Journal of Computational Physics 517 (2024) 113347

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Acoustic shape optimization using energy stable curvilinear finite differences

Gustav Eriksson [*], Vidar Stiernström [1]

Department of Information Technology, Uppsala University, PO Box 337, S-751 05 Uppsala, Sweden

ARTICLE INFO

Keywords:
Shape optimization
Acoustic wave equation
Finite differences
Summation-by-parts
Adjoint method

ABSTRACT

A gradient-based method for shape optimization problems constrained by the acoustic wave equation is presented. The method makes use of high-order accurate finite differences with summation-by-parts properties on multiblock curvilinear grids to discretize in space. Representing the design domain through a coordinate mapping from a reference domain, the design shape is obtained by inverting for the discretized coordinate map. The adjoint state framework is employed to efficiently compute the gradient of the loss functional. Using the summation-by-parts properties of the finite difference discretization, we prove stability and dual consistency for the semi-discrete forward and adjoint problems. Numerical experiments verify the accuracy of the finite difference scheme and demonstrate the capabilities of the shape optimization method on two model problems with real-world relevance.

1. Introduction

Partial differential equation (PDE) constrained shape optimization is a highly important topic for applications in research and engineering. Together with topology optimization, it constitutes a central tool for computer-aided optimal design problems. To obtain an efficient algorithm for PDE-constrained optimization problems, gradient-based methods are usually employed. In the context of PDE-constrained optimization, the adjoint framework has proven to be a very efficient approach for computing gradients of the loss functional, especially if the number of design variables is large [1,2]. Usually, one has to choose whether to apply the adjoint method on the continuous or discretized equations. In the first approach, *optimize-then-discretize* (OD), the continuous gradient and the forward and adjoint equations are discretized with methods of choice, potentially unrelated to one another. OD is typically easier to analyze and implement since it does not have to consider the particularities of the discretization methods. However, the computed gradient will in general not be the exact gradient of the discrete loss functional, which can lead to convergence issues in the optimization [3]. The alternative approach, *discretize-then-optimize* (DO), computes the exact gradient (up to round-off) with respect to the discrete forward and adjoint problems. However, depending on how the forward problem is discretized, the DO approach can lead to stability issues in the discrete adjoint problem [4]. In this work, we discretize space using a finite difference scheme that is *dual-consistent*. Disregarding the discretization of time, this means that the scheme from the DO approach is a stable and consistent approximation of the scheme from the OD approach, such that the two approaches are equivalent. Dual-consistent discretizations and superconvergent

* Corresponding author.
E-mail addresses: gustav.eriksson@it.uu.se (G. Eriksson), cstierns@stanford.edu (V. Stiernström).
[1] Now at the Department of Geophysics, Stanford University, Stanford, USA.

https://doi.org/10.1016/j.jcp.2024.113347
Received 29 September 2023; Received in revised form 8 May 2024; Accepted 8 August 2024

Available online 12 August 2024

approximate functionals from PDEs are well-known concepts in many disciplines where integral quantities are of interest [5–7]. See also [8,9] for recent examples from the finite difference community.

Given that the computation of directional derivatives with respect to the design geometry is a key part of gradient-based shape optimization, the question of how the geometry is represented as a variable is important. Various approaches exist in the literature. Perhaps the most obvious and standard approach is based on domain transformations, where diffeomorphisms are used to map coordinates between the physical domain subject to optimization and a fixed reference domain [10]. The coordinate mapping then provides a straightforward way to express integrals and derivatives in the physical domain in terms of integrals and derivatives in a fixed reference domain, greatly simplifying the computation of the gradient. Although the simplicity of this approach is highly attractive, it can lead to robustness issues, such that re-meshing is required to obtain an accurate solution [11]. An alternative approach is to discretize the problem using a so-called fictitious-domain method, where the domain of interest is embedded in a larger domain of simpler shape [12]. In [13] a fictitious-domain method is used together with level-set functions and CutFEM to solve a shape optimization problem constrained by the Helmholtz equation. In the present study, we discretize the PDE using finite differences defined on boundary-conforming structured grids. For this reason, we take the first approach and represent the geometry through a coordinate mapping between the physical domain and a rectangular reference domain. As demonstrated by the numerical results in Section 5, the issue of poor mesh quality and re-meshing can be avoided by the use of regularization.

In the present study, we consider shape optimization problems constrained by the acoustic wave equation. Due to the hyperbolic nature of this PDE, high-order finite difference methods are well-suited discretization methods in terms of accuracy and efficiency [14,15]. One way to obtain robust and provably stable high-order finite difference methods is to use difference stencils with summation-by-parts (SBP) properties [16,17]. SBP finite differences have been used in multiple studies of the second-order wave equation in the past [18,19]. With careful treatment of boundary conditions, these methods allow for semi-discrete stability proofs and energy estimates that are equivalent to the continuous energy balance equations. Methods for imposing boundary conditions include, e.g., the simultaneous-approximation-term method (SBP-SAT) [20,21], the projection method (SBP-P) [22,23], the ghost-point method (SBP-GP) [24], and a combination of SAT and the projection method (SBP-P-SAT), recently presented in [23]. To the best of our knowledge, SBP finite differences employed to solve acoustic shape optimization problems have not yet been presented in the literature.

Although the shape optimization methodology developed in the present study focuses on acoustic waves it is also applicable to other types of wave equations. In addition, from a mathematical point of view, the method closely resembles the work in [25], where an SBP-SAT method is used for seismic full waveform inversion. The difference essentially lies in what the coefficients in the discretized equations represent. Here they are metric coefficients derived from the coordinate transformation between the physical and reference domains, while in [25] they are unknown material properties. Indeed, the metric coefficients may be viewed as the material properties in an anisotropic wave equation on a fixed domain. Other studies where SBP difference methods have been employed for adjoint-based optimization include aerodynamic shape optimization [26], optimization of turbulent flows [27] and optimization of gas networks [28].

The paper is developed as follows: In Section 2 the model problem is presented. Then, in Section 3, we analyze the forward problem and present the SBP finite difference discretization. In Section 4 the optimization problem is considered, including the derivation of the semi-discrete adjoint problem and corresponding gradient expression. We evaluate the performance of the method using three numerical experiments in Section 5. Finally, the study is concluded in Section 6.

2. Problem setup

The general type of shape optimization problems considered in this paper are of the form

$$\min_p \mathrm{J}(u,p), \quad \text{such that} \tag{1a}$$

$$\begin{aligned} u_{tt} &= c^2\Delta u + F(\mathbf{x},t), && \mathbf{x}\in\Omega_p, \quad t\in[0,T], \\ Lu &= g(\mathbf{x},t), && \mathbf{x}\in\partial\Omega_p, \ t\in[0,T], \\ u &= 0, \quad u_t = 0, && \mathbf{x}\in\Omega_p, \quad t=0, \end{aligned} \tag{1b}$$

where $\mathrm{J}(u,p)$ is the loss functional, $F(\mathbf{x},t)$ is a forcing function, c is the wave speed, and the linear operator L together with boundary data $g(\mathbf{x},t)$ defines the boundary conditions. Essentially, the problem consists of finding the control variable p determining the shape of the domain $\Omega_p\in\mathbb{R}^2$ such that $\mathrm{J}(u,p)$ is minimized while u satisfies the acoustic wave equation (1b). We use the subscript p to indicate the domain's dependency on the control parameter.

To make the analysis easier to follow, consider the model problem given by

$$\min_p \mathrm{J}(u,p) = \int_0^T (u^+(\mathbf{x}_r,t) - u_d(t))^2\, dt, \quad \text{such that} \tag{2a}$$

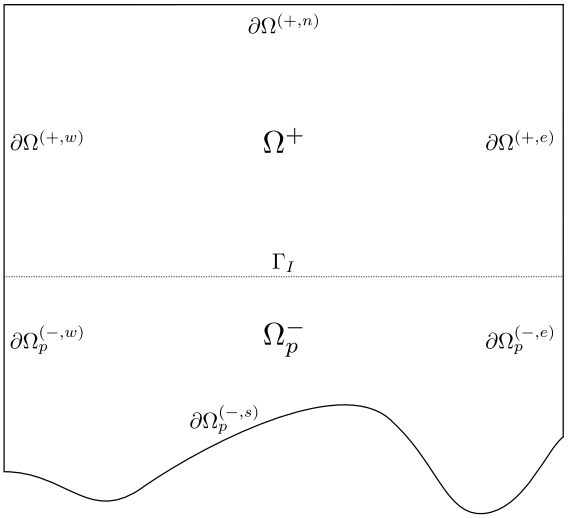


Fig. 1. Boundary conditions of the bathymetry problem. The dotted line indicate the interface between Ω^+ and Ω_p^-.

$$
\begin{array}{lll}
u_{tt}^+ = c^2\Delta u^+ + f(t)\hat{\delta}(\mathbf{x}-\mathbf{x_s}), & \mathbf{x}\in\Omega^+, & t\in[0,T],\\
u_{tt}^- = c^2\Delta u^-, & \mathbf{x}\in\Omega_p^-, & t\in[0,T],\\
u_t^\pm + c\mathbf{n}^\pm\cdot\nabla u^\pm = 0, & \mathbf{x}\in\partial\Omega_p^{(\pm,w,e)}, & t\in[0,T],\\
\mathbf{n}^-\cdot\nabla u^- = 0, & \mathbf{x}\in\partial\Omega_p^{(-,s)}, & t\in[0,T],\\
u^+ = 0, & \mathbf{x}\in\partial\Omega_p^{(+,n)}, & t\in[0,T],\\
u^+ - u^- = 0, & \mathbf{x}\in\Gamma_I, & t\in[0,T],\\
\mathbf{n}^+\cdot\nabla u^+ + \mathbf{n}^-\cdot\nabla u^- = 0, & \mathbf{x}\in\Gamma_I, & t\in[0,T],\\
u^\pm = 0, \quad u_t^\pm = 0, & \mathbf{x}\in\Omega_p, & t=0,
\end{array}
\tag{2b}
$$

with domain Ω_p as depicted in Fig. 1. We assume that the top half of the domain is known and therefore split it into two blocks $\Omega_p = \Omega^+ \cup \Omega_p^-$, where only Ω_p^- is subject to the optimization. Superscripts $+$ and $-$ are used to indicate variables defined on the respective domains. This model problem can be thought of as determining the bathymetry in a segment of a lake or sea, given a time series of recorded pressure u_d at a receiver located in $\mathbf{x_r}$ originating from an acoustic point source at $\mathbf{x_s}$, where $\hat{\delta}$ denotes the Dirac delta function. For simplicity, we also assume that we have one receiver and one source and that they are located in Ω^+. Extensions to multiple sources and receivers in Ω^+ follow straightforwardly. At the top boundary, a homogeneous Dirichlet boundary condition is prescribed, corresponding to a constant surface pressure, and at the curved bottom boundary we prescribe a fully reflecting homogeneous Neumann boundary condition. At the sides, first-order outflow boundary conditions [29] are prescribed. For well-posedness, we further require continuity of the solutions $u^\pm$ and the flux $\mathbf{n}^\pm\cdot\nabla u^\pm$ across the interface $\Gamma_I = \Omega^+\cap\Omega_p^-$, where $\mathbf{n}^\pm$ are the outwards-pointing normals of the respective domain.

3. The forward problem

Initially, consider only the forward problem (2b) and its spatial discretization. In Section 4 we will return to the optimization problem. The acoustic wave equation on curvilinear or multiblock domains discretized using SBP finite differences has been treated on multiple occasions in the past, see e.g., [30,24,31,21,32,20,23]. In this work, we formulate discretizations of the Laplace operator directly on the physical domains Ω^+, and Ω_p^-. As previously mentioned, on the non-rectangular domain Ω_p^- this is achieved through a coordinate map to a square reference domain. By incorporating the metric transformation, a process referred to as 'encapsulation' in [33], the discrete Laplace operator satisfies SBP properties on Ω_p^-. Although this analysis is not new, it is reiterated here without proofs, since it is needed later in the context of optimizing for p. Using the discrete SBP Laplace operators we then enforce boundary and interface conditions through the SBP-P-SAT method.

3.1. Continuous analysis

We begin by considering the well-posedness of (2b). For simplicity, the wave speed c is assumed to be constant in Ω_p, but the extension to spatially variable $c(\mathbf{x})$ is straightforward. Let

$$
(u,v)_\Omega \equiv \int_\Omega uv\,d\mathbf{x}, \quad \|u\|_\Omega^2 \equiv (u,u)_\Omega, \quad \text{and} \quad \langle u,v\rangle_{\partial\Omega} \equiv \int_{\partial\Omega} uv\,ds. \tag{3}
$$

We begin by proving stability of (2b), where it is sufficient to only consider the homogeneous problem, i.e. $f(t)$ is set to zero. Multiplying the first equation in (2b) by u_t^+, the second equation by u_t^-, integrating over the domains, adding the results, and using Green's first identity leads to the energy equation

$$\frac{dE}{dt} = 2c^2\langle u_t^+, \mathbf{n}^+ \cdot \nabla u^+ \rangle_{\partial\Omega^+} + 2c^2\langle u_t^-, \mathbf{n}^- \cdot \nabla u^- \rangle_{\partial\Omega_p^-}, \tag{4}$$

where E is an energy given by

$$E = \|u_t^+\|^2_{\Omega^+} + c^2\|\nabla u^+\|^2_{\Omega_p^+} + \|u_t^-\|^2_{\Omega_p^-} + c^2\|\nabla u^-\|^2_{\Omega_p^-} \geq 0. \tag{5}$$

Inserting the boundary and interface conditions leads to

$$\frac{dE}{dt} = -2c \sum_{k=w,e} \langle u_t^+, u_t^+ \rangle_{\partial\Omega^{(+,k)}} + \langle u_t^-, u_t^- \rangle_{\partial\Omega_p^{(-,k)}} \leq 0, \tag{6}$$

which proves stability. From (6) it is clear that energy is dissipated through the east and west boundaries, due to the outflow boundary conditions.

The finite difference operators considered in the present study are defined on Cartesian grids. Therefore, for a general domain Ω_p^-, we introduce a coordinate mapping from a reference domain $\tilde{\Omega}^- \in [0,1]^2$ to Ω_p^-. Note that Ω^+ is already rectangular, hence no coordinate transformation is needed in this part of the domain. Let

$$x = x(\xi, \eta; p) \quad \text{and} \quad y = y(\xi, \eta; p), \tag{7}$$

define a smooth one-to-one coordinate mapping (diffeomorphism) from the reference domain $\xi, \eta \in \tilde{\Omega}^-$ to the physical domain $x, y \in \Omega_p^-$. Note that the functions x and y are parametrized by p. There are many ways one can derive the mappings (7). Here we use linear transfinite interpolation [34], which explicitly defines $x(\xi, \eta; p)$ and $y(\xi, \eta; p)$ given the coordinates of the boundary $\partial\Omega_p^-$. Letting subscripts denote partial differentiation, the Jacobian determinant (or the area element) of the mapping is given by

$$J = x_\xi y_\eta - x_\eta y_\xi. \tag{8}$$

By the chain rule, derivatives with respect to x and y are given by

$$\begin{aligned} u_x^- &= J^{-1}(y_\eta u_\xi^- - y_\xi u_\eta^-), \\ u_y^- &= J^{-1}(-x_\eta u_\xi^- + x_\xi u_\eta^-). \end{aligned} \tag{9}$$

By extension, the Laplace operator in terms of derivatives in the reference domain is given by

$$\Delta u^- = J^{-1}((\alpha_1 u_\xi^-)_\xi + (\beta u_\xi^-)_\eta + (\beta u_\eta^-)_\xi + (\alpha_2 u_\eta^-)_\eta), \tag{10}$$

where the metric coefficients are given by

$$\begin{aligned} \alpha_1 &= J^{-1}(x_\eta^2 + y_\eta^2), \\ \beta &= -J^{-1}(x_\xi x_\eta + y_\xi y_\eta), \\ \alpha_2 &= J^{-1}(x_\xi^2 + y_\xi^2). \end{aligned} \tag{11}$$

Further, the transformed normal derivatives satisfy

$$\mathbf{n}^- \cdot \nabla u^- = \begin{cases} \frac{\alpha_1 u_\xi^- + \beta u_\eta^-}{W_1}, & \xi, \eta \in \partial\tilde{\Omega}^{(-,w,e)}, \\ \frac{\alpha_2 u_\eta^- + \beta u_\xi^-}{W_2}, & \xi, \eta \in \partial\tilde{\Omega}^{(-,s,n)}, \end{cases} \tag{12}$$

where W_1 and W_2 are boundary scaling factors given by

$$W_1 = \sqrt{x_\xi^2 + y_\xi^2} \quad \text{and} \quad W_2 = \sqrt{x_\eta^2 + y_\eta^2}. \tag{13}$$

The inner products on the physical and reference domains are linked through the following equation:

$$(u^-, v^-)_{\Omega_p^-} = (u^-, Jv^-)_{\tilde{\Omega}^-}. \tag{14}$$

Similarly, the relation between boundary inner products on the physical and reference domains is given by

$$\begin{aligned} \langle u^-, v^- \rangle_{\partial\Omega_p^{(-,w,e)}} &= \langle u^-, W_1 v^- \rangle_{\partial\tilde{\Omega}_p^{(-,w,e)}}, \\ \langle u^-, v^- \rangle_{\partial\Omega_p^{(-,s,n)}} &= \langle u^-, W_2 v^- \rangle_{\partial\tilde{\Omega}_p^{(-,s,n)}}. \end{aligned} \tag{15}$$

3.2. One-dimensional SBP operators

We begin by presenting SBP finite difference operators in one spatial dimension, to be used as building blocks in Sections 3.3 and 3.4. Let the interval $[x_l, x_r]$ be discretized by m equidistant grid points $\mathbf{x} = [x_0, x_1, ..., x_{m-1}]$, such that

$$x_i = x_l + ih, \quad i = 0, 1, ..., m-1, \tag{16}$$

where $h = \frac{x_r - x_l}{m-1}$ is the grid step size. SBP operators are associated with a norm matrix (or quadrature) H, defining an inner product and norm given by

$$(\mathbf{u}, \mathbf{v})_H = \mathbf{u}^\top H \mathbf{v} \quad \text{and} \quad \|\mathbf{u}\|_H^2 = (\mathbf{u}, \mathbf{u})_H, \quad \forall \mathbf{u}, \mathbf{v} \in \mathbb{R}^m. \tag{17}$$

In this work we only consider diagonal-norm SBP operators, i.e. operators where H is a diagonal matrix, for which the inner product in (17) is simply the Euclidean dot product weighted by the diagonal of H. Later in the paper we shall use the same notation (inner product with a diagonal matrix subscript) to denote other discrete inner products defined equivalently, e.g., two-dimensional inner products. We will also use boundary restriction vectors given by

$$e_l^\top = [1, 0, ..., 0] \quad \text{and} \quad e_r^\top = [0, ..., 0, 1], \tag{18}$$

and identity matrices I_m of dimension $m \times m$.

