $\hat{N} \times \hat{N}$ and the vectors $\mathbf{e}$, b, and c are of size $\hat{N}$. The Hessian matrix of $J(\mathbf{e})$, including second derivatives, is $\nabla^2 J(\mathbf{e}) = \mathsf{A}$. As the matrix A is symmetric and positive definite, $J(\mathbf{e})$ is convex and it has a unique minimum at the point where the first derivative, the gradient of $J(\mathbf{e})$, is zero. Since the minimum value of the functional $J(\mathbf{e})$ is obtained at the minimum point, or minimizer,

$$\mathbf{e}^* = \arg \min_{\mathbf{e} \in \mathbb{R}^{\hat{N}}} J(\mathbf{e}),$$

such that $\nabla J(\mathbf{e}^*) = \mathsf{A}\mathbf{e}^* - \mathsf{b} = 0$, minimizing (175) is equivalent to solving a linear system

$$\mathsf{A}\mathbf{e} = \mathsf{b}. \tag{176}$$

Consequently, the minimum value of the functional is

$$J(\mathbf{e}^*) = \min_{\mathbf{e}\in\mathbb{R}^{\hat{N}}} J(\mathbf{e}) = -\mathsf{b}^T\mathsf{A}^{-1}\mathsf{b}/2 + \mathsf{c}.$$

In principle, if A and b are known explicitly, the linear system could be solved by some direct method, such as LU decomposition. However, the main problem with direct solution methods is how to find, for a large sparse matrix, a decomposition with only a few non-zero elements to improve the efficiency of the method. This is the main reason why iterative methods, beginning with an initial guess and generating a sequence of estimates until reaching the solution, are preferred especially for solving large sparse linear systems.

There are several well known iterative methods, such as Newton and quasi-Newton methods, steepest descent, and conjugate gradient method, that are suitable for solving unconstrained optimization problems. If we focus on large problems, the set of feasible optimization algorithms is restricted to the methods with small memory requirements. If the Hessian matrix is not sparse, storing it may consume a considerable amount of memory when large problems are considered. In addition, constructing the Hessian matrix (or its inverse) might be computationally expensive. Since the Hessian matrix or its approximation is needed to be stored in Newton-type methods, we neglect these methods at this point.

Gradient information is worth utilizing in optimization if the gradient can be constructed in a computationally efficient manner. In the gradient-based iterative methods, the initial value $\mathbf{e}^0$ is usually chosen arbitrarily, and the next solution candidate is a point $\mathbf{e}^{i+1}$ which minimizes the objective functional from the previous point $\mathbf{e}^i$, $i = 0, 1, \ldots$ to the given direction $\mathbf{w}^i$ such that

$$\mathbf{e}^{i+1} = \mathbf{e}^i + \eta^i\mathbf{w}^i, \quad i = 0, 1, \ldots \tag{177}$$

where $\eta^i \in \mathbb{R}$ is a step length chosen to minimize $J(\mathbf{e}^{i+1})$. Once the direction is determined, finding the proper η^i is a minimization of a function of a single variable, in other words, a line search problem in which the step length η^i can be solved by setting the derivative of $J(\mathbf{e}^i + \eta^i\mathbf{w}^i)$, with respect to η^i, to zero.

In the steepest descent method, the successive search directions are perpendicular to the previous direction. Consequently, the convergence slows down near the minimum point. In the conjugate gradient method, choosing the new direction is improved by adding a portion of the previous direction to the steepest descent direction. Hence, the method uses gradient information but searches for the solution as a sequence of conjugate directions needing only a small amount of storage space. It is a an orthogonal projection technique working in the Krylov subspace $K_j(\mathbf{g}^0, \mathsf{A}) = span(\mathbf{g}^0, \mathsf{A}\mathbf{g}^0, \ldots, \mathsf{A}^{j-1}\mathbf{g}^0)$, where $\mathbf{g}^0 = \nabla J(\mathbf{e}^0)$ is the initial residual. Moreover, the CG method is sufficiently easy to implement. Thus, the CG method provides a well-suited optimization algorithm for a sufficiently large problem with quadratic objective functional and linear state equation. To guarantee the smooth initial approximation for the CG algorithm we use a transition procedure, suggested by Mur [158], which is described in Section 7.1. The main principles of the CG method are presented in Section 7.2. For speeding up the convergence rate of the CG algorithm, we use the graph-based multigrid method introduced in [153]. We consider the preconditioned conjugate gradient algorithm and present a brief overview of the multigrid method in Section 7.3.

7.1 Initial approximation

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., [33]), it is important to have smooth initial approximations for $\mathbf{e}_0$ and $\mathbf{e}_1$, which satisfy the boundary conditions. In [33], a special procedure suggested by Mur in [158, p. 950] 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:

$$\theta(t) = \begin{cases} \left(2 - \sin\left(\pi t / 2T_{\text{tr}}\right)\right) \sin\left(\pi t / 2T_{\text{tr}}\right), & \text{if } 0 \leq t \leq T_{\text{tr}}, \\ 1, & \text{if } t \geq T_{\text{tr}}. \end{cases} \tag{178}$$

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. Then, for example in the case of the interaction between acoustic and elastic waves, with the velocity potential -based formulation in the fluid domain, we solve the following initial value problem:

$$\frac{1}{c(\mathbf{x})^2}\frac{\partial^2 \phi}{\partial t^2} - \nabla^2 = \theta(t) f_\phi, \qquad \text{in } Q_f = \Omega_f \times [0, T_{\text{tr}}], \tag{179}$$

$$\phi = 0, \qquad \text{on } \gamma_{0\text{f}} = \Gamma_{0\text{f}} \times [0, T_{\text{tr}}], \tag{180}$$

$$\frac{1}{c(\mathbf{x})}\frac{\partial \phi}{\partial t} + \frac{\partial \phi}{\partial \mathbf{n}_f} = \theta(t) y_{\phi\text{ext}}, \qquad \text{on } \gamma_{ef} = \Gamma_{ef} \times [0, T_{\text{tr}}], \tag{181}$$

$$\frac{\partial \mathbf{u_s}}{\partial t} \cdot \mathbf{n}_s + \frac{\partial \phi}{\partial \mathbf{n}_f} = 0, \qquad \text{on } \gamma_i = \Gamma_i \times [0, T_{\text{tr}}], \tag{182}$$

$$\rho_s(\mathbf{x})\frac{\partial^2 \mathbf{u_s}}{\partial t^2} - \nabla \cdot \sigma(\mathbf{u_s}) = \theta(t)\mathbf{f}, \qquad \text{in } Q_s = \Omega_s \times [0, T_{\text{tr}}], \tag{183}$$

$$\mathbf{u_s} = \mathbf{0}, \qquad \text{on } \gamma_{0\text{s}} = \Gamma_{0\text{s}} \times [0, T_{\text{tr}}], \tag{184}$$

$$\rho_s(\mathbf{x})\mathbf{B}_s\frac{\partial \mathbf{u_s}}{\partial t} + \sigma(\mathbf{u_s})\mathbf{n}_s = \theta(t)\mathbf{g}_{\text{ext}}, \qquad \text{on } \gamma_{es} = \Gamma_{es} \times [0, T_{\text{tr}}], \tag{185}$$

$$\sigma(\mathbf{u_s})\mathbf{n}_s + \rho_f(\mathbf{x})\frac{\partial \phi}{\partial t}\mathbf{n}_f = \mathbf{0}, \qquad \text{on } \gamma_i = \Gamma_i \times [0, T_{\mathrm{tr}}], \tag{186}$$

$$\phi(\mathbf{x}, 0) = \mathbf{0}, \qquad \text{in } \Omega_f, \tag{187}$$

$$\frac{\partial \phi}{\partial t}(\mathbf{x}, 0) = \mathbf{0}, \qquad \text{in } \Omega_f, \tag{188}$$

$$\mathbf{u_s}(\mathbf{x}, 0) = \mathbf{0}, \qquad \text{in } \Omega_s, \tag{189}$$

$$\frac{\partial \mathbf{u_s}}{\partial t}(\mathbf{x}, 0) = \mathbf{0}, \qquad \text{in } \Omega_s. \tag{190}$$

After solving the problem (179)-(190), 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

$$\mathbf{e}_0^0 = \left(\phi(\mathbf{x}, T_{\mathrm{tr}}), \mathbf{u_s}(\mathbf{x}, T_{\mathrm{tr}})\right)^T,$$

$$\mathbf{e}_1^0 = \left(\frac{\partial \phi}{\partial t}(\mathbf{x}, T_{\mathrm{tr}}), \frac{\partial \mathbf{u_s}}{\partial t}(\mathbf{x}, T_{\mathrm{tr}})\right)^T.$$

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.

7.2 Minimization using the conjugate gradient algorithm

The CG method was originally introduced by Hestenes and Stiefel in 1952 [111] for solving linear systems of the form (176), where A is a symmetric and positive definite matrix of size $\hat{N} \times \hat{N}$ and the vectors $\mathbf{e}$ and b are of size $\hat{N}$. In exact arithmetics, the CG method was proposed as a direct method which converges in at most $\hat{N}$ steps. In practice, there are some rounding errors in the numerical procedure implying that the number of iterations might be greater than $\hat{N}$. However, with a favorable spectrum of A, the solution is possible to achieve with the number of iterations far fewer than $\hat{N}$.

The major advantages of the conjugate gradient method are its speed and simplicity. It converges much faster than steepest descent and does not suffer from the inefficiencies and possible instabilities that arise from using a fixed step size. Although it might be slower than the second-order Newton and quasi-Newton minimization methods, it requires only a small amount of additional storage space; only the current and previous gradient and search vectors must be stored in addition to the weights.

The method can be generalized for solving a linear system $\Lambda \mathbf{e} = \hat{\mathsf{b}}$ with non-symmetric and non-self-adjoint coefficient matrix Λ as well. In that case, the linear system is multiplied by the adjoint of the coefficient matrix, that is, Λ^*, leading to a linear system $\Lambda^* \Lambda \mathbf{e} = \Lambda^* \hat{\mathsf{b}}$ having symmetric and positive definite coefficient $\mathsf{A} = \Lambda^* \Lambda$ and right-hand side vector $\mathsf{b} = \Lambda^* \hat{\mathsf{b}}$. In the context of the control problem at hand, Λ can be interpreted as the mapping related to the state equation, whereas Λ^* represents the adjoint mapping.

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 $\mathbf{w}^i$, $i = 1, 2, \ldots$ are generated to be conjugant with the matrix A. Thus, the inner product of $\mathbf{w}^i$ and $\mathsf{A}\mathbf{w}^j$ is zero, that is, $(\mathbf{w}^i, \mathsf{A}\mathbf{w}^j) = 0, i \neq j$ meaning that the vectors $\mathbf{w}^0, \mathbf{w}^1, \ldots, \mathbf{w}^{\hat{N}-1}$ are said to be A-conjugate. A set of non-zero 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 A-conjugate set $\{\mathbf{w}^0, \mathbf{w}^1, \ldots, \mathbf{w}^{\hat{N}-1}\}$. Since the vectors $\mathbf{w}^0, \mathbf{w}^1, \ldots, \mathbf{w}^{\hat{N}-1}$ 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

$$\mathbf{e}^* = \mathbf{e}^0 + \sum_{i=0}^{\hat{N}-1} \eta^i \mathbf{w}^i = \mathbf{e}^k + \sum_{i=k}^{\hat{N}-1} \eta^i \mathbf{w}^i, \tag{191}$$

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

$$(\mathbf{w}^j)^T (\mathsf{b} - \mathsf{A}\mathbf{e}^k) = \sum_{i=k}^{\hat{N}-1} \eta^i (\mathbf{w}^j)^T \mathsf{A}\mathbf{w}^i. \tag{192}$$

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

$$\eta^i = \frac{-(\mathbf{w}^i)^T \mathbf{g}^i}{(\mathbf{w}^i)^T \mathsf{A}\mathbf{w}^i}, \tag{193}$$

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

```
Compute the initial value $\mathbf{e}^0 = (\mathbf{e}_0^0, \mathbf{e}_1^0)^T$.
Compute the gradient $\mathbf{g}^0 = \mathsf{A}\mathbf{e}^0 - \mathsf{b}$.
Set $\mathbf{w}^0 = -\mathbf{g}^0$.
Set $c_0 = -(\mathbf{w}^0, \mathbf{g}^0)$, $c = c_0$ and $i = 1$.
Repeat until $\sqrt{\frac{c}{c_0}} < \varepsilon$
    Compute the gradient update $\mathbf{v}^{i-1} = \mathsf{A}\mathbf{w}^{i-1}$.
    Compute $\eta^{i-1} = \frac{c}{(\mathbf{w}^{i-1}, \mathbf{v}^{i-1})}$.
    Update the control vector $\mathbf{e}^i = \mathbf{e}^{i-1} + \eta^{i-1}\mathbf{w}^{i-1}$.
    Update the residual vector $\mathbf{g}^i = \mathbf{g}^{i-1} + \eta^{i-1}\mathbf{v}^{i-1}$.
    Set $\mathbf{v}^i = -\mathbf{g}^i$.
    Compute $\gamma^{i-1} = \frac{1}{c}$, $c = -(\mathbf{v}^i, \mathbf{g}^i)$, $\gamma^{i-1} = c\gamma^{i-1}$.
```

```
    Update minimizing direction $\mathbf{w}^i = \mathbf{v}^i + \gamma^{i-1}\mathbf{w}^{i-1}$.
    Set $i = i + 1$,
```

where i refers to the number of iterations, the residual is $-\mathbf{g}$, $\mathbf{w}$ is the search direction, and $\sqrt{\frac{c}{c_0}}$ is the relative euclidean norm of the residual, which is the gradient of the functional J. Values of the control variables $\mathbf{e}$ at the ith iteration are denoted by $\mathbf{e}_0^i$ and $\mathbf{e}_1^i$. Smooth initial approximations $\mathbf{e}^0 = (\mathbf{e}_0^0, \mathbf{e}_1^0)^T$ for the algorithm are computed with a transition procedure, which is presented in Section 7.1.

Since the matrix A is symmetric and positive definite, it defines an inner product $\mathsf{x}^T\mathsf{A}\mathsf{y}$ between vectors x and y in $\mathbb{R}^{\hat{N}}$, which is referred to as an A-inner product. The convergence of the CG method is dependent on the spectrum of the eigenvalues of the matrix A. The convergence rate is bounded by the condition number κ which is given by the ratio of the largest and smallest eigenvalue for symmetric and positive definite matrix A by using the A-norm $||\mathbf{e}||_{\mathsf{A}} = \sqrt{(\mathbf{e}, \mathsf{A}\mathbf{e})}$. The theoretical convergence speed of the CG method after i iterations is given by

$$||\mathbf{e}^* - \mathbf{e}^i||_{\mathsf{A}} \leq 2||\mathbf{e}^* - \mathbf{e}^0||_{\mathsf{A}} \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^i, \tag{194}$$

in which $\mathbf{e}^*$ is the solution of the minimization problem and $\mathbf{e}^0$ is an arbitrary initial condition [93]. It is worth mentioning that the formula gives only the upper limit for the convergence, but from it we can at least conclude that the converge is fast if the value of the condition number is close to one.

