Journal of Computational Physics 354 (2018) 26–42

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

Numerical solution of the wave equation with variable wave speed on nonconforming domains by high-order difference potentials ☆

S. Britt [a], S. Tsynkov [b,c], E. Turkel [a,*]

[a] School of Mathematical Sciences, Tel Aviv University, Ramat Aviv, Tel Aviv 69978, Israel
[b] Department of Mathematics, North Carolina State University, Box 8205, Raleigh, NC 27695, USA
[c] Moscow Institute of Physics and Technology, Dolgoprudny, 141700, Russia

ARTICLE INFO

Article history:
Received 8 May 2017
Received in revised form 24 October 2017
Accepted 25 October 2017
Available online 27 October 2017

Keywords:
Method of difference potentials (MDP)
Compact finite difference scheme
Implicit scheme
Regular structured grid
Nonconforming boundary
High-order accuracy
High-order MDP

ABSTRACT

We solve the wave equation with variable wave speed on nonconforming domains with fourth order accuracy in both space and time. This is accomplished using an implicit finite difference (FD) scheme for the wave equation and solving an elliptic (modified Helmholtz) equation at each time step with fourth order spatial accuracy by the method of difference potentials (MDP). High-order MDP utilizes compact FD schemes on regular structured grids to efficiently solve problems on nonconforming domains while maintaining the design convergence rate of the underlying FD scheme. Asymptotically, the computational complexity of high-order MDP scales the same as that for FD.

1. Introduction

We consider an initial boundary value problem for the wave (d'Alembert) equation:

$$u_{tt} = c^2 \Delta u + F, \qquad \vec{x} \in \Omega \tag{1a}$$

$$u\left(\vec{x}, 0\right) = \phi_0\left(\vec{x}\right) \tag{1b}$$

$$u_t\left(\vec{x}, 0\right) = \phi_1\left(\vec{x}\right) \tag{1c}$$

$$\ell(u)|_\Gamma = \psi(t) \tag{1d}$$

where $\Gamma = \partial\Omega$ is the boundary, the wave speed c is a variable function of the spatial coordinates (assumed smooth in the current work, although this limitation can be lifted as explained in Section 5), and F is an inhomogeneous term. The

☆ Work supported by the United States–Israel Binational Science Foundation (BSF) under grant # 2014048 and by Army Research Office (ARO) under grant # W911NF-16-1-0115. S. Britt was supported by the Raymond and Beverly Sackler Post-Doctoral Scholarship at Tel Aviv University and a Fulbright Postdoctoral Scholarship funded by the US–Israel Educational Foundation.

* Corresponding author.
E-mail addresses: darrellstevenbritt@gmail.com (S. Britt), tsynkov@math.ncsu.edu (S. Tsynkov), turkel@post.tau.ac.il (E. Turkel).
URLs: http://www4.ncsu.edu/~stsynkov/ (S. Tsynkov), http://www.math.tau.ac.il/~turkel/ (E. Turkel).

https://doi.org/10.1016/j.jcp.2017.10.049
0021-9991/

boundary condition (1d) in this work is taken to be either Dirichlet ($\ell = 1$) or Neumann ($\ell = \frac{\partial}{\partial \boldsymbol{n}}$). In our earlier work [1, 2] that discussed the Helmholtz equation (i.e., the time-harmonic wave equation), we have considered a variety of more general boundary conditions as well.

Equation (1a) is an established model for a broad range of problems in acoustics and electromagnetism. The key feature of all these problems is their linearity. The numerical methods that we are developing hereafter are not designed for solving the nonlinear problems. We rather consider our main challenge as to compute the solution over large and generally shaped regions with high fidelity and robustness.

Finite difference (FD) methods are known to lead to inexpensive and efficient algorithms for computing smooth solutions on regular domains/grids. Their primary disadvantage is in dealing with more complicated geometries and solutions with low regularity. The finite element method (FEM) and its extensions, as well as the discontinuous Galerkin method (DG), may help alleviate these two constraints pertinent to FD. Yet in practical problems of wave propagation, especially in 3D, both FD and FEM have serious limitations because of their relatively high "points-per-wavelength" requirement, as well as numerical pollution (the dispersion error), see [3,4] and [5, Section 4.6.1]. The numerical phase velocity of the wave in these methods depends on the wavenumber. Therefore, a propagating packet of waves with different frequencies gets distorted in the simulation. Furthermore, the numerical error strongly depends on the frequency [6,5].

This drawback can be (partially) overcome by high-order FD schemes. They, however, usually need a wider stencil, which complicates the boundary conditions. A class of schemes aimed at reducing the phase error are the dispersion relation preserving schemes [7,8]. Yet they need an even wider stencil than conventional schemes of the same order of accuracy.

There is also a special type of high-order schemes that do not require a wider stencil. These schemes rely on a targeted approximation of the class of solutions rather than of a much broader class of generic sufficiently smooth functions. The equation-based compact schemes that we have developed in [9–11] for the Helmholtz equation are in this category; other similar methods include [12–15]. A recent extension of compact equation-based schemes to the time domain is given in [16]. Such schemes reduce pollution while keeping the treatment of the boundary conditions simple. However, geometry still remains a hurdle.

In FEM, on the other hand, a high-order accurate approximation can be built for arbitrary boundaries with the help of isoparametric elements [17]. These methods require a grid generation which can be nontrivial for complex geometries and interfaces. In DG, discontinuous enrichment methods, and generalized FEM, high-order accuracy also requires additional degrees of freedom. The disadvantage of these methods for the linear problems with smooth solutions is their substantial redundancy, which entails additional computational costs.

A group of methods known to provide a very considerable flexibility from the standpoint of geometry are the boundary element methods (BEM). They typically apply to steady-state or time-harmonic problems (elliptic PDEs). In these methods, linear boundary value problems are reduced to boundary integral equations (BIE) with respect to equivalent boundary sources. BEM impose practically no limitations on the shape of the boundary and automatically account for the correct far field behavior of the solution. However, these methods rely on the explicit knowledge of the fundamental solution (and so they are not easily compatible with variable propagation speed), and the treatment of the boundary conditions requires care in choosing the boundary sources so as to maintain the equivalence of the reduction and well-posedness of the resulting boundary representation. In doing so, the cases that involve resonances of the complementary domain require special attention, see, e.g., [18].

Standard BEM cannot be used directly for unsteady problems of wave propagation (hyperbolic PDEs). Their time-dependent applications are rather limited to combined problems with a clearly identifiable elliptic component, such as slow speed flows of viscous fluid [19,20] or water waves [21].

A special class of BIEs called the retarded potential boundary integral equations (RPBIE), see [22,23], provide a venue toward extending the BEM from elliptic to hyperbolic PDEs. However, the corresponding time domain numerical methods [24–27] are not nearly as popular as their frequency domain counterparts. One difficulty is that many time domain discretizations of RPBIEs appear prone to instabilities, even if the well-posedness of the RPBIE per se can be guaranteed in the first place (some aspects of stability have recently been studied in [28]). For the most part, however, the reason is that as the time elapses the boundary extends and the computation of convolutions involved in RPBIEs that typically relies on Laplace transform methods [29,30] becomes progressively more expensive. In that regard, we also mention work [31,32] that uses RPBIEs and convolution quadratures [29,30] for the development and analysis of far-field boundary conditions.