To discretize the spatial derivatives we will use the first derivative SBP operators and the second derivative variable coefficient compatible SBP operators of interior orders four and six, first derived in [35,19]. The SBP properties of these operators are given in the following definitions:

Definition 1. A difference operator $D_1 \approx \frac{\partial}{\partial x}$ is said to be a first derivative SBP operator if, for the norm H, the relation

$$H D_1 + D_1^\top H = -e_l e_l^\top + e_r e_r^\top, \tag{19}$$

holds.

Definition 2. Let D_1 be a first derivative SBP operator and $\mathbf{c}$ be the restriction of a function $c(x)$ on the grid $\mathbf{x}$. A difference operator $D_2^{(\mathbf{c})} \approx \frac{\partial}{\partial x} c(x) \frac{\partial}{\partial x}$, is said to be a compatible second derivative SBP operator if, for the norm H, the relation

$$H D_2^{(\mathbf{c})} = -M^{(\mathbf{c})} - e_l e_l^\top \mathbf{c} d_l^\top + e_r e_r^\top \mathbf{c} d_r^\top, \tag{20}$$

holds, where

$$M^{(\mathbf{c})} = D_1^\top H \bar{\mathbf{c}} D_1 + R^{(\mathbf{c})}, \tag{21}$$

with $R^{(\mathbf{c})}$ symmetric and semi-positive definite, $\bar{\mathbf{c}} = \mathtt{diag}(\mathbf{c})$, and

$$d_l^\top = e_l^\top \hat{D}_1 \quad \text{and} \quad d_r^\top = e_r^\top \hat{D}_1, \tag{22}$$

where the first and last rows of $\hat{D}_1$ approximates $\frac{\partial}{\partial x}$.

Note that the remainder term $R^{(\mathbf{c})}$ is small (zero to the order of accuracy) [19]. We also define

$$D_2 \equiv D_2^{(\mathbf{1})} \quad \text{and} \quad R \equiv R^{(\mathbf{1})}, \tag{23}$$

where $\mathbf{1}$ is a vector of length m with only ones.

3.3. Two-dimensional operators on Ω^+

Since Ω^+ is rectangular by construction, operators on Ω^+ are directly obtained through tensor products of the one-dimensional operators. Let m_x^+ and m_y^+ denote the number of grid points in the x- and y-directions, respectively, and let $\mathbf{v}^+ \in \mathbb{R}^{N^+}$, where $N^+ = m_x^+ m_y^+$, denote a column-major ordered solution vector. The two-dimensional SBP operators are constructed using tensor products as follows:

$$\begin{aligned}
D_x^+ &= (D_1 \otimes I_{m_y^+}), & D_y^+ &= (I_{m_x^+} \otimes D_1), \\
D_{xx}^+ &= (D_2 \otimes I_{m_y^+}), & D_{yy}^+ &= (I_{m_x^+} \otimes D_2), \\
H_x^+ &= (H \otimes I_{m_y^+}), & H_y^+ &= (I_{m_x^+} \otimes H), \\
e_w^+ &= (e_l^\top \otimes I_{m_y^+}), & e_e^+ &= (e_r^\top \otimes I_{m_y^+}), \\
e_s^+ &= (I_{m_x^+} \otimes e_l^\top), & e_n^+ &= (I_{m_x^+} \otimes e_r^\top),
\end{aligned}$$

$$d_w^+ = -(d_l^\top \otimes I_{m_y^+}), \qquad d_e^+ = (d_r^\top \otimes I_{m_y^+}),$$
$$d_s^+ = -(I_{m_x^+} \otimes d_l^\top), \qquad d_n^+ = (I_{m_x^+} \otimes d_r^\top). \tag{24}$$

For notational clarity, we have used the same symbols for 1D operators in the x- and y-directions in (24), although they are different in general (since they depend on the number of grid points and the grid spacing).

The discrete Laplace operator in Ω^+ is

$$D_L^+ = D_{xx}^+ + D_{yy}^+. \tag{25}$$

We also have the following norm matrices:

$$\begin{aligned} H^+ &= H_x^+ H_y^+, \\ H_{w,e,s,n}^+ &= H, \end{aligned} \tag{26}$$

with inner products $(\cdot,\cdot)_{H^+}$ and $(\cdot,\cdot)_{H_{w,e,s,n}^+}$ defined as in (17). Using Definitions 1 and 2, D_L^+ can be shown to satisfy

$$\begin{aligned} (\mathbf{u}, D_L^+ \mathbf{v})_{H^+} = &-(D_x^+ \mathbf{u}, D_x^+ \mathbf{v})_{H^+} - (D_y^+ \mathbf{u}, D_y^+ \mathbf{v})_{H^+} - (\mathbf{u}, \mathbf{v})_{R^+} \\ &+ \sum_{k=w,e,s,n} (e_k^+ \mathbf{u}, d_k^+ \mathbf{v})_{H_k^+}, \quad \forall \mathbf{u}, \mathbf{v} \in \mathbb{R}^{N^+}, \end{aligned} \tag{27}$$

where

$$(\mathbf{u}, \mathbf{v})_{R^+} = \mathbf{u}^\top (R_x^+ H_y^+ + R_y^+ H_x^+) \mathbf{v}, \tag{28}$$

with $R_x^+ = (R \otimes I_{m_y^+})$ and $R_y^+ = (I_{m_x^+} \otimes R)$. Note that $R_x^+ \approx 0$, $R_y^+ \approx 0$ (since $R \approx 0$), and

$$(\mathbf{u}, \mathbf{v})_{R^+} = (\mathbf{v}, \mathbf{u})_{R^+} \quad \text{and} \quad \|\mathbf{u}\|_{R^+}^2 := (\mathbf{u}, \mathbf{u})_{R^+} \geq 0, \quad \forall \mathbf{u}, \mathbf{v} \in \mathbb{R}^{N^+}. \tag{29}$$

3.4. Curvilinear operators on Ω_p^-

We begin by discretizing $\tilde{\Omega}^-$ using m_ξ^- and m_η^- equidistant grid points in the ξ- and η-directions, respectively. As in Ω^+, we let $\mathbf{v}^- \in \mathbb{R}^{N^-}$, where $N^- = m_\xi^- m_\eta^-$, denote a column-major ordered solution vector. Using tensor products, we have the following two-dimensional SBP operators in $\tilde{\Omega}^-$:

$$D_\xi^- = (D_1 \otimes I_{m_\eta^-}), \qquad D_\eta^- = (I_{m_\xi^-} \otimes D_1),$$
$$\hat{D}_\xi^- = (\hat{D}_1 \otimes I_{m_\eta^-}), \qquad \hat{D}_\eta^- = (I_{m_\xi^-} \otimes \hat{D}_1),$$
$$H_\xi^- = (H \otimes I_{m_\eta^-}), \qquad H_\eta^- = (I_{m_\xi^-} \otimes H),$$
$$e_w^- = (e_l^\top \otimes I_{m_\eta^-}), \qquad e_e^- = (e_r^\top \otimes I_{m_\eta^-}),$$
$$e_s^- = (I_{m_\xi^-} \otimes e_l^\top), \qquad e_n^- = (I_{m_\xi^-} \otimes e_r^\top), \tag{30}$$

where $\hat{D}_1$ is the boundary derivative operator in Definition 2. Note that with a general variable coefficients vector $\mathbf{c} \in \mathbb{R}^{N^-}$, tensor products can not be used to construct the two-dimensional variable coefficient operators $D_{\xi\xi}^{(\mathbf{c})}$ and $D_{\eta\eta}^{(\mathbf{c})}$. Instead, the one-dimensional operators are built line-by-line with the corresponding values of $\mathbf{c}$ and stitched together to form the two-dimensional operators. See [30] for more details on this.

Next, we use the relations in Section 3.1 to derive finite difference operators in Ω_p^-. As previously mentioned, this derivation can be found with more detail in, e.g., [30,21]. Since the mappings $x(\xi,\eta;p)$ and $y(\xi,\eta;p)$ are not generally known analytically, the metric derivatives x_ξ, x_η, y_ξ, and y_η are computed using the first derivative SBP operators D_ξ^- and D_η^-. If $\mathbf{x}$ and $\mathbf{y}$ are vectors containing the coordinates on the physical grid, we have the discrete metric coefficient diagonal matrices

$$\begin{aligned} \mathbf{X}_\xi &= \mathtt{diag}(D_\xi^- \mathbf{x}), \\ \mathbf{X}_\eta &= \mathtt{diag}(D_\eta^- \mathbf{x}), \\ \mathbf{Y}_\xi &= \mathtt{diag}(D_\xi^- \mathbf{y}), \\ \mathbf{Y}_\eta &= \mathtt{diag}(D_\eta^- \mathbf{y}), \end{aligned} \tag{31}$$

and

$$\begin{aligned}\mathbf{J} &= \mathbf{X}_\xi \mathbf{Y}_\eta - \mathbf{X}_\eta \mathbf{Y}_\xi,\\ \alpha_1 &= \mathbf{J}^{-1}(\mathbf{X}_\eta^2 + \mathbf{Y}_\eta^2),\\ \beta &= -\mathbf{J}^{-1}(\mathbf{X}_\xi \mathbf{X}_\eta + \mathbf{Y}_\xi \mathbf{Y}_\eta),\\ \alpha_2 &= \mathbf{J}^{-1}(\mathbf{X}_\xi^2 + \mathbf{Y}_\xi^2).\end{aligned} \tag{32}$$

Using (9), (10), and (12), we get the first derivative operators

$$\begin{aligned}D_x^- &= J^{-1}(\mathbf{Y}_\eta D_\xi^- - \mathbf{Y}_\xi D_\eta^-),\\ D_y^- &= J^{-1}(-\mathbf{X}_\eta D_\xi^- + \mathbf{X}_\xi D_\eta^-),\end{aligned} \tag{33}$$

the two-dimensional curvilinear Laplace operator

$$D_L^- = J^{-1}(D_{\xi\xi}^{(\alpha_1)} + D_\eta^- \beta D_\xi^- + D_\xi^- \beta D_\eta^- + D_{\eta\eta}^{(\alpha_2)}), \tag{34}$$

and the discrete normal derivative operators

$$\begin{aligned}d_w^- &= -e_w^- \mathbf{W}_2^{-1}(\alpha_1 e_w^{-\top} e_w^- \hat{D}_\xi^- + \beta e_w^{-\top} e_w^- D_\eta^-),\\ d_e^- &= e_e^- \mathbf{W}_2^{-1}(\alpha_1 e_e^{-\top} e_e^- \hat{D}_\xi^- + \beta e_e^{-\top} e_e^- D_\eta^-),\\ d_s^- &= -e_s^- \mathbf{W}_1^{-1}(\alpha_2 e_s^{-\top} e_s^- \hat{D}_\eta^- + \beta e_s^{-\top} e_s^- D_\xi^-),\\ d_n^- &= e_n^- \mathbf{W}_1^{-1}(\alpha_2 e_n^{-\top} e_n^- \hat{D}_\eta^- + \beta e_n^{-\top} e_n^- D_\xi^-),\end{aligned} \tag{35}$$

where

$$\mathbf{W}_1 = \sqrt{\mathbf{X}_\xi^2 + \mathbf{Y}_\xi^2} \quad \text{and} \quad \mathbf{W}_2 = \sqrt{\mathbf{X}_\eta^2 + \mathbf{Y}_\eta^2}. \tag{36}$$

We also have the following norm matrices:

$$\begin{aligned}H^- &= H_\xi^- H_\eta^- J,\\ H_w^- &= H e_w \mathbf{W}_2 e_w^\top,\\ H_e^- &= H e_e \mathbf{W}_2 e_e^\top,\\ H_s^- &= H e_s \mathbf{W}_1 e_s^\top,\\ H_n^- &= H e_n \mathbf{W}_1 e_n^\top,\end{aligned} \tag{37}$$

with inner products $(\cdot,\cdot)_{H^-}$ and $(\cdot,\cdot)_{H_{w,e,s,n}^-}$ defined as in (17). Note that the power of two and square roots in (32) and (36) are evaluated elementwise. Using Definitions 1 and 2, the Laplace operator D_L^- can be shown to satisfy (see [21])

$$\begin{aligned}(\mathbf{u}, D_L^- \mathbf{v})_{H^-} = &-(D_x^- \mathbf{u}, D_x^- \mathbf{v})_{H^-} - (D_y^- \mathbf{u}, D_y^- \mathbf{v})_{H^-} - (\mathbf{u}, \mathbf{v})_{R^-}\\ &+ \sum_{k=w,e,s,n} (e_k^- \mathbf{u}, d_k^- \mathbf{v})_{H_k^-}, \quad \forall \mathbf{u}, \mathbf{v} \in \mathbb{R}^{N^-},\end{aligned} \tag{38}$$

where

$$(\mathbf{u}, \mathbf{v})_{R^-} = \mathbf{u}^\top (R_\xi^{(\alpha_1)} H_\eta^- + R_\eta^{(\alpha_2)} H_\xi^-)\mathbf{v}, \tag{39}$$

with $R_\xi^{(\alpha_1)}$ and $R_\eta^{(\alpha_2)}$ created the same way as $D_{\xi\xi}^{(\alpha_1)}$ and $D_{\eta\eta}^{(\alpha_2)}$. Note that $R_\xi^{(\alpha_1)} \approx 0$, $R_\eta^{(\alpha_2)} \approx 0$ (since $R^{(\mathbf{c})} \approx 0$), and

$$(\mathbf{u}, \mathbf{v})_{R^-} = (\mathbf{v}, \mathbf{u})_{R^-} \quad \text{and} \quad \|\mathbf{u}\|_{R^-}^2 := (\mathbf{u}, \mathbf{u})_{R^-} \geq 0, \quad \forall \mathbf{u}, \mathbf{v} \in \mathbb{R}^{N^-}. \tag{40}$$

Remark 1. The two domains Ω^+ and Ω_p^- are conforming at the interface Γ_I, and for convenience reasons, we shall use the same number of grid points on both sides of the interface, i.e. $m_{\Gamma_I} = m_x^+ = m_\xi^-$. Therefore,

$$(\mathbf{u}, \mathbf{v})_{H_s^+} = (\mathbf{u}, \mathbf{v})_{H_n^-}, \quad \forall \mathbf{u}, \mathbf{v} \in \mathbb{R}^{m_{\Gamma_I}}. \tag{41}$$

3.5. SBP-P-SAT discretization

We now return to the forward problem (2b). By replacing all spatial derivatives in (2b) by their corresponding SBP operators, we obtain a constrained initial value problem given by,

$$
\begin{aligned}
&\mathbf{v}_{tt}^{+} = c^2 D_L^{+}\mathbf{v}^{+} + f(t)\tilde{\mathbf{d}}_s, && t>0,\\
&\mathbf{v}_{tt}^{-} = c^2 D_L^{-}\mathbf{v}^{-}, && t>0,\\
&e_{w,e}^{\pm}\mathbf{v}_t^{\pm} + c d_{w,e}^{\pm}\mathbf{v}^{\pm} = 0, && t>0,\\
&d_s^{-}\mathbf{v}^{-} = 0, && t>0,\\
&e_n^{+}\mathbf{v}^{+} = 0, && t>0,\\
&e_s^{+}\mathbf{v}^{+} - e_n^{-}\mathbf{v}^{-} = 0, && t>0,\\
&d_s^{+}\mathbf{v}^{+} + d_n^{-}\mathbf{v}^{-} = 0, && t>0,\\
&\mathbf{v}^{\pm} = 0, \quad \mathbf{v}_t^{\pm} = 0, && t=0,
\end{aligned}
\tag{42}
$$

where

$$
\tilde{\mathbf{d}}_s = (H^{+})^{-1}(\delta_x^{(s)} \otimes \delta_y^{(s)}). \tag{43}
$$

Here $\delta_x^{(s)}$ and $\delta_y^{(s)}$ are discrete one-dimensional point sources discretized as in [36], using the same number of moment conditions as the order of the SBP operators and no smoothness conditions.