Consequently, 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 $\mathsf{A}\mathbf{e} = \mathsf{b}$ by a matrix $\mathcal{L}^{-1}$ meaning that instead of seeking a solution from the space $\mathbf{e}^0 + K_j(\mathbf{g}^0, \mathsf{A})$, where K_j is the j:th Krylov subspace defined as $K_j(\mathbf{g}^0, \mathsf{A}) = span(\mathbf{g}^0, \mathsf{A}\mathbf{g}^0, \ldots, \mathsf{A}^{j-1}\mathbf{g}^0)$, we are seeking the solution from the space $\mathbf{e}^0 + K_j(\mathcal{L}^{-1}\mathbf{g}^0, \mathcal{L}^{-1}\mathsf{A})$. 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., [93]). The preconditioned version of the conjugate gradient (CG) algorithm for solving the time-harmonic wave problems is as follows:

Algorithm 2 *Preconditioned CG algorithm*

```
Compute the initial value $\mathbf{e}^0 = (\mathbf{e}_0^0, \mathbf{e}_1^0)^T$.
Solve the state equation $s(\mathbf{e}^0, \hat{\mathbf{y}}(\mathbf{e}^0)) = 0$.
Solve the adjoint state equation $\left(\frac{\partial s(\mathbf{e}^0, \hat{\mathbf{y}}(\mathbf{e}^0))}{\partial \hat{\mathbf{y}}(\mathbf{e}^0)}\right)^T \hat{\mathbf{z}} = \left(\frac{\partial J(\mathbf{e}^0, \hat{\mathbf{y}}(\mathbf{e}^0))}{\partial \hat{\mathbf{y}}(\mathbf{e}^0)}\right)^T$.
Compute the gradient $\mathbf{g}^0 = (g_0^0, g_1^0)^T$ by the formulas (168)-(169) or (171)-(172).
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 $i = 1$.
Repeat until $\sqrt{\frac{c}{c_0}} < \varepsilon$
```

```
Solve the state equation $s_0(\mathbf{w}^{i-1}, \hat{\mathbf{y}}(\mathbf{w}^{i-1})) = 0$.
Solve the adjoint state equation $\left(\frac{\partial s(\mathbf{w}^{i-1}, \hat{\mathbf{y}}(\mathbf{w}^{i-1}))}{\partial \hat{\mathbf{y}}(\mathbf{w}^{i-1})}\right)^T \hat{\mathbf{z}} = \left(\frac{\partial J(\mathbf{w}^{i-1}, \hat{\mathbf{y}}(\mathbf{w}^{i-1}))}{\partial \hat{\mathbf{y}}(\mathbf{w}^{i-1})}\right)^T$.
Compute the gradient update $\mathbf{v}^{i-1} = (v_0^{i-1}, v_1^{i-1})^T$ by the formulas (168)-(169) or (171)-(172) .
Compute $\eta^{i-1} = \frac{c}{(\mathbf{w}^{i-1}, \mathbf{v}^{i-1})}$.
Update the control vector $\mathbf{e}^i = \mathbf{e}^{i-1} + \eta^{i-1}\mathbf{w}^{i-1}$.
Update the residual vector $\mathbf{g}^i = \mathbf{g}^{i-1} + \eta^{i-1}\mathbf{v}^{i-1}$.
Solve linear system with the preconditioner $\mathcal{L}\mathbf{v}^i = -\mathbf{g}^i$.
Compute $\gamma^{i-1} = \frac{1}{c}$, $c = -(\mathbf{v}^i, \mathbf{g}^i)$, $\gamma^{i-1} = c\gamma^{i-1}$.
Update minimizing direction $\mathbf{w}^i = \mathbf{v}^i + \gamma^{i-1}\mathbf{w}^{i-1}$.
Set $i = i + 1$.
```

In the above, $\hat{\mathbf{y}}$ is the solution of the state equation (127) or (144), $\hat{\mathbf{z}} = (\mathbf{p}^0, \frac{\partial \mathbf{p}^0}{\partial t})$ is the solution of the adjoint state equation (164) or (170), and the gradient variable $\mathbf{g} = (g_0, g_1)$ is computed by the formulas (168)-(169) or (171)-(172). By $s_0(\mathbf{w}, \hat{\mathbf{y}}(\mathbf{w})) = 0$ we denote the state equation (127) or (144), where $\mathcal{F}^i = 0$ for all i.

Each conjugate gradient iteration step requires computation of the gradient of the least-squares functional, ∇J by the formulas (168)-(169) or (171)-(172), which involves the solution of the state equation (127) or (144) and the corresponding adjoint equation (164) or (170), the solution of a linear system with the preconditioner, and some other matrix-vector operations.

7.3 Preconditioning using the graph-based multigrid

The preconditioning is seen in the algorithm so that instead of the linear system $\mathsf{A}\mathbf{v} = \mathsf{b}$, we solve $\mathsf{A}\mathcal{L}^{-1}\mathbf{g} = \mathsf{b}$. A good preconditioner $\mathcal{L}$ approximates the matrix A, but solving the preconditioned problem is easier than solving the original linear system. Probably the simplest choice for a preconditioner is a diagonal preconditioner $\mathcal{L}$ having the same diagonal entries than the matrix A. Thus, multiplying by $\mathcal{L}^{-1}$ is simply dividing elementwisely by its corresponding diagonal element. With $\mathcal{L} = \mathcal{I}$ we have the unpreconditioned situation which is easy to solve but suffers from poor convergence, and the number of iterations grows rapidly with the order of elements [146]. The special case $\mathcal{L} = \mathsf{A}^{-1}$ returns to the same level of difficulty than the original problem, but it can be solved by one iteration. While choosing a preconditioner, we, in general, need to choose between a small number of iterations and long computing time for the preconditioning step or a large number of iterations with a time-efficient preconditioner.

From the formula (157), we see that the functional depends on the initial conditions both directly and indirectly through the solution of the linear wave equation. By substituting $\hat{\mathbf{y}} = \mathsf{B}\mathbf{e} + \mathsf{d}$ into (157) and comparing with the formula (175), we get

$$\mathsf{A} = (\mathsf{B} - \mathcal{I})^T \begin{pmatrix} \mathcal{K} & \mathbf{0} \\ \mathbf{0} & \mathcal{M} \end{pmatrix} (\mathsf{B} - \mathcal{I}), \tag{195}$$

$$\mathsf{b} = -\mathsf{d}^T \begin{pmatrix} \mathcal{K} & \mathbf{0} \\ \mathbf{0} & \mathcal{M} \end{pmatrix} (\mathsf{B} - \mathcal{I}), \qquad \mathsf{c} = \frac{1}{2}\mathsf{d}^T \begin{pmatrix} \mathcal{K} & \mathbf{0} \\ \mathbf{0} & \mathcal{M} \end{pmatrix} (\mathsf{B} - \mathcal{I})\mathsf{d}. \tag{196}$$

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

We use a block-diagonal preconditioner

$$\mathcal{L} = \begin{pmatrix} \mathcal{K} & \mathbf{0} \\ \mathbf{0} & \mathcal{M} \end{pmatrix}, \tag{197}$$

which corresponds to the energy formulation (92). 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 GBMG method introduced in [153]. As the name of the method indicates, a number of different grid levels are used on the domain, ranging between fine and coarse levels. A sequence of linear problems

$$\mathcal{K}_l \tilde{\mathbf{w}}_l = \tilde{\mathbf{G}}_l \tag{198}$$

is generated, corresponding to grid levels $l = 0, \ldots, \tilde{k}$, where $\tilde{k}$ represents the coarsest level. Each GBMG iteration starts with the finest level matrix $\mathcal{K}_0$, a right-hand side vector $\tilde{\mathbf{G}}_0$, 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 $\mathcal{K}_l \tilde{e}_l = \tilde{r}_l$. Unlike the classical geometric multigrid methods [96, 187], in the GBMG the actual coarsening of the given mesh is not needed for finding coarser grid levels.

The coarsening, that is, selecting the unknowns for coarser levels, is based on the graph of the stiffness matrix, rather than on the actual values stored in the stiffness matrix. This approach ensures fast computation of coarser level components. The coarsening process operates in a geometric fashion by sequentially choosing a coarse node and eliminating the neighboring nodes of the graph. In selecting the unknowns for coarser levels, the primary criterion is to take the node with minimum degree when eliminations are taken into account. The secondary criterion is to follow the original node numbering.

The use of the multigrid methods for spectral elements has recently been studied in [113]. The number of connections between unknowns of the problem increases when higher-order elements are used. In this case, the coarsening strategy described above leads to unacceptably coarse systems and the convergence factor of the multigrid approach degrades rapidly as the order of the approximation polynomials increases. We overcome this problem by employing a graph constructed so that unknowns are connected to each other as if low-order finite elements were used in the discretization process. Only the unknowns corresponding to the nearest neighboring Gauss–Lobatto points are connected to each other. Additionally, in vector-valued problems it is necessary to prevent mixing 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.

The grid transfer operators are the restriction operator $\widetilde{\mathbf{R}}$ and the prolongation operator $\widetilde{\mathbf{P}}$. The matrices $\mathcal{K}_l$, which are used at multigrid levels $l = 0, \ldots, \tilde{k}$, are set as an initialization step of the GBMG algorithm. For this purpose, we need the restriction operator $\widetilde{\mathbf{R}}_l^{l+1}$ from the fine level l to the coarse level $(l+1)$

$$\widetilde{\mathbf{R}}_l^{l+1} = \begin{pmatrix} \mathbf{R}_l^{l+1} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_l^{l+1} \end{pmatrix}, \tag{199}$$

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

$$\left(\mathbf{R}_l^{l+1}\right)_{ij} = \begin{cases} 1 & \text{for a fine grid point } j \text{ which is a coarse grid point } i, \\ \frac{1}{k} & \text{for a fine grid point } j \text{ which is a neighbor of coarse grid} \\ & \text{point } i \text{ and has } k \text{ neighboring coarse grid points,} \\ 0 & \text{otherwise.} \end{cases} \tag{200}$$

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 prolongation operator $\widetilde{\mathbf{P}}_{l+1}^{l}$ from the coarse level $(l+1)$ to the fine level l is chosen to be the transpose of the restriction,

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

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 W-cycle [96, 179] is utilized as a multigrid iteration until the residual norm of the solution is smaller than 10^{-6}.

Example 7.3.1 For simplicity, we consider the acoustic wave problem,

$$-\omega^2\Phi - \nabla^2\Phi = 0, \qquad \text{in } \Omega_f, \tag{201}$$

$$\Phi = \exp(i\boldsymbol{\omega}\cdot\mathbf{x}), \qquad \text{on } \Gamma_{0f}, \tag{202}$$

$$i\omega\Phi + \frac{\partial\Phi}{\partial\mathbf{n}_f} = i\left(\omega_1 n_{f1} + \omega_2 n_{f2} - \omega\right)\exp(i\boldsymbol{\omega}\cdot\mathbf{x}), \qquad \text{on } \Gamma_{ef}, \tag{203}$$

where $\boldsymbol{\omega} = \frac{1}{\sqrt{2}}(1,-1)\,\omega$ with angular frequency $\omega = 2\pi$ and the outward pointing normal vector $\mathbf{n}_f = (n_{f1}, n_{f2})$. The domain $\Omega = \Omega_f$, consisting of a fluid with density

$\rho_f = 1$ and wave speed $c = 1$, is defined so that the boundary surrounding it, Γ_{ef}, coincides with the boundary of the rectangle $[0,0] \times [4,4]$. We have set a square obstacle, having a side length of 2 and boundary Γ_{0f} in the center of the domain Ω_f. The function $\Phi = \exp(i\boldsymbol{\omega} \cdot \mathbf{x})$ satisfies the problem (201)-(203), and $\phi = \cos(\omega t - \boldsymbol{\omega} \cdot \mathbf{x})$ is the solution of the corresponding time-dependent equation, that is,

$$\frac{\partial^2 \phi}{\partial t^2} - \nabla^2 \phi = 0, \qquad \text{in } Q_f = \Omega_f \times (0,T), \tag{204}$$

$$\phi = \cos(\omega t - \boldsymbol{\omega} \cdot \mathbf{x}), \qquad \text{on } \gamma_{0f} = \Gamma_{0f} \times (0,T), \tag{205}$$

$$\frac{\partial \phi}{\partial t} + \frac{\partial \phi}{\partial \mathbf{n}_f} = (\omega - \boldsymbol{\omega} \cdot \mathbf{n}_f) \sin(\omega t - \boldsymbol{\omega} \cdot \mathbf{x}), \qquad \text{on } \gamma_{ef} = \Gamma_{ef} \times (0,T), \tag{206}$$

with the initial conditions

$$\mathbf{e}_0 = \phi(\mathbf{x}, 0) = \cos(\boldsymbol{\omega} \cdot \mathbf{x}), \tag{207}$$

$$\mathbf{e}_1 = \frac{\partial \phi(\mathbf{x}, 0)}{\partial t} = \omega \sin(\boldsymbol{\omega} \cdot \mathbf{x}). \tag{208}$$

The system (204)-(208) is the state equation which can be presented in the form (127). Further, we use the corresponding adjoint state equation (164) and compute the gradient by the formula (168)-(169).

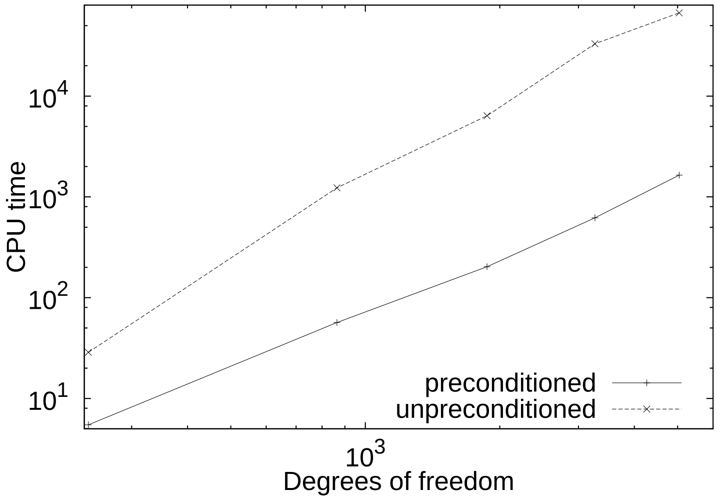


FIGURE 19 Comparison between the computational effort of preconditioned and unpreconditioned CG algorithm. CPU time in seconds is presented with respect to the number of degrees of freedom.

To show the benefit of preconditioning, we compare the behavior of the implementation of Algorithm 1 and Algorithm 2. The computations corresponding to r-refinement with $h = 1/4$ are carried out on an AMD Opteron 885 processor at 2.6 GHz. The length of the time interval $T_{\text{tr}} = 2T$ with $T = 1$ is used for computing the initial values. The stopping criterion is $\varepsilon = 10^{-5}$, and the time domain is divided into $N = 300$ steps for each element order r.

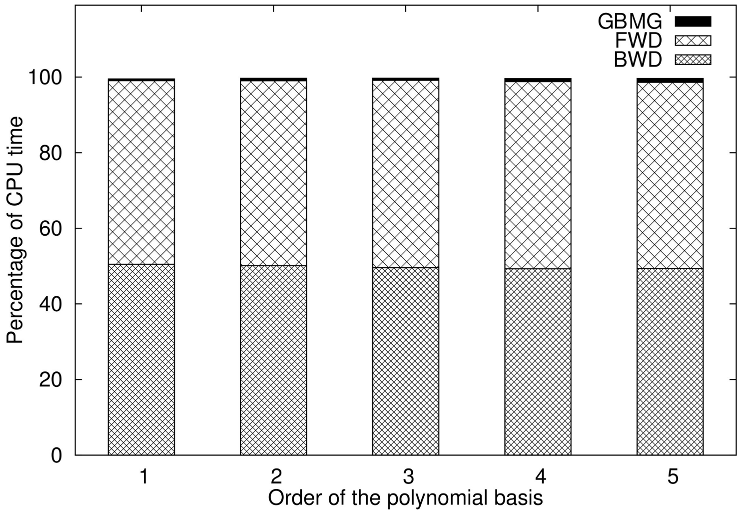


FIGURE 20 Proportion of CPU time (in percent) required by the GBMG cycles and computing state (FWD) and adjoint state (BWD) in one CG iteration.

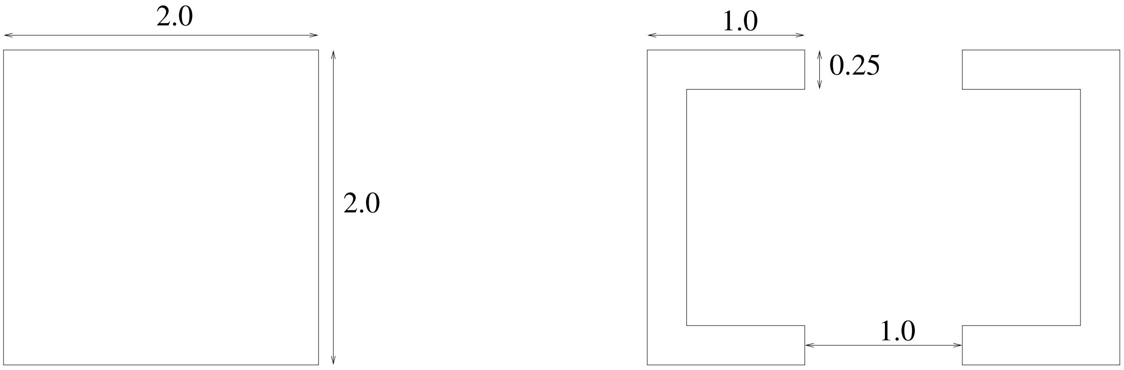
When the order of the polynomial basis increases or the mesh stepsize becomes smaller, systems to be solved become larger, which causes the increase in CPU time. The preconditioned minimization seems to be at least an order of magnitude faster than the unpreconditioned one (see Figure 19). It is observed that the number of CG iterations required to attain the stopping criterion grows with the number of degrees of freedom in the unpreconditioned case, while in the preconditioned case the number of iterations remains approximately constant. Consequently, the number of iterations is significantly smaller with the preconditioning. On the other hand, at each iteration an additional linear system is solved by GBMG. Nevertheless, the most of the CPU time is used for solving state and adjoint state equations, and CPU time required by the GBMG preconditioner is only a few percent of the CPU time for the whole algorithm. This is seen in Figure 20, which 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, a negligible amount of CPU time is used for matrix-vector multiplications at each iteration. Thus, significant savings result from the GBMG preconditioner.