In our earlier work on the Helmholtz equation [33,1,34,35,2], we have employed the method of difference potentials (MDP) developed by Ryaben'kii [36–39]. The MDP can be viewed as a discrete analog of Calderon's potentials and Calderon's boundary equations with projections in functional analysis [40,41]. Its capacity of handling the boundaries of general shape is comparable to that of BIEs. Yet the MDP does not require fundamental solutions and automatically guarantees the equivalence of the reduced boundary problem and the original one. It uses discretizations on regular structured grids and can maintain high-order accuracy for non-conforming boundaries. Difference potentials for the Helmholtz equation [33,1,34,35,2] were built using compact equation-based schemes [9–11] that enable high-order accuracy while avoiding the extensive redundancy inherent in high-order FEM and DG methods.

In the current paper, we extend the previously developed MDP-based approach for time-harmonic waves to the genuinely time-dependent formulation (1). Our goal is to achieve the same geometric flexibility and high-oder accuracy as we have obtained for the Helmholtz equation [33,1,34]. Fundamentally, there may be two ways of pursuing this goal. One can build a full-fledged MDP algorithm in 3+1 dimensional space-time. In doing so, like in the case of RPBIEs, computing the operators

may appear increasingly costly as the time elapses. To eliminate this increase, in 3D one can employ the strong Huygens' principle and lacunae [42] in the solutions of equation (1a), and this method is analyzed in [43]. The idea of the approach of [43] is very close to what we have used previously when building the lacunae-based outer boundary conditions for the numerical simulation of unsteady waves [44–51]. Note that the phase filter of [52] can also be thought of in terms of the Huygens' principle.

Alternatively, one can first discretize equation (1a) in time by means of an implicit scheme. This discretization yields an elliptic (i.e., steady-state) equation that needs to be solved on the upper time level at every time step. This can be done by means of MDP similarly to how the Helmholtz equation was solved in [33,1,34]. It is this approach that we develop and analyze in the current paper. The compact discretization that we use was recently introduced in [16]. Earlier, a similar methodology for parabolic equations was explored in [53,54], but the finite difference stencils are not compact and only Dirichlet boundary conditions are considered.

For simplicity, in this paper we consider the wave equation (1a) in two dimensions and take Ω to be a disk centered at the origin, so that the normal direction to Γ coincides with the polar radius. A Cartesian FD scheme will be used, so that the boundary does not conform to the grid. A treatment of general smooth boundary shapes is demonstrated for the Helmholtz equation in [55], and this case therefore represents no loss of generality.

In Section 2, we introduce an implicit finite difference scheme for equation (1a), see [16] for additional detail. This FD scheme becomes a component of the high-order MDP algorithm for the initial boundary value problem (1) described in Section 3. Numerical results verifying fourth order convergence in space and time are presented in Section 4.

2. Implicit time discretization

Let h_t be the uniform time step and let δ_t^2 be the second order central difference in time. We consider the following semi-discrete approximation to equation (1a) centered at the time t_n using a free parameter θ (sometimes referred to as a θ-scheme, see, e.g., [56,57]):

$$\frac{1}{h_t^2}\delta_t^2 u^n = c^2\Delta u^n + \theta c^2 \delta_t^2 \Delta u^n + F^n + \theta\delta_t^2 F^n. \tag{2}$$

Rearranging (2) to gather the upper time level terms yields the implicit formulation:

$$\begin{aligned}\Delta u^{n+1} - \frac{1}{\theta c^2 h_t^2}u^{n+1} &= 2\left(\Delta u^n - \frac{1}{\theta c^2 h_t^2}u^n\right) - \left(\Delta u^{n-1} - \frac{1}{\theta c^2 h_t^2}u^{n-1}\right) - \frac{1}{\theta}\Delta u^n \\ &\quad - \frac{1}{\theta c^2}F^n - \frac{1}{c^2}\delta_t^2 F^n \\ &\equiv f^{n+1}.\end{aligned} \tag{3}$$

By definition, $f^{n+1} = \Delta u^{n+1} - \frac{1}{\theta c^2 h_t^2}u^{n+1}$. Examining (3), we may use this definition at prior time levels to express f^{n+1} by the recursive formula

$$f^{n+1} = 2f^n - f^{n-1} - \frac{1}{\theta}\Delta u^n - \tilde{F}^{n+1}, \tag{4}$$

where $\tilde{F}^{n+1} \equiv \frac{1}{\theta c^2}F^n + \frac{1}{c^2}\delta_t^2 F^n$ consists of the given inhomogeneous terms. The terms f^0 and f^1 are known from the initial conditions (1b) of the wave equation. In Section 3.3, it will be demonstrated how to compute the remaining term Δu^n when solving the elliptic equation (3) numerically.

As shown in [16], the accuracy and stability of the semi-discrete approximation (3) depends on the parameter θ, so that for

- $\theta = \frac{1}{12}$, (3) is 4th order accurate in time and conditionally stable,
- $\theta \geq \frac{1}{4}$, (3) is 2nd order accurate in time and unconditionally stable (this includes the Crank–Nicholson scheme, $\theta = \frac{1}{2}$),
- $\theta = 0$, (3) is explicit and 2nd order accurate in time.

Britt et al. [16] proved stability based on the construction of an appropriate norm. They showed that the scheme is energy conserving in this norm.

Moreover, even though the scheme is implicit and requires a matrix inversion at the upper time level, it is superior in complexity to a low-order explicit scheme and comparable to an explicit scheme of the same order of accuracy. The reason is that the inversion can be done by an efficient technique, such as FFT or multigrid. The issue of complexity has been thoroughly studied in [16].

From the initial condition (1b), u^0 is known. Because (3) contains two backwards time levels, we approximate u^1 by the Taylor expansion

$$u^1 = u^0 + h_t u_t^0 + \frac{h_t^2}{2} u_{tt}^0 + \frac{h_t^3}{6} u_{ttt}^0 + \frac{h_t^4}{24} u_{tttt}^0 + \mathcal{O}\left(h_t^5\right). \tag{5}$$

The term u_t^0 is also known from the initial condition (1c), while the higher order terms may be found by substituting the PDE (1a) at the initial time step as follows:

$$\begin{aligned} u_{tt}^0 &= c^2 \Delta u^0 + F^0 \\ u_{ttt}^0 &= c^2 \Delta u_t^0 + F_t^0 \\ u_{tttt}^0 &= c^2 \Delta u_{tt}^0 + F_{tt}^0 = c^2 \Delta (c^2 \Delta u^0 + F^0) + F_{tt}^0. \end{aligned} \tag{6}$$

It is assumed that the derivatives of F^0 and the spatial derivatives of u^0 in (6) can either be computed analytically or by finite differences with the desired resolution.