We impose the boundary and interface conditions in (42) using a combination of the SAT method [37,38,21] and the projection method [23,39]. SATs are used to weakly impose the outflow and Neumann boundary conditions and to couple the fluxes across the interface, while the projection method strongly imposes the Dirichlet boundary conditions and continuity of the solution across the interface. This choice is made to keep the projection operator as simple as possible while avoiding the so-called borrowing trick necessary for an SBP-SAT discretization [40]. A hybrid SBP-P-SAT method for the second-order wave equation on Cartesian multiblock domains with non-conforming interfaces was presented in [23]. Indeed, the scheme presented here is the same as in [23] if the interpolation operators are replaced with identity matrices and Ω_p^{-} is Cartesian. More details on the projection method can be found in [41,42]. Let

$$
\mathbf{w} = \begin{bmatrix} \mathbf{v}^{+} \\ \mathbf{v}^{-} \end{bmatrix} \in \mathbb{R}^N, \tag{44}
$$

where $N = N^{+} + N^{-}$, denote the global solution vector and

$$
\bar{H} = \begin{bmatrix} H^{+} & 0 \\ 0 & H^{-} \end{bmatrix}, \tag{45}
$$

a global norm matrix with associated inner product defined as in (17). A consistent SBP-P-SAT discretization is then given by

$$
\begin{aligned}
&\mathbf{w}_{tt} = D\mathbf{w} + E\mathbf{w}_t + f(t)\mathbf{d}_s, && t>0,\\
&\mathbf{w} = 0, \quad \mathbf{w}_t = 0, && t=0,
\end{aligned}
\tag{46}
$$

where

$$
\begin{aligned}
D &= c^2 P\left(\begin{bmatrix} D_L^{+} & 0 \\ 0 & D_L^{-} \end{bmatrix} + SAT_{BC_1} + SAT_{IC}\right)P,\\
E &= cPSAT_{BC_2}P,\\
SAT_{BC_1} &= -\bar{H}^{-1}\left(\sum_{k=w,e}\begin{bmatrix} e_k^{+\top}H_k^{+}d_k^{+} & 0 \\ 0 & e_k^{-\top}H_k^{-}d_k^{-} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & e_s^{-\top}H_s^{-}d_s^{-} \end{bmatrix}\right),\\
SAT_{BC_2} &= -\bar{H}^{-1}\sum_{k=w,e}\begin{bmatrix} e_k^{+\top}H_k^{+}e_k^{+} & 0 \\ 0 & e_k^{-\top}H_k^{-}e_k^{-} \end{bmatrix},\\
SAT_{IC} &= -\bar{H}^{-1}\begin{bmatrix} e_s^{+\top}H_s^{+}d_s^{+} & e_s^{+\top}H_s^{+}d_n^{-} \\ 0 & 0 \end{bmatrix}, \quad \text{and}\\
\mathbf{d}_s &= \begin{bmatrix} \tilde{\mathbf{d}}_s \\ 0 \end{bmatrix}.
\end{aligned}
\tag{47}
$$

The projection operator is given by

$$
P = I - \bar{H}^{-1}L^{\top}(L\bar{H}^{-1}L^{\top})^{-1}L, \tag{48}
$$

where

$$
L = \begin{bmatrix} e_n^{+} & 0 \\ e_s^{+} & -e_n^{-} \end{bmatrix}. \tag{49}
$$

This corresponds to imposing the conditions

$$\begin{aligned} &e^{\pm}_{w,e}\mathbf{v}^{\pm}_t + cd^{\pm}_{w,e}\mathbf{v}^{\pm} = 0, \quad t>0,\\ &d^-_s\mathbf{v}^- = 0, \qquad\qquad t>0,\\ &d^+_s\mathbf{v}^+ + d^-_n\mathbf{v}^- = 0, \qquad t>0, \end{aligned} \tag{50}$$

weakly using the SAT method and the conditions

$$\begin{aligned} &e^+_n\mathbf{v}^+ = 0, \qquad\qquad t>0,\\ &e^+_s\mathbf{v}^+ - e^-_n\mathbf{v}^- = 0, \quad t>0, \end{aligned} \tag{51}$$

strongly using the projection method.

We now prove three Lemmas. The first one is with regards to the stability of the scheme (46) while the second and third are on its self-adjointness properties. Similar self-adjointness properties of SBP-SAT discretizations of the acoustic and elastic wave equation have been shown in e.g. [43,25]. However, to the best of our knowledge, the derivation is new for the SBP-P-SAT discretization presented herein.

Lemma 1. *The SBP-P-SAT scheme* (46) *is stable.*

Proof. We prove stability using the energy method. Since data does not influence stability, we set $f(t)=0$. Taking the inner product between $\mathbf{w_t}$ and (46) gives

$$\begin{aligned} \frac{1}{2}\frac{d}{dt}\|\mathbf{w}_t\|^2_{\tilde{H}} &= c^2(\tilde{\mathbf{v}}^+_t, D^+_L\tilde{\mathbf{v}}^+)_{H^+} + c^2(\tilde{\mathbf{v}}^-_t, D^-_L\tilde{\mathbf{v}}^-)_{H^-}\\ &\quad - \sum_{k=w,e}\left(c\|e^+_k\tilde{\mathbf{v}}^+_t\|^2_{H^+_k} + c^2(e^+_k\tilde{\mathbf{v}}^+_t, d^+_k\tilde{\mathbf{v}}^+)_{H^+_k}\right.\\ &\quad \left. + c\|e^-_k\tilde{\mathbf{v}}^-_t\|^2_{H^-_k} + c^2(e^-_k\tilde{\mathbf{v}}^-_t, d^-_k\tilde{\mathbf{v}}^-)_{H^+_k}\right)\\ &\quad - c^2(e^-_s\tilde{\mathbf{v}}^-_t, d^-_s\tilde{\mathbf{v}}^-)_{H^-_s} - c^2(e^+_s\tilde{\mathbf{v}}^+_t, d^+_s\tilde{\mathbf{v}}^+ + d^-_n\tilde{\mathbf{v}}^-)_{H^+_s}, \end{aligned} \tag{52}$$

where $P\mathbf{w} = \tilde{\mathbf{w}} = \begin{bmatrix}\tilde{\mathbf{v}}^+\\ \tilde{\mathbf{v}}^-\end{bmatrix}$ denotes the projected solution vector. Using (27) and (38) and rearranging terms lead to

$$\begin{aligned} \frac{d}{dt}(E^+ + E^-) &= -2c\sum_{k=w,e}\|e^-_k\tilde{\mathbf{v}}^-_t\|^2_{H^-_k} + \|e^+_k\tilde{\mathbf{v}}^+_t\|^2_{H^+_k}\\ &\quad + 2c^2(e^-_n\tilde{\mathbf{v}}^-_t - e^+_s\tilde{\mathbf{v}}^+_t, d^-_n\tilde{\mathbf{v}}^-)_{H^{(-)}_N} + 2c^2(e^+_n\tilde{\mathbf{v}}^+_t, d^+_n\tilde{\mathbf{v}}^+)_{H^+_N}, \end{aligned} \tag{53}$$

where

$$E^{\pm} = \|\mathbf{v}^{\pm}_t\|^2_{H^{\pm}} + c^2\|D^{\pm}_x\tilde{\mathbf{v}}^{\pm}\|^2_{H^{\pm}} + c^2\|D^{\pm}_y\tilde{\mathbf{v}}^{\pm}\|^2_{H^{\pm}} + c^2\|\tilde{\mathbf{v}}^{\pm}\|^2_{R^{\pm}}. \tag{54}$$

Since $L\tilde{\mathbf{w}} = LP\mathbf{w} = 0$ holds exactly by the definition of P, we have

$$e^+_n\tilde{\mathbf{v}}^+ = 0, \quad \text{and} \quad e^-_n\tilde{\mathbf{v}}^- = e^+_s\tilde{\mathbf{v}}^+. \tag{55}$$

Inserted into (53) results in

$$\frac{d}{dt}(E^+ + E^-) = -2c\sum_{k=w,e}\|e^-_k\tilde{\mathbf{v}}^-_t\|^2_{H^-_k} + \|e^+_k\tilde{\mathbf{v}}^+_t\|^2_{H^+_k} \le 0, \tag{56}$$

which is the discrete equivalent to the continuous energy equation (6) and proofs stability of (46). □

Lemma 2. *The matrix D in* (47) *is self-adjoint with respect to $\tilde{H}$, i.e.*

$$(\mathbf{u}, D\mathbf{v})_{\tilde{H}} = (D\mathbf{u}, \mathbf{v})_{\tilde{H}}, \quad \forall\mathbf{u},\mathbf{v}\in\mathbb{R}^N. \tag{57}$$

Proof. Let $P\mathbf{u} = P\begin{bmatrix}\mathbf{u}^+\\ \mathbf{u}^-\end{bmatrix} = \tilde{\mathbf{u}} = \begin{bmatrix}\tilde{\mathbf{u}}^+\\ \tilde{\mathbf{u}}^-\end{bmatrix}$ and $P\mathbf{v} = P\begin{bmatrix}\mathbf{v}^+\\ \mathbf{v}^-\end{bmatrix} = \tilde{\mathbf{v}} = \begin{bmatrix}\tilde{\mathbf{v}}^+\\ \tilde{\mathbf{v}}^-\end{bmatrix}$, then

$$\begin{aligned} (\mathbf{u}, D\mathbf{v})_{\tilde{H}} &= c^2(\tilde{\mathbf{u}}^+, D^+_L\tilde{\mathbf{v}}^+)_{H^+} + c^2(\tilde{\mathbf{u}}^-, D^-_L\tilde{\mathbf{v}}^-)_{H^-}\\ &\quad - c^2\sum_{k=w,e}(e^+_k\tilde{\mathbf{u}}^+, d^+_k\tilde{\mathbf{v}}^+)_{H^+_k} + (e^-_k\tilde{\mathbf{u}}^-, d^-_k\tilde{\mathbf{v}}^-)_{H^+_k}\\ &\quad - c^2(e^-_s\tilde{\mathbf{u}}^-, d^-_s\tilde{\mathbf{v}}^-)_{H^-_s} - c^2(e^+_s\tilde{\mathbf{u}}^+, d^+_s\tilde{\mathbf{v}}^+ + d^-_n\tilde{\mathbf{v}}^-)_{H^+_s}. \end{aligned} \tag{58}$$

Using (27) and (38) and rearranging terms lead to

$$
\begin{aligned}
(\mathbf{u}, D\mathbf{v})_{\bar{H}} = &-c^2(D_x^+\tilde{\mathbf{u}}^+, D_x^+\tilde{\mathbf{v}}^+)_{H^+} - c^2(D_y^+\tilde{\mathbf{u}}^+, D_y^+\tilde{\mathbf{v}}^+)_{H^+} \\
&- c^2(D_x^-\tilde{\mathbf{u}}^-, D_x^-\tilde{\mathbf{v}}^-)_{H^-} - c^2(D_y^-\tilde{\mathbf{u}}^-, D_y^-\tilde{\mathbf{v}}^-)_{H^-} \\
&- c^2(\tilde{\mathbf{u}}^+, \tilde{\mathbf{v}}^+)_{R^+} - c^2(\tilde{\mathbf{u}}^-, \tilde{\mathbf{v}}^-)_{R^-} \\
&+ c^2(e_n^-\tilde{\mathbf{u}}^- - e_s^+\mathbf{u}^+, d_n^-\tilde{\mathbf{v}}^-)_{H_n^-} + c^2(e_n^+\tilde{\mathbf{u}}^+, d_n^+\tilde{\mathbf{v}}^+)_{H_n^+}.
\end{aligned} \tag{59}
$$

Since $LP\mathbf{u} = 0$ hold exactly by the definition of P, we have

$$
e_n^+\tilde{\mathbf{u}}^+ = 0, \quad \text{and} \quad e_n^-\tilde{\mathbf{u}}^- = e_s^+\tilde{\mathbf{u}}^+. \tag{60}
$$

Inserted into (59) results in

$$
\begin{aligned}
(\mathbf{u}, D\mathbf{v})_{\bar{H}} = &-c^2(D_x^+\tilde{\mathbf{u}}^+, D_x^+\tilde{\mathbf{v}}^+)_{H^+} - c^2(D_y^+\tilde{\mathbf{u}}^+, D_y^+\tilde{\mathbf{v}}^+)_{H^+} \\
&- c^2(D_x^-\tilde{\mathbf{u}}^-, D_x^-\tilde{\mathbf{v}}^-)_{H^-} - c^2(D_y^-\tilde{\mathbf{u}}^-, D_y^-\tilde{\mathbf{v}}^-)_{H^-} \\
&- c^2(\tilde{\mathbf{u}}^+, \tilde{\mathbf{v}}^+)_{R^+} - c^2(\tilde{\mathbf{u}}^-, \tilde{\mathbf{v}}^-)_{R^-}.
\end{aligned} \tag{61}
$$

Since all the terms on the right-hand side of (61) are symmetric, i.e. we can swap $\mathbf{u}$ and $\mathbf{v}$ and obtain the same expression, we have

$$
(\mathbf{u}, D\mathbf{v})_{\bar{H}} = (\mathbf{v}, D\mathbf{u})_{\bar{H}} = (D\mathbf{u}, \mathbf{v})_{\bar{H}}, \tag{62}
$$

which proves the lemma. □

Lemma 3. *The matrix E in (47) is self-adjoint with respect to $\bar{H}$, i.e.*

$$
(\mathbf{u}, E\mathbf{v})_{\bar{H}} = (E\mathbf{u}, \mathbf{v})_{\bar{H}}, \quad \forall \mathbf{u}, \mathbf{v} \in \mathbb{R}^N. \tag{63}
$$

Proof. Let $P\mathbf{u} = P\begin{bmatrix}\mathbf{u}^+\\ \mathbf{u}^-\end{bmatrix} = \tilde{\mathbf{u}} = \begin{bmatrix}\tilde{\mathbf{u}}^+\\ \tilde{\mathbf{u}}^-\end{bmatrix}$ and $P\mathbf{v} = P\begin{bmatrix}\mathbf{v}^+\\ \mathbf{v}^-\end{bmatrix} = \tilde{\mathbf{v}} = \begin{bmatrix}\tilde{\mathbf{v}}^+\\ \tilde{\mathbf{v}}^-\end{bmatrix}$, then

$$
(\mathbf{u}, E\mathbf{v})_{\bar{H}} = -c \sum_{k=w,e} (e_k^+\tilde{\mathbf{u}}^+, e_k^+\tilde{\mathbf{v}}^+)_{H_k^+} + (e_k^-\tilde{\mathbf{u}}^-, e_k^-\tilde{\mathbf{v}}^-)_{H_k^-}. \tag{64}
$$

Since all the terms on the right-hand side of (64) are symmetric, i.e. we can swap $\mathbf{u}$ and $\mathbf{v}$ and obtain the same expression, we have

$$
(\mathbf{u}, E\mathbf{v})_{\bar{H}} = (\mathbf{v}, E\mathbf{u})_{\bar{H}} = (E\mathbf{u}, \mathbf{v})_{\bar{H}}, \tag{65}
$$

which proves the lemma. □

4. The discrete optimization problem

We now return to the optimization problem (2). As discussed in Section 1, when analyzing PDE-constrained optimization problems of this kind one typically has two choices. Either you derive the adjoint equations and the associated gradient of the loss functional in the continuous setting and then discretize (OD). Or, you discretize the forward problem before deriving the discrete adjoint equations and gradient (DO). Here we do a compromise and discretize in space, but leave time continuous, and then proceed with the optimization. As we shall see, in the semi-discrete setting the spatial discretization guarantees that DO and OD are equivalent, due to Lemmas 2 and 3. With space discretized, we have the ODE-constrained optimization problem

$$
\min_{\mathbf{p}} \mathcal{J}(\mathbf{w}, \mathbf{p}) = \frac{1}{2}\int_0^T r(\mathbf{w}, t)^2\, dt + \frac{1}{2}\gamma\|D_2\mathbf{p}\|_H^2, \quad \text{such that} \tag{66a}
$$

$$
\begin{aligned}
&\mathbf{w}_{tt} = D\mathbf{w} + E\mathbf{w}_t + f(t)\mathbf{d}_s, \quad t > 0 \\
&\mathbf{w} = \mathbf{w}_t = 0, \qquad\qquad\qquad\quad t = 0,
\end{aligned} \tag{66b}
$$

where $r(\mathbf{w}, t)$ is the residual of the loss given by

$$
r(\mathbf{w}, t) = (\mathbf{d}_r, \mathbf{w}(t))_{\bar{H}} - \mathbf{v}_d(t), \tag{67}
$$

where $\mathbf{v}_d(t)$ is the target data in the receiver and $\mathbf{d}_r$ is constructed analogously to $\mathbf{d}_s$. Note that the term $(\mathbf{d}_r, \mathbf{w}(t))_{\bar{H}}$ is an interpolation of $\mathbf{w}$ onto the receiver coordinates $\mathbf{x}_r$.

4.1. Shape parameterization and regularization

The shape of the bottom boundary can be parametrized in many ways, see [44] for an overview of common methods. Here we let $\mathbf{p} = [p_1, p_2, p_3, \dots, p_{m_{\Gamma_I}}]$ be a vector containing the y-coordinates of the bottom for each grid point in the x-direction, see Fig. 3. To

combat the ill-posedness of (2) a regularization term $\frac{1}{2}\gamma\|D_2\mathbf{p}\|_H^2$ is added to (66a). This additional term penalizes spurious oscillations in $\mathbf{p}$ which helps us find smooth shapes (i.e. reduces the risk of getting stuck in a non-regular local minima to (66)). The parameter γ is chosen experimentally, a small value will give rise to oscillations and lead to a non-regular solution, whereas a large value will restrict $\mathbf{p}$ and lead to suboptimal solutions.

Using linear transfinite interpolation, the elements in discrete column-major ordered coordinate vectors $\mathbf{x}$, and $\mathbf{y}$ in Ω_p^- are given by

$$x_{i,j} = x_l + (x_r - x_l)\xi_i \quad \text{and} \quad y_{i,j} = p_i + (L_I - p_i)\eta_j, \tag{68}$$

where ξ_i, η_j are the elements in the 1D coordinate vectors in $\tilde{\Omega}_p^-$, and L_I is the y-coordinate of the interface, here $L_I = 0.5$.