When solving the state equation (127) or (144), $\mathcal{M}^{-1}$ is the only matrix inversion which is involved in time-stepping. By $\hat{N}_f$ we denote the number of degrees of freedom in the fluid domain, whereas $\hat{N}_s$ is the number of degrees of freedom in the structure domain. If the matrix $\mathcal{M}$ is diagonal, it is inverted simply by inverting each of its diagonal elements. This requires only $\hat{N} = \hat{N}_s + \hat{N}_f$ floating point operations, which is the number of degrees of freedom in the space discretization. The operation count of a matrix-vector product with any one of the matrices $\mathcal{M}$, $\mathcal{M}^{-1}$, $\mathcal{S}$, or $\mathcal{S}^{-1}$ (or some linear combination of these) is of order $O(\hat{N})$. In the matrix-vector multiplication involving the sparse stiffness matrix $\mathcal{K}$, only non-zero matrix entries are multiplied, which requires the order of $r^2\hat{N}$ operations. Besides these, some additions and multiplications are needed at each time step. Thus, solving the state equation needs $O(r^2\hat{N})$ floating point operations at each time step in the CD and the RK time-steppings. From this, we can conclude that the computational demand for computing the solution for the state equation with N time steps is $O(Nr^2\hat{N})$. The number of computational operations needed for solving the adjoint state equation is obviously of the same order as that needed 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 time steps N is fixed, the number of iterations is approximately constant, and the element order r has small integer values, the computational demand for the overall CG algorithm is $O(\hat{N})$.

Example 7.3.2 We compare the simulation results with acoustic scattering by a sound-soft obstacle with scattering by an elastic obstacle. We consider the acoustic wave equation with incident plane wave $\varphi_{\text{inc}}(\mathbf{x},t) = \cos(\boldsymbol{\omega} \cdot \mathbf{x} - \omega t)$ implying $y_{\varphi\text{ext}} = (\omega - \boldsymbol{\omega} \cdot \mathbf{n}_s)\sin(\boldsymbol{\omega} \cdot \mathbf{x} - \omega t)$. Furthermore, there are no other source terms, that is, $f_\varphi = 0$, $\mathbf{f} = \mathbf{0}$, and $\mathbf{g}_{\text{ext}} = \mathbf{0}$.

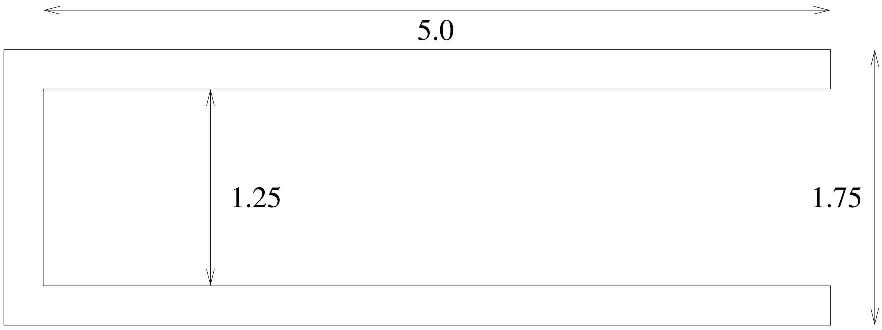
The infinite domain is truncated by the boundary Γ_{ef}, and the computational domain is divided into square-elements, each having a side length h. The domain Ω with a square obstacle with side length 2 (see Figure 21(a)) was defined such that the surrounding boundary Γ_{ef} coincided with the border of the square $[0,4] \times [0,4]$. Scattering by two semi-open obstacles (see Figure 21(b)) 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. In simulations with one non-convex semi-open obstacle (see Figure 21(c)), 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 obstacles are centered in the computational domain and located at the perpendicular distance of 1 from the boundary Γ_{ef}. Thus, the lower left corner of the obstacle is at the point $(1,1)$ in each case. If the scatterer is assumed to be sound-soft, the obstacles are surrounded by the boundary $\Gamma_{0\text{f}}$. When an elastic obstacle is involved, the obstacles defining the domain Ω_s are surrounded by the boundary Γ_i.

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. Since the velocity is higher in the solid medium than in the fluid medium, we need to use more timesteps to satisfy the stability conditions when the elastic obstacle is considered (see Table 6). The propagation direction is chosen to be $\boldsymbol{\omega} = \omega\big(-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\big)$. 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 $c_s(\mathbf{x}) = 3.12$. Thus, the fluid domain approximates to a water-like liquid, whereas the solid domain is assumed to consist of a metallic material like aluminum. Time discretization is realized with central finite differences. The stopping criterion is set with the relative norm $\sqrt{c/c_0}$ and $\varepsilon = 10^{-5}$.



(a) Square obstacle.

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



(c) Non-convex semi-open obstacle.

FIGURE 21 Geometrical shapes of the obstacles.

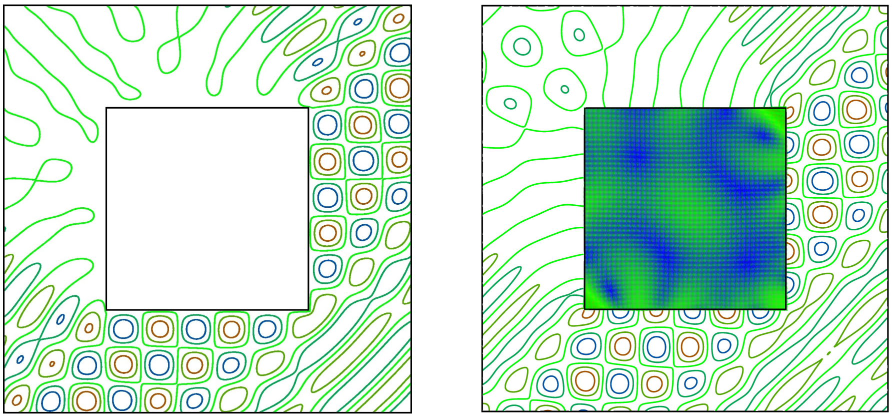
TABLE 6 Mesh stepsizes and number of timesteps for different spectral orders with $\omega = 4\pi$.

r			1	2	3	4	5
	Mesh stepsize h		$1/80$	$1/40$	$1/28$	$1/20$	$1/16$
Number of timesteps N	CD	sound-soft obstacle	90	270	300	320	320
	CD	elastic obstacle	400	800	1500	1700	2000
	RK	sound-soft obstacle	60	100	140	150	150
	RK	elastic obstacle	300	360	480	540	600

We start by a special case of fluid-structure interaction, in which the structure is assumed to be sound-soft. That is, we can model the interface of the two domains by simply replacing the boundary Γ_i by the boundary Γ_{ef}. Thus, only the fluid variables are involved in the computations when a sound-soft obstacle is considered. Then, we replace the sound-soft obstacle by an elastic one with the material parameters $\rho_s(\mathbf{x}) = 2.7$, $c_p(\mathbf{x}) = 5.95$ and $c_s(\mathbf{x}) = 3.12$ corresponding to aluminum. The number of iterations with different scatterers is reported in Table 7.

TABLE 7 The number of iterations of the preconditioned CG algorithm in the case of the CD and the RK time discretization with different scatterers.

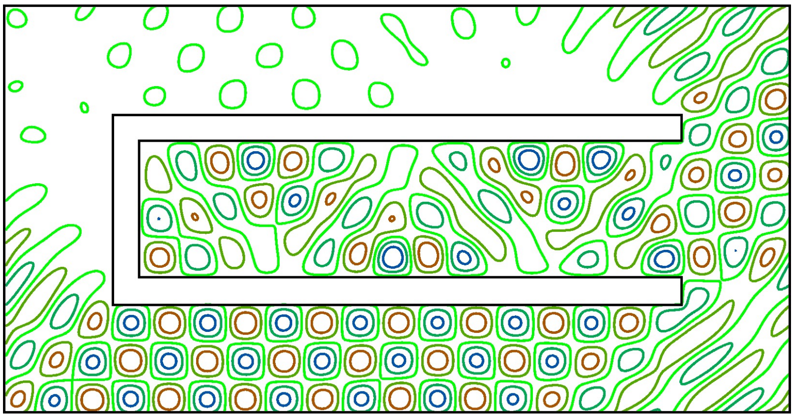
Type of the obstacle and time discretization			Element order r 1	2	3	4	5
convex obstacle (square)	sound-soft	CD	216	208	142	143	178
	elastic	CD	107	105	101	102	104
	sound-soft	RK	59	75	74	76	75
	elastic	RK	111	115	107	107	109
non-convex semi-open obstacle	sound-soft	CD	217	208	188	229	332
	elastic	CD	199	192	192	193	193
	sound-soft	RK	211	300	301	300	299
	elastic	RK	189	188	188	188	188
two non-convex semi-open obstacles	sound-soft	CD	238	252	186	286	268
	elastic	CD	177	166	165	165	165
	sound-soft	RK	123	146	145	145	145
	elastic	RK	133	165	163	144	164



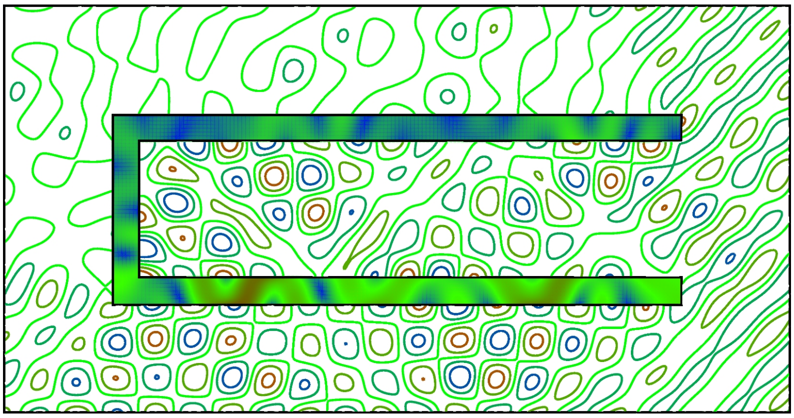
(a) Sound-soft convex obstacle. (b) Elastic convex obstacle.

FIGURE 22 Scattering by a convex obstacle with $r = 3$ and $h = 1/28$.

As we can see, the number of iterations is substantially smaller 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

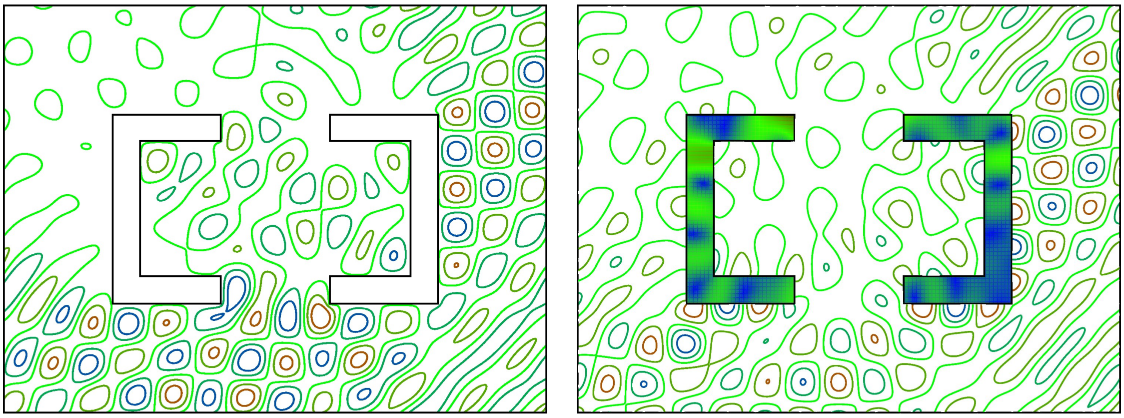


(a) Sound-soft non-convex semi-open obstacle.



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

FIGURE 23 Scattering by a non-convex semi-open obstacle with $r = 3$ and $h = 1/28$.



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

FIGURE 24 Scattering by a system of two non-convex semi-open obstacles with $r = 3$ and $h = 1/28$.

r. Contour plots of the numerical solutions with the RK time discretization and $r = 3$ are presented in Figures 22-24. As we can see, there are differences in the wave motion depending on whether the scatterer is modeled as a sound-soft or an elastic obstacle.

8 EFFICIENCY CONSIDERATIONS

In this chapter, we show numerical results of time-harmonic scattering applications in order to validate the controllability method discussed in previous chapters and to demonstrate some properties of the proposed algorithm. For each element order r, we construct square-element meshes, which are matching on the interface Γ_i. The practical realization of the algorithm is implemented in Fortran 95. Numerical experiments are carried out on an AMD Opteron 885 processor at 2.6 GHz.

The specifications of the test cases are based on the test examples presented in the previous chapters of this thesis, as well as on the earlier research results concerning spectral elements and exact controllability approaches with acoustic and elastic wave equations solved in separate domains. During the computations of acoustic and elastic problems in separate domains, we have observed that with the higher-order spectral element method, a certain error level can be reached with lower computational work than with the conventional FEM also when the central finite difference scheme is considered for time discretization. Furthermore, the computational effort of the method has linear dependence on the number of nonzero elements in the stiffness matrix, and the number of preconditioned CG iterations appears to be independent of the order of the spectral element basis, which confirms the efficiency of the GBMG preconditioner and makes the solver feasible for higher-order discretizations.

Previously, in [106], we used the central finite difference scheme for time discretization of an acoustic scattering by a sound-soft obstacle. The spectral element discretization used in that article results in a global mass matrix which is diagonal by construction. That time discretization 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. Only matrix-vector products are needed in time-dependent simulation, which leads to a very efficient implementation of the control algorithm. As a drawback, the scheme needs to satisfy the CFL condition, which limits the length of the time step (see [44] for details).

When higher-order spectral elements are used with the second-order time discretization, very small time steps are needed to eliminate the temporal error. Since the second-order accurate time discretization scheme restricted the efficiency of the overall control approach with higher-order elements, the central finite difference scheme was compared with the fourth-order accurate Runge–Kutta method for the acoustic and elastic wave equations in [107, 162], respectively.

In [4] it was confirmed that the method based on exact controllability and spectral elements performs well when compared with shifted-Laplacian preconditioning. The quadrilateral spectral elements required fewer discretization nodes than triangular finite elements to obtain the same accuracy level. Consequently, the control method with spectral elements gave more accurate results and used less memory than the shifted-Laplacian method with triangular higher-order finite elements. Nevertheless, the CPU time consumption for the controllability approach was larger than for the shifted-Laplacian method.

Now, the main focus of the numerical experiments is on the coupled fluid-structure interaction problems. That is, we test how the coupling between the domains affects the efficiency of the control method. Some preliminary experiments discussing acousto-elastic scattering problems with exact controllability and spectral elements are published in [160]. Coupling between the pressure and the displacement was discussed. It turned out that finding a proper least-squares functional for the minimization problem of the non-symmetric formulation is not a straightforward task in the two-dimensional domain. That is why we minimized the functional related to the symmetric formulation arising from the coupling between the velocity potential and the displacement. Because of the poor choice of the least-squares functional, the number of iterations required to attain the stopping criterion was extremely large. This is further verified in Section 8.1, and in the latter test cases we prefer the symmetric formulation.

By the efficiency of the method we refer to the ratio between accuracy and the computational work required to achieve that level of accuracy. The main goal of the numerical experiments presented in the further sections is to study the accuracy of 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.

In the articles [107, 162] we constructed artificial problems, the solutions of which are known to be the plane wave, for acoustic and elastic problems separately. That is how we eliminated the error caused by the absorbing boundary condition and were able to study the effect of the spatial discretization. We presented the accuracy of the approximation and showed how to define the size of the timestep which eliminates the temporal error. In this thesis, we slightly change the viewpoint since the accuracy issues are well covered with the transient fluid-structure interaction problems in Chapter 5. Therefore, in this chapter we focus on the influence of the control algorithm to the overall accuracy.

TABLE 8 Mesh stepsizes, number of timesteps, and number of degrees of freedom for different spectral orders with $\omega = 4\pi$.

r	1	2	3	4	5
Mesh stepsize	$1/20$	$1/10$	$1/7$	$1/5$	$1/4$
Number of timesteps to satisfy the stability condition	100	100	130	140	160
Number of degrees of freedom	1323	1323	1452	1323	1323

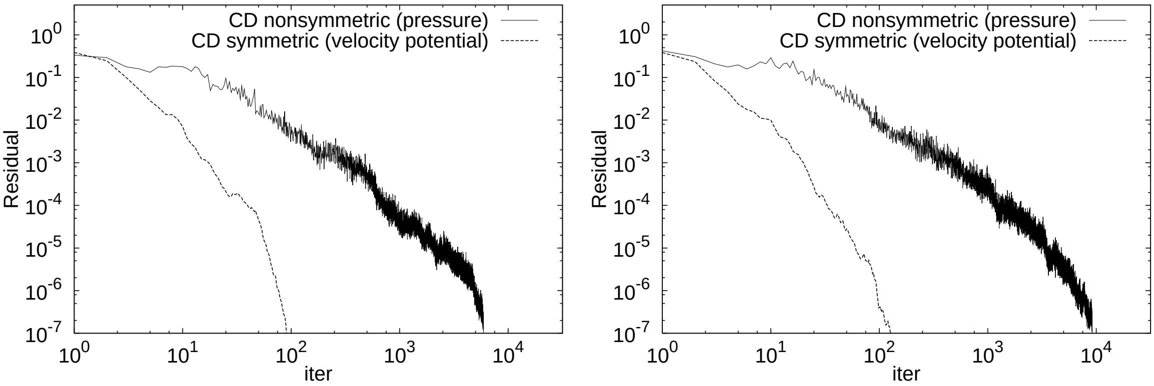
Since geometries with curved boundaries can not be represented exactly by a rectangular mesh, we use polygonal boundaries in the test experiments.