The elliptic equation (3) at time step t_{n+1} is the modified Helmholtz equation, for which we formulate the boundary value problem (BVP) at each time step:

$$\Delta u^{n+1} - k^2 u^{n+1} = f^{n+1}, \qquad \vec{x} \in \Omega, \tag{7a}$$

$$\ell(u^{n+1})|_\Gamma = \psi(t_{n+1}), \tag{7b}$$

where $k^2 = \frac{1}{\theta c^2 h_t^2}$ and f^{n+1} contains the previously computed time steps as well as inhomogeneous terms from the wave equation, see (3). In the next section, we describe a solution method for the semi-discrete BVP (7) at each time step by difference potentials.

3. Difference potentials

To solve the elliptic BVP (7) at each time step, we embed the general domain Ω within a simpler auxiliary domain Ω_0, which in 2D we take to be a square. We then formulate an auxiliary problem on the square Ω_0 that can be solved efficiently by a finite difference scheme on the Cartesian grid and will be used in constructing the difference potential. To demonstrate the capability of the high-order MDP algorithm to treat nonconforming domains, we take Ω to be a disk centered at the origin, the simplest nonconforming shape for a Cartesian scheme.

The statement of the auxiliary problem on the square Ω_0 and its parameters, along with its solution method by finite differences, is described in Section 3.1. Section 3.2 establishes notation and definitions of various sets of grid nodes which will be needed when defining the difference potential. Section 3.3 specifies how to calculate the right-hand side f^{n+1} of the elliptic equation on the appropriate grid set as required to formulate the difference potential. In Section 3.4 we define the difference potential itself and introduce the boundary equation with projection.

In Section 3.5, we describe an equation-based extension procedure which ensures a high-order approximation of the continuous solution to the elliptic equation (7a) by difference potential. This is based on a result by Reznik [58,59] which provides sufficient conditions for the difference potential to approximate a continuous Calderon potential [41,40]. Next, we introduce a basis for pairs of functions along the continuous boundary Γ in Section 3.6, with which we will approximate the boundary data $\left(u, \frac{\partial u}{\partial \boldsymbol{n}}\right)\big|_\Gamma$ of the elliptic BVP (7).

Section 3.7 describes the high-order MDP algorithm for solving the wave equation using the implicit time discretization (3).

3.1. The auxiliary problem

The auxiliary problem (AP) for the method of difference potentials must satisfy three conditions: the auxiliary domain Ω_0 must contain the domain of interest Ω, the equation specified on Ω_0 must coincide with the PDE on $\Omega \subset \Omega_0$, and the PDE specified on Ω_0 must be well-posed. Therefore let Ω_0 be a square of side-length $s > 2$ centered at the origin, so that Ω_0 contains Ω (i.e., the disk of radius 1 centered at the origin.) In our MDP formulation for the wave equation using the implicit time discretization (3), the PDE for the AP on the square Ω_0 comes from the elliptic equation (7a),

$$\begin{aligned} \Delta u - k^2 u &= g, \qquad -\frac{s}{2} \le x, y \le \frac{s}{2} \\ u|_{x=\pm\frac{s}{2}, y=\pm\frac{s}{2}} &= 0, \end{aligned} \tag{8}$$

where $k^2 = \frac{1}{\theta c^2 h_t^2}$ and g is an arbitrary right-hand side that will be specified by the MDP algorithm. As $c(x, y)$ in the wave equation (1a) is defined only on Ω, we require a twice continuously differentiable extension of c to the larger domain Ω_0 — this can be done, e.g., by polynomial extrapolation. The MDP algorithm requires the solution of multiple APs with different right-hand sides. Critically, the AP (8) is well-posed.

For the purposes of the high-order MDP algorithm for the wave equation, it is sufficient to consider only zero Dirichlet boundary conditions for the FD scheme on the square, even when the boundary conditions on the nonconforming domain

of interest (i.e., the disk) are not Dirichlet. This is because problem (8) remains well-posed on the square for any value of $k^2 > 0$ and any right-hand side g with Dirichlet BCs. The usual Helmholtz equation with Dirichlet BCs on the square is prone to resonances for certain values of k^2, and in our prior work this was avoided by taking a more complicated Sommerfeld-type BC at the left and right edges of the auxiliary domain [1]. The boundary conditions for the wave equation are therefore significantly more straightforward.

We numerically solve the AP (8) by the compact finite difference scheme described in [14]. On a 2D Cartesian grid which is equally spaced in both directions with step size $h_x = h_y$, we define the fourth order FD scheme $\boldsymbol{L}^{(h)}u = g$ as follows. Denote by L_0 the coefficient of the center node of a nine point stencil, by L_s the coefficient of the four side nodes, and by L_c the coefficient of the four corner points. Then the left-hand side operator $\boldsymbol{L}^{(h)}$ on the 9-point stencil has coefficients

$$L_0 = -\frac{10}{3} - \frac{2}{3}k^2h_x^2, \quad L_s = \frac{2}{3} - \frac{k^2h_x^2}{12}, \quad L_c = \frac{1}{6}. \tag{9}$$

The compact scheme uses a right-hand side operator $\boldsymbol{B}^{(h)}$ defined by coefficients B_i corresponding to the L_i:

$$B_0 = \frac{2h_x^2}{3}, \quad B_s = \frac{h_x^2}{12}, \quad B_c = 0. \tag{10}$$

When solving the BVP (7) by difference potentials, the AP (8) will need to be solved repeatedly for the right-hand sides g supplied by the MDP algorithm. These right-hand sides are defined in Section 3.4 and can be thought of as if they already include the application of the operator $\boldsymbol{B}^{(h)}$. The AP (8) will also need to be solved for the "physical" right-hand sides f^{n+1} of (7a), in which case the operator $\boldsymbol{B}^{(h)}$ will be applied explicitly. Eventually, in either case the discrete equation that we solve has the form $\boldsymbol{L}^{(h)}u = g$.

We define the Courant–Friedrichs–Lewy (CFL) number to be $\lambda = \frac{ch_t}{h_x}$, where h_x is the uniform spatial grid step. The stability condition for the fourth order time discretization $\theta = \frac{1}{12}$ depends on the spatial discretization. Let $\boldsymbol{L}_0^{(h)}$ denote the negative definite discrete approximation to the Laplacian ($k = 0$ in formula (9)), for which $0 < (-h_x^2\boldsymbol{L}_0^{(h)}u, u) \le L_{\text{upper}}\|u\|^2$. Then, the stability condition is given by $\lambda^2 \le \frac{6}{L_{\text{upper}}}$, which becomes $\lambda^2 \le \frac{3}{8}$ for scheme (9)–(10), see [16].

Let $\boldsymbol{G}^{(h)}$ denote the solution operator of the discrete AP, so that the numerical solution of (8) is formally $u = \boldsymbol{G}^{(h)}g$ on the square, although the matrix $\boldsymbol{G}^{(h)}$ is never explicitly computed. For variable coefficients, the discrete AP can be solved by either direct or iterative methods in 2D, but direct methods such as LU or Cholesky factorization become prohibitively expensive in 3D. In [16], we compare the efficiency of a direct LU solver with conjugate gradient and multigrid iterative solvers in 2D. The MDP algorithm will require multiple solutions of the AP in the preprocessing steps as well as two solutions of the AP at each time step, but in every instance only the right-hand side g of the AP (8) changes while all other parameters are fixed. If solving by a direct matrix factorization method such as LU or Cholesky, the matrix decomposition is performed once at the beginning and then reused so that the subsequent numerical solutions of the AP are computationally inexpensive.