4.2. The adjoint equation method

To solve (66) efficiently gradient-based optimization will be employed, and to this end, we require $\frac{\partial \mathcal{J}}{\partial p_i}, i = 1, 2, ..., m_{\Gamma_I}$, given by

$$\begin{aligned}\frac{\partial \mathcal{J}}{\partial p_i} &= \frac{1}{2}\int_0^T \frac{\partial}{\partial p_i} r(\mathbf{w},t)^2\, dt + \gamma(D_2\mathbf{e}_i, D_2\mathbf{p})_H = \\ &= \int_0^T (r(\mathbf{w},t)\mathbf{d}_r, \frac{\partial \mathbf{w}}{\partial p_i})_{\bar{H}}\, dt + \gamma(D_2\mathbf{e}_i, D_2\mathbf{p})_H,\end{aligned} \tag{69}$$

where $\mathbf{e}_i$ is a column vector with the entry 1 at position i and zeros elsewhere. Here we have used that $\frac{\partial \mathbf{d}_r}{\partial p_i} = 0$ (since the receiver is located in Ω^+). Clearly, naively computing (69) requires evaluating $\frac{\partial \mathbf{w}}{\partial p_i}$. Approximating $\frac{\partial \mathbf{w}}{\partial p_i}$ using first-order finite differences, e.g., would require $m_{\Gamma_I} + 1$ solves of (66b) which quickly becomes costly for large problem sizes. Instead, we introduce a Lagrange multiplier $\lambda \in \mathbb{R}^N$ and use the adjoint framework [1,2], which allows us to compute $\frac{\partial \mathcal{J}}{\partial p_i}$ without evaluating or approximating $\frac{\partial \mathbf{w}}{\partial p_i}$. The method is summarized in the following Lemma:

Lemma 4. *Let* $\mathbf{w}$ *be a solution to* (66b) *and* λ *a solution to the adjoint equation*

$$\begin{aligned}&\lambda_{\tau\tau} = D\lambda + E\lambda_\tau - r(\mathbf{w},\tau)\mathbf{d}_r, \quad 0 \le \tau \le T, \\ &\lambda = \lambda_\tau = 0, \quad \tau = 0,\end{aligned} \tag{70}$$

where $\tau = T - t$. *Then the gradient is given by*

$$\frac{\partial \mathcal{J}}{\partial p_i} = \int_0^T -(\lambda, \frac{\partial D}{\partial p_i}\mathbf{w})_{\bar{H}} + (\lambda_t, \frac{\partial E}{\partial p_i}\mathbf{w})_{\bar{H}}\, dt + \gamma(D_2\mathbf{e}_i, D_2\mathbf{p})_H. \tag{71}$$

Proof. First, we define the Lagrangian functional

$$\mathcal{L}(\mathbf{w},\lambda) = \mathcal{J} + \int_0^T (\lambda, \mathbf{w}_{tt} - D\mathbf{w} - E\mathbf{w}_t - f(t)\mathbf{d}_s)_{\bar{H}}\, dt, \tag{72}$$

and note that $\mathcal{J} = \mathcal{L}$ and thus $\frac{\partial \mathcal{J}}{\partial p_i} = \frac{\partial \mathcal{L}}{\partial p_i}$ for any λ whenever $\mathbf{w}$ is a solution to (66b). Consider the gradient of $\mathcal{L}$, given by

$$\frac{\partial \mathcal{L}}{\partial p_i} = \frac{\partial \mathcal{J}}{\partial p_i} + \int_0^T (\lambda, \frac{\partial \mathbf{w}_{tt}}{\partial p_i} - \frac{\partial D\mathbf{w}}{\partial p_i} - \frac{\partial E\mathbf{w}_t}{\partial p_i})_{\bar{H}}\, dt, \tag{73}$$

where we have used that $\frac{\partial \mathbf{d}_s}{\partial p_i} = 0$ (since the source is located in Ω^+). The first term in (73) is given by (69). The other terms are treated using integration by parts in time, resulting in

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial p_i} = \frac{\partial \mathcal{J}}{\partial p_i} &+ \int_0^T (\lambda_{tt}, \frac{\partial \mathbf{w}}{\partial p_i})_{\bar{H}} - (\lambda, \frac{\partial D\mathbf{w}}{\partial p_i})_{\bar{H}} + (\lambda_t, \frac{\partial E\mathbf{w}}{\partial p_i})_{\bar{H}}\, dt \\ &+ \left[(\lambda, \frac{\partial \mathbf{w}_t}{\partial p_i})_{\bar{H}} - (\lambda_t, \frac{\partial \mathbf{w}}{\partial p_i})_{\bar{H}} - (\lambda, \frac{\partial E\mathbf{w}}{\partial p_i})_{\bar{H}}\right]_0^T\end{aligned}$$

$$= \frac{\partial \mathcal{J}}{\partial p_i} + \int_0^T (\lambda_{tt}, \frac{\partial \mathbf{w}}{\partial p_i})_{\bar{H}} - (\lambda, \frac{\partial D\mathbf{w}}{\partial p_i})_{\bar{H}} + (\lambda_t, \frac{\partial E\mathbf{w}}{\partial p_i})_{\bar{H}}\, dt, \tag{74}$$

where we have used the initial conditions for $\mathbf{w}$ and $\mathbf{w}_t$ and prescribed the following terminal conditions for λ and λ_t:

$$\lambda = 0, \quad \lambda_t = 0, \quad t = T. \tag{75}$$

Using (69) and Lemmas 2 and 3 we get

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial p_i} &= \int_0^T (r(\mathbf{w}, t)\mathbf{d}_r + \lambda_{tt} - D\lambda + E\lambda_t, \frac{\partial \mathbf{w}}{\partial p_i})_{\bar{H}}\, dt \\ &+ \int_0^T -(\lambda, \frac{\partial D}{\partial p_i}\mathbf{w})_{\bar{H}} + (\lambda_t, \frac{\partial E}{\partial p_i}\mathbf{w})_{\bar{H}}\, dt + \gamma(D_2 e_i, D_2\mathbf{p})_{\bar{H}}. \end{aligned} \tag{76}$$

If λ satisfies (70) we get the following formula for the gradient:

$$\frac{\partial \mathcal{L}}{\partial p_i} = \int_0^T -(\lambda, \frac{\partial D}{\partial p_i}\mathbf{w})_{\bar{H}} + (\lambda_t, \frac{\partial E}{\partial p_i}\mathbf{w})_{\bar{H}}\, dt + \gamma(D_2\mathbf{e}_i, D_2\mathbf{p})_H, \tag{77}$$

and since $\frac{\partial \mathcal{L}}{\partial p_i} = \frac{\partial \mathcal{J}}{\partial p_i}$ we get (71). □

Note that the only difference between (70) and (66b) is the forcing function, and hence stability of (70) follows immediately from Lemma 1. The matrices $\frac{\partial D}{\partial p_i}$ and $\frac{\partial E}{\partial p_i}$ can be computed analytically and are presented in Appendix A. Essentially, the matrices consist of the SBP operators together with derivatives of the metric coefficients given in Section 3.4. The derivation involves repeated application of the product rule but is otherwise straightforward.

Remark 2. In this model problem, we have assumed that the source and receiver are located in Ω^+, which simplifies the analysis since $\frac{\partial \mathbf{d}_s}{\partial p_i} = \frac{\partial \mathbf{d}_r}{\partial p_i} = 0$ holds. If this is not the case, one would have to evaluate the derivative of the discrete Dirac delta function, which is not well-defined everywhere for the discrete Dirac delta functions used here. In [45], point source discretizations that are continuously differentiable everywhere are derived. In principle, we could use this discretization instead and allow $\mathbf{x}_s$ and $\mathbf{x}_r$ to be located in Ω_p^-, but this is out of scope for the present work.

4.3. Dual consistency

In the continuous setting it is well-established that (2b) is self-adjoint under time reversal [46,2,21,25] such that for the continuous adjoint state variables $\lambda^\pm$ the adjoint (or dual) equations are given by

$$\begin{aligned} &\lambda^+_{\tau\tau} = c^2\Delta\lambda^+ - r(u,\tau)\hat{\delta}(\mathbf{x}-\mathbf{x}_r), && \mathbf{x}\in\Omega^+, && \tau\in[0,T],\\ &\lambda^-_{\tau\tau} = c^2\Delta\lambda^-, && \mathbf{x}\in\Omega_p^-, && \tau\in[0,T],\\ &\lambda^\pm_\tau + c\mathbf{n}^\pm\cdot\nabla\lambda^\pm = 0, && \mathbf{x}\in\partial\Omega_p^{(\pm,w,e)}, && \tau\in[0,T],\\ &\mathbf{n}^-\cdot\nabla\lambda^- = 0, && \mathbf{x}\in\partial\Omega_p^{(-,s)}, && \tau\in[0,T],\\ &\lambda^+ = 0, && \mathbf{x}\in\partial\Omega_p^{(+,n)}, && \tau\in[0,T],\\ &\lambda^+ - \lambda^- = 0, && \mathbf{x}\in\Gamma_I, && \tau\in[0,T],\\ &n^+\cdot\nabla\lambda^+ + n^-\cdot\nabla\lambda^- = 0, && \mathbf{x}\in\Gamma_I, && \tau\in[0,T],\\ &\lambda^\pm = 0, \quad \lambda^\pm_\tau = 0, && \mathbf{x}\in\Omega, && \tau = 0. \end{aligned} \tag{78}$$

Note that the semi-discrete adjoint problem (70) is a consistent approximation of (78), and thus (66b) is a dual consistent semi-discretization of the forward problem [9,47]. Moreover, the gradient to the continuous optimization problem (2) is given by the following lemma:

Lemma 5. *Let λ^- satisfy (78). Then the gradient $\frac{\delta \mathrm{J}}{\delta p}$ to (2), is given by*

$$\frac{\delta \mathrm{J}}{\delta p} = \int_0^T G_\lambda + G_{\lambda_t} dt, \tag{79}$$

where

$$
\begin{aligned}
G_\lambda = -c^2\Big(& -(\lambda^-, J^{-1}\frac{\delta J}{\delta p}\Delta u^-)_{\Omega^-} \\
& +(\lambda^-, J^{-1}((\frac{\delta \alpha_1}{\delta p}u_\xi^- + \frac{\delta \beta}{\delta p}u_\eta^-)_\xi + (\frac{\delta \alpha_2}{\delta p}u_\eta^- + \frac{\delta \beta}{\delta p}u_\xi^-)_\eta))_{\Omega^-} \\
& -\langle \lambda^-, W_1^{-1}(\frac{\delta \alpha_1}{\delta p}u_\xi^- + \frac{\delta \beta}{\delta p}u_\eta^-)\rangle_{\partial\Omega^{(-,e)}} \\
& +\langle \lambda^-, W_1^{-1}(\frac{\delta \alpha_1}{\delta p}u_\xi^- + \frac{\delta \beta}{\delta p}u_\eta^-)\rangle_{\partial\Omega^{(-,w)}} \\
& -\langle \lambda^-, W_2^{-1}(\frac{\delta \alpha_2}{\delta p}u_\eta^- + \frac{\delta \beta}{\delta p}u_\xi^-)\rangle_{\partial\Omega^{(-,n)}} \\
& +\langle \lambda^-, W_2^{-1}(\frac{\delta \alpha_2}{\delta p}u_\eta^- + \frac{\delta \beta}{\delta p}u_\xi^-)\rangle_{\partial\Omega^{(-,s)}}\Big),
\end{aligned} \tag{80}
$$

and

$$
G_{\lambda_t} = c\Big(\langle \lambda_t^-, W_1^{-1}\frac{\delta W_1}{\delta p}u^-\rangle_{\partial\Omega^{(-,e)}} + \langle \lambda_t^-, W_1^{-1}\frac{\delta W_1}{\delta p}u^-\rangle_{\partial\Omega^{(-,w)}}\Big), \tag{81}
$$

with the metric coefficients J, α_1, α_2, β, W_1 and W_1, given in Section 3.1.

Proof. See Appendix B. □

Lemma 5 is the continuous counterpart of Lemma 4, where (71) is the semi-discrete version of (79), with the addition of regularization. This is seen by comparing the explicit expressions for the operator derivatives presented in Appendix A to (80) - (81) where $G_\lambda \approx (\lambda, \frac{\partial D}{\partial p_i}\mathbf{w})_{\bar{H}}$ and $G_{\lambda_t} \approx (\lambda_t, \frac{\partial E}{\partial p_i}\mathbf{w})_{\bar{H}}$.

4.4. Temporal discretization

For time discretization we use the 4th-order explicit Runge–Kutta method (RK4) and for computing the time integrals a 6th-order accurate SBP quadrature. The time step is chosen as

$$
\Delta t = k\Delta t_{max}, \tag{82}
$$

where

$$
\Delta t_{max} = \frac{2.8}{\sqrt{\rho(D)}}, \tag{83}
$$

is approximately the stability limit of RK4 applied to (46) (written on first-order form), $\rho(D)$ is the spectral radius of D, and $k < 1$ is a positive CFL constant. The influence of E on the stability limit Δt_{max} is in this case very small since it scales as h^{-1}, where h is the spatial step size, compared to D that scales as h^{-2}.

Since the temporal discretization used here is not self-adjoint, the computed gradients will contain an approximation error. In Section 5.2 we verify the accuracy of the gradient and show that it decreases with k. For the numerical experiments presented in this paper, we use $k = 0.1$, which seems to be sufficient for the optimization algorithm to work well on the problems we solve. There exist self-adjoint temporal discretization schemes that would lead to exact gradients, e.g., symplectic Runge–Kutta methods [48] or SBP in time [49] where the temporal derivatives are also approximated using SBP operators, but implementing these are out of scope in the present work.

4.5. Optimization algorithm

There are many optimization algorithms that could be employed for these types of problems. Here we use the BFGS algorithm [50] as implemented in the Matlab function *fminunc*, which is a quasi-newton method that approximates the Hessian using only the gradient of the loss function. For larger problems (e.g., with higher grid resolutions or 3D problems) the L-BFGS method would be a suitable alternative. Each iteration of the BFGS method requires the loss $\mathcal{J}$ and the gradient $\nabla_{\mathbf{p}}\mathcal{J}$, which are computed as follows:

1. Solve the forward problem (66b).
2. Compute the loss (66a) using the solution to the forward problem.
3. Solve the adjoint problem (70) using the solution to the forward problem.
4. For each $i = 1, 2, ..., m_{\Gamma_I}$, compute $\frac{\partial \mathcal{J}}{\partial p_i}$ using (71) and form the gradient vector

$$
\nabla_{\mathbf{p}}\mathcal{J} = \left[\frac{\partial \mathcal{J}}{\partial p_1}, \frac{\partial \mathcal{J}}{\partial p_2}, ..., \frac{\partial \mathcal{J}}{\partial p_{m_{\Gamma_I}}}\right]. \tag{84}
$$

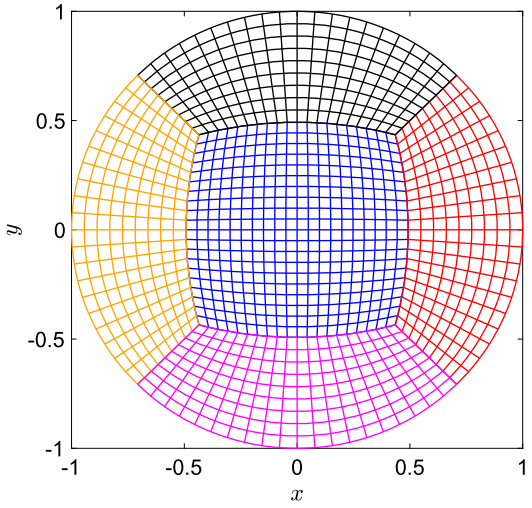


Fig. 2. Grid of the circular domain. The different colors of the grid indicate the block decomposition. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

Note that we only have to solve the forward and adjoint problems one time each per iteration, independently of the number of optimization parameters m_{Γ_I}. This is one of the main advantages of the adjoint method. However, evaluating (71) in step 4 requires storing the vectors $\mathbf{w}$, λ, and λ_t for all time steps, which naturally may be very memory consuming, especially for long-time simulations in realistic 3D applications. In such cases, one often must resort to check-pointing (storing the vectors at certain time intervals and recomputing intermediate results) or storing the vectors on disk.

5. Numerical experiments

5.1. Accuracy study

Since the SBP-P-SAT method is new in this setting (wave equations for multiblock, curvilinear domains), we briefly verify the accuracy of the method by performing a convergence study on the forward problem with a known analytical solution. We consider a circular domain Ω decomposed into five blocks as depicted in Fig. 2, and solve the PDE

$$\begin{aligned} u_{tt} &= \Delta u, && \mathbf{x} \in \Omega, && t \in [0,T], \\ u &= g(\mathbf{x},t), && \mathbf{x} \in \partial\Omega, && t \in [0,T], \\ u &= u_0(\mathbf{x}), && \mathbf{x} \in \Omega, && t = 0, \\ u_t &= 0, && \mathbf{x} \in \Omega, && t = 0, \end{aligned} \tag{85}$$

where $g(\mathbf{x},t)$ and $u_0(\mathbf{x})$ are chosen so that u satisfies the standing-wave solution

$$u(\mathbf{x},t) = \sin(3\pi x)\sin(4\pi y)\cos(5\pi t). \tag{86}$$

The problem (85) is discretized using the SBP-P-SAT method described in Section 3, with interface conditions imposed using the hybrid projection and SAT method and the Dirichlet boundary conditions imposed using the projection method.

In Table 1 the L_2-error and convergence rate at final time $T = 1$ is presented for the 4th- and 6th-order accurate SBP operators, where each row corresponds to a different number of total grid points N. With the 4th-order accurate operators we obtain the convergence rate 4 while with the 6th-order accurate operators the convergence rate is slightly above 5, which is in line with previous observations [22].

5.2. Bathymetry optimization

We now consider the optimization problem (2). The domain is given by Fig. 3 and the wave speed is $c = 1$. The source is located at $[0.25, 0.8]$ with the time-dependent function given by the Ricker wavelet function

$$f(t) = \frac{2}{\sqrt{3\sigma}\pi^{1/4}}\left(1 - \left(\frac{t}{\sigma}\right)^2\right)e^{-\frac{t^2}{2\sigma^2}}, \tag{87}$$

Table 1
L_2-error and convergence rate with varying total degrees of freedom N for standing-wave problem with 4th- and 6th-order accurate SBP operators.