8.1 Convergence rate

First we show how important it is to choose a sufficient functional for minimization. We use the functional (157), which is derived via the energy consideration for the symmetric formulation, in which the velocity potential is used as a fluid variable. By minimizing this functional, we solve the exact controllability problem in the case of the CD time discretization with both the symmetric and the non-symmetric formulations, where the state equations and the domains are the same as the ones used in Example 5.1.2. When higher-order elements are used, good efficiency with high accuracy can be achieved by using sufficiently large mesh stepsize [2]. This is why we use coarser mesh with higher element order. In these computations, the resolution of the spatial discretization, that is, degrees of freedom per wavelength, is approximately constant such that $r/h \approx 20$. The number of timesteps is chosen such that the CFL condition is satisfied (see Table 8).

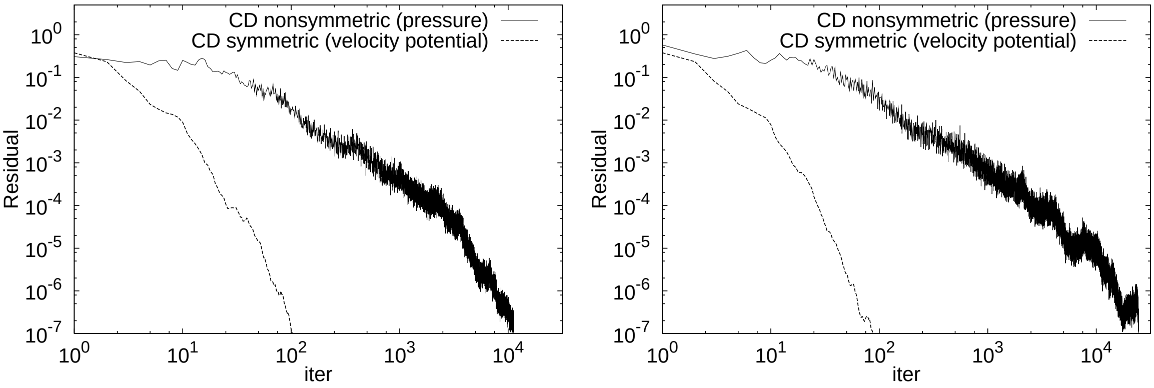
The convergence history, shown in Figure 25, impresses that it is not reasonable in conjunction with the non-symmetric formulation to use the functional derived to the symmetric formulation. Although it seems to be possible to attain the solution also in that way, the convergence rate is really slow. For instance, if we use the stopping criterion $\varepsilon = 10^{-7}$, the non-symmetric scheme needs thousands of iterations, whereas approximately one hundred iterations are needed in the case of the symmetric formulation to attain the stopping criterion.

From Figure 26 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. Thus, the natural energy functional seems to be a sufficient choice for minimization. The larger number of iterations means heavier computational cost. Consequently, the efficiency of the method is lost if the functional for minimization is not chosen properly. As deriving, implementing and computing an appropriate functional for the non-symmetric formulation is rather complicated, we conclude that the non-symmetric formulation is not well suited for the control algorithm. However, it may work well with other solution ap-



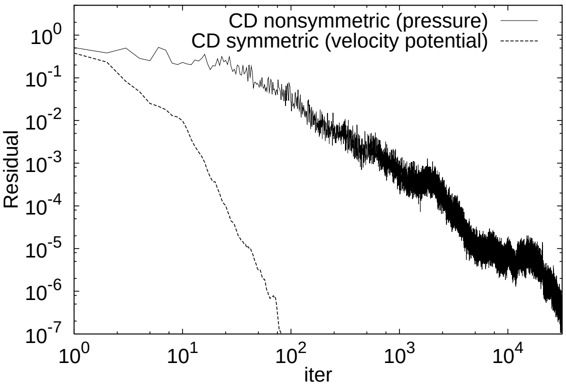
(a) r=1

(b) r=2



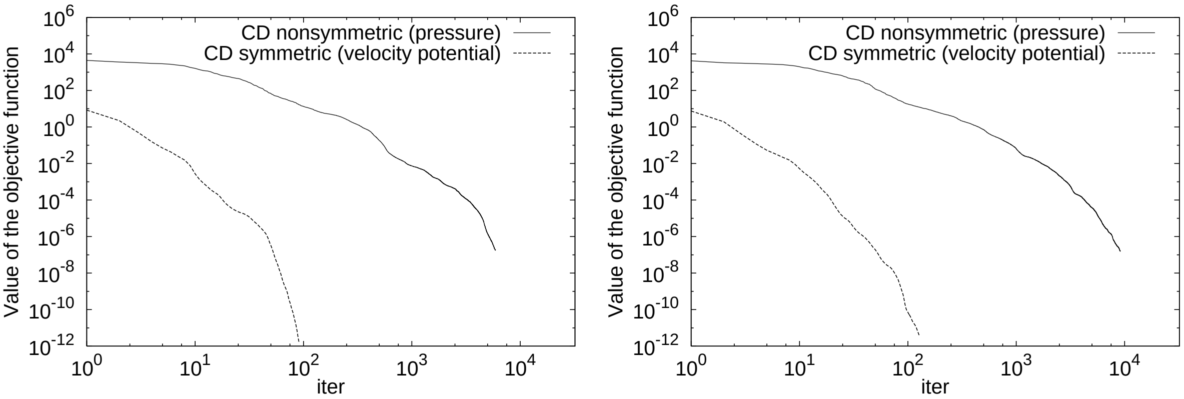
(c) r=3

(d) r=4

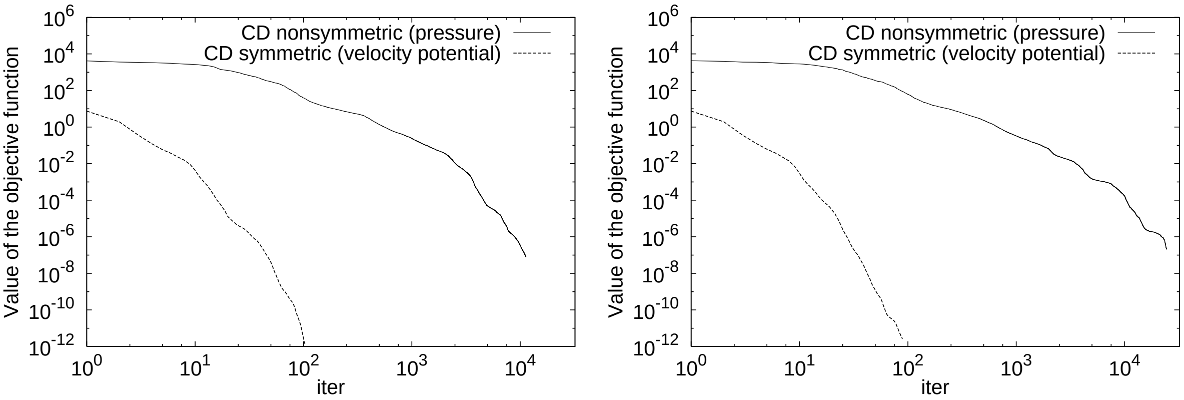


(e) r=5

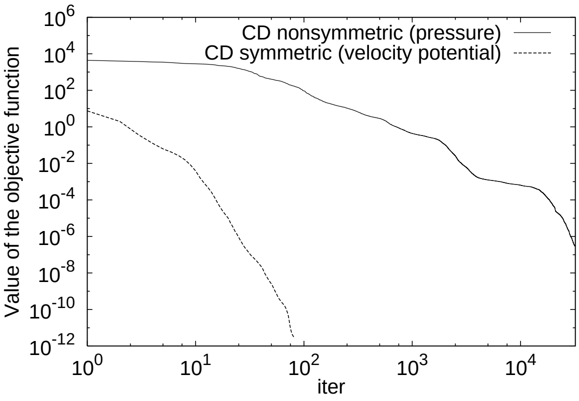
FIGURE 25 Comparison between the convergence histories of the relative euclidean norm of the residual with respect to the number of iterations when the functional (157) is minimized by the preconditioned CG algorithm with the non-symmetric and the symmetric formulation.



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



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

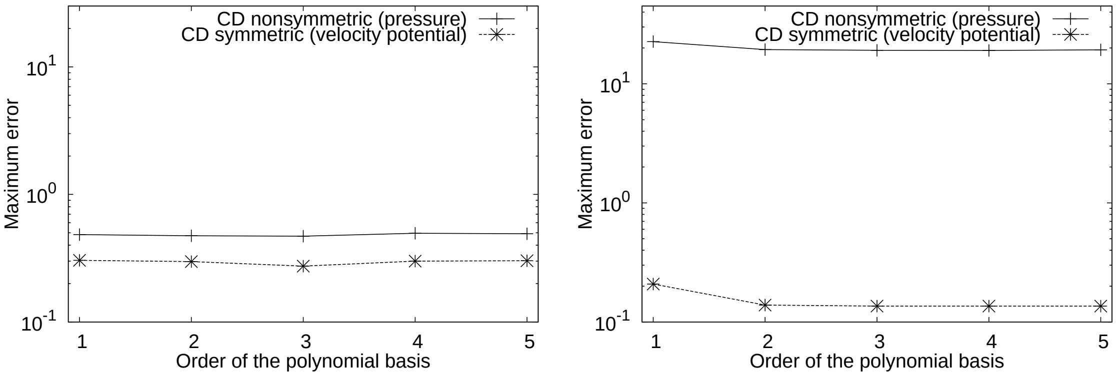


(e) r=5

FIGURE 26 Values of the functional (157) with respect to the number of iterations in the case of non-symmetric and symmetric formulations.

proaches such that preconditioned iterative methods, such as GMRES or Bi-CGSTAB, applied straight to the time-harmonic equations.

As we have shown in Example 5.1.2, presented in Chapter 5, the solutions achieved by solving the time-dependent state equation is even more accurate in the case of the symmetric than non-symmetric formulation. This seems to be valid also for the time-harmonic solutions computed via the control algorithm. That is, the error, depicted in Figure 27, is larger in the case of non-symmetric formulation especially in the fluid domain. Thus, the control algorithm is not ruining the phenomenon. What is remarkable, the error is not decreasing significantly when the higher-order polynomial basis is used. Summing up these features, we decide to concentrate on the examples considering the symmetric formulation in the following numerical experiments.



(a) Solid domain.

(b) Fluid domain.

FIGURE 27 Errors between the solution of the control algorithm and the analytical solution measured in the L^∞ norm.

8.2 Analytical solution for time-harmonic fluid-structure scattering problems

Analytical solutions for scattering problems in the field of fluid-structure interaction exist only for some simple geometries, like a circle. However, using them is sometimes reasonable for testing the accuracy of numerical methods. In this section, we consider scattering by an elastic circle Ω_s, having radius $a_s = 1$, embedded in the center of an acoustic domain Ω_f which is truncated by a circular absorbing boundary of radius $a_f = 2$ (see

Figure 28). In the time-harmonic frame the problem can be modeled as

$$-\kappa(\mathbf{x})^2\Phi - \nabla^2\Phi = 0, \qquad \text{in } \Omega_f, \tag{209}$$

$$-i\kappa(\mathbf{x})\Phi + \frac{\partial\Phi}{\partial\mathbf{n}_f} = Y_{\Phi\mathrm{ext}}, \qquad \text{on } \Gamma_{ef}, \tag{210}$$

$$i\omega\mathbf{U_s}\cdot\mathbf{n}_s + \frac{\partial\Phi}{\partial\mathbf{n}_f} = 0, \qquad \text{on } \Gamma_i, \tag{211}$$

$$-\omega^2\rho_s(\mathbf{x})\mathbf{U_s} - \nabla\cdot\sigma(\mathbf{U_s}) = \mathbf{0}, \qquad \text{in } \Omega_s, \tag{212}$$

$$\sigma(\mathbf{U_s})\mathbf{n}_s + i\omega\rho_f(\mathbf{x})\Phi\mathbf{n}_f = \mathbf{0}, \qquad \text{on } \Gamma_i, \tag{213}$$

where i is the imaginary unit, ω is the angular frequency, and $\kappa = \omega/c$ is the wavenumber describing how many waves there are in the fluid domain for a 2π unit. Respectively, we can define wavenumbers in the solid domain $\kappa_p = \omega/c_P$ and $\kappa_S = \omega/c_S$, where c_p and c_s represent the speed of the pressure waves (P-waves) and the speed of the shear waves (S-waves).

We test the accuracy of the control method in a circular domain with the incident plane wave $\phi_{\mathrm{inc}}(\mathbf{x},t) = \cos(\kappa x_1 - \omega t)$ implying $y_{\phi\mathrm{ext}} = \kappa(1 - n_{f1})\sin(\kappa x_1 - \omega t)$. The time-dependence is presented in the form $e^{i\omega t}$, meaning that the relations between time-dependent and time-harmonic variables are $\phi(\mathbf{x},t) = \mathrm{Re}(\Phi(\mathbf{x})e^{-i\omega t})$, leading to $Y_{\Phi\mathrm{ext}}(\mathbf{x}) = -i\kappa(1 - n_{f1})e^{i\kappa x_1}$, and $\mathbf{u_s}(\mathbf{x},t) = \mathrm{Re}(\mathbf{U_s}(\mathbf{x})e^{-i\omega t})$. An analytical solution of the time-harmonic problem, accomplished by using the separation of variables, can be presented in polar coordinates (r,α) by a Fourier series as follows (see, e.g., [115])

$$\Phi_r(r,\alpha) = e^{i\kappa r\cos\alpha} + \sum_{n=0}^{\infty}\left(A_n H_n^{(1)}(\kappa r) + B_n H_n^{(2)}(\kappa r)\right)\cos(n\alpha), \tag{214}$$

$$\mathbf{U_s}_r(r,\alpha) = \begin{pmatrix} \sum_{n=0}^{\infty}\left(C_n\left(J_{n-1}(\kappa_p r) - \frac{n}{\kappa_p r}J_n(\kappa_p r)\right)\kappa_p + D_n\frac{n}{r}J_n(\kappa_s r)\right)\cos(n\alpha) \\ \sum_{n=0}^{\infty}\left(-C_n\frac{n}{r}J_n(\kappa_p r) - D_n\left(J_{n-1}(\kappa_s r) - \frac{n}{\kappa_s r}J_n(\kappa_s r)\right)\kappa_s\right)\sin(n\alpha) \end{pmatrix}, \tag{215}$$

where $e^{i\kappa r\cos\alpha} = J_0(\kappa r) + \sum_{n=1}^{\infty} 2i^n J_n(\kappa r)\cos(n\alpha)$ is the incident plane wave, J_n represents the Bessel functions and $H_n^{(1)}$ and $H_n^{(2)}$ are the Hankel functions. The coefficients

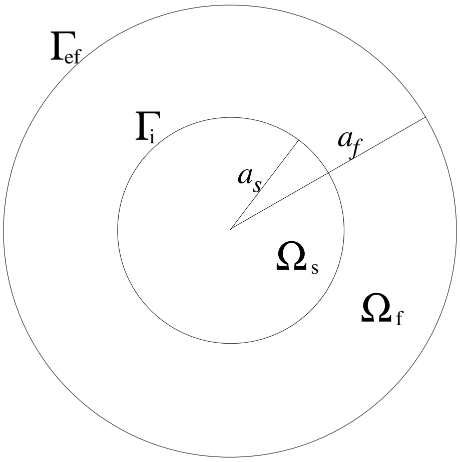
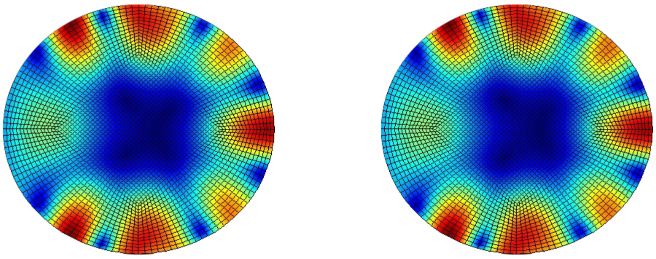


FIGURE 28 The elastic circle Ω_s surrounded by the acoustic media Ω_f.