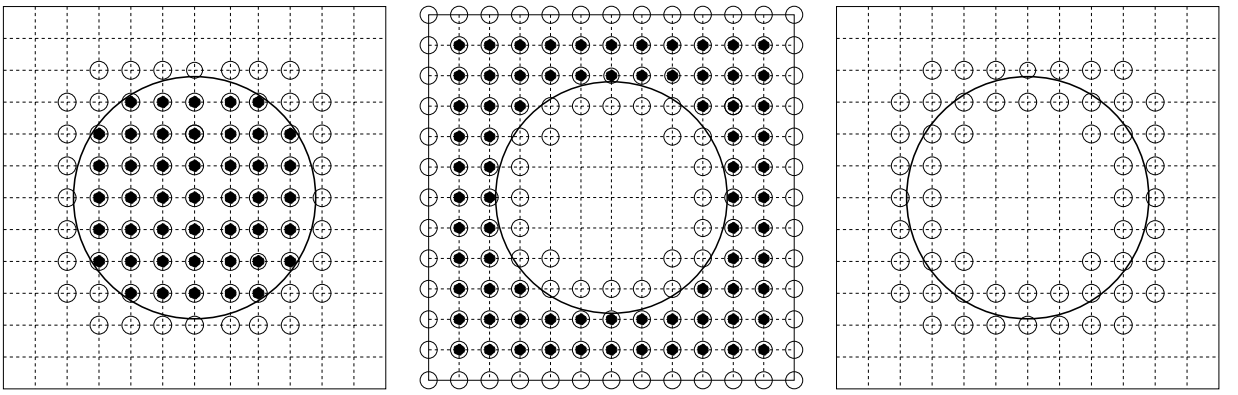
3.2. Grid sets

The definition of the difference potential involves subsets of grid points within the auxiliary domain which we now specify. Let $\mathbb{N}_0$ be the set of all Cartesian grid nodes with step size $h_x = h_y$ on the auxiliary square Ω_0, and let $\mathbb{M}_0$ be the set of interior nodes of the grid. Denote by $\mathbb{M}^+ \subset \mathbb{M}_0$ the set of grid points inside of the disk and by $\mathbb{M}^- \subset \mathbb{M}_0$ the nodes outside of the disk but excluding the boundary nodes of the square. We define corresponding sets $\mathbb{N}^+ \subset \mathbb{N}_0$ and $\mathbb{N}^- \subset \mathbb{N}_0$ which consist of all nodes touched by the 9-point stencil (see equation (9)) acting on $\mathbb{M}^+$ and $\mathbb{M}^-$ respectively. The intersection $\gamma = \mathbb{N}^+ \cap \mathbb{N}^-$ is then a nonempty subset of grid points that straddle the continuous boundary Γ, and we therefore refer to γ as the discrete boundary.

The grid sets $\mathbb{M}^+$, $\mathbb{M}^-$, $\mathbb{N}^+$, $\mathbb{N}^-$, and γ are schematically shown in Fig. 1. Having a given node from $\mathbb{M}^0$ lying precisely at the boundary $\Gamma = \partial\Omega$ rather than strictly inside or outside Ω presents no problem. Such nodes can be included either in $\mathbb{M}^+$ or $\mathbb{M}^-$. The corresponding convention needs to be made ahead of time and followed throughout the development of the algorithm, in which case the particular choice, $\mathbb{M}^+$ or $\mathbb{M}^-$, will not make a difference.

3.3. Computing $\boldsymbol{B}^{(h)}f^{n+1}$ on $\mathbb{M}^+$

In the recurrence relation (4) for f^{n+1}, all terms are known on Ω by assumption except for Δu^n. For the method of difference potentials, we require the quantity $\boldsymbol{B}^{(h)}f^{n+1}$ only at the grid nodes $\mathbb{M}^+ \subset \Omega$; however, direct application of the FD stencil of $\boldsymbol{B}^{(h)}$ on $\mathbb{M}^+$ (see equation (10)) involves values of f^{n+1} that lie outside of Ω, namely, at some of the nodes from the set $\mathbb{N}^+ \setminus \mathbb{M}^+$. We must therefore specify how to compute $\boldsymbol{B}^{(h)}f^{n+1}$ on $\mathbb{M}^+$ at each time step. Our approach first treats the initial time steps so that the values of $\boldsymbol{B}^{(h)}f^0$ and $\boldsymbol{B}^{(h)}f^1$ are known on $\mathbb{M}^+$ and then proceeds by the recurrence relation (4) to obtain $\boldsymbol{B}^{(h)}f^{n+1}$ on $\mathbb{M}^+$. For clarity, we introduce the notation $f_B^{n+1} := \boldsymbol{B}^{(h)}f^{n+1}$, which is a sequence of values on $\mathbb{M}^+$ which formally is defined by the recurrence relation (4) operated on from the left by the stencil $\boldsymbol{B}^{(h)}$ but in practice does not require values of f^n outside of Ω after the initial steps.



(a) • — $\mathbb{M}^+$, ◯ — $\mathbb{N}^+$ (b) • — $\mathbb{M}^-$, ◯ — $\mathbb{N}^-$ (c) ◯ — γ

Fig. 1. Interior and exterior grid subsets and the grid boundary.

From the initial conditions (1b) of the wave equation, f^0 and f^1 are known on Ω, and at these initial values we may employ a continuously differentiable extension of f^0 and f^1 from the domain Ω to the adjacent nodes of $\mathbb{N}^+ \setminus \mathbb{M}^+$ (e.g., by a Taylor series of sufficient order). We then directly apply the stencil $\boldsymbol{B}^{(h)}$ to f^0 and f^1 using the extended values on $\mathbb{N}^+ \setminus \mathbb{M}^+$, which yields the initial quantities f_B^0, f_B^1 of the recurrence for f_B^{n+1} on $\mathbb{M}^+$.