N	$\log_{10}(e_4)$	q_4	$\log_{10}(e_6)$	q_6
4797	-2.98	-	-3.25	-
10553	-3.71	-4.29	-4.24	-5.77
18549	-4.22	-4.18	-4.94	-5.68
28381	-4.61	-4.21	-5.47	-5.77
40777	-4.93	-4.10	-5.86	-5.04
72289	-5.44	-4.08	-6.57	-5.68
112761	-5.83	-4.05	-7.08	-5.30

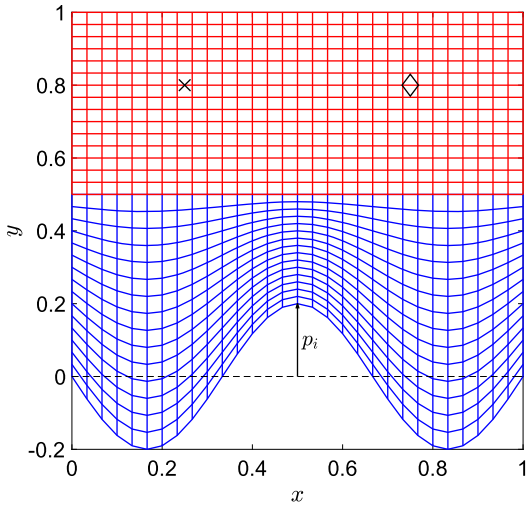


Fig. 3. Discretization of bathymetry problem with parameterization of the seabed. The different colors of the grid indicate the block decomposition. The cross (×) and diamond (◊) indicate the location of the source and the receiver, respectively.

with $\sigma = 0.1$. The receiver is located at $[0.75, 0.8]$. We use synthetic receiver data, produced by simulating the forward problem with a bottom boundary (seabed) given in Fig. 3, using the 6th-order accurate operators with $m_\xi^+ = m_x^- = 401$ and $m_\eta^+ = m_y^- = 201$ (total degrees of freedom $N = 161202$). In the optimization, we use the 4th-order operators and $m_\xi^+ = m_x^- = 41$ and $m_\eta^+ = m_y^- = 21$ (total degrees of freedom $N = 1722$). Linear interpolation of the synthetic receiver data is used when there is no data matching the time level of the numerical solution. The final time is set to $T = 4$ and the regularization parameter to $\gamma = 10^{-5}$. The initial guess of the seabed is chosen as a straight line, i.e. $\mathbf{p} = [0, 0, ..., 0]$.

We begin by investigating the accuracy of the gradient (71) by comparing it to the more common, but expensive, finite difference approach. Let

$$(D^+\mathcal{J}(\mathbf{p}))_i = \frac{\mathcal{J}(\mathbf{p} + \Delta p \mathbf{e}_i) - \mathcal{J}(\mathbf{p})}{\Delta p}, \tag{88}$$

denote the i:th component of the gradient vector approximated using first-order finite differences, where $\mathbf{e}_i$ is the vector with a 1 at the i:th element and zeroes elsewhere and Δp is the step size of finite difference approximation. Further, let

$$e(\Delta p) = \frac{\|\nabla_{\mathbf{p}}\mathcal{J}(\mathbf{p}) - D^+\mathcal{J}(\mathbf{p})\|_2}{\|\nabla_{\mathbf{p}}\mathcal{J}(\mathbf{p})\|_2}, \tag{89}$$

denote the relative L_2-error of the finite difference gradient assuming that $\nabla_{\mathbf{p}}\mathcal{J}(\mathbf{p})$ is exact. In Fig. 4, the error $e(\Delta p)$ is plotted against Δp for various CFL constants k. The results show that the error initially decreases with a first-order rate, but for small enough Δp it plateaus at a constant error that decreases for smaller time steps. Hence we have a discretization error in the gradient arising from the temporal discretization. For $k = 0.1$, which is what we use in the optimization, the relative error in the gradient is approximately 1%. For $k = 0.001$ and $k = 0.0001$ and very small Δp we also start seeing the effects of cancellation errors in the finite difference gradient.

We now turn to the actual optimization. In Fig. 5 the shape of the seabed after 0, 5, 20, and 177 iterations are presented. After 177 iterations the optimization stops with the chosen tolerance 10^{-6}. We can conclude that the method manages to reconstruct the shape of the seabed accurately with only one source and one receiver, and that a 1% error in the gradient seems sufficiently small for this problem.

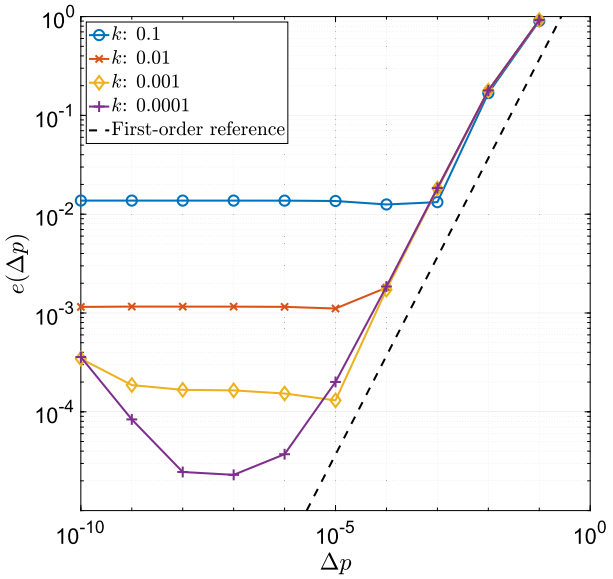


Fig. 4. Relative error between gradient approximated using finite differences and gradient computed using the adjoint method for varying CFL constants k.

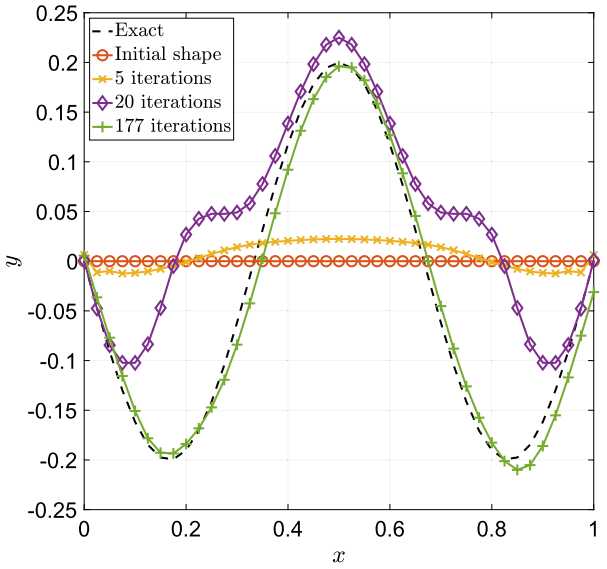


Fig. 5. Shape of the seabed after 0, 5, 20, and 177 iterations.

5.3. Air horn shape optimization

For the final numerical example, we use a similar setup as in [51], where the shape of the mouth of an acoustic horn is optimized to minimize the reflected sound, see Fig. 6. The problem setup, including the domain decomposition and boundary conditions, is presented in Fig. 7. On the walls of the horn, Γ_W and $\Gamma_{H,p}$, we prescribe fully reflecting homogeneous Neumann boundary conditions. At the outflow Γ_O we use first-order accurate outflow boundary conditions. To reduce the computational costs by half, we utilize the horizontal symmetry and prescribe symmetry boundary conditions (homogeneous Neumann conditions) at Γ_S. Note that $\Gamma_{H,p}$ is parametrized by the optimization parameter p. At the inflow boundary Γ_I, a time-dependent inflow function is prescribed using inhomogeneous Dirichlet boundary conditions. The wave speed is $c = 340$ m/s and the final time $T = 0.4$ s.

The main difference between [51] and the present work is that we solve the acoustic wave equation in the time domain, whereas in [51] the optimization is performed in the frequency domain by solving Helmholtz equation. The advantage of our approach is that we can use any time-dependent inflow function we wish with very small effects on the computational costs, whereas in the frequency domain the Helmholtz equation must be solved one frequency at a time for complex signals. However, for simple inflow signals, the

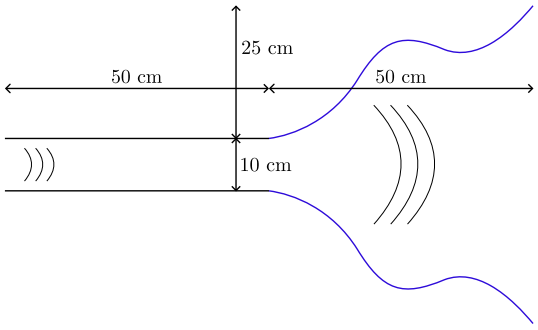
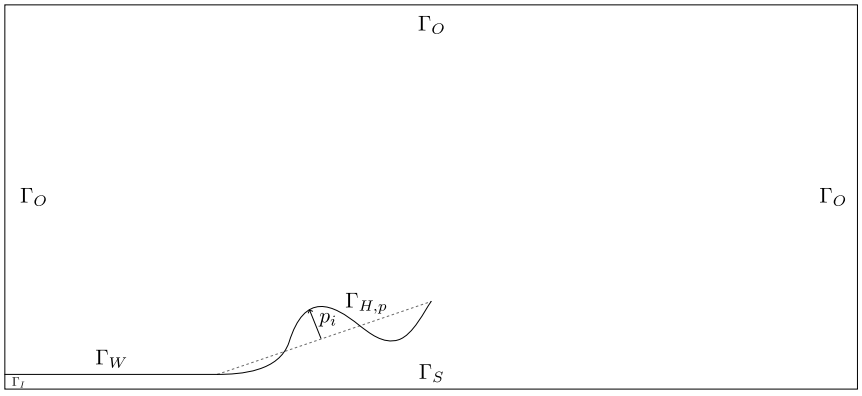
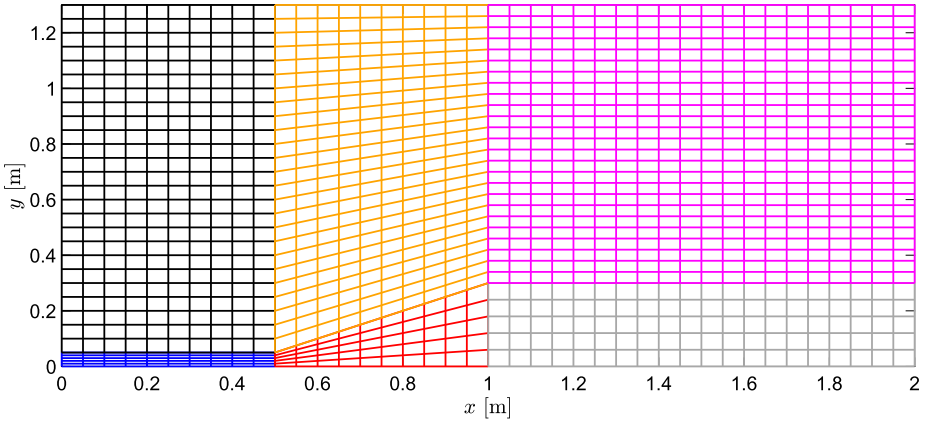


Fig. 6. Dimensions of the acoustic horn. The blue section of the horn is subject to shape optimization.



(a) Boundary conditions and horn parameterization.



(b) Coarse grid of the acoustic horn problem. The different colors indicate the block decomposition.

Fig. 7. Discretization of the acoustic horn problem.

approach taken in [51] is likely to be the more computational efficient option. In this work we present results for two inflow functions at Γ_I, first a single-frequency 300 Hz signal and then a broadband signal consisting of frequencies between 300 Hz and 400 Hz.

Designing suitable objective functions for optimization problems is a non-trivial task. In [51], a specific inflow boundary condition is derived that allows for straightforward computation of reflected waves, i.e. the left-going waves in the horn. In the optimization process, the mouth of the horn is modified to minimize the amplitudes of these waves. Here we take inspiration from [51] and design an objective function that also indicates left-going waves in the horn, but adapted for time-domain discretizations. This is achieved by minimizing the outgoing characteristic $u_t - c\mathbf{n} \cdot \nabla u$ integrated over the inflow boundary Γ_I and time. Assuming that the flow in the channel from the inflow boundary to the mouth of the horn is predominately one-dimensional, the boundary integral $\|u_t - c\mathbf{n} \cdot \nabla u\|^2_{\Gamma_I}$

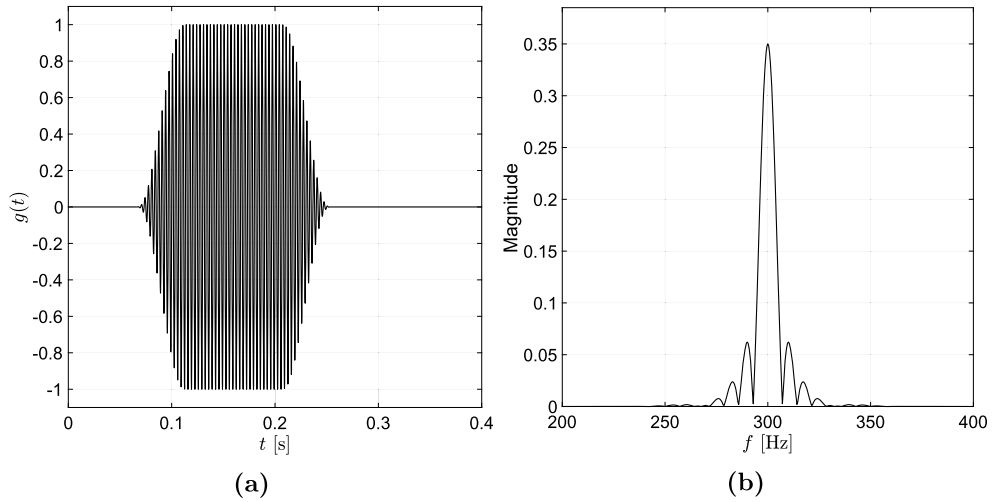


Fig. 8. Single frequency 300 Hz input signal in time domain **(a)** and frequency domain **(b)**.

measures the outgoing energy flux at Γ_I. Therefore, minimizing $\int_0^T \|u_t - c\mathbf{n}\cdot\nabla u\|^2_{\Gamma_I} dt$, minimizes the energy attributed to reflected waves in the horn. The unregularized optimization problem in the continuous setting then reads

$$\min_p \mathrm{J}(u,p) = \frac{1}{2}\int_0^T \|u_t - c\mathbf{n}\cdot\nabla u\|^2_{\Gamma_I}\, dt, \tag{90a}$$

such that

$$\begin{aligned}
u_{tt} &= c^2\Delta u, && \mathbf{x}\in\Omega_p, && t\in[0,T],\\
\mathbf{n}\cdot\nabla u &= 0, && \mathbf{x}\in\Gamma_W,\Gamma_{H,p},\Gamma_S, && t\in[0,T],\\
u &= g(t), && \mathbf{x}\in\Gamma_I, && t\in[0,T],\\
u_t + c\mathbf{n}\cdot\nabla u &= 0, && \mathbf{x}\in\Gamma_O, && t\in[0,T],\\
u = 0,\quad u_t &= 0, && \mathbf{x}\in\Omega_p, && t=0,
\end{aligned} \tag{90b}$$

where $g(t)$ is the inflow function.

The shape of the mouth of the horn is parametrized by the deviation from a straight line between the two fixed points $(0.5, 0.05)$ and $(1, 0.3)$, see Fig. 7a. The problem (90) is discretized using the SBP-P-SAT method described in Section 3, generalized to the multiblock setting in Fig. 7. As for the bathymetry problem, the adjoint problem to (90b) is derived which allows for efficient computation of the gradient of the loss function $\mathcal{J}$. To regularize the optimization problem we add an additional term to $\mathcal{J}$ as described in Section 4.1, with regularization parameter $\gamma = 10^{-5}$. To retain space we refrain from presenting the discretization in detail.

Two numerical experiments are performed to evaluate the method. First, we consider the problem with a single frequency input signal truncated in time. Explicitly, the input data is given by

$$g(t) = \sigma(t - t_0)\sin(2\pi f t), \tag{91}$$

where $f = 300$ Hz, $t_0 = 0.16$, and $\sigma(t)$ is a tapering window function given by Equation (7) in [52], with $\epsilon = 0.3$ (smoothness) and $T = 0.2$ (size of window). The input signal $g(t)$ is plotted in time and in frequency domain in Fig. 8. The optimal solution (with tolerance 10^{-6}) is found after 70 iterations. In Fig. 9a the shape of the horn after 30 and 70 iterations is shown. The solution to the forward problem with the optimized shape at $t = 0.08$ s, $t = 0.15$ s, $t = 0.2$ s, and $t = 0.35$ s is plotted in Fig. 10. Note that almost no wave propagation is present in the domain at $t = 0.35$ s. To evaluate the properties of the optimized horn the unregularized loss (given in (90a)) is computed for a range of input signals, one frequency at a time, and plotted in Fig. 9b. The results show that the optimized shape decreases the reflected sound at 300 Hz by more than four orders of magnitude compared to the initial shape.

For the second experiment, we optimize the horn for a broadband signal consisting of frequencies between 300 Hz and 400 Hz. The signal is constructed in time domain utilizing sinc functions multiplied by a Gaussian pulse, which results in a near-rectangular function in the frequency domain, see Fig. 11. Explicitly, the input data is given by

$$g(t) = \frac{2}{f_r - f_l} e^{-(\frac{t-t_0}{\sigma})^2} \left(f_l \,\mathrm{sinc}\left(2f_l(t-t_0)\right) - f_r \,\mathrm{sinc}\left(2f_r(t-t_0)\right)\right), \tag{92}$$

where $t_0 = 0.2$, $\sigma = 0.075$, $f_l = 300$ Hz, $f_r = 400$ Hz, and

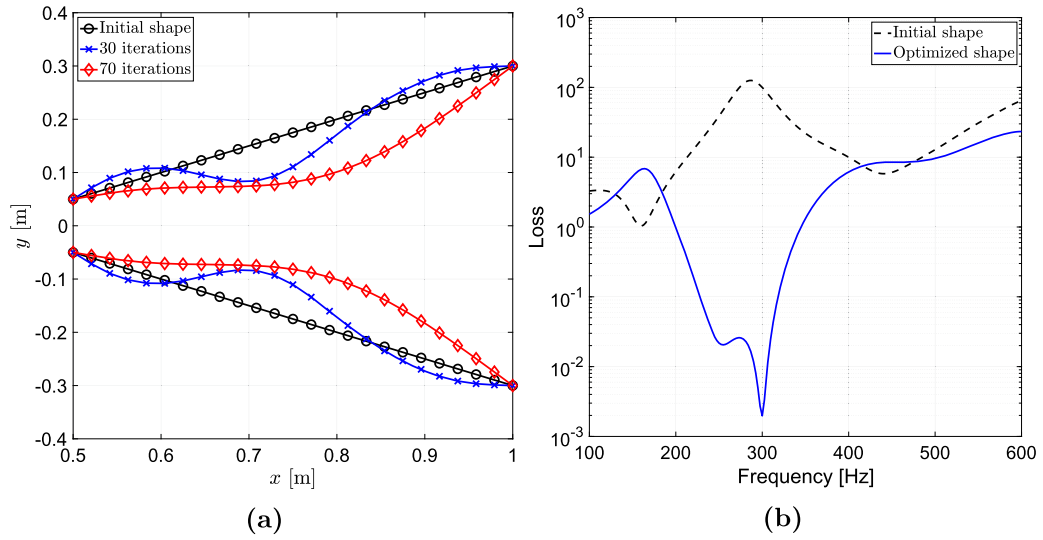


Fig. 9. Results optimizing for 300 Hz. **(a)** Shape of the horn for different optimization iterations. **(b)** Unregularized loss (90a) for varying frequencies.