TABLE 9 Number of timesteps and space discretization points for different spectral orders with $\omega = 5\pi$ in the case of scattering by elastic obstacles in a circular domain.

r	1	2	3	4	5
Timesteps N for the CD time discretization	90	330	850	1370	1980
Timesteps N for the RK time discretization	60	140	270	430	600
Number of discretization point in Ω_s	801	3137	7009	12417	19361
Number of discretization point in Ω_f	832	3200	7104	12544	19520



(a) Control algorithm solution $\sqrt{\mathbf{u_s}_1^2 + \mathbf{u_s}_2^2}$

(b) Fourier series solution $\sqrt{\mathrm{Re}(\mathbf{U}_{s1})^2 + \mathrm{Re}(\mathbf{U}_{s2})^2}$

FIGURE 29 Displacement amplitude solutions in the solid domain.

A_n, B_n, C_n, and D_n can be solved from the linear system

$$H_n^{'(1)}(\kappa a_s)A_n + H_n^{'(2)}(\kappa a_s)B_n - \frac{i\omega}{a_s^2}\left(\kappa_p a_s J_{n-1}(\kappa_p a_s) - nJ_n(\kappa_p a_s)\right)C_n - \frac{i\omega}{a_s^2} nJ_n(\kappa_s a_s)D_n$$
$$= \begin{cases} -i^n J_n^{'}(\kappa a_s), & \text{for } n = 0, \\ -2i^n J_n^{'}(\kappa a_s), & \text{for } n = 1, \dots, \infty, \end{cases}$$

$$(H_n^{'(1)}(\kappa a_f) - i\kappa H_n^{(1)}(\kappa a_f))A_n + (H_n^{'(2)}(\kappa a_f) - i\kappa H_n^{(2)}(\kappa a_f))B_n = 0,$$

$$- i\omega\rho_f H_n^{(1)}(\kappa a_s)A_n - i\omega\rho_f H_n^{(2)}(\kappa a_s)B_n + \frac{2\mu_s}{a_s^2}\left(\left(n^2 + n - \frac{\kappa_s^2 a_s^2}{2}\right)J_n(\kappa_p a_s) - \kappa_p a_s J_{n-1}(\kappa_p a_s)\right)C_n$$
$$+ \frac{2\mu_s}{a_s^2} n\left(\left(-(n+1)\right)J_n(\kappa_s a_s) + \kappa_s a_s J_{n-1}(\kappa_s a_s)\right)D_n = \begin{cases} \omega\rho_f i^{n+1} J_n(\kappa a_s), & \text{for } n = 0, \\ 2\omega\rho_f i^{n+1} J_n(\kappa a_s), & \text{for } n = 1, \dots, \infty, \end{cases}$$

$$- n\left(\left(-(n+1)\right)J_n(\kappa_p a_s) + \kappa_p a_s J_{n-1}(\kappa_p a_s)\right)C_n - \left(n^2 + n - \frac{\kappa_s^2 a_s^2}{2}\right)J_n(\kappa_s a_s) - \kappa_s a_s J_{n-1}(\kappa_s a_s)D_n = 0,$$

constructed by substituting the formulas (214) and (215) into the boundary and coupling conditions (210), (211), and (213) in polar coordinates. Derivatives of the Bessel and Hankel functions are denoted as $J_n^{'}$ and $H_n^{'}$, respectively.

The test simulation was run with angular frequency $\omega = 5\pi$, element orders $r = 1, \dots, 5$ ($r = 1$ corresponds to bilinear finite elements), and material parameters $\rho_f(\mathbf{x}) = 1$ and $\rho_s(\mathbf{x}) = 2.7$, $c(\mathbf{x}) = 1.5$, $c_p(\mathbf{x}) = 6.20$ and $c_s(\mathbf{x}) = 3.12$. The computational domains are presented by rectangular meshes. The minimum and maximum mesh

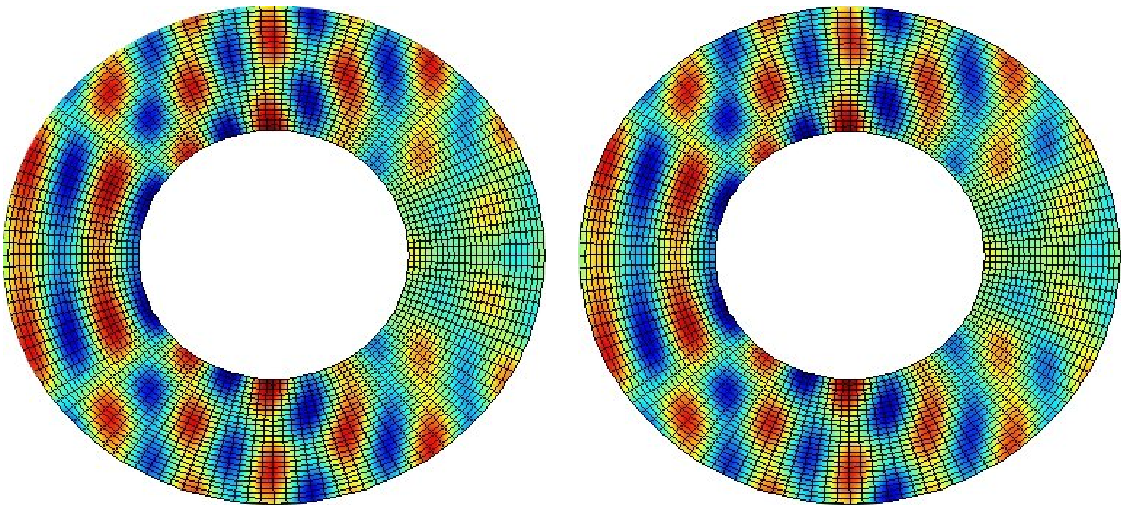
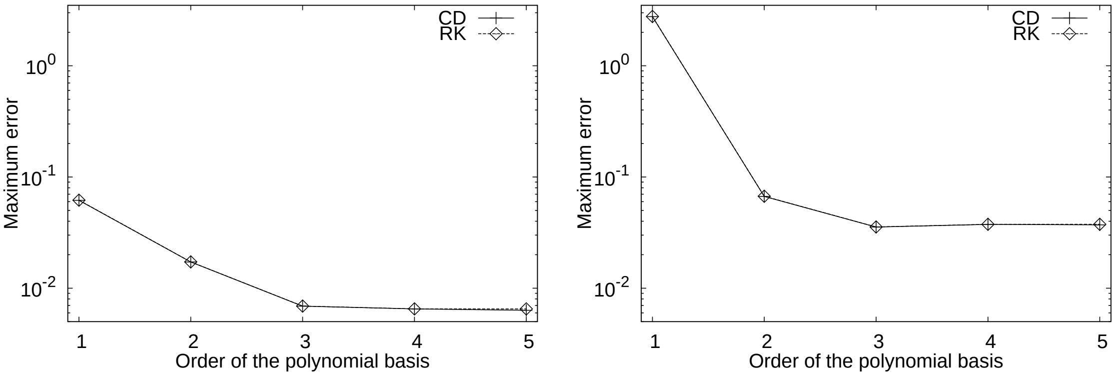

(a) Control algorithm solution ϕ (b) Fourier series solution $\mathrm{Re}(\Phi)$

FIGURE 30 Velocity potential solutions in the fluid domain.

stepsizes in the fluid domain are $h_{f,min} \approx 0.0833$ and $h_{f,max} \approx 0.1963$. Respectively, the minimum and maximum mesh stepsizes in the solid domain are $h_{s,min} \approx 0.0530$ and $h_{s,max} \approx 0.0982$. The stopping criterion of the control algorithm is achieved when the relative euclidean norm of the gradient of the least-squares functional is below 10^{-5}, whereas the Fourier modes n are computed until the relative difference attained by adding the next mode to the series is below 10^{-5}. The number of timesteps used for solving the time-dependent state and adjoint equations is shown in Table 9.

The control algorithm solution in the solid domain with the RK time discretization and $r = 2$ is illustrated in Figure 29(a), whereas the real part of the Fourier series solution in the solid domain is shown in Figure 29(b). Respectively, the control algorithm solution with the RK time discretization and $r = 2$ and the Fourier series solution of the values of the velocity potential in the fluid domain are presented in Figures 30(a) and 30(b). Naturally, there is no qualitative difference between the solutions attained by the control algorithm and the Fourier series solution.

To take a closer look at the accuracy, we consider the quantitative difference between the control algorithm and the Fourier series solution. These errors, measured using the L^∞ norm, are presented in the case of the CD and the RK time discretization in Figure 31. In principle, the accuracy of the spatial discretization increases with the element order. The horizontal line in the error curves represents the level of the dominating error source, which is caused by some factor other than the spatial discretization. Since the Fourier series solution is defined to satisfy the artificial boundary condition on Γ_{ef}, the influence of the radiation boundary condition to the error can be excluded. The possible sources interrupting the convergence of the error level include the accuracy of time discretization, the stopping criterion of the Fourier series solution, and approximation of the geometrical shapes. It is noteworthy that in this case both time discretization schemes give equally accurate results for all the element orders. We also run some tests by tightening the stopping criterion of the Fourier series solution and using smaller timesteps with both time-discretization schemes. Since these attempts did not improve the accuracy, we



(a) Error $||\mathrm{Re}(\mathbf{U_s}) - \mathbf{u_s}||_\infty$ in the solid domain. (b) Error $||\mathrm{Re}(\Phi) - \phi||_\infty$ in the fluid domain.

FIGURE 31 Errors between the control algorithm and the Fourier series solution measured in the L^∞ norm.

assume that approximating the curvilinear shapes of the domain by quadrilateral elements limits the overall accuracy. In the next sections we take a closer look at the accuracy and efficiency and isolate the different error sources.

8.3 Influence of the stopping criterion

According to the results shown in Table 3, we can choose the timestep to examine the spatial discretization such that the error of time discretization is negligible. When using the control algorithm with a constant stopping criterion for all element orders r, so small timesteps are not reasonable. This is the case especially when concerning higher-order elements with a sufficiently large stopping criterion ε. The reason for this is that the stopping criterion limits the accuracy of the control method. Therefore, our aim is to consider the stopping criterion and the length of the timestep for each spectral order.

To show how the stopping criterion of the algorithm influences the accuracy, we solve with the control method the time-harmonic elasticity problem

$$-\omega^2 \rho_s(\mathbf{x})\mathbf{U_s} - \nabla \cdot \sigma(\mathbf{U_s}) = \mathbf{F}, \qquad \text{in } \Omega_s, \tag{216}$$

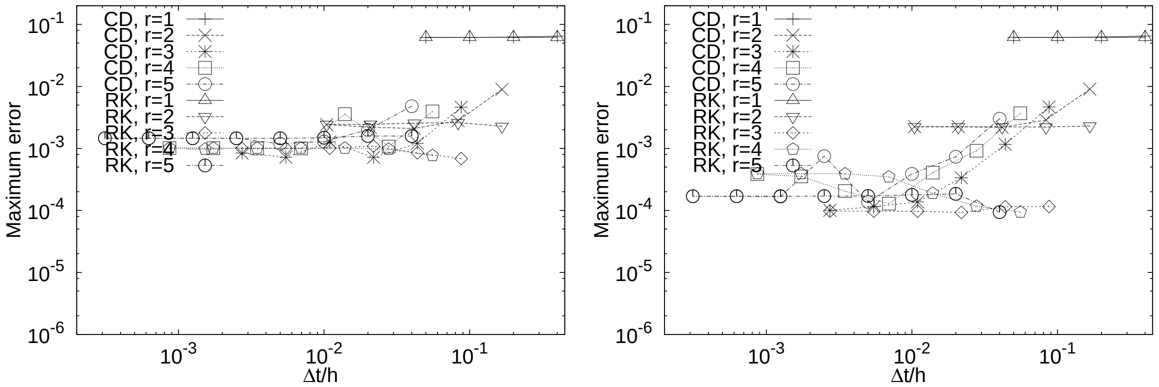
$$\mathbf{U_s} = \mathbf{G}_0, \qquad \text{on } \Gamma_{0s}, \tag{217}$$

$$i\omega \rho_s(\mathbf{x})\mathbf{B}_s\mathbf{U_s} + \sigma(\mathbf{U_s})\mathbf{n}_s = \mathbf{G}_{\text{ext}}, \qquad \text{on } \Gamma_{es}, \tag{218}$$

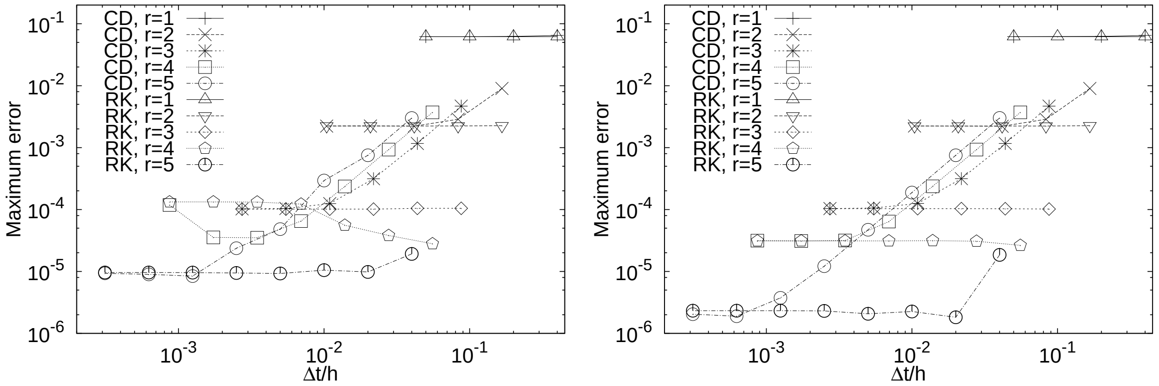
with $\omega = 2\pi$, $\boldsymbol{\omega} = \left(-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right)$, $\rho = 2.7$, $c_p = 2$, and $c_s = 1$. The right-hand side functions $\mathbf{F}$, $\mathbf{G}_0$, and $\mathbf{G}_{\text{ext}}$ are defined to satisfy

$$\mathrm{Re}(\mathbf{U_s}) = \begin{pmatrix} \omega_1 \cos(\frac{\omega}{c_p}\vec{\omega} \cdot \mathbf{x}) + \omega_2 \cos(\frac{\omega}{c_s}\vec{\omega} \cdot \mathbf{x}) \\ \omega_2 \cos(\frac{\omega}{c_p}\vec{\omega} \cdot \mathbf{x}) - \omega_1 \cos(\frac{\omega}{c_s}\vec{\omega} \cdot \mathbf{x}) \end{pmatrix}, \tag{219}$$

$$\mathrm{Im}(\mathbf{U_s}) = \begin{pmatrix} \omega_1 \sin(\frac{\omega}{c_p}\vec{\omega} \cdot \mathbf{x}) + \omega_2 \sin(\frac{\omega}{c_s}\vec{\omega} \cdot \mathbf{x}) \\ \omega_2 \sin(\frac{\omega}{c_p}\vec{\omega} \cdot \mathbf{x}) - \omega_1 \sin(\frac{\omega}{c_s}\vec{\omega} \cdot \mathbf{x}) \end{pmatrix}. \tag{220}$$



(a) $\varepsilon = 10^{-4}$ (b) $\varepsilon = 10^{-5}$



(c) $\varepsilon = 10^{-6}$ (d) $\varepsilon = 10^{-7}$

FIGURE 32 Maximum errors obtained in the case of the CD and the RK time-stepping with four different stopping criteria ε.

The boundary surrounding the domain $\Omega = \Omega_s$, Γ_{es}, coincides with the boundary of the rectangle $[0,0] \times [4,4]$. We have set a square obstacle, having a side length of 2 and boundary $\Gamma_{0\mathrm{s}}$ in the center of the domain Ω_s. The domain is divided into rectangular elements, and the ratio between the order of elements r and the mesh stepsize h is $r/h \approx 20$.

We have started computations with the largest stable timestep (see Table 2) 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. Errors between the analytical solution and the experimental result are computed as L^{∞}-norms. Accuracy of the numerical solution in the structure domain is shown in Figure 32 as a function of the ratio between the time step Δt and the mesh step size h for both the CD and the RK time-steppings with five element orders r and four different stopping criteria ε.

We have eliminated the error of approximation of the geometry by using polygonal boundaries and the right-hand side functions satisfy the absorbing boundary condition. Hence, as the order of the approximation in space increases, the solution becomes more accurate until the effect of the stopping criterion or the error of time or space discretization becomes dominant. For the element order $r = 1$, the error of space discretization is

TABLE 10 Number of space discretization points for different element orders in pollution tests.