Recall that $\boldsymbol{L}_0^{(h)}$ denotes the FD operator defined by (9)–(10) for the case $k = 0$. The numerical solution at previous time steps, u^n, is assumed to be known by difference potentials on the grid $\mathbb{N}^+$ so that application of this FD operator yields $\boldsymbol{L}_0^{(h)} u^n \approx \boldsymbol{B}^{(h)} \Delta u^n + \mathcal{O}\left(h_x^4\right)$ (note that the coefficients of $\boldsymbol{B}^{(h)}$ do not depend on k, see (10)). For an inhomogeneous equation, the source term $\tilde{F}^{n+1}$ is known on Ω and can also be defined at the required nodes of $\mathbb{N}^+ \setminus \mathbb{M}^+$ by a continuously differentiable extension at each time step,[1] so that we may compute $\boldsymbol{B}^{(h)} \tilde{F}^{n+1}$ at $\mathbb{M}^+$ by direct application of the operator. Altogether, we have the following at the nodes $\mathbb{M}^+$:

$$\begin{aligned} f_B^{n+1} &= 2 f_B^n - f_B^{n-1} - \frac{1}{\theta} \boldsymbol{B}^{(h)} \Delta u^n - \boldsymbol{B}^{(h)} \tilde{F}^{n+1} \\ &\approx 2 f_B^n - f_B^{n-1} - \frac{1}{\theta} \boldsymbol{L}_0^{(h)} u^n - \boldsymbol{B}^{(h)} \tilde{F}^{n+1}. \end{aligned} \tag{11}$$

Again, we emphasize that the terms $f_B^n = \boldsymbol{B}^{(h)} f^n$ do not involve application of the operator but are calculated by recurrence starting from the initial values f_B^0, f_B^1 as described above, and this eliminates the need for special treatment of f^n at the nodes $\mathbb{N}^+ \setminus \mathbb{M}^+$ which lie outside of Ω.

3.4. Difference potentials and the boundary equation with projection

Let ξ_γ be a function with values on the discrete boundary γ and define the auxiliary function $w|_\gamma = \xi_\gamma$ on $\mathbb{N}_0$ with $w|_{\mathbb{N}_0 \setminus \gamma} = 0$. Then the **difference potential with density** ξ_γ is the operator $\boldsymbol{P}_{\mathbb{N}^+} : \gamma \to \mathbb{N}^+$ defined by

$$\boldsymbol{P}_{\mathbb{N}^+} \xi_\gamma := w - \boldsymbol{G}^{(h)} \left[\mathbb{1}_{\mathbb{M}_0}(\mathbb{M}^+) \left(\boldsymbol{L}^{(h)} w \right) \right], \tag{12}$$

where $\mathbb{1}_{\mathbb{M}_0}$ is the indicator function on $\mathbb{M}_0$ and the right-hand side of (12) is considered only on the grid $\mathbb{N}^+ \subset \mathbb{N}_0$. As $\boldsymbol{G}^{(h)}$ is the solution operator of the discrete AP (8), the construction of the difference potential of density ξ_γ requires solving an auxiliary problem with the right-hand side $g = \mathbb{1}_{\mathbb{M}_0}(\mathbb{M}^+) \left(\boldsymbol{L}^{(h)} w \right)$. Therefore the cost of computing a difference potential on $\mathbb{N}^+$ is the same as solving a finite difference problem on the auxiliary square $\mathbb{N}_0$. We define the **difference projection of** ξ_γ as

$$\boldsymbol{P}_\gamma \xi_\gamma := \left(\boldsymbol{P}_{\mathbb{N}^+} \xi_\gamma \right) \big|_\gamma \tag{13}$$

in order to formulate the discrete **boundary equation with projection** (BEP) on γ at time step t_{n+1}:

$$\boldsymbol{P}_\gamma \xi_\gamma + \left(\boldsymbol{G}^{(h)} \tilde{f}_B^{n+1} \right) \Big|_\gamma = \xi_\gamma, \tag{14}$$

[1] The extension of the known function $\tilde{F}^{n+1}$ from Ω to nearby grid nodes is rendered by a low order Taylor formula and proves and entails a computationally negligible cost.

where $\tilde{f}_B^{n+1}$ is equal to f_B^{n+1} (see (11)) on the interior nodes $\mathbb{M}^+$ and zero on $\mathbb{M}^-$. A central result in the theory of difference potentials (see [37]) is that a density ξ_γ satisfies the BEP (14) if and only if it coincides on γ with a solution to the difference equation $\boldsymbol{L}^{(h)}u = \tilde{f}_B^{n+1}$ on $\mathbb{N}^+$. Therefore, we may think of the BEP as a reduction of the discrete problem from the discretized domain $\mathbb{N}^+$ to the discrete boundary γ.

Incorporating the boundary conditions (7b) on the circle (see Sections 3.5–3.6) to supplement the BEP (14) yields the unique solution ξ_γ, which allows us to approximate the solution to the BVP (7) on the disk Ω as follows:

$$u|_{\mathbb{N}^+} = \boldsymbol{P}_{\mathbb{N}^+}\xi_\gamma + \left(\boldsymbol{G}^{(h)}\tilde{f}_B^{n+1}\right)\Big|_{\mathbb{N}^+}. \tag{15}$$

One may note that the definition of the difference potential (12) depends on the operator $\boldsymbol{G}^{(h)}$ and therefore on the specific choice of the auxiliary problem (8) (i.e., the shape of the auxiliary domain and its boundary conditions). However, as long as the AP has a unique solution for every right-hand side g and is well-posed, one can show that the range of the projection operator $\boldsymbol{P}_\gamma$ of (13) does not depend on the choice of the AP, see [37].

It is important to note that the operators $\boldsymbol{P}_{\mathbb{N}^+}$ and $\boldsymbol{P}_\gamma$, see (12)–(13), as well as the key equation (14), are constructed independently at the discrete level. They, however, can be considered as formally "mirroring" the continuous development that leads to Calderon's potentials and Calderon's boundary equations with projections, see [40,41]. The parallel between the discrete and continuous constructs is explained, e.g., in our recent work [33]. For a more detailed analysis and rigorous justification of all the results we refer the reader to work by Ryaben'kii [36,37]. The discrete potentials and projections (12)–(13) approximate their continuous counterparts provided that there is a certain relation between the respective densities, as discussed in Section 3.5.

3.5. Equation-based extension from the continuous to discrete boundary

While the argument (i.e., density) of the difference potential (12) is a function defined on the boundary nodes of the discrete boundary γ, the density of the continuous Calderon potential is a pair of functions defined on the continuous boundary Γ [37,41,40]. In the case that this function pair is the solution of the modified Helmholtz equation (7a) and its normal derivative at the continuous boundary Γ, the Calderon potential with density $\left(u, \frac{\partial u}{\partial \boldsymbol{n}}\right)\big|_\Gamma$ will reconstruct this solution on the interior of Ω. The equation-based extension procedure will exploit the continuous wave equation (1a) to obtain normal derivatives of its solution in order to enforce a special relationship between the discrete density ξ_γ and the continuous function pair $\left(u, \frac{\partial u}{\partial \boldsymbol{n}}\right)\big|_\Gamma$ so that the resulting difference potential (15) will approximate the solution to the boundary value problem (7) on $\mathbb{N}^+$ at each time step.