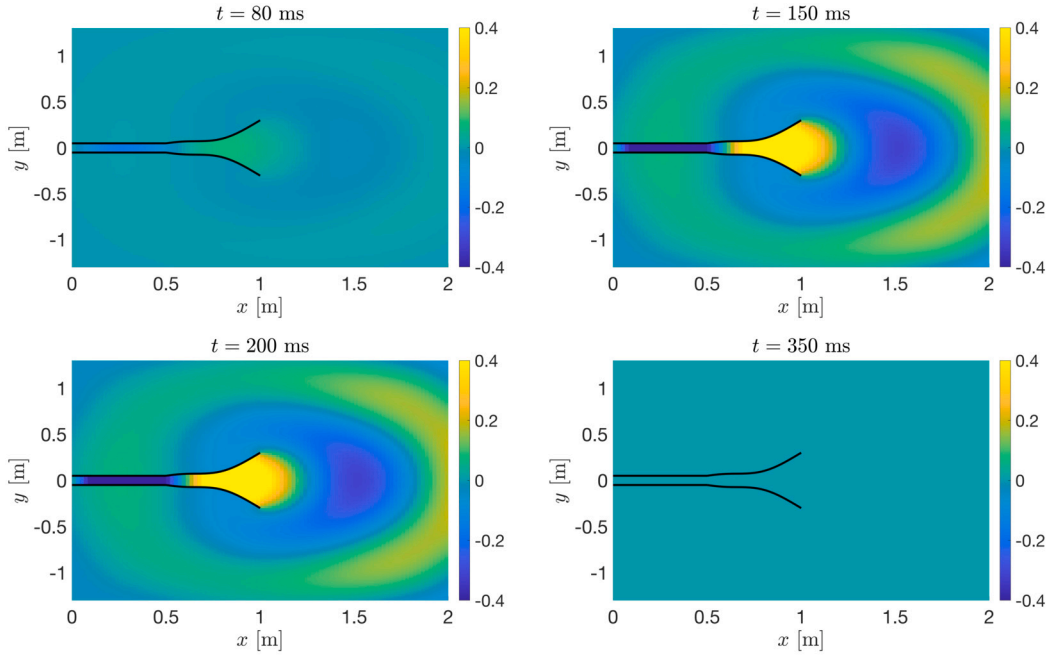


Fig. 10. Snapshots of the solution to the forward problem with the air horn optimized for 300 Hz.

$$\operatorname{sinc}(t) = \begin{cases} \frac{\sin(\pi t)}{\pi t} & t \neq 0 \\ 1 & t = 0. \end{cases} \tag{93}$$

Note that the computational cost of evaluating the loss and gradient with the broadband input signal is almost identical to the costs of that of a single frequency. This is one advantage of solving the problem in the time domain. The optimal solution (with tolerance 10^{-6}) is found after 76 iterations. In Fig. 12a and 12b the shape of the horn after 30 and 76 iterations and the unregularized loss for varying frequencies are presented, respectively. Once again we can observe that the reflected sound in the frequencies of interest is smaller by approximately two orders of magnitude for the horn with an optimized shape as compared to the initial shape.

6. Conclusions

In this paper, we present a method for solving shape optimization problems constrained by the acoustic wave equation using energy stable finite differences of high-order accuracy with SBP properties. The optimization method is based on the inversion of

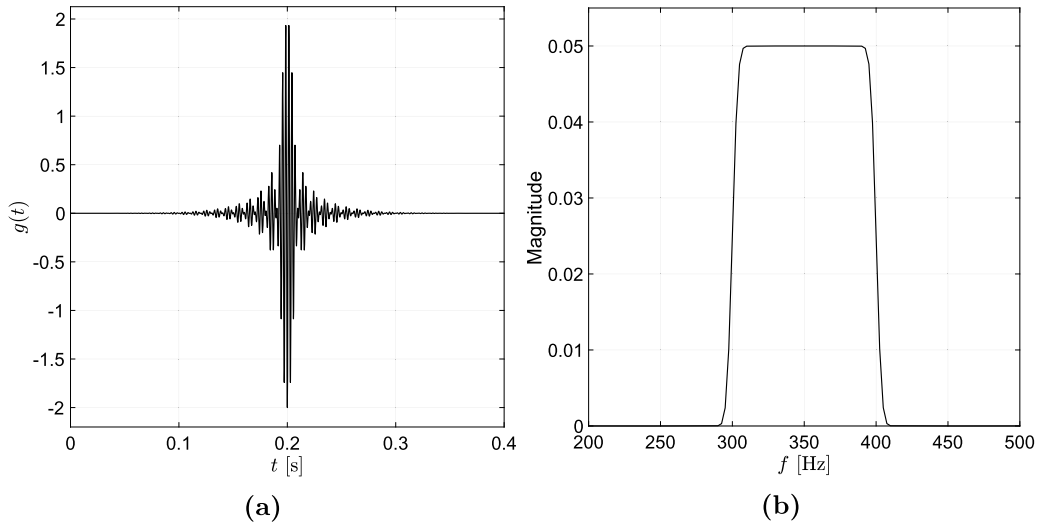


Fig. 11. Broadband input signal in time domain **(a)** and frequency domain **(b)**.

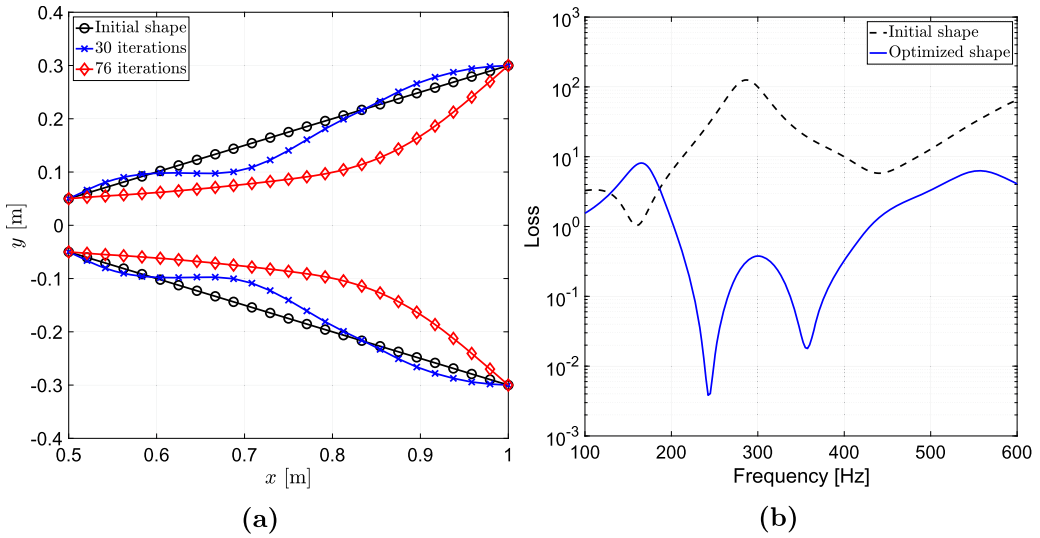


Fig. 12. Results optimizing for broadband signal consisting of frequencies between 300 Hz and 400 Hz. **(a)** Shape of the horn for different optimization iterations. **(b)** Unregularized loss (90a) for varying frequencies.

coordinate transformations, where the design domain is transformed to a rectangular reference domain and the resulting metric coefficients are optimized to minimize the loss functional. A gradient-based optimization algorithm is used, computing the gradient of the loss functional through the adjoint framework. From a numerical point of view, the problem is similar to full waveform inversion in seismic imaging, where the objective is to find material parameters in the medium. The problem is solved in the time domain, meaning that any time-dependent source and receiver data can be used. Using a combination of weak and strong imposition of boundary conditions through the SAT and projection methods, the scheme of the forward problem is shown to be dual consistent.

Three numerical experiments are performed. First, the accuracy of the forward scheme is verified, demonstrating that the expected convergence rates are obtained. In the second numerical experiment, we show that the method can be used to reconstruct the shape of a seabed with only one source and one receiver close to the surface. In the final numerical experiment, we optimize the shape of the mouth of an air horn with the objective of minimizing reflected sound in the horn. For this last example, a complicated source function is used consisting of many frequencies to emphasize the advantages of solving the problem in the time domain.

Interesting topics for future research include, e.g., extensions of the method to three spatial dimensions and the impact of more sophisticated time-stepping schemes on the convergence behavior of the optimization problem. From a theoretical standpoint, the extension to 3D is straightforward, the issues lie in the efficient implementation of the solver and the computation of the gradient. Similarly, time integration using a self-adjoint method, e.g., the Runge–Kutta methods presented in [48] or SBP in time [49], is mostly an implementation challenge. The error study of the adjoint-based gradient in Section 5.2 shows that choosing the time step

a factor of 10 below the CFL stability limit (i.e., setting CFL constant $k = 0.1$) results in a 1% relative error attributed to the time discretization error. Moreover, the error in the gradient seems to decrease linearly with k, implying that time steps close to the stability limit will result in approximately 10% relative error. Using a self-adjoint time integrator could therefore allow for larger time steps without incurring an error. For self-adjoint Runge–Kutta methods, this entails evaluating the time integral in the adjoint-based gradient using the Runge–Kutta quadrature, thereby requiring storing also the Runge–Kutta stage values of the forward and adjoint solutions, resulting in increased memory costs [48]. Whether the benefits of obtaining the exact gradient to the fully discrete optimization problem through dual consistency in space and time outweigh the benefits in computational costs of the simpler methods remains to be answered.

CRediT authorship contribution statement

Gustav Eriksson: Conceptualization, Formal analysis, Investigation, Methodology, Software, Visualization, Writing – original draft, Writing – review & editing. **Vidar Stiernström:** Conceptualization, Investigation, Writing – original draft, Writing – review & editing, Formal analysis, Methodology.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

Acknowledgements

G. Eriksson was supported by Formas (Grant No. 2018-00925). V. Stiernström was supported by the Swedish Research Council (grant 2017-04626 VR).

The computations were enabled by resources provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS) and the Swedish National Infrastructure for Computing (SNIC) at UPPMAX partially funded by the Swedish Research Council through grant agreements no. 2022-06725 and no. 2018-05973.

Appendix A. Operator derivatives

Note that all operators in Ω^+ are independent of $\mathbf{p}$, hence we only have to consider the operators in Ω_p^-. We shall begin with the derivatives of the coordinates of the grid points in Ω_p^-. Using (68), we get

$$\begin{aligned}\frac{\partial x_{i,j}}{\partial p_i} &= 0,\\ \frac{\partial y_{i,j}}{\partial p_i} &= \mathbf{e}_i - \mathbf{e}_i \eta_j,\end{aligned} \tag{A.1}$$

where $\mathbf{e}_i$ is the i:th column of $I_{m_{\Gamma_I}}$. We also have the metric derivatives

$$\begin{aligned}\frac{\partial \mathbf{X}_\xi}{\partial p_i} &= \mathtt{diag}(D_\xi^- \frac{\partial \mathbf{x}}{\partial p_i}), \quad \frac{\partial \mathbf{Y}_\xi}{\partial p_i} = \mathtt{diag}(D_\xi^- \frac{\partial \mathbf{y}}{\partial p_i}),\\ \frac{\partial \mathbf{X}_\eta}{\partial p_i} &= \mathtt{diag}(D_\eta^- \frac{\partial \mathbf{x}}{\partial p_i}), \quad \frac{\partial \mathbf{Y}_\eta}{\partial p_i} = \mathtt{diag}(D_\eta^- \frac{\partial \mathbf{y}}{\partial p_i}),\end{aligned} \tag{A.2}$$

and metric coefficients

$$\begin{aligned}\frac{\partial \mathbf{J}}{\partial p_i} &= \frac{\partial \mathbf{X}_\xi}{\partial p_i}\mathbf{Y}_\eta + \mathbf{X}_\xi \frac{\partial \mathbf{Y}_\eta}{\partial p_i} - \frac{\partial \mathbf{X}_\eta}{\partial p_i}\mathbf{Y}_\xi - \mathbf{X}_\eta \frac{\partial \mathbf{Y}_\xi}{\partial p_i},\\ \frac{\partial \alpha_1}{\partial p_i} &= -\frac{\partial \mathbf{J}}{\partial p_i}\mathbf{J}^{-2}(\mathbf{X}_\eta^2 + \mathbf{Y}_\eta^2) + 2\mathbf{J}^{-1}(\frac{\partial \mathbf{X}_\eta}{\partial p_i}\mathbf{X}_\eta + \frac{\partial \mathbf{Y}_\eta}{\partial p_i}\mathbf{Y}_\eta),\\ \frac{\partial \beta}{\partial p_i} &= \frac{\partial \mathbf{J}}{\partial p_i}\mathbf{J}^{-2}(\mathbf{X}_\xi\mathbf{X}_\eta + \mathbf{Y}_\xi\mathbf{Y}_\eta) - \mathbf{J}^{-1}(\frac{\partial \mathbf{X}_\xi}{\partial p_i}\mathbf{X}_\eta + \frac{\partial \mathbf{Y}_\xi}{\partial p_i}\mathbf{Y}_\eta + \mathbf{X}_\xi\frac{\partial \mathbf{X}_\eta}{\partial p_i} + \mathbf{Y}_\xi\frac{\partial \mathbf{Y}_\eta}{\partial p_i}),\\ \frac{\partial \alpha_2}{\partial p_i} &= -\frac{\partial \mathbf{J}}{\partial p_i}\mathbf{J}^{-2}(\mathbf{X}_\xi^2 + \mathbf{Y}_\xi^2) + 2\mathbf{J}^{-1}(\frac{\partial \mathbf{X}_\xi}{\partial p_i}\mathbf{X}_\xi + \frac{\partial \mathbf{Y}_\xi}{\partial p_i}\mathbf{Y}_\xi).\end{aligned} \tag{A.3}$$

Using this, we can construct the derivatives of the operators in Ω_p^- as follows

$$\begin{aligned}\frac{\partial D_L^-}{\partial p_i} &= -\frac{\partial \mathbf{J}}{\partial p_i}\mathbf{J}^{-2}(D_{\xi\xi}^{(\alpha_1)} + D_\eta^-\beta D_\xi^- + D_\xi^-\beta D_\eta^- + D_{\eta\eta}^{(\alpha_2)}) \\ &\quad + J^{-1}(D_{\xi\xi}^{(\frac{\partial \alpha_1}{\partial p_i})} + D_\eta^-\frac{\partial \beta}{\partial p_i}D_\xi^- + D_\xi^-\frac{\partial \beta}{\partial p_i}D_\eta^- + D_{\eta\eta}^{(\frac{\partial \alpha_2}{\partial p_i})}),\end{aligned} \tag{A.4}$$

where we have used that

$$\frac{\partial D_2^{(\mathbf{c}(\mathbf{p}))}}{\partial p_i} = H^{-1}(-M^{(\frac{\partial \mathbf{c}(\mathbf{p})}{\partial p_i})} - e_l e_l^\top \frac{\partial \mathbf{c}}{\partial p_i} d_l^\top + e_r e_r^\top \frac{\partial \mathbf{c}}{\partial p_i} d_r^\top) = D_2^{(\frac{\partial \mathbf{c}(\mathbf{p})}{\partial p_i})}, \tag{A.5}$$

since

$$\frac{\partial M^{(\mathbf{c}(\mathbf{p}))}}{\partial p_i} = D_1^\top H \frac{\partial \bar{\mathbf{c}}}{\partial p_i} D_1 + R^{(\frac{\partial \mathbf{c}(\mathbf{p})}{\partial p_i})} = M^{(\frac{\partial \mathbf{c}(\mathbf{p})}{\partial p_i})}, \tag{A.6}$$

and

$$\frac{\partial R^{(\mathbf{c}(\mathbf{p}))}}{\partial p_i} = R^{(\frac{\partial \mathbf{c}(\mathbf{p})}{\partial p_i})}, \tag{A.7}$$

for any $\mathbf{p}$-dependent vector $\mathbf{c}(\mathbf{p})$, due to the specific structure of $R^{(\mathbf{c}(\mathbf{p}))}$ (see, e.g., [19,21,20]). We also have