	Element order r				
	1	2	3	4	ω
Number of space discretization points	5040	5040	5544	5040	2π
	19680	19680	21672	19680	4π
	77760	77760	85680	77760	8π
	309120	309120	340704	309120	16π
Stopping criteria ε	10^{-3}	10^{-4}	10^{-5}	10^{-6}	

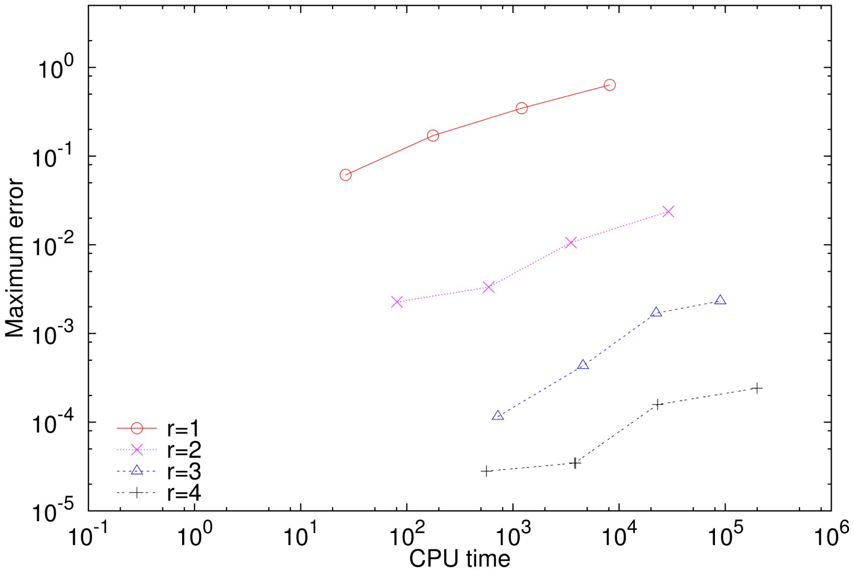


FIGURE 33 Errors with respect to CPU time (in seconds) with angular frequencies $\omega = \{2\pi, 4\pi, 8\pi, 16\pi\}$ such that $\omega h = r\pi/10$.

dominant over the error of time discretization for all stable lengths of the timestep. Then, using the second-order and the fourth-order accurate time schemes leads to equally accurate results. When higher-order elements are used, we can see the benefit of the RK time discretization. We also notice that using a tight stopping criterion is just a waste of time when low-order elements are used. For the element order r, the results would have been equally accurate if the stopping criterion $\varepsilon = 10^{-(r+2)}$ had been used.

8.4 Pollution effect

Even though we have eliminated the main error sources, the pollution effect deteriorates the accuracy of solutions with high frequency. That is, 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 with several different angular frequencies and the resolutions of the mesh such that $\omega h = r\pi/10$ (see Table 10). Furthermore, we use the stopping criterion $\varepsilon = 10^{-(r+2)}$, which is not limiting the accuracy of the numerical solutions computed with the CG algorithm for the element order r. The number of timesteps is chosen to eliminate the temporal error according to Table 3.

The accuracy of the solution with the RK time discretization is presented in Figure 33 with respect to the CPU time. The CPU time for the algorithm grows with the wavenumber. Naturally, the reason for this is the increase in the number of CG iterations. As the wavenumber grows, the error increases for the all element orders. In the case of classical finite element discretization, that is, $r = 1$, 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 also for high angular frequencies. Thus, better accuracy is achieved with less work, when higher-order elements are used.

9 CONCLUSIONS

This study was concentrated on developing numerical solution techniques for simulating the interaction between acoustic and elastic waves. The focus was in particular on the mutual interaction between two time-harmonic linear wave equations: the scalar-valued Helmholtz equation concerning the propagation of acoustic waves and the vector-valued Navier-Cauchy equation describing the propagation of waves in an elastic medium. Several formulations of the corresponding equations were presented, and the practicality of them was considered mostly from an implementational point of view.

We considered the spectral element space discretization for both time-dependent and time-harmonic equations considering acoustic and elastic wave propagation and the fluid-structure interaction. The spectral element formulation, based on high-degree polynomials, used in this thesis results in a global mass matrix that is diagonal by construction, which leads to an efficient implementation. This is an advantage compared with the classical finite element method.

The temporal discretization for time-dependent equations was made by the second-order accurate central finite difference scheme or the fourth-order accurate Runge–Kutta or Adams–Bashforth approaches. The performance of the different time discretization schemes was compared numerically. We found out that the Adams–Bashforth scheme has such strict stability conditions that it is not a feasible choice at this stage. For the other two time discretization schemes we defined the constants k_r, for the element orders $r = 1, \dots, 5$. It can be used for computing the length of the timestep that eliminates the temporal error but is still large enough to maintain the efficiency.

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 the order of the element is greater than the order of the time discretization method used. The second-order central finite difference time discretization method is efficient with finite elements, but when high accuracy is needed, it is best to use the Runge–Kutta time discretization. It is also worth mentioning that with the orders orders $r \geq 5$ we should use even higher-order time discretization schemes than considered in this thesis. However, in the simulations independent of the time, the accuracy can be increased efficiently with the element orders higher than 5.

We observed that with the higher-order spectral element method, a certain error level can be reached with lower computational work than with the conventional finite element method also when the central finite difference scheme is considered for time discretization. The spectral element method requires fewer grid points per wavelength to the same accuracy as the spectral element method. Consequently, accurate results are reached by solving smaller systems, that is, fewer computational operations, which saves the computing time.

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 Périaux. 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. The function to be minimized has shown to play a key role in the efficiency of the method, and at this stage we applied the natural energy formulation. We computed the gradient by first discretizing the wave equation and the function to be minimized. Then, we computed the gradient directly for the discretized problem by following the adjoint equation technique. The practical realization of the method was implemented in Fortran 95.

The validation of the accuracy of the control approach is done by comparing the results with the known analytical solutions. In the case of wave propagation, the exact time-harmonic solution was computed by using the Hankel and Bessel functions. The accuracy of the spatial discretization is shown to increase with the element order until an error factor, such as time discretization, approximation of the geometrical shapes or a stopping criterion, disturbs the approach. The test problems were chosen such that as many error factors as possible were eliminated. With curvilinear geometries, the shapes are not approximated accurately with the discretization presented here by the quadrilateral elements. Thus, this presentation gives accurate results for the polygonal geometries, and curvilinear elements should be applied when curvilinear geometries are involved.

The simulation results show that the number of iterations required to attain the stopping criterion is independent of the element order. Furthermore, the computational effort of the method seems to have linear dependence on the number of degrees of freedom, and the number of preconditioned conjugate gradient iterations appears to be independent of the order of the spectral element basis, which confirms also the efficiency of the graph-based multigrid preconditioner and makes the solver feasible for higher-order elements. We also noticed that it is feasible to use a less tight stopping criterion for the low-order elements and, on the other hand, a more tight stopping criterion for the higher-order elements. This results in better efficiency without loss of accuracy. With higher-order elements, the pollution effect is not eliminated, but better accuracy is achieved with less work, when higher-order elements are used.

REFERENCES

[1] M. S. Abbes, S. Bouaziz, F. Chaari, M. Maatar, and M. Haddar. An acoustic-structural interaction modelling for the evaluation of a gearbox-radiated noise. *International Journal of Mechanical Sciences*, 50(3):569 – 577, 2008.

[2] 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.

[3] 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.

[4] T. Airaksinen and S. Mönkölä. Comparison between shifted-Laplacian preconditioning and controllability method for computational acoustics. *Journal of Computational and Applied Mathematics*, 6(234):1796–1802, 2010.

[5] T. Airaksinen, A. Pennanen, and J. Toivanen. A damping preconditioner for time-harmonic wave equations in fluid and elastic material. *Journal of Computational Physics*, 228:1466–1479, 2009.

[6] M. Akköse, S. Adanur, A. Bayraktar, and A. A. Dumanoglu. Elasto-plastic earthquake response of arch dams including fluid-structure interaction by the lagrangian approach. *Applied Mathematical Modelling*, 32(11):2396 – 2412, 2008.

[7] O. Andrianarison and 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):1958–1972, 2006. Fluid-Structure Interaction.

[8] 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.

[9] F. Axisa, F. Axisa, and J. Antunes. *Modelling of mechanical systems: Fluid structure interaction*. Butterworth-Heinemann, 2006.

[10] I. Babuska and U. Banerjee. Stable generalized finite element method (sgfem). Technical Report ICES REPORT 11-07, The Institute for Computational Engineering and Sciences, The University of Texas at Austin, USA, 2011.

[11] I. Babuska and J. M. Melenk. The partition of unity method. *International Journal of Numerical Methods in Engineering*, 40:727–758, 1996.

[12] 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.

[13] 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.

[14] K.-J. Bathe, C. Nitikitpaiboon, and X. Wang. A mixed displacement-based finite element formulation for acoustic fluid-structure interaction. *Computers & Structures*, 56(2/3):225–237, 1995.

[15] K.-J. Bathe and H. Zhang. Finite element developments for general fluid flows with structural interactions. *International Journal for Numerical Methods in Engineering*, 60(1):213–32, 2004.

[16] A. Bayliss, M. Gunzburger, and E. Turkel. Boundary conditions for the numerical solution of elliptic equations in exterior regions. *SIAM Journal on Applied Mathematics*, 42(2):430–451, 1982.

[17] 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.

[18] T. B. Belytschko and J. M. Kennedy. A fluid-structure finite element method for the analysis of reactor safety problems. *Nuclear Engineering and Design*, 38(1):71–81, 1976.

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

[20] 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.

[21] J.-P. Berenger. A perfectly matched layer for the absorption of electromagnetic waves. *Journal of Computational Physics*, 114(2):185 – 200, 1994.

[22] A. Bermúdez, P. Gamallo, L. Hervella-Nieto, and E. Rodríguez. Finite element analysis of pressure formulation of the elastoacoustic problem. *Numerische Mathematik*, 95(1):29–51, 2003.

[23] A. Bermúdez, P. Gamallo, P. Hervella-Nieto, R. Rodríguez, and D. Santamarina. *Computational Acoustics of Noise Propagation in Fluids. Finite and Boundary Element Methods.*, chapter Fluid-structure acoustic interaction, pages 253–286. Springer, 2008.

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

[25] 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.

[26] F. J. Blom. A monolithical fluid-structure interaction algorithm applied to the piston problem. *Computer Methods in Applied Mechanics and Engineering*, 167(3-4):369–391, 1998.

[27] 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.

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

[29] W. L. Briggs, V. E. Henson, and S. F. McCormick. *A multigrid tutorial*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2 edition, 2000.

[30] 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.

[31] 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.

[32] 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.

[33] 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.

[34] H. J. Bungartz and M. Schäfer. *Fluid-structure interaction: modelling, simulation, optimisation*. Springer, 2006.

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

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

[37] P. Causin, J. F. Gerbeau, and F. Nobile. Added-mass effect in the design of partitioned algorithms for fluid-structure problems. *Computer Methods in Applied Mechanics and Engineering*, 194(42-44):4506 – 4527, 2005.

[38] 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.

[39] E. Chaljub, Y. Capdeville, and 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):457–491, 2003.

[40] L. Cheng, R. D. White, and K. Grosh. Three-dimensional viscous finite element formulation for acoustic fluid-structure interaction. *Computer Methods in Applied Mechanics and Engineering*, 197(49-50):4160 – 4172, 2008.

[41] M. J. S. Chin-Joe-Kong, W. A. Mulder, and M. van Veldhuizen. Higher-order triangular and tetrahedral finite elements with mass lumping for solving the wave equation. *Journal of Engineering Mathematics*, 35:405–426, 1999. 10.1023/A:1004420829610.

[42] 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.

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

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

[45] G. Cohen and M. Duruflé. Mixed spectral elements for the Helmholtz equation. In G. Cohen, E. Heikkola, P. Joly, and P. Neittaanmäki, editors, *Mathematical and Numerical Aspects of Wave Propagation, Proceedings of WAVES 2003*, pages 743–748, Jyväskylä, Finland, 2003. Springer-Verlag.

[46] G. Cohen and S. Fauqueux. Mixed finite elements with mass-lumping for the transient wave equation. *Journal of Computational Acoustics*, 8(1):171–188, 2000.

[47] G. Cohen and S. Fauqueux. Mixed spectral finite elements for the linear elasticity system in unbounded domains. *SIAM Journal on Scientific Computing*, 26(3):844–863, 2005.

[48] 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.

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

[50] R. Courant, K. Friedrichs, and H. Lewy. On the partial difference equations of mathematical physics. *IBM Journal*, pages 215–234, 1967. English translation of the 1928 German original.

[51] J. M. Crolet and R. Ohayon. *Computational methods for fluid-structure interaction*. Chapman & Hall/CRC, 1994.

[52] P. Cummings and X. Feng. Domain decomposition methods for a system of coupled acoustic and elastic Helmholtz equations. In *Proceedings of the Eleventh International Conference on Domain Decomposition Methods*, pages 203–210. DDM.org, 1999.

[53] C. Czygan and O. von Estorff. Fluid-structure interaction by coupling bem and nonlinear fem. *Engineering Analysis with Boundary Elements*, 26(9):773–779, 2002.

[54] L. Demkowicz. Asymptotic convergence in finite and boundary element methods: part 1: theoretical results. *Computers & Mathematics with Applications*, 27(12):69–84, 1994.

[55] L. Demkowicz. Asymptotic convergence in finite and boundary element methods: part 2: The lbb constant for rigid and elastic scattering problems. *Computers & Mathematics with Applications*, 28(6):93–109, 1994.

[56] L. Demkowicz and J. T. Oden. Application of hp-adaptive be/fe methods to elastic scattering. *Computer Methods in Applied Mechanics and Engineering*, 133(3-4):287–317, 1996.

[57] W. Desmet, B. Pluymers, and P. Sas. Vibro-acoustic analysis procedures for the evaluation of the sound insulation characteristics of agricultural machinery cabins. *Journal of Sound and Vibration*, 266(3):407 – 441, 2003.

[58] B. Després, P. Joly, and J. E. Roberts. A domain decomposition for the harmonic maxwell's equations. In *MACS international symposium on iterative methods in linear algebra*, pages 475–484, Amsterdam, 1991. Elsevier Science Publisher B.V.

[59] E. H. Dowell and K. C. Hall. Modeling of fluid-structure interaction. *Annual Review of Fluid Mechanics*, 33:445–490, 2001.

[60] 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.

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

[62] 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.

[63] B. Engquist and A. Majda. Absorbing boundary conditions for numerical simulation of waves. *Mathematics of Computation*, 31:629–651, 1977.

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

[65] Y. A. Erlangga, C. W. Oosterlee, and C. Vuik. A novel multigrid based preconditioner for heterogeneous Helmholtz problems. *SIAM Journal on Scientific Computing*, 27(4):1471–1492, 2006.

[66] Y. A. Erlangga, C. Vuik, and C. W. Oosterlee. On a class of preconditioners for solving the Helmholtz equation. *Applied Numerical Mathematics*, 50(3-4):409–425, 2004.

[67] G. C. Everstine. Finite element formulatons of structural acoustics problems. *Computers & Structures*, 65(3):307–321, 1997.

[68] 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.

[69] 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.

[70] J. J. Faran. Sound scattering by solid cylinders and spheres. *Journal of the Acoustical Society of America*, 23(4):405–418, 1951.

[71] 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.

[72] 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.

[73] C. Farhat, M. Lesoinne, P. LeTallec, K. Pierson, and D. Rixen. FETI-DP: a dual-primal unified FETI method. i. a faster alternative to the two-level FETI method. *International Journal for Numerical Methods in Engineering*, 50(7):1523–1544, 2001.

[74] 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.

[75] 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.

[76] C. Farhat, C. Macedo, M. Lesoinne, F.-X. Roux, F. Magoulès, and A. de La Bourdonnaie. Two-level domain decomposition methods with lagrange multipliers for the fast iterative solution of acoustic scattering problems. *Computer Methods in Applied Mechanics and Engineering*, 184(2–4):213–239, 2000.

[77] C. Farhat and F.-X. Roux. An unconventional domain decomposition method for an efficient parallel solution of large-scale finite element systems. *SIAM Journal on Scientific and Statistical Computing - Special issue on iterative methods in numerical linear algebra*, 13(1):379–396, 1992.

[78] 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.

[79] 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.

[80] J. A. Fay. *Introduction to fluid mechanics*. MIT Press, 1994.

[81] X. Feng. Analysis of finite element methods and domain decomposition algorithms for a fluid-solid interaction problem. *SIAM Journal on Numerical Analysis*, 38(4):1312–1336, 2000.

[82] X. Feng, P. Lee, and Y. Wei. Mathematical analysis of a fluid-solid interaction problem. *Applicable Analysis*, 80(3):409–429, 2001.

[83] P. F. Fischer and J. W. Lottes. Hybrid Schwarz-multigrid methods for the spectral element method: Extensions to Navier-Stokes. In R. Kornhuber, R.and Hoppe, J. Périaux, O. Pironneau, O.and Widlund, and J. Xu, editors, *Domain Decomposition Methods in Science and Engineering*, volume 40 of *Lecture Notes in Computational Science and Engineering*, 2004.

[84] B. Flemisch, M. Kaltenbacher, and B. I. Wohlmuth. Elasto-acoustic and acoustic-acoustic coupling on non-matching grids. *International Journal for Numerical Methods in Engineering*, 67(13):1791–1810, 2006.

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

[86] E. Givelberg and J. Bunn. A comprehensive three-dimensional model of the cochlea. *Journal of Computational Physics*, 191(2):377 – 391, 2003.