Given a function pair at the continuous boundary, $\left(v, \frac{\partial v}{\partial \boldsymbol{n}}\right)\big|_\Gamma$, the Taylor expansion to a near-boundary node of γ takes the form:

$$v_\gamma \overset{\text{def}}{=} \boldsymbol{Ex}\left(v, \frac{\partial v}{\partial \boldsymbol{n}}\right)\Big|_\Gamma = v_\Gamma + \rho\,\frac{\partial v}{\partial \boldsymbol{n}}\Big|_\Gamma + \frac{\rho^2}{2}\,\frac{\partial^2 v}{\partial \boldsymbol{n}^2}\Big|_\Gamma + \frac{\rho^3}{6}\,\frac{\partial^3 v}{\partial \boldsymbol{n}^3}\Big|_\Gamma + \mathcal{O}\left(\rho^4\right), \tag{16}$$

where ρ represents the distance from the boundary nodes of γ to the continuous curve Γ. The 2nd order and higher normal derivatives in (16) will be replaced by expressions derived from the wave equation (1a). In the case of a disk centered at the origin, the normal direction coincides with the polar radius, $\boldsymbol{n} = r$. For more general boundary shapes, similar results are obtained in normal and tangential coordinates, see [55]. Rearranging the continuous wave equation (1a) for v and differentiating in r yields:

$$\Delta v = \frac{1}{c^2}\left(v_{tt} - F\right), \tag{17a}$$

$$\frac{\partial\left(\Delta v\right)}{\partial r} = \frac{1}{c^2}\left(\frac{\partial v_{tt}}{\partial r} - \frac{\partial F}{\partial r}\right) - \frac{2}{c^3}\frac{\partial c}{\partial r}\left(v_{tt} - F\right). \tag{17b}$$

By assumption, the wave speed c and the inhomogeneous term F of the wave equation (1a) are given and can be computed with sufficient accuracy along with their radial derivatives.

Using the polar form of the Laplacian, we can write:

$$\frac{\partial^2 v}{\partial r^2} = -\left(\frac{1}{r}\frac{\partial v}{\partial r} + \frac{1}{r^2}\frac{\partial^2 v}{\partial \theta^2}\right) + \Delta v. \tag{18a}$$

Differentiating with respect to r, we have

$$\frac{\partial^3 v}{\partial r^3} = \frac{1}{r^2}\frac{\partial v}{\partial r} - \frac{1}{r}\frac{\partial^2 v}{\partial r^2} + \frac{2}{r^3}\frac{\partial^2 v}{\partial \theta^2} - \frac{1}{r^2}\frac{\partial^3 v}{\partial r \partial \theta^2} + \frac{\partial(\Delta v)}{\partial r}. \tag{18b}$$

Therefore, to compute the extension (16) with accuracy $\mathcal{O}\left(\rho^4\right)$ using (17) and (18), it remains to show how to compute the terms v_{tt} and $\frac{\partial v_{tt}}{\partial r}$ in (17). These terms are needed along the continuous boundary Γ at time level t_{n+1}. We follow the approach of [53,54] that involves backwards differences.

Formulae (17) will be evaluated at the time level t_{n+1} using a relation in the form of a 2nd order backwards difference formula in time for v_{tt},

$$v_{tt}^{n+1} := \frac{1}{h_t^2}\left(2v - 5u^n + 4u^{n-1} - u^{n-2}\right), \tag{19}$$

where v is, in general, an input function that is not dependent on time and the backwards time levels of u are known terms. In the special case that $v = u^{n+1}$, then equation (19) is exactly a second order backwards difference formula for $u_{tt}^{n+1} = v_{tt}^{n+1}$. Notice that (19) contains one more backwards time level than the time discretization (3), and so at the first time step u^{-1} is computed by a similar Taylor expansion as for u^1 in (5) (note that u^{-1} only needs to be approximated at the continuous boundary Γ for the extension (16)).

We observe that while the core finite difference scheme (2)–(3) involves three time steps, having the backward difference (19) included into the extension operator makes the overall time marching algorithm a four step scheme. The energy stability proof given in [16] applies to the core scheme (2)–(3) without boundary conditions or with Dirichlet boundary conditions. We have not conducted a theoretical stability analysis in the case with other types of boundary conditions, and it may be a challenging task. However, our numerical simulations presented in Section 4.2 show no indication of instability that can be attributed to the boundary conditions (in fact, no indication of instability at all as long as the stability constraints discussed in Section 2 are met). Thus, we conclude that using the four step backward difference (19) as a part of the extension (16) does not affect the overall stability justified for the original three step scheme.

Note also that (19) appears in the $\mathcal{O}\left(\rho^2\right)$ term of (16). Since the previous time levels are assumed to be known with only $\mathcal{O}\left(h_t^4\right)$ accuracy, the accuracy of formula (19) becomes $\mathcal{O}\left(h_t^2\right)$, bringing the overall accuracy of (16) to $\mathcal{O}\left(\rho^2 h_t^2\right)$. Assuming $h_t \approx \rho$, this results in at best $\mathcal{O}\left(\rho^4\right)$ accuracy of the extension (16) — this assumption is discussed in more detail in the remarks at the end of this section. A higher order backwards difference formula for v_{tt} will still contain a multiplication by h_t^{-2} while using $\mathcal{O}\left(h_t^4\right)$ solutions from prior time levels, and so would result in the same overall accuracy. Thus, the highest order extension that can be expected by this procedure is $\mathcal{O}\left(\rho^4\right)$.

Therefore, for the special case when $\frac{\partial v}{\partial r} = \frac{\partial u}{\partial r}^{n+1}$, the term $\frac{\partial v_{tt}}{\partial r}$ in (17) needs only to be approximated to $\mathcal{O}(\rho)$ to maintain $\mathcal{O}\left(\rho^4\right)$ accuracy overall. The central difference in time at the previous time level is sufficient:

$$\frac{\partial v_{tt}}{\partial r}^{n+1} = \frac{1}{h_t^2}\left(\frac{\partial v}{\partial r} - 2\frac{\partial u^n}{\partial r} + \frac{\partial u^{n-1}}{\partial r}\right), \tag{20}$$

where again $\frac{\partial v}{\partial r}$ can be interpreted as a general input function, while the backwards time levels are known. Assuming that the radial derivative at previous time steps are known with at least $\mathcal{O}\left(h_t^3\right)$ accuracy, the approximation (20) is $\mathcal{O}(h_t) \approx \mathcal{O}(\rho)$. Equation (20) requires the radial derivatives along Γ at the first two time steps, which are computed from the initial condition by finite differences in r for u^0 and u^1. If desired, we may increase the accuracy of the approximation to $\frac{\partial v_{tt}}{\partial r}$ to $\mathcal{O}\left(h_t^2\right)$ by replacing (20) by the same backwards difference formula as for u_{tt}^{n+1} in (19).