$$\begin{aligned}\frac{\partial d_w^-}{\partial p_i} &= e_w^- \frac{\partial \mathbf{W}_2}{\partial p_i}\mathbf{W}_2^{-2}(\alpha_1 e_w^{-\top} e_w^- D_\xi^- + \beta e_w^{-\top} e_w^- D_\eta^-) \\ &\quad - e_w^- \mathbf{W}_2^{-1}(\frac{\partial \alpha_1}{\partial p_i} e_w^{-\top} e_w^- D_\xi^- + \frac{\partial \beta}{\partial p_i} e_w^{-\top} e_w^- D_\eta^-), \\ \frac{\partial d_e^-}{\partial p_i} &= -e_e^- \frac{\partial \mathbf{W}_2}{\partial p_i}\mathbf{W}_2^{-2}(\alpha_1 e_e^{-\top} e_e^- D_\xi^- + \beta e_e^{-\top} e_e^- D_\eta^-) \\ &\quad + e_e^- \mathbf{W}_2^{-1}(\frac{\partial \alpha_1}{\partial p_i} e_e^{-\top} e_e^- D_\xi^- + \frac{\partial \beta}{\partial p_i} e_e^{-\top} e_e^- D_\eta^-), \\ \frac{\partial d_s^-}{\partial p_i} &= e_s^- \frac{\partial \mathbf{W}_1}{\partial p_i}\mathbf{W}_1^{-2}(\alpha_2 e_s^{-\top} e_s^- D_\eta^- + \beta e_s^{-\top} e_s^- D_\xi^-) \\ &\quad - e_s^- \mathbf{W}_1^{-1}(\frac{\partial \alpha_2}{\partial p_i} e_s^{-\top} e_s^- D_\eta^- + \frac{\partial \beta}{\partial p_i} e_s^{-\top} e_s^- D_\xi^-), \\ \frac{\partial d_n^-}{\partial p_i} &= -e_n^- \frac{\partial \mathbf{W}_1}{\partial p_i}\mathbf{W}_1^{-2}(\alpha_2 e_n^{-\top} e_n^- D_\eta^- + \beta e_n^{-\top} e_n^- D_\xi^-) \\ &\quad + e_n^- \mathbf{W}_1^{-1}(\frac{\partial \alpha_2}{\partial p_i} e_n^{-\top} e_n^- D_\eta^- + \frac{\partial \beta}{\partial p_i} e_n^{-\top} e_n^- D_\xi^-),\end{aligned} \tag{A.8}$$

with

$$\frac{\partial \mathbf{W}_1}{\partial p_i} = \frac{\frac{\partial \mathbf{X}_\xi}{\partial p_i}\mathbf{X}_\xi + \frac{\partial \mathbf{Y}_\xi}{\partial p_i}\mathbf{Y}_\xi}{\sqrt{\mathbf{X}_\xi^2 + \mathbf{Y}_\xi^2}} \quad \text{and} \quad \frac{\partial \mathbf{W}_2}{\partial p_i} = \frac{\frac{\partial \mathbf{X}_\eta}{\partial p_i}\mathbf{X}_\eta + \frac{\partial \mathbf{Y}_\eta}{\partial p_i}\mathbf{Y}_\eta}{\sqrt{\mathbf{X}_\eta^2 + \mathbf{Y}_\eta^2}}, \tag{A.9}$$

and the differentiated inner product matrices

$$\frac{\partial \bar{H}}{\partial p_i} = \begin{bmatrix} 0 & 0 \\ 0 & \frac{\partial H^-}{\partial p_i} \end{bmatrix}, \quad \text{with} \quad \frac{\partial H^-}{\partial p_i} = H_\xi^- H_\eta^- \frac{\partial J}{\partial p_i}, \tag{A.10}$$

and

$$\begin{aligned}\frac{\partial H_w^-}{\partial p_i} &= H e_w \frac{\partial \mathbf{W}_2}{\partial p_i} e_w^\top, \quad \frac{\partial H_e^-}{\partial p_i} = H e_e \frac{\partial \mathbf{W}_2}{\partial p_i} e_e^\top, \\ \frac{\partial H_s^-}{\partial p_i} &= H e_s \frac{\partial \mathbf{W}_1}{\partial p_i} e_s^\top, \quad \frac{\partial H_n^-}{\partial p_i} = H e_n \frac{\partial \mathbf{W}_1}{\partial p_i} e_n^\top.\end{aligned} \tag{A.11}$$

Using this, we get

$$\begin{aligned}\frac{\partial SAT_{BC_1}}{\partial p_i} &= \frac{\partial \bar{H}}{\partial p_i}\bar{H}^{-2}\left(\sum_{k=w,e}\begin{bmatrix} e_k^{+\top}H_k^+ d_k^+ & 0 \\ 0 & e_k^{-\top}H_k^- d_k^- \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & e_s^{-\top}H_s^- d_s^- \end{bmatrix}\right) \\ &- \bar{H}^{-1}\sum_{k=w,e}\begin{bmatrix} 0 & 0 \\ 0 & e_k^{-\top}(\frac{\partial H_k^-}{\partial p_i}d_k^- + H_k^-\frac{\partial d_k^-}{\partial p_i}) \end{bmatrix} \\ &- \bar{H}^{-1}\begin{bmatrix} 0 & 0 \\ 0 & e_s^{-\top}(\frac{\partial H_s^-}{\partial p_i}d_s^- + H_s^-\frac{\partial d_s^-}{\partial p_i}) \end{bmatrix},\end{aligned} \tag{A.12}$$

$$\begin{aligned}\frac{\partial SAT_{BC_2}}{\partial p_i} &= \frac{\partial \bar{H}}{\partial p_i}\bar{H}^{-2}\sum_{k=w,e}\begin{bmatrix} e_k^{+\top}H_k^+ e_k^+ & 0 \\ 0 & e_k^{-\top}H_k^- e_k^- \end{bmatrix} \\ &- \bar{H}^{-1}\sum_{k=w,e}\begin{bmatrix} 0 & 0 \\ 0 & e_k^{-\top}\frac{\partial H_k^-}{\partial p_i}e_k^- \end{bmatrix},\end{aligned} \tag{A.13}$$

$$\begin{aligned}\frac{\partial SAT_{IC}}{\partial p_i} &= \frac{\partial \bar{H}}{\partial p_i}\bar{H}^{-2}\begin{bmatrix} e_s^{+\top}H_s^+ d_s^+ & e_s^{+\top}H_s^+ d_n^- \\ 0 & 0 \end{bmatrix} \\ &- \bar{H}^{-1}\begin{bmatrix} 0 & e_s^{+\top}H_s^+\frac{\partial d_n^-}{\partial p_i} \\ 0 & 0 \end{bmatrix},\end{aligned} \tag{A.14}$$

and

$$\begin{aligned}\frac{\partial P}{\partial p_i} &= \frac{\partial \bar{H}}{\partial p_i}\bar{H}^{-2}L^\top(L\bar{H}^{-1}L^\top)^{-1}L \\ &- \bar{H}^{-1}L^\top(L\bar{H}^{-1}L^\top)^{-1}L\frac{\partial \bar{H}}{\partial p_i}\bar{H}^{-2}L^\top(L\bar{H}^{-1}L^\top)^{-1}L \\ &= P\frac{\partial \bar{H}}{\partial p_i}\bar{H}^{-1}(I-P).\end{aligned} \tag{A.15}$$

We are now ready to write down the equations for $\frac{\partial D}{\partial p_i}$ and $\frac{\partial E}{\partial p_i}$, we have

$$\begin{aligned}\frac{\partial D}{\partial p_i} &= c^2\frac{\partial P}{\partial p_i}\left(\begin{bmatrix} D_L^+ & 0 \\ 0 & D_L^- \end{bmatrix} + SAT_{BC_1} + SAT_{IC}\right)P \\ &+ c^2 P\left(\begin{bmatrix} 0 & 0 \\ 0 & \frac{\partial D_L^-}{\partial p_i} \end{bmatrix} + \frac{\partial SAT_{BC_1}}{\partial p_i} + \frac{\partial SAT_{IC}}{\partial p_i}\right)P \\ &+ c^2 P\left(\begin{bmatrix} D_L^+ & 0 \\ 0 & D_L^- \end{bmatrix} + SAT_{BC_1} + SAT_{IC}\right)\frac{\partial P}{\partial p_i},\end{aligned} \tag{A.16}$$

and

$$\frac{\partial E}{\partial p_i} = c\frac{\partial P}{\partial p_i}SAT_{BC_2}P + cP\frac{\partial SAT_{BC_2}}{\partial p_i}P + cPSAT_{BC_2}\frac{\partial P}{\partial p_i}. \tag{A.17}$$

Remark 3. The specific form of $\frac{\partial x_{i,j}}{\partial p_i}$ and $\frac{\partial y_{i,j}}{\partial p_i}$ in (A.1) would allow us to significantly simplify the expressions in (A.2)-(A.17). But, for clarity and completeness, we present the derivatives of the operators for a general mapping.

Appendix B. Proof of Lemma 5

Start by considering the first variation of $\mathtt{J}$ in (2) with respect to the parameterization p. We have,

$$\begin{aligned}
\delta \mathtt{J} &= \delta\left(\frac{1}{2}\int_0^T r^2\, dt\right) \\
&= \int_0^T \int_{\Omega^+} r\frac{\delta r(u^+,\mathbf{x}_r,t)}{\delta u^+(\mathbf{x},t)}\delta u^+(\mathbf{x},t)\, d\mathbf{x}dt \\
&= \int_0^T \int_{\Omega^+} r\frac{\delta u^+(\mathbf{x}_r,t)}{\delta u^+(\mathbf{x},t)}\delta u^+(\mathbf{x},t)\, d\mathbf{x}dt \\
&= \int_0^T \int_{\Omega^+} r\hat{\delta}(\mathbf{x}-\mathbf{x}_r)\delta u^+(\mathbf{x},t)\, d\mathbf{x}dt \\
&= \int_0^T (r\hat{\delta}(\mathbf{x}-\mathbf{x}_r),\delta u^+)_{\Omega^+}\, dt,
\end{aligned} \tag{B.1}$$

where $r = r(u^+,\mathbf{x}_r,t) = u^+(\mathbf{x}_r,t) - u_d(t)$. The Lagrangian loss functional of (2) is

$$\mathtt{L} = \mathtt{J} + \int_0^T (\lambda^+, u_{tt}^+ - c^2\Delta u^+ - f(t)\hat{\delta}(\mathbf{x}-\mathbf{x}_s)_{\Omega^+} + (\lambda^-, u_{tt}^- - c^2\Delta u^-))_{\Omega^-}\, dt. \tag{B.2}$$

Note that $\mathtt{L} = \mathtt{J}$ such that $\delta\mathtt{L} = \delta\mathtt{J}$ for any $u^\pm$ satisfying (2b), and we therefore proceed with analyzing $\delta\mathtt{L}$. As a first step (B.2) is recast into a suitable form through integration by parts, starting with the contribution from Ω^+. Note that $\delta f(t)\hat{\delta}(\mathbf{x}-\mathbf{x_s}) = 0$ (since Ω^+ is independent of p) and we therefore disregard the point source in the following analysis. Integrating the second term in (B.2) by parts in time and space twice results in

$$\int_0^T (\lambda^+, u_{tt}^+ - c^2\Delta u^+)_{\Omega^+}\, dt = VT^+ + IT^+, \tag{B.3}$$

where

$$VT^+ = \int_0^T (\lambda_{tt}^+ - c^2\Delta\lambda^+, u^+)_{\Omega^+}\, dt, \tag{B.4}$$

are the volume terms and

$$IT^+ = \int_0^T -\langle\lambda^+, c^2\mathbf{n}^+\cdot\nabla u^+\rangle_{\delta\Omega^{(+,s)}} + \langle c^2\mathbf{n}^+\cdot\nabla\lambda^+, u^+\rangle_{\delta\Omega^{(+,s)}}\, dt, \tag{B.5}$$

are the interface terms. Any boundary terms vanish by the forward and adjoint initial and boundary conditions in (2b), (78). Due to the p-dependency of Ω_p^-, the third term in (B.2) requires a slightly different treatment. Integrating in time twice and space once and using the forward and adjoint initial and boundary conditions yields results in

$$\int_0^T (\lambda^-, u_{tt}^- - c^2\Delta u^-)_{\Omega_p^-}\, dt = VT^- + BT^- + IT^-, \tag{B.6}$$

where

$$VT^- = \int_0^T (\lambda_{tt}^-, u^-)_{\Omega_p^-} + (\nabla\lambda^-, c^2\nabla u^-)_{\Omega_p^-}\, dt, \tag{B.7}$$

$$BT^- = \int_0^T \langle c\lambda_t^-, u^-\rangle_{\partial\Omega_p^{(-,e)}} + \langle c\lambda_t^-, u^-\rangle_{\partial\Omega_p^{(-,w)}}\, dt, \tag{B.8}$$

and

$$IT^- = \int_0^T -\langle \lambda^-, c^2 \mathbf{n}^- \cdot \nabla u^- \rangle_{\partial\Omega_p^{(-,n)}} \, dt. \tag{B.9}$$

BT^- is obtained by using the boundary conditions $c\mathbf{n}^- \cdot \nabla u^- = u_t^-$ and integrating by parts in time, canceling boundary terms using the forward and adjoint initial conditions. Adding the interface terms (B.5) and (B.9) (exchanging $\partial\Omega^{(+,s)}$, $\partial\Omega_p^{(-,n)}$ for Γ_I) yields

$$\begin{aligned} IT &= IT^+ + IT^- \\ &= \int_0^T \Big(-\langle \lambda^+, c^2 \mathbf{n}^+ \cdot \nabla u^+ \rangle_{\Gamma_I} + \langle c^2 \mathbf{n}^+ \cdot \nabla \lambda^+, u^+ \rangle_{\Gamma_I} \\ &\qquad -\langle \lambda^-, c^2 \mathbf{n}^- \cdot \nabla u^- \rangle_{\Gamma_I} \Big) \, dt \\ &= \int_0^T \langle c^2 \mathbf{n}^+ \cdot \nabla \lambda^+, u^+ \rangle_{\Gamma_I} \, dt, \end{aligned} \tag{B.10}$$

where the last equality was obtained using the forward and adjoint interface conditions in (2b), (78). To summarize, we have arrived at

$$\mathrm{L} = \mathrm{J} + VT^+ + VT^- + BT^- + IT, \tag{B.11}$$

and will now consider $\delta\mathrm{L}$.

To start with, by (B.4) $\delta VT^+ = \int_0^T (\lambda_{tt}^+ - c^2 \Delta \lambda^+, \delta u^+)_{\Omega^+} \, dt$ such that

$$\delta(\mathrm{J} + VT^+) = \int_0^T (\lambda_{tt}^+ - c^2 \Delta \lambda^+ + r\hat{\delta}(\mathbf{x} - \mathbf{x_r}), \delta u^+)_{\Omega^+} \, dt = 0, \tag{B.12}$$

due to (B.1) and (78). To obtain δVT^-, we first transform (B.7) to the reference domain $\tilde{\Omega}^-$ using (9), (11) and (12), resulting in

$$\begin{aligned} VT^- = \int_0^T \Big(&(\lambda_{tt}^-, Ju^-)_{\tilde{\Omega}^-} + c^2 (\lambda_\xi^-, \alpha_1 u_\xi^- + \beta u_\eta^-)_{\tilde{\Omega}^-} \\ &+ c^2 (\lambda_\eta^-, \alpha_2 u_\eta^- + \beta u_\xi^-)_{\tilde{\Omega}^-} \Big) \, dt. \end{aligned} \tag{B.13}$$

Taking the first variation, applying the product rule and grouping terms containing δu^- and variations of metric terms (using that $\delta u_{\xi,\eta}^- = (\delta u^-)_{\xi,\eta}$), results in

$$\delta VT^- = VT_{\delta u}^- + VT_{\delta X}^-, \tag{B.14}$$

with

$$\begin{aligned} VT_{\delta u}^- = \int_0^T \Big(&(\lambda_{tt}^-, J\delta u^-)_{\tilde{\Omega}^-} + c^2 (\lambda_\xi^-, \alpha_1 (\delta u^-)_\xi + \beta (\delta u^-)_\eta)_{\tilde{\Omega}^-} \\ &+ c^2 (\lambda_\eta^-, \alpha_2 (\delta u^-)_\eta + \beta (\delta u^-)_\xi)_{\tilde{\Omega}^-} \Big) \, dt, \end{aligned} \tag{B.15}$$

and

$$\begin{aligned} VT_{\delta X}^- = \int_0^T \Big(&(\lambda_{tt}^-, \delta J u^-)_{\tilde{\Omega}^-} + c^2 (\lambda_\xi^-, \delta\alpha_1 u_\xi^- + \delta\beta u_\eta^-)_{\tilde{\Omega}^-} \\ &+ c^2 (\lambda_\eta^-, \delta\alpha_2 u_\eta^- + \delta\beta u_\xi^-)_{\tilde{\Omega}^-} \Big) \, dt. \end{aligned} \tag{B.16}$$

Integrating $VT_{\delta u}^-$ in (B.15) by parts, raising δu^-, and transforming back to the physical domain yields

$$\begin{aligned} VT_{\delta u}^- &= \int_0^T (\lambda_{tt}^- - c^2 \Delta \lambda^-, \delta u^-)_{\Omega_p^-} + \langle c^2 \mathbf{n}^- \cdot \nabla \lambda^-, \delta u^- \rangle_{\partial\Omega_p^-} \, dt \\ &= BT_{\delta u}^- + IT_{\delta u}^-, \end{aligned} \tag{B.17}$$

where (78) is used to cancel the volume and boundary terms such that the remaining terms are

$$BT^-_{\delta u} = \int_0^T \langle c^2 \mathbf{n}^- \cdot \nabla \lambda^-, \delta u^- \rangle_{\Omega_p^{(-,w)}} + \langle c^2 \mathbf{n}^- \cdot \nabla \lambda^-, \delta u^- \rangle_{\Omega_p^{(-,e)}} \, dt, \tag{B.18}$$

and

$$IT^-_{\delta u} = \int_0^T \langle c^2 \mathbf{n}^- \cdot \nabla \lambda^-, \delta u^- \rangle_{\Gamma_I} \, dt. \tag{B.19}$$