[87] D. Givoli. High-order local non-reflecting boundary conditions: a review. *Wave Motion*, 39(4):319 – 326, 2004. New computational methods for wave propagation.

[88] G. M. L. Gladwell. A variational formulation of damped acousto structural vibration problems. *Journal of Sound and Vibration*, 4(2):172–186, 1966.

[89] G. M. L. Gladwell and V. Mason. Variational finite element calculation of the acoustic response of a rectangular panel. *Journal of Sound and Vibration*, 14(1):115–135, 1971.

[90] G. M. L. Gladwell and G. Zimmermann. On energy and complementary energy formulations of acoustic and structural vibration problems. *Journal of Sound and Vibration*, 3(3):233–241, 1966.

[91] 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.

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

[93] G. H. Golub and C. F. van Loan. *Matrix computations*. John Hopkins University Press, 1996.

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

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

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

[97] B. Han, X. Zhou, and J. Liu. Adaptive multigrid method for numerical solutions of elastic wave equation. *Applied Mathematics and Computation*, 133(1-2):609–614, 2002.

[98] 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.

[99] 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.

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

[101] 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.

[102] E. Heikkola, K. Miettinen, and P. Nieminen. Multiobjective optimization of an ultrasonic transducer using NIMBUS. *Ultrasonics*, 2006.

[103] E. Heikkola, S. Mönkölä, A. Pennanen, and T. Rossi. Controllability method for the Helmholtz equation with spectral elements. In *Proceedings of WAVES 2005*, 2005.

[104] E. Heikkola, S. Mönkölä, A. Pennanen, and T. Rossi. Controllability method for the solution of linear elastic wave equations. In *III European Conference on Computational Mechanics - Solids, Structures and Coupled Problems in Engineering*, Lisbon, Portugal, 5-9 June 2006.

[105] E. Heikkola, S. Mönkölä, A. Pennanen, and T. Rossi. Controllability techniques for the Helmholtz equation with spectral elements. In *European Conference on Computational Fluid Dynamics*, The Netherlands, 5-8 September 2006.

[106] 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.

[107] 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.

[108] E. Heikkola, S. Mönkölä, T. Rossi, and J. Toivanen. Efficient iterative solution of the Helmholtz equation with an optimized PML. In *Proceedings of Forum Acusticum*, Sevilla, 2002.

[109] 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.

[110] M. Heil. An efficient solver for the fully coupled solution of large-displacement fluid-structure interaction problems. *Computer Methods in Applied Mechanics and Engineering*, 193(1–2):1–23, 2004.

[111] M. R. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *Journal of Research of the National Bureau of Standards*, 49(6):409–436, 1952.

[112] 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.

[113] 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.

[114] T. Huttunen, J. 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.

[115] T. Huttunen, J. Kaipio, and P. Monk. An ultra-weak method for acoustic fluid-solid interaction. *Journal of Computational and Applied Mathematics*, 213(1):166–185, 2008.

[116] B. Hübner, E. Walhorn, and D. Dinkler. A monolithic approach to fluid-structure interaction using space-time finite elements. *Computer Methods in Applied Mechanics and Engineering*, 193(23–26):2087–2104, 2004.

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

[118] 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.

[119] J. Ikonen and S. Pohjolainen. Modeling and simulating tractor cabin acoustics under external excitations by using FEM/BEM. In *ISMA2002 international Conference on Noise and Vibration Engineering*, 2002.

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

[121] E. Järvinen, P. Råback, M. Lyly, and Salenius J.-P. A method for partitioned fluid-structure interaction computation of flow in arteries. *Medical Engineering & Physics*, 30(7):917–923, 2008.

[122] J. B. Keller and D. Givoli. Exact non-reflecting boundary conditions. *Journal of Computational Physics*, 82(1):172–192, 1989.

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

[124] 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.

[125] A. Klawonn, L. F. Pavarino, and O. Rheinbach. Spectral element feti-dp and bddc preconditioners with multi-element subdomains. *Computer Methods in Applied Mechanics and Engineering*, 198(3–4):511–523, 2008.

[126] D. Komatitsch, C. Barnes, and J. Tromp. Wave propagation near a fluid-solid interface: A spectral-element approach. *Geophysics*, 65(2):623–631, 2000.

[127] 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.

[128] D. Komatitsch and J. Tromp. Spectral-element simulations of global seismic wave propagation II: 3-d models, oceans, rotation, and gravity. *Geophysical Journal International*, 150:303–318, 2002.

[129] 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.

[130] 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.

[131] G. H. Koopman and J. B Fahnline. *Quiet structures. A sound minimization approach.* Academic Press, 1997.

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

[133] S. Kähkönen, R. Glowinski, T. Rossi, and R. Mäkinen. Solution of time-periodic wave equation using mixed finite-elements and controllability techniques. *Journal of Computational Acoustics*, 2011. DOI No: 10.1142/S0218396X11004377.

[134] 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.

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

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

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

[138] J. Lilja, T. Marjomäki, J. Jurvelius, T. Rossi, and E. Heikkola. Simulation and experimental measurement of side-aspect target strength of atlantic salmon (salmo salar) at high frequency. *Canadian Journal of Fisheries and Aquatic Sciences*, 61:2227–2236, 2004.

[139] J. L. Lions. *Optimal Control of Systems Governed by Partial Differential Equations.* Springer-Verlag, Berlin, 1971.

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

[141] 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.

[142] P. L. Lions. On the Schwartz alternating method I. In *Proceedings of the First International Symposium on Domain Decomposition Methods for PDEs*, pages 1–42, Philadelphia, 1988. SIAM.

[143] P. L. Lions. On the Schwartz alternating method II: Stochastic interpretation and order properties. In *Proceedings of the Second International Symposium on Domain Decomposition Methods for PDEs*, pages 47–70, Philadelphia, 1989. SIAM.

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

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

[146] 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.

[147] 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.

[148] C. Makridakis, F. Ihlenburg, and I. Babuska. Analysis and finite element methods for fluid-structure interaction problem in one dimension. *Mathematical Models and Methods in Applied Sciences*, 6(8):1119–1141, 1996.

[149] M. Malinen. *Computational methods for optimal control in ultrasound therapy*. PhD thesis, Department of applied physics, University of Kuopio, 2004.

[150] J. Mandel. An iterative substructing method for coupled fluid-solid acoustic problems. *Journal of Computational Physics*, 177:95–116, 2002.

[151] J. Mandel and M. O. Popa. Iterative solvers for coupled fluid-solid scattering. *Applied Numerical Mathematics*, 54(2):194–207, 2005.

[152] S. Marburg and B. Nolte. *Computational acoustics of noise propagation in fluids: finite and boundary element methods*. Springer, 2008.

[153] 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.

[154] E. S. Di Martino, G. Guadagni, A. Fumero, G. Ballerini, R. Spirito, P. Biglioli, and A. Redaelli. Fluid-structure interaction within realistic three-dimensional models of the aneurysmatic aorta as a guidance to assess the risk of rupture of the aneurysm. *Medical Engineering & Physics,*, 23(9):647–655, 11 2001.

[155] H. G. Matthies and J. Steindorf. Partitioned strong coupling algorithms for fluid-structure interaction. *Computers & Structures*, 81(8-11):805 – 812, 2003. K.-J. Bathe 60th Anniversary Issue.

[156] 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.

[157] E. Mesquita and R. Pavanello. Numerical methods for the dynamics of unbounded domains. *Computational & Applied Mathematics*, 24:1–26, 2005.

[158] 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.

[159] S. Mönkölä. Time-harmonic solution for linear elastic wave equation with controllability and higher-order discretizations. In *Proceedings of the 8th International Conference on Mathematical and Numerical Aspects of Waves, WAVES 2007*, pages 342–344, 23-27 July 2007.

[160] S. Mönkölä. Time-harmonic solution for acousto-elastic interaction with controllability and spectral elements. *Journal of Computational and Applied Mathematics*, 234(6):1904–1911, 2010.

[161] S. Mönkölä, E. Heikkola, A. Pennanen, and T. Rossi. Controllability method for the Navier equation with higher order discretizations. In J. Haataja, R. Stenberg, J. Périaux, P. Råback, and P. Neittaanmäki, editors, *Numerical analysis and scientific computing for PDEs and their challenging applications*, Barcelona, Spain, 2007. CIMNE.

[162] S. Mönkölä, E. Heikkola, A. Pennanen, and T. Rossi. Time-harmonic elasticity with controllability and higher order discretization methods. *Journal of Computational Physics*, 227(11):5513–5534, 2008.

[163] S. Mönkölä, E. Heikkola, and T. Rossi. Controllability method for time-harmonic acousto-elastic interaction. In *5th European Congress on Computational Methods in Applied Sciences and Engineering*, 2008.

[164] 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.

[165] R. Ohayon and C. Soize. *Structural acoustics and vibration: Mechanical models, variational formulations and discretization*. Academic Press, San Diego, 1999.

[166] L. G. Olson and K.-J. Bathe. Analysis of fluid-structure interactions. A direct symmetric coupled formulation based on the fluid velocity potential. *Computers & Structures*, 21:21–32, 1985.

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

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

[169] D. Pauly and T. Rossi. Theoretical considerations on the computation of generalized time-periodic waves. *Advances in Mathematical Sciences and Applications*, 21(1), 2011.

[170] 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.

[171] L. F. Pavarino and E. Zampieri. Overlapping Schwarz and spectral element methods for linear elasticity and elastic waves. *Journal of Scientific Computing*, 27(1-3):51–73, 2006.

[172] R. E. Plessix. A review of the adjoint-state method for computing the gradient of a functional with geophysical applications. *Geophysical Journal International*, 167(2):495–503, 2006.

[173] 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.

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

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

[176] S. W. Rienstra and A. Hirschberg. An introduction to acoustics, 2010.

[177] M. Ross, M. A. Sprague, C. A. Felippa, and K. C. Park. Treatment of acoustic fluid-structure interaction by localized lagrange multipliers and comparison to alternative interface-coupling methods. *Computer Methods in Applied Mechanics and Engineering*, 198:986–1005, 2009.

[178] 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.

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

[180] S. H. Schot. Eighty years of Sommerfeld's radiation condition. *Historia mathematica*, 19:385–401, 1992.

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

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

[183] 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.

[184] 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.

[185] 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.

[186] L. L. Thompson and 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*, 132:195–227, 1996.

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

[188] A. Valencia and F. Solis. Blood flow dynamics and arterial wall interaction in a saccular aneurysm model of the basilar artery. *Computers & Structures,*, 84(21):1326–1337, 2006.

[189] T. Varslot and S.-E. Måsøy. Forward propagation of acoustic pressure pulses in 3D soft biological tissue. *Modeling, Identification and Control*, 27(3):181–190, 2006.

[190] X. Wang and 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):2001–2017, 1997.

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

[192] 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.

[193] 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.

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

[195] 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.

[196] B. I. Wohlmuth. *Discretization methods and iterative solvers based on domain decomposition*. Springer-Verlag, Berlin, 2001.

[197] E. Zampieri and L. F. Pavarino. An explicit second order spectral element method for acoustic waves. *Advances in Computational Mathematics*, 25(4):381–401, 2006.

[198] X. Zeng and J. Bielak. Stable symmetric finite element boundary integral coupling methods for fluid structure interface problems. *Engineering Analysis with Boundary Elements*, 15:79–91, 1995.

[199] 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.

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

YHTEENVETO (FINNISH SUMMARY)

Tämä väitöskirja, suomenkieliseltä nimeltään *Akustisten ja elastisten aaltojen välisen neste–rakenne-vuorovaikutuksen numeerinen simulointi*, käsittelee tehokkaiden ratkaisumenetelmien kehittämistä nesteen ja kiinteän aineen välisen vuorovaikutuksen tietokonesimulointiin. Tutkimus keskittyy akustisten ja elastisten aaltojen etenemistä kuvaavien Helmholtzin yhtälön ja Navier–Cauchy-mallin muodostamaan kytkettyyn systeemiin, jota voidaan hyödyntää monimutkaisten käytännön sovellusten tietokonesimuloinnissa. Näiden akustisia painekenttiä ja elastisia materiaaleja kuvaavien osittaisdifferentiaaliyhtälöiden avulla voidaan mallintaa ja simuloida esimerkiksi vedenalaisten rakenteiden kestävyyttä, maanjäristysaaltojen etenemistä sekä työkoneiden ohjaamojen tai konserttisalien akustisia ominaisuuksia. Lääketieteessä ultraäänitutkimusten antamaa tietoa käyttäen voidaan laskennallisesti mallintaa kudosten rakennetta. Ultraääntä sovelletaan myös sikiötutkimuksessa, lihas- ja kudosvaurioiden hoitamisessa, kiputilojen lievittämisessä sekä leikkaus- ja syöpähoidoissa. Kaikuluotauksessa ultraäänen avulla mitataan etäisyyksiä sekä tutkitaan vesistön pohjaa ja kalaparvia.

Aikaharmonisten aalto-ongelmien ratkaisemisessa käytetään yleensä kompleksiarvoisia aikaharmonisia malleja. Mallien käsitteleminen tietokoneella vaatii diskretointia, laskenta-alueen jakamista laskentapisteisiin, joka on perinteisesti toteutettu äärellisten elementtien menetelmällä. Tämä johtaa suuriin indefiniitteihin yhtälöryhmiin, joille on hankala kehittää tehokkaita 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 diskretointia, mikä puolestaan vaatii runsaasti laskentaresursseja.

Tässä työssä aikaharmonisten aalto-ongelmien ratkaisemiseen sovelletaan säätöteoriaan perustuvaa tekniikkaa sekä modernimpaa, korkeamman asteen, elementtimenetelmää. Tehtävän ratkaisemisessa ei käytetä aikaharmonisia aaltoyhtälöitä, vaan aikaharmoninen tehtävä esitetään tarkan säädettävyyden tehtävänä ajasta riippuvan mallin avulla. Tämän jälkeen tehtävä muotoillaan pienimmän neliön optimointiongelmaksi, joka ratkaistaan liittogradienttimenetelmällä. Gradientin laskenta, joka on menetelmän keskeinen osa, suoritetaan liittotilatekniikalla. Menetelmän tehokkuuden lisäämiseksi liittogradienttialgoritmi pohjustetaan lohkodiagonaalisella matriisilla, ja pohjustuksen yhteydessä muodostuvat jäykkyysmatriisin sisältävät yhtälöt ratkaistaan graafipohjaisella monihilamenetelmällä.