In the course of the MDP implementation, the extension (16) will need to be applied to an arbitrary pair of functions $(\xi_0, \xi_1)\big|_\Gamma$ that substitute for $\left(v, \frac{\partial v}{\partial \boldsymbol{n}}\right)\big|_\Gamma$. In doing so, ξ_0 will also replace v on the right-hand side of (19) and ξ_1 will replace $\frac{\partial v}{\partial r}$ on the right-hand side of (20). Following the approach for inhomogeneous elliptic equations (see, e.g., [1,33]), it will be useful to separate the extension operator into two components: $\boldsymbol{Ex}_H$ and $\boldsymbol{Ex}_I$. The component $\boldsymbol{Ex}_H$ will rely on the input functions (ξ_0, ξ_1) while $\boldsymbol{Ex}_I^{n+1}$ is independent of the inputs but will change in time, so that the extension can be thought of as an affine operator:

$$\boldsymbol{Ex}(\xi_0, \xi_1) = \boldsymbol{Ex}_H(\xi_0, \xi_1) + \boldsymbol{Ex}_I^{n+1}. \tag{21}$$

In formula (21), the linear operator $\boldsymbol{Ex}_H$ consists of those and only those terms that involve ξ_0, ξ_1 (or v and $\frac{\partial v}{\partial n}$), and their tangential derivatives, while the function $\boldsymbol{Ex}_I^{n+1}$ contains known quantities such as the source term F of the wave equation (1a) or the numerically computed solution at previous time steps that appear in the expressions (19) and (20). Note, that the finite differences for u_{tt}^{n+1} and $\frac{\partial u_{tt}^{n+1}}{\partial r}$ involve the current time level t_{n+1} which is unknown. However, in the equation-based extension the upper time level coincides with the input functions $v = \xi_0$ and $\frac{\partial v}{\partial r} = \xi_1$ and are thus a part of $\boldsymbol{Ex}_H$.

The motivation for this decomposition is that $\boldsymbol{Ex}_H$ and $\boldsymbol{Ex}_I^{n+1}$ are computed separately to avoid redundancy. For elliptic problems, this allows for the efficient solution of multiple BVPs with shared geometry since $\boldsymbol{Ex}_H(\xi_0, \xi_1)$ is computed once, so that only $\boldsymbol{Ex}_I$ needs to be recomputed when solving problems with different boundary conditions or different inhomogeneous terms. This same idea contributes to an efficient time marching in the high-order MDP algorithm for the wave equation, since $\boldsymbol{Ex}_H$ is completely determined by time-invariant input functions and can thus be accomplished in a pre-processing stage (see Sections 3.6 and 3.7), while $\boldsymbol{Ex}_I^{n+1}$ is updated at each time step. In contrast to our work on elliptic equations, $\boldsymbol{Ex}_I^{n+1}$ is not known analytically.

Denote the coefficients of the backwards difference formula (19) by d_0^{n-j}, $j = -1, 0, 1, 2$, and those of the (20) by d_1^{n-j}, $j = -1, 0, 1$. Altogether, the extension is given by $\boldsymbol{Ex}(\xi_0, \xi_1) = \boldsymbol{Ex}_H(\xi_0, \xi_1) + \boldsymbol{Ex}_I^{n+1}$ with

$$\begin{aligned}\boldsymbol{Ex}_H(\xi_0,\xi_1) &= \xi_0 + \rho\xi_1 + \frac{\rho^2}{2}\left(-\frac{1}{r}\xi_1 - \frac{1}{r^2}\frac{\partial^2\xi_0}{\partial\theta^2} + \frac{d_0^{n+1}}{c^2}\xi_0\right)\\ &\quad + \frac{\rho^3}{6}\left[\frac{2}{r^2}\xi_1 + \frac{3}{r^3}\frac{\partial^2\xi_0}{\partial\theta^2} - \frac{1}{r^2}\frac{\partial^2\xi_1}{\partial\theta^2}\right.\\ &\quad \left.+\frac{1}{c^2}d_1^{n+1}\xi_1 + \left(-\frac{1}{rc^2} - \frac{2}{c^3}\frac{\partial c}{\partial r}\right)d_0^{n+1}\xi_0\right], \end{aligned} \tag{22a}$$

$$\begin{aligned}\boldsymbol{Ex}_I^{n+1} &= \frac{\rho^2}{2c^2}\left(\sum_{j=0}^{4} d_0^{n-j}u^{n-j} - F^{n+1}\right) + \frac{\rho^3}{6}\left[-\frac{1}{c^2 r}\left(\sum_{j=0}^{4} d_0^{n-j}u^{n-j} - F^{n+1}\right)\right.\\ &\quad \left.-\frac{2}{c^3}\frac{\partial c}{\partial r}\left(d_1^n\frac{\partial u}{\partial r}^n + d_1^{n-1}\frac{\partial u}{\partial r}^{n-1} - \frac{\partial F}{\partial r}^{n+1}\right)\right]. \end{aligned} \tag{22b}$$

While the coefficients d_0^j and d_1^j in formulae (22) are specified by (19) and (20), respectively, they may be replaced by other difference formula if desired. It is assumed that the tangential derivatives of the input functions ξ_0 and ξ_1 are known. Provided that these input functions do not change with time, $\boldsymbol{Ex}_H$ in (22a) does not change in time either. In Section 3.6, we specify a time-independent basis for pairs of functions that will be the input functions. Meanwhile, $\boldsymbol{Ex}_I^{n+1}$ in (22b) changes with each time step and is not dependent on the input functions.

It was shown that $\mathcal{O}(\rho^4)$ is the highest order of the extension (16) that can be achieved by our method for the wave equation (1a) using a 4th order FD scheme. This approach is valid for boundary conditions of any type on the circle and makes use of the wave equation (1a) to derive the equation-based expressions, and the numerical results of Section 4 verify the overall 4th order accuracy of the algorithm for Dirichlet BCs, while the results for Neumann BCs have a slightly lower convergence rate but are still high-order accurate.

It is also possible to proceed by equation-based substitution of the modified Helmholtz (steady-state) equation (7a). In that case, a $\mathcal{O}(\rho^3)$ extension can be obtained for Dirichlet boundary conditions, but it is unclear how to obtain a higher order extension or how to treat general boundary conditions. Due to these limitations, the derivation of the equation-based extension (16) using the modified Helmholtz equation (7a) is omitted.

Remarks on accuracy

- As a consequence of Reznik's theorem [58,59], for a scheme of order $p = 4$ and a PDE of order $q = 2$, an extension operator based on the Taylor formula of order $p + q = 6$ is sufficient to maintain the overall fourth order accuracy of the MDP algorithm. Our previous work on the Helmholtz equation [33,1,35] has shown experimentally that a Taylor expansion of order 4, i.e., only $\mathcal{O}(\rho^5)$, is sufficient to maintain the overall 4th order accuracy with a 4th order FD scheme. For the wave equation (1a), formulae (22) specify an equation-based operator of the form (16) with an accuracy of $\mathcal{O}(\rho^4)$. Yet the numerical results in Section 4 show overall 4th order accuracy for the wave equation (1a) with Dirichlet boundary conditions and also high-order accuracy for Neumann boundary conditions, even with variable wave speed. Reducing to a $\mathcal{O}(\rho^3)$ extension resulted in a diminished convergence rate for the wave equation.
- The calculation of $\boldsymbol{Ex}_I^{n+1}$ in (22b) uses the numerical solution and its numerical normal derivative on Γ found at previous time steps by the high-order MDP algorithm. In order to simplify the analysis, we assume these quantities are known with $\mathcal{O}(h_t^4 + h_x^4)$ accuracy. The accuracy of the overall scheme depends on the accuracy of the extension, and likewise the accuracy of the extension depends on the accuracy of the overall scheme at prior times. Provided that the $\mathcal{O}(\rho^4)$ extension is sufficient to maintain fourth order accuracy in space and time and that the FD scheme for the wave equation is stable, then sufficient accuracy in the initial conditions ensures that the design convergence rate of the scheme is maintained at each step.