For $VT^-_{\delta X}$ in (B.16) we integrate by parts, raising λ^-, resulting in

$$VT^-_{\delta X} = V^-_{VT} + BT^-_{\delta X}, \tag{B.20}$$

where

$$V^-_{VT} = \int_0^T c^2 (\lambda^-, \delta J \Delta u^- - (\delta\alpha_1 u^-_\xi + \delta\beta u^-_\eta)_\xi - (\delta\alpha_2 u^-_\eta + \delta\beta u^-_\xi)_\eta)_{\tilde{\Omega}^-} \, dt, \tag{B.21}$$

and

$$\begin{aligned} BT^-_{\delta X} = \int_0^T c^2 \Big(& \langle \lambda^-, \delta\alpha_1 u^-_\xi + \delta\beta u^-_\eta \rangle_{\partial\tilde{\Omega}^{(-,e)}} \\ & - \langle \lambda^-, \delta\alpha_1 u^-_\xi + \delta\beta u^-_\eta \rangle_{\partial\tilde{\Omega}^{(-,w)}} \\ & + \langle \lambda^-, \delta\alpha_2 u^-_\eta + \delta\beta u^-_\xi \rangle_{\partial\tilde{\Omega}^{(-,n)}} \\ & - \langle \lambda^-, \delta\alpha_2 u^-_\eta + \delta\beta u^-_\xi \rangle_{\partial\tilde{\Omega}^{(-,s)}} \Big) \, dt. \end{aligned} \tag{B.22}$$

To obtain (B.21) we used $u^-_{tt} = c^2 \Delta u^-$ (where Δu^- is given by (10)) to substitute the second derivative in time. Focusing on the remaining boundary terms, we return to (B.8) and transform to the reference domain

$$BT^- = \int_0^t \langle \lambda^-_t, cW_1 u^- \rangle_{\partial\tilde{\Omega}^{(-,e)}} + \langle \lambda^-_t, cW_1 u^- \rangle_{\partial\tilde{\Omega}^{(-,w)}} \, dt. \tag{B.23}$$

By the product rule $\delta(W_1 u^-) = \delta W_1 u^- + W_1 \delta u^-$, it follows that

$$\begin{aligned} \delta BT^- = \int_0^T \Big(& \langle \lambda^-_t, c\delta W_1 u^- \rangle_{\partial\tilde{\Omega}^{(-,e)}} + \langle \lambda^-_t, c\delta W_1 u^- \rangle_{\partial\tilde{\Omega}^{(-,w)}} \\ & + \langle c\lambda^-_t, \delta u^- \rangle_{\partial\Omega^{(-,e)}} + \langle c\lambda^-_t, \delta u^- \rangle_{\partial\Omega^{(-,w)}} \Big) \, dt, \end{aligned} \tag{B.24}$$

where the two latter terms are transformed back to the physical domain. Considering $BT^-_{\delta u} + \delta BT^-$ (given by (B.18), (B.24)) the terms on the physical domain cancel since by the adjoint outflow conditions in (78), $\langle \lambda^-_t + c\mathbf{n}^- \cdot \nabla \lambda^-, \delta u^- \rangle_{\partial\Omega^{(-,w)}} = 0$ due to $\lambda_\tau = -\lambda_t$ (and similar for the east boundary). Additionally including the term $BT^-_{\delta X}$ in (B.22), the variation of the boundary terms are

$$\begin{aligned} V^-_{BT} = & BT^-_{\delta X} + BT^-_{\delta u} + \delta BT^- = \\ & \int_0^T \Big(c^2 (\langle \lambda^-, \delta\alpha_1 u^-_\xi + \delta\beta u^-_\eta \rangle_{\partial\tilde{\Omega}^{(-,e)}} - \langle \lambda^-, \delta\alpha_1 u^-_\xi + \delta\beta u^-_\eta \rangle_{\partial\tilde{\Omega}^{(-,w)}}) \\ & + c^2 (\langle \lambda^-, \delta\alpha_2 u^-_\eta + \delta\beta u^-_\xi \rangle_{\partial\tilde{\Omega}^{(-,n)}} - \langle \lambda^-, \delta\alpha_2 u^-_\eta + \delta\beta u^-_\xi \rangle_{\partial\tilde{\Omega}^{(-,s)}}) \\ & + c (\langle \lambda^-_t, \delta W_1 u^- \rangle_{\partial\tilde{\Omega}^{(-,w)}} + \langle \lambda^-_t, \delta W_1 u^- \rangle_{\partial\tilde{\Omega}^{(-,e)}}) \Big) \, dt. \end{aligned} \tag{B.25}$$

For the variation of interface terms, first note that by (B.10) $\delta IT = (c^2 \mathbf{n}^+ \cdot \nabla \lambda^+, \delta u^+)_{\Gamma_I}$. Then by (B.19), $\delta IT + IT^-_{\delta u} = 0$, due to the forward and adjoint interface conditions of (2b) and (78). We have therefore arrived at $\delta \mathbb{L} = V^-_{VT} + V^-_{BT}$, given by (B.21), and (B.25). Grouping terms in V^-_{VT}, and V^-_{BT} by inner products with λ^- and λ^-_t and transforming back to the physical domain results in

$$\delta \mathbb{L} = V_\lambda + V_{\lambda_t}, \tag{B.26}$$

where

$$
\begin{aligned}
V_\lambda = -c^2\Big(& -(\lambda^-, J^{-1}\delta J \Delta u^-)_{\Omega^-} \\
& +(\lambda^-, J^{-1}((\delta\alpha_1 u^-_\xi + \delta\beta u^-_\eta)_\xi + (\delta\alpha_2 u^-_\eta + \delta\beta u^-_\xi)_\eta))_{\Omega^-} \\
& -\langle \lambda^-, W_1^{-1}(\delta\alpha_1 u^-_\xi + \delta\beta u^-_\eta)\rangle_{\partial\Omega^{(-,e)}} \\
& +\langle \lambda^-, W_1^{-1}(\delta\alpha_1 u^-_\xi + \delta\beta u^-_\eta)\rangle_{\partial\Omega^{(-,w)}} \\
& -\langle \lambda^-, W_2^{-1}(\delta\alpha_2 u^-_\eta + \delta\beta u^-_\xi)\rangle_{\partial\Omega^{(-,n)}} \\
& +\langle \lambda^-, W_2^{-1}(\delta\alpha_2 u^-_\eta + \delta\beta u^-_\xi)\rangle_{\partial\Omega^{(-,s)}} \Big),
\end{aligned}
\tag{B.27}
$$

and

$$
V_{\lambda_t} = c\Big(\langle \lambda_t^-, W_1^{-1}\delta W_1 u^-\rangle_{\partial\Omega^{(-,e)}} + \langle \lambda_t^-, W_1^{-1}\delta W_1 u^-\rangle_{\partial\Omega^{(-,w)}} \Big). \tag{B.28}
$$

Since $\delta \mathrm{J} = \delta \mathrm{L} = V_\lambda + V_{\lambda_t}$, we arrive at $\frac{\delta \mathrm{J}}{\delta p} = \frac{\delta \mathrm{L}}{\delta p} = G_\lambda + G_{\lambda_t}$, i.e., (79). This completes the proof.

References

[1] K.C. Giannakoglou, D.I. Papadimitriou, Adjoint Methods for Shape Optimization, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 79–108.
[2] R.-E. Plessix, A review of the adjoint-state method for computing the gradient of a functional with geophysical applications, Geophys. J. Int. 167 (2) (2006) 495–503.
[3] R. Glowinski, J. He, On Shape Optimization and Related Issues, Birkhäuser Boston, Boston, MA, 1998, pp. 151–179.
[4] S.S. Collis, M. Heinkenschloss, Analysis of the streamline upwind/petrov galerkin method applied to the solution of optimal control problems, CAAM TR02-01 108, 2002.
[5] I. Babuška, A. Miller, The post-processing approach in the finite element method—part 1: calculation of displacements, stresses and other higher derivatives of the displacements, Int. J. Numer. Methods Eng. 20 (6) (1984) 1085–1109, https://doi.org/10.1002/nme.1620200610.
[6] N.A. Pierce, M.B. Giles, Adjoint recovery of superconvergent functionals from pde approximations, SIAM Rev. 42 (2) (2000) 247–264, https://doi.org/10.1137/S0036144598349423.
[7] J. Lu, An a posteriori error control framework for adaptive precision optimization using discontinuous galerkin finite element method, Ph.D. thesis, Massachusetts Institute of Technology, 2005.
[8] J.E. Hicken, D.W. Zingg, Superconvergent functional estimates from summation-by-parts finite-difference discretizations, SIAM J. Sci. Comput. 33 (2) (2011) 893–922, https://doi.org/10.1137/100790987.
[9] J. Berg, J. Nordström, Superconvergent functional output for time-dependent problems using finite differences on summation-by-parts form, J. Comput. Phys. 231 (20) (2012) 6846–6860, https://doi.org/10.1016/j.jcp.2012.06.032.
[10] M.C. Delfour, J.P. Zolésio, Shapes and Geometries, 2nd edition, Society for Industrial and Applied Mathematics, 2011.
[11] M. Berggren, Shape calculus for fitted and unfitted discretizations: domain transformations vs. boundary-face dilations, Commun. Optim. Theory 2023 (2023) 27.
[12] R. Glowinski, T.-W. Pan, J. Périaux, A fictitious domain method for Dirichlet problem and applications, Comput. Methods Appl. Mech. Eng. 111 (3–4) (1994) 283–303, https://doi.org/10.1016/0045-7825(94)90135-X.
[13] A. Bernland, E. Wadbro, M. Berggren, Acoustic shape optimization using cut finite elements, Int. J. Numer. Methods Eng. 113 (3) (2018) 432–449, https://doi.org/10.1002/nme.5621.
[14] H.-O. Kreiss, J. Oliger, Comparison of accurate methods for the integration of hyperbolic equations, Tellus 24 (1972) 199–215, https://doi.org/10.3402/tellusa.v24i3.10634.
[15] H.-O. Kreiss, G. Scherer, Finite element and finite difference methods for hyperbolic partial differential equations, in: Mathematical Aspects of Finite Elements in Partial Differential Equations, 1974, pp. 195–212.
[16] M. Svärd, J. Nordström, Review of summation-by-parts schemes for initial-boundary-value problems, J. Comput. Phys. 268 (2014) 17–38, https://doi.org/10.1016/j.jcp.2014.02.031.
[17] D.C. Del Rey Fernández, J.E. Hicken, D.W. Zingg, Review of summation-by-parts operators with simultaneous approximation terms for the numerical solution of partial differential equations, https://doi.org/10.1016/j.compfluid.2014.02.016, 2014.
[18] K. Mattsson, F. Ham, G. Iaccarino, Stable boundary treatment for the wave equation on second-order form, J. Sci. Comput. 41 (3) (2009) 366–383, https://doi.org/10.1007/s10915-009-9305-1.
[19] K. Mattsson, Summation by parts operators for finite difference approximations of second-derivatives with variable coefficients, J. Sci. Comput. 51 (3) (2012) 650–682, https://doi.org/10.1007/s10915-011-9525-z.
[20] V. Stiernström, M. Almquist, K. Mattsson, Boundary-optimized summation-by-parts operators for finite difference approximations of second derivatives with variable coefficients, J. Comput. Phys. 491 (2023) 112376, https://doi.org/10.1016/j.jcp.2023.112376, 24.
[21] M. Almquist, E.M. Dunham, Non-stiff boundary and interface penalties for narrow-stencil finite difference approximations of the Laplacian on curvilinear multi-block grids, J. Comput. Phys. 408 (2020) 109294, https://doi.org/10.1016/j.jcp.2020.109294, 33.
[22] K. Mattsson, J. Nordström, High order finite difference methods for wave propagation in discontinuous media, J. Comput. Phys. 220 (1) (2006) 249–269, https://doi.org/10.1016/j.jcp.2006.05.007.
[23] G. Eriksson, Non-conforming interface conditions for the second-order wave equation, J. Sci. Comput. 95 (3) (2023) 92, https://doi.org/10.1007/s10915-023-02218-1, 20.
[24] S. Wang, N.A. Petersson, Fourth order finite difference methods for the wave equation with mesh refinement interfaces, SIAM J. Sci. Comput. 41 (5) (2019) A3246–A3275, https://doi.org/10.1137/18M1211465.
[25] M. Bader, M. Almquist, E.M. Dunham, Modeling and inversion in acoustic-elastic coupled media using energy-stable summation-by-parts operators, Geophysics 88 (3) (2023) T137–T150, https://doi.org/10.1190/geo2022-0195.1.
[26] J.E. Hicken, D.W. Zingg, Aerodynamic optimization algorithm with integrated geometry parameterization and mesh movement, AIAA J. 48 (2) (2010) 400–413, https://doi.org/10.2514/1.44033.
[27] A. Kord, J. Capecelatro, A discrete-adjoint framework for optimizing high-fidelity simulations of turbulent reacting flows, Proc. Combust. Inst. 39 (4) (2023) 5375–5384, https://doi.org/10.1016/j.proci.2022.06.021.
[28] S. Hossbach, M. Lemke, J. Reiss, Finite-difference-based simulation and adjoint optimization of gas networks, Math. Methods Appl. Sci. 45 (7) (2022) 4035–4055, https://doi.org/10.1002/mma.8030.

[29] B. Engquist, A. Majda, Absorbing boundary conditions for the numerical simulation of waves, Math. Comput. 31 (139) (1977) 629–651, https://doi.org/10.2307/2005997.
[30] M. Almquist, I. Karasalo, K. Mattsson, Atmospheric sound propagation over large-scale irregular terrain, J. Sci. Comput. 61 (2) (2014) 369–397, https://doi.org/10.1007/s10915-014-9830-4.
[31] M. Almquist, S. Wang, J. Werpers, Order-preserving interpolation for summation-by-parts operators at nonconforming grid interfaces, SIAM J. Sci. Comput. 41 (2) (2019) A1201–A1227, https://doi.org/10.1137/18M1191609.
[32] O. O'Reilly, N.A. Petersson, Energy conservative SBP discretizations of the acoustic wave equation in covariant form on staggered curvilinear grids, J. Comput. Phys. 411 (2020) 109386, https://doi.org/10.1016/j.jcp.2020.109386, 22.
[33] O. Ålund, J. Nordström, Encapsulated high order difference operators on curvilinear non-conforming grids, J. Comput. Phys. 385 (2019) 209–224, https://doi.org/10.1016/j.jcp.2019.02.007.
[34] R.E. Smith, Algebraic grid generation, Appl. Math. Comput. 10–11 (1982) 137–170, https://doi.org/10.1016/0096-3003(82)90190-4.
[35] B. Strand, Summation by parts for finite difference approximations for d/dx, J. Comput. Phys. 110 (1) (1994) 47–67, https://doi.org/10.1006/jcph.1994.1005.
[36] N.A. Petersson, O. O'Reilly, B. Sjögreen, S. Bydlon, Discretizing singular point sources in hyperbolic wave propagation problems, J. Comput. Phys. 321 (2016) 532–555, https://doi.org/10.1016/j.jcp.2016.05.060.
[37] K. Mattsson, F. Ham, G. Iaccarino, Stable and accurate wave-propagation in discontinuous media, J. Comput. Phys. 227 (19) (2008) 8753–8767, https://doi.org/10.1016/j.jcp.2008.06.023.
[38] S. Wang, An improved high order finite difference method for non-conforming grid interfaces for the wave equation, J. Sci. Comput. 77 (2) (2018) 775–792, https://doi.org/10.1007/s10915-018-0723-9.
[39] G. Eriksson, J. Werpers, D. Niemelä, N. Wik, V. Zethrin, K. Mattsson, Boundary and interface methods for energy stable finite difference discretizations of the dynamic beam equation, J. Comput. Phys. 476 (2023) 111907, https://doi.org/10.1016/j.jcp.2023.111907, 20.
[40] S. Wang, K. Virta, G. Kreiss, High order finite difference methods for the wave equation with non-conforming grid interfaces, J. Sci. Comput. 68 (3) (2016) 1002–1028, https://doi.org/10.1007/s10915-016-0165-1.
[41] P. Olsson, Summation by parts, projections, and stability. I, Math. Comput. 64 (211) (1995) 1035–1065, https://doi.org/10.2307/2153482, S23–S26.
[42] P. Olsson, Summation by parts, projections, and stability. II, Math. Comput. 64 (212) (1995) 1473–1493, https://doi.org/10.2307/2153366.
[43] M. Almquist, E.M. Dunham, Elastic wave propagation in anisotropic solids using energy-stable finite differences with weakly enforced boundary and interface conditions, J. Comput. Phys. 424 (2021) 109842, https://doi.org/10.1016/j.jcp.2020.109842, 33.
[44] J.A. Samareh, Survey of shape parameterization techniques for high-fidelity multidisciplinary shape optimization, AIAA J. 39 (5) (2001) 877–884.
[45] B. Sjögreen, N.A. Petersson, Source estimation by full wave form inversion, J. Sci. Comput. 59 (1) (2014) 247–276, https://doi.org/10.1007/s10915-013-9760-6.
[46] O. Gauthier, J. Virieux, A. Tarantola, Two-dimensional nonlinear inversion of seismic waveforms: numerical results, Geophysics 51 (7) (1986) 1387–1403, https://doi.org/10.1190/1.1442188.
[47] J.E. Hicken, D.W. Zingg, Dual consistency and functional accuracy: a finite-difference perspective, J. Comput. Phys. 256 (2014) 161–182, https://doi.org/10.1016/j.jcp.2013.08.014.
[48] J.M. Sanz-Serna, Symplectic Runge-Kutta schemes for adjoint equations, automatic differentiation, optimal control, and more, SIAM Rev. 58 (1) (2016) 3–33, https://doi.org/10.1137/151002769.
[49] J. Nordström, T. Lundquist, Summation-by-parts in time: the second derivative, SIAM J. Sci. Comput. 38 (3) (2016) A1561–A1586, https://doi.org/10.1137/15M103861X.
[50] C.G. Broyden, The convergence of a class of double-rank minimization algorithms 1. General considerations, IMA J. Appl. Math. 6 (1) (1970) 76–90, https://doi.org/10.1093/imamat/6.1.76.
[51] E. Bängtsson, D. Noreland, M. Berggren, Shape optimization of an acoustic horn, Comput. Methods Appl. Mech. Eng. 192 (11) (2003) 1533–1571, https://doi.org/10.1016/S0045-7825(02)00656-4.
[52] D.J.A. McKechan, C. Robinson, B.S. Sathyaprakash, A tapering window for time-domain templates and simulated signals in the detection of gravitational waves from coalescing compact binaries, Class. Quantum Gravity 27 (8) (2010) 084020, https://doi.org/10.1088/0264-9381/27/8/084020.