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

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

APPENDIX 1 TIME-HARMONIC COUPLED PROBLEM

With pressure-displacement formulation the coupled problem is

$$-\frac{\kappa(\mathbf{x})^2}{\rho_f(\mathbf{x})}P_f - \nabla\cdot\left(\frac{1}{\rho_f(\mathbf{x})}\nabla P_f\right) = F, \qquad \text{in } \Omega_f, \tag{221}$$

$$P_f = 0, \qquad \text{on } \Gamma_{0f}, \tag{222}$$

$$i\kappa(\mathbf{x})P_f + \frac{\partial P_f}{\partial \mathbf{n}_f} = Y_{\text{ext}}, \qquad \text{on } \Gamma_{ef}, \tag{223}$$

$$\rho_f(\mathbf{x})\omega^2\mathbf{U_s}\cdot\mathbf{n}_s + \frac{\partial P_f}{\partial \mathbf{n}_f} = 0, \qquad \text{on } \Gamma_i, \tag{224}$$

$$-\omega^2\rho_s(\mathbf{x})\mathbf{U_s} - \nabla\cdot\sigma(\mathbf{U_s}) = \mathbf{F}, \qquad \text{in } \Omega_s, \tag{225}$$

$$\mathbf{U_s} = \mathbf{0}, \qquad \text{on } \Gamma_{0s}, \tag{226}$$

$$i\omega\rho_s(\mathbf{x})\mathbf{B}_s\mathbf{U_s} + \sigma(\mathbf{U_s})\mathbf{n}_s = \mathbf{G}_{\text{ext}}, \qquad \text{on } \Gamma_{es}, \tag{227}$$

$$\sigma(\mathbf{U_s})\mathbf{n}_s - P_f\mathbf{n}_f = \mathbf{0}, \qquad \text{on } \Gamma_i. \tag{228}$$

If we use the velocity potential Φ instead of pressure in the fluid domain, the coupled form is

$$-\kappa(\mathbf{x})^2\Phi - \nabla^2\Phi = F_\Phi, \qquad \text{in } \Omega_f, \tag{229}$$

$$\Phi = 0, \qquad \text{on } \Gamma_{0f}, \tag{230}$$

$$i\kappa(\mathbf{x})\Phi + \frac{\partial \Phi}{\partial \mathbf{n}_f} = Y_{\Phi\text{ext}}, \qquad \text{on } \Gamma_{ef}, \tag{231}$$

$$i\omega\mathbf{U_s}\cdot\mathbf{n}_s + \frac{\partial \Phi}{\partial \mathbf{n}_f} = 0, \qquad \text{on } \Gamma_i, \tag{232}$$

$$-\omega^2\rho_s(\mathbf{x})\mathbf{U_s} - \nabla\cdot\sigma(\mathbf{U_s}) = \mathbf{F}, \qquad \text{in } \Omega_s, \tag{233}$$

$$\mathbf{U_s} = \mathbf{0}, \qquad \text{on } \Gamma_{0s}, \tag{234}$$

$$i\omega\rho_s(\mathbf{x})\mathbf{B}_s\mathbf{U_s} + \sigma(\mathbf{U_s})\mathbf{n}_s = \mathbf{G}_{\text{ext}}, \qquad \text{on } \Gamma_{es}, \tag{235}$$

$$\sigma(\mathbf{U_s})\mathbf{n}_s + i\omega\rho_f(\mathbf{x})\Phi\mathbf{n}_f = \mathbf{0}, \qquad \text{on } \Gamma_i. \tag{236}$$

When also the fluid domain is modeled by the displacements, we get the equations

$$-\omega^2\rho_f(\mathbf{x})\mathbf{U_f} - c^2\rho_f(\mathbf{x})\nabla\cdot(\nabla\cdot\mathbf{U_f}) = \mathbf{F}_f, \qquad \text{in } \Omega_f, \tag{237}$$

$$\mathbf{U_f} = \mathbf{0}, \qquad \text{on } \Gamma_{0f}, \tag{238}$$

$$i\omega\rho_f(\mathbf{x})\mathbf{B}_f\mathbf{U_f} + c^2\rho_f(\mathbf{x})(\nabla\cdot\mathbf{U_f})\mathbf{n}_f = \mathbf{Y}_{f\text{ext}}, \qquad \text{on } \Gamma_{ef}, \tag{239}$$

$$\mathbf{U_s}\cdot\mathbf{n}_s + \mathbf{U_f}\cdot\mathbf{n}_f = 0, \qquad \text{on } \Gamma_i, \tag{240}$$

$$-\omega^2\rho_s(\mathbf{x})\mathbf{U_s} - \nabla\cdot\sigma(\mathbf{U_s}) = \mathbf{F}, \qquad \text{in } \Omega_s, \tag{241}$$

$$\mathbf{U_s} = \mathbf{0}, \qquad \text{on } \Gamma_{0s}, \tag{242}$$

$$i\omega\rho_s(\mathbf{x})\mathbf{B}_s\mathbf{U_s} + \sigma(\mathbf{U_s})\mathbf{n}_s = \mathbf{G}_{\text{ext}}, \qquad \text{on } \Gamma_{es}, \tag{243}$$

$$\sigma(\mathbf{U_s})\mathbf{n}_s + c^2\rho_f(\mathbf{x})(\nabla\cdot\mathbf{U_f})\mathbf{n}_f = \mathbf{0}, \qquad \text{on } \Gamma_i. \tag{244}$$

APPENDIX 2 GLOSSARY OF ACRONYMS

AB	Adams–Bashforth (discretization)
AMG	Algebraic multigrid
BDDC	Balancing domain decomposition by constraints
BWD	Backward (adjoint state)
Bi-CGSTAB	Biconjugate gradient stabilized (method)
CD	Central finite difference (discretization)
CFL	Courant–Friedrichs–Lewy (stability condition)
CG	Conjugate gradient
CPU	Central processing unit
DDM	Domain decomposition method
DOF	(Number of) degrees of freedom
FDM	Finite difference method
FEM	Finite element method
FETI	Finite element tearing and interconnecting
FSI	Fluid-structure interaction
FWD	Forward (state)
GBMG	Graph-based multigrid
GL	Gauss–Lobatto quadrature
GMG	Geometric multigrid
GMRES	Generalized minimal residual (method)
HUM	Hilbert uniqueness method
IFEM	Infinite element method
LAPACK	Linear algebra package
ODE	Ordinary differential equation
PDE	Partial differential equation
PML	Perfectly matched layer
PUM	Partition of unity method
RK	Runge–Kutta (discretization)
SEM	Spectral element method
SOR	Successive over-relaxation
UWVF	Ultra weak variational formulation

APPENDIX 3 GLOSSARY OF SYMBOLS

Absorbing boundary in fluid	Γ_{ef}
Absorbing boundary in solid	Γ_{es}
Accuracy constant for element order r	k_r
Acoustic admittance	β
Acoustic sound intensity	I
Adjoint state variable (including $\mathbf{p}$, in some cases also $\frac{\partial \mathbf{p}}{\partial t}$, at each timestep)	$\hat{\mathbf{z}}$
Affine element mapping to the ith reference element	$\mathcal{G}_i$
Adiabatic compressibility	$\varkappa$
Angle	α
Angular frequency	ω
Bessel functions	J_n
Body force in fluid (time-dependent)	f
Body force in fluid (time-harmonic)	F
Body force in solid (time-dependent)	$\mathbf{f}$
Body force in solid (time-harmonic)	$\mathbf{F}$
Boundary condition operator	$\mathcal{W}$
Boundary of Ω	$\partial\Omega$
Bounded obstacle	Θ
Coarsest multigrid level	$\tilde{k}$
Condition number	κ
Control space	Z
Control variable	$\mathbf{e}_0$
Control variable	$\mathbf{e}_1$
Coordinate system in the reference element	(ξ, ζ)
Cost function (objective functional, least-squares functional)	J
Coupling interface (time-dependent)	$\gamma_{\mathrm{i}} = \Gamma_{\mathrm{i}} \times [0, T]$
Coupling interface (time-harmonic)	Γ_{i}
Coupling matrix from fluid to solid	$\mathcal{A}_{\mathrm{sf}} = ((\mathcal{A}_{\mathrm{sf}})_1, (\mathcal{A}_{\mathrm{sf}})_2)^T$
Coupling matrix from solid to fluid	$\mathcal{A}_{\mathrm{fs}} = ((\mathcal{A}_{\mathrm{fs}})_1, (\mathcal{A}_{\mathrm{fs}})_2)^T$
Damping matrix in fluid	$\mathcal{S}_f$
Damping matrix in solid	$\mathcal{S}_s$
Damping matrix in fluid and solid	$\mathcal{S}$
Density	ρ
Density at the steady-state	ρ_0
Density at the steady-state in the fluid	ρ_{f0}
Density at the steady-state in the solid	ρ_{s0}
Density difference $\rho - \rho_0$	$\hat{\rho}$
Density in fluid	ρ_f
Density in solid	ρ_s
Derivative of the rth degree Legendre polynomial	L_r'
Displacement in fluid (time-dependent)	$\mathbf{u_f} = (\mathbf{u_{f1}}, \mathbf{u_{f2}})^T$
Displacement in solid (time-dependent)	$\mathbf{u_s} = (\mathbf{u_{s1}}, \mathbf{u_{s2}})^T$

Displacement in fluid (time-dependent)	$\mathbf{U}_f$
Displacement in solid (time-harmonic)	$\mathbf{U}_\mathbf{s} = (\mathbf{U}_{\mathbf{s}1}, \mathbf{U}_{\mathbf{s}2})^T$
Displacement in solid, incident (time-harmonic)	$\mathbf{U}_{\mathbf{s}\mathrm{inc}}$
Domain	Ω
Domain of the fluid media (time-dependent)	Q_f
Domain of the fluid media (time-harmonic)	Ω_f
Domain of the solid media (time-dependent)	Q_s
Domain of the solid media (time-harmonic)	Ω_s
Elastic moduli tensor	C
Element	Ω_i
Elementary vector	$\hat{e}$
Energy	E
Energy	$E_{\phi s f}$
Exterior domain	G
Finest multigrid level	$\tilde{0}$
Frequency	f
Function space, infinite, scalar-valued	V
Function space, infinite, vector-valued	$\mathbf{V}$
Function space, infinite, vector-valued	$\mathbf{V}_f$
Function space, finite, scalar-valued	V_h^r
Function space, finite, vector-valued	$\mathbf{V}_h^r$
Gradient	$\mathbf{g} = (g_0, g_1)^T$
Gradient update in optimization algorithm	$\mathbf{v} = (v_0, v_1)^T$
Gravity	$\mathbf{g}$
Hankel function	H_n
Identity matrix	$\mathcal{I}$
Imaginary part	Im
Imaginary unit	$i = \sqrt{-1}$
Initial condition	$\mathbf{e} = (\mathbf{e}_0, \mathbf{e}_1)^T$
Initial residual in optimization	$\mathbf{g}^0 = \nabla J(\mathbf{e}^0)$
Kronecker delta	δ_{ij}
Krylov subspace	K
Lamé parameter λ in fluid	λ_f
Lamé parameter λ in solid	λ_s
Lamé parameter μ in fluid	μ_f
Lamé parameter μ in solid	μ_s
Largest mesh stepsizes in fluid	$h_{f,max}$
Largest mesh stepsizes in solid	$h_{s,max}$
Legendre polynomial of degree r	L_r
Mass	m
Mass matrix in fluid	$\mathcal{M}_f$
Mass matrix in solid	$\mathcal{M}_s$
Mass matrix in fluid and solid	$\mathcal{M}$
Matrix coefficient in quadratic functional (non-self-adjoint)	$\mathsf{\Lambda}$
Matrix coefficient in quadratic functional (self-adjoint)	A

Mesh stepsize	h
Number of degrees of freedom, total	$\hat{N} = 2N_s + N_f$
Number of degrees of freedom in fluid	$\hat{N}_f = N_f$
Number of degrees of freedom in solid	$\hat{N}_s = 2N_s$
Number of diagonal elements in the matrix $\mathcal{M}$	$\hat{n}$
Number of discretization points in fluid	N_f
Number of discretization points in solid	N_s
Number of elements	N_e
Number of timesteps	N
Objective functional	J
Order of the basis functions	r
Optimal control (solution of minimization)	$\mathbf{e}^*$
Optimal state	$\hat{\mathbf{y}}(\mathbf{e}^*)$
Outward unit normal vector	$\mathbf{n} = (n_1, n_2)^T$
Outward unit normal vector in fluid	$\mathbf{n}_f = (n_{f1}, n_{f2})^T$
Outward unit normal vector in solid	$\mathbf{n}_s = (n_{s1}, n_{s2})^T$
Pointwise values of the fluid and structure adjoint state variables	$\mathbf{p}$
Pointwise values of the fluid and structure state variables (time-dependent)	$\mathbf{u}$
Pointwise values of the fluid and structure state variables (time-harmonic)	$\mathbf{U}$
Poisson ratio	ν
Preconditioner	$\mathcal{L}$
Pressure (time-dependent)	p_f
Pressure, incident (time-dependent)	$p_{f_{\text{inc}}}$
Pressure, scattered (time-dependent)	$p_{f_{\text{scat}}}$
Pressure (time-harmonic)	P_f
Pressure, incident (time-harmonic)	$P_{f_{\text{inc}}}$
Pressure, scattered (time-harmonic)	$P_{f_{\text{scat}}}$
Prolongation operator from level $(l+1)$ to finer level l	$\widetilde{\mathbf{P}}^{l}_{l+1}$
Propagation direction	$\boldsymbol{\omega} = (\omega_1, \omega_2)$
Radius	r
Radius of circular fluid domain	a_f
Radius of circular solid domain	a_s
Real part	Re
Reference element	$\Omega_{\text{ref}} = [0,1]^2$
Restriction operator from level l to coarser level $(l+1)$	$\widetilde{\mathbf{R}}_{l}^{l+1}$
Right-hand side of state equation (time-dependent)	$\mathcal{F}$
Right-hand side of state equation (time-harmonic)	F
Right-hand side vector in fluid (time-dependent)	$\mathbf{f}_f$
Right-hand side vector in fluid (time-harmonic)	$\mathbf{F}_f$
Right-hand side vector in solid (time-dependent)	$\mathbf{f}_s = ((\mathbf{f}_s)_1, (\mathbf{f}_s)_2)^T$
Right-hand side vector in solid (time-harmonic)	$\mathbf{F}_s = ((\mathbf{F}_s)_1, (\mathbf{F}_s)_2)^T$
Scaling factor in optimization	γ

Search direction in optimization	$\mathbf{w}$
Set of Gauss–Lobatto points	$\xi_i, i = 1, \ldots, r+1$
Set of Gauss–Lobatto weights	$w_i, i = 1, \ldots, r+1$
Set of Lagrange interpolants	$\hat{\varphi}_j(\xi_i), j = 1, \ldots, r+1$
Set of polynomial basis functions of order r	Q^r
Spatial variable	$\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$
Smallest mesh stepsizes in fluid	$h_{f,min}$
Smallest mesh stepsizes in solid	$h_{s,min}$
Speed of pressure waves (P-waves) in solid	c_p
Speed of shear waves (S-waves) in solid	c_s
Speed of sound in fluid	c
Stability constant for element order r	α_r
State equation	s
State equation with zero right-hand side	s_0
Step length to optimization direction	η
Stiffness matrix in fluid	$\mathcal{K}_f$
Stiffness matrix in solid	$\mathcal{K}_s$
Stiffness matrix in fluid and solid	$\mathcal{K}$
Stopping criterion in optimization algorithm	ε
Strain tensor	ϵ
Stress tensor	σ
Source function on boundary γ_{0f} (time-dependent)	y_0
Source function on boundary Γ_{0f} (time-harmonic)	Y_0
Source function on boundary γ_{ef} (time-dependent)	y_{ext}
Source function on boundary Γ_{ef} (time-harmonic)	Y_{ext}
Source function on boundary γ_{0s} (time-dependent)	$\mathbf{g}_0$
Source function on boundary Γ_{0s} (time-harmonic)	$\mathbf{G}_0$
Source function on boundary γ_{es} (time-dependent)	$\mathbf{g}_{\text{ext}}$
Source function on boundary Γ_{es} (time-harmonic)	$\mathbf{G}_{\text{ext}}$
State variable (including $\mathbf{u}$, in some cases also $\frac{\partial \mathbf{u}}{\partial t}$, at each timestep)	$\hat{\mathbf{y}}$
Thermodynamical internal energy	u
Test function in space V	v
Test function in space $\mathbf{V}$	$\mathbf{v}$
Test function in space $\mathbf{V}_f$	$\mathbf{v}_f$
Time	t
Time derivative of the pointwise values of the fluid and structure variables	$v = \frac{\partial \mathbf{u}}{\partial t}$
Timestep length	Δt
Total time	T
Transition function	θ
Transition time interval length	T_{tr}
Vector coefficient in quadratic functional	b
Vector coefficient in quadratic functional	c
Velocity potential (time-dependent)	ϕ

Velocity potential, incident (time-dependent)	ϕ_{inc}
Velocity potential, scattered (time-dependent)	ϕ_{scat}
Velocity potential (time-harmonic)	Φ
Velocity potential, incident (time-harmonic)	Φ_{inc}
Velocity potential, scattered (time-harmonic)	Φ_{scat}
Velocity	$\mathbf{v}$
Velocity at the steady-state	$\mathbf{v}_0$
Velocity at the steady-state in the fluid	$\mathbf{v}_{f0}$
Velocity at the steady-state in the solid	$\mathbf{v}_{s0}$
Velocity difference $\mathbf{v} - \mathbf{v}_0$	$\hat{\mathbf{v}}$
Velocity in fluid	$\mathbf{v}_f$
Velocity in solid	$\mathbf{v}_s$
Vertical distance from zero level	z
Wavelength	ℓ
Wavenumber in fluid	κ
Wavenumber of pressure waves (P-waves) in solid	κ_p
Wavenumber of shear waves (S-waves) in solid	κ_s
Young modulus	E

APPENDIX 4 GLOSSARY OF NOTATIONS

Derivatives	$f'(t,y(t))$	$= \frac{df(t,y(t))}{dt}$
Partial derivatives	$f_{xy}(x,y,z)$ $f^{(i,j)}(x,y)$	$= \frac{\partial}{\partial y}(\frac{\partial f(x,y,z)}{\partial x})$ $= \frac{\partial^j}{\partial y^j}(\frac{\partial^i f(x,y,z)}{\partial x^i})$
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

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

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.

The space of square-integrable functions,

$$L^2(\Omega) = \{v : \Omega \to \mathbb{R}^n : \int\limits_\Omega |v|^2 < \infty\}.$$

The space of functions and derivatives in $L^2(\Omega)$,

$$H^1(\Omega) = \{v \in L^2(\Omega), \nabla v \in L^2(\Omega)\}.$$

The space of functions in $H^1(\Omega)$ vanishing on Γ, which is a part of the boundary of Ω,

$$H^1_\Gamma(\Omega) = \{v \in H^1(\Omega), v|_\Gamma = 0\}.$$

The subspace of zero-mean functions in $H^1(\Omega)$,

$$\overset{\circ}{H}{}^1(\Omega) = \{v \in H^1(\Omega), \int\limits_\Omega v = 0\}.$$

The space of vector fields and divergences in $L^2(\Omega)$,

$$H(div, \Omega) = \{v \in L^2(\Omega), \nabla \cdot v \in L^2(\Omega)\}.$$