3.6. Basis functions on Γ

Let us now introduce a basis for function pairs on the continuous boundary Γ. The coefficients of the solution to the BVP (7) and its normal derivative in this basis will be obtained by substituting extension (22) into the discrete BEP (14) and incorporating the boundary condition (7b) on the circle. This yields a system of linear equations for the unknown coefficients of $\left(u, \frac{\partial u}{\partial \boldsymbol{n}}\right)|_\Gamma$ at each time step.

Since Γ is a closed curve (i.e., the circle), we assume that the boundary data $\left(u, \frac{\partial u}{\partial n}\right)\big|_\Gamma$ will be periodic at each time step. We introduce a set of $2N + 1$ complex Fourier basis functions on Γ:

$$\left\{e^{ij\theta}, \theta \in [0, 2\pi), j = -N, .., N\right\}, \tag{23}$$

and form pairs $\left\{\left(e^{ij\theta}, 0\right)\right\}$ and $\left\{\left(0, e^{ij\theta}\right)\right\}$, so that altogether there are $2(2N + 1)$ basis functions for approximating a general function pair (ξ_0, ξ_1) on Γ. In particular, for each time step we may write

$$\left(u, \frac{\partial u}{\partial r}\right)\Big|_{\Gamma} \approx \sum_{j=-N}^{N} \left[c_j^{(0)} \left(e^{ij\theta}, 0\right) + c_j^{(1)} \left(0, e^{ij\theta}\right)\right], \tag{24}$$

where the time level superscripts are omitted and understood to be at the current time t_{n+1}. The number of basis functions should be chosen such that the accuracy of (24) will exceed the accuracy of the overall algorithm. Series expansions using Fourier basis functions converge rapidly for smooth functions, and thus we expect the number of basis functions to be small compared to the number of points in the grid boundary, $4N + 2 \ll |\gamma|$.

Observe that the component $\boldsymbol{Ex}_H(\xi_0, \xi_1)$ of the extension operator (22a) is linear since all of the expressions involved in (16) and formulae (18) are linear differential operators. Therefore, applying the extension operator $\boldsymbol{Ex}$ to the basis expansion (24), we represent an approximation to u^{n+1} at the grid boundary γ by

$$u_\gamma^{n+1} = \sum_{j=-N}^{N} \left[c_j^{(0)} \boldsymbol{Ex}_H \left(e^{ij\theta}, 0\right) + c_j^{(1)} \boldsymbol{Ex}_H \left(0, e^{ij\theta}\right)\right] + \boldsymbol{Ex}_I^{n+1}. \tag{25}$$

Let $\boldsymbol{I}_\gamma$ denote the identity operator on γ. We obtain a system of linear equations for the coefficients of the boundary data by substituting (25) into the BEP (14),

$$\boldsymbol{P}_\gamma u_\gamma + \left(\boldsymbol{G}^{(h)} \tilde{f}_B^{n+1}\right)\Big|_\gamma = u_\gamma^{n+1},$$

or, equivalently,

$$(\boldsymbol{P}_\gamma - \boldsymbol{I}_\gamma) u_\gamma^{n+1} = -\left(\boldsymbol{G}^{(h)} \tilde{f}_B^{n+1}\right)\Big|_\gamma. \tag{26}$$

Let $\boldsymbol{Q}_H(\xi_0, \xi_1) = \left(\boldsymbol{P}_\gamma - \boldsymbol{I}_\gamma\right) \boldsymbol{Ex}_H(\xi_0, \xi_1)$ and $\boldsymbol{Q}_I^{n+1} = \left(\boldsymbol{P}_\gamma - \boldsymbol{I}_\gamma\right) \boldsymbol{Ex}_I^{n+1}$. Then, substituting the right-hand side of (25) into (26) and rearranging, we have

$$\sum_{j=-N}^{N} \left[c_j^{(0)} \boldsymbol{Q}_H \left(e^{ij\theta}, 0\right) + c_j^{(1)} \boldsymbol{Q}_H \left(0, e^{ij\theta}\right)\right] = -\left(\boldsymbol{G}^{(h)} \tilde{f}_B^{n+1}\right)\Big|_\gamma - \boldsymbol{Q}_I^{n+1}. \tag{27}$$

Let $\boldsymbol{Q} = [\boldsymbol{Q}_0, \boldsymbol{Q}_1]$ be the $|\gamma| \times 2(2N+1)$ matrix whose first $2N+1$ columns $\boldsymbol{Q}_0$ are given by $\boldsymbol{Q}_H\left(e^{ij\theta}, 0\right)$, $-N \le j \le N$, and last $2N+1$ columns $\boldsymbol{Q}_1$ by $\boldsymbol{Q}_H\left(0, e^{ij\theta}\right)$, $-N \le j \le N$, and let $\boldsymbol{c} = \left[c_{-N}^{(0)}, \ldots, c_N^{(0)}, c_{-N}^{(1)}, \ldots, c_N^{(1)}\right]^T$. Then we may write the linear system (27) with $|\gamma|$ equations and $4N+2$ unknowns as

$$\boldsymbol{Q}\boldsymbol{c} = -\left(\boldsymbol{G}^{(h)} \tilde{f}_B^{n+1}\right)\Big|_\gamma - \boldsymbol{Q}_I^{n+1}. \tag{28}$$

Since N will be relatively small for smooth functions, we will typically have $4N + 2 < |\gamma|$ or even $4N + 2 \ll |\gamma|$ so that system (28) is overdetermined. Because the BEP (14) does not incorporate the boundary conditions on the disk, the system (28) requires additional information in order to specify a unique solution before it can be solved. For example, in the case that (1d) is a Dirichlet condition, at time t_{n+1} we compute the Fourier expansion of the Dirichlet data

$$u^{n+1}\big|_\Gamma = \phi_0^{n+1}(\theta) = \sum_{j=-N}^{N} c_j^{(0)} e^{ij\theta}, \tag{29}$$

so that $\boldsymbol{c}^{(0)}$ is known. Substituting into (28) yields

$$\boldsymbol{Q}_1 \boldsymbol{c}^{(1)} = -\left(\boldsymbol{G}^{(h)} \tilde{f}_B^{n+1}\right)\Big|_\gamma - \boldsymbol{Q}_I^{n+1} - \boldsymbol{Q}_0 \boldsymbol{c}^{(0)}. \tag{30a}$$

The same argument follows for a Neumann boundary condition and gives

$$\boldsymbol{Q}_0 \boldsymbol{c}^{(0)} = -\left(\boldsymbol{G}^{(h)} \tilde{f}_B^{n+1}\right)\Big|_\gamma - \boldsymbol{Q}_I^{n+1} - \boldsymbol{Q}_1 \boldsymbol{c}^{(1)}. \tag{30b}$$

It is shown in [1] how to impose the proper constraints for general Robin boundary conditions, but only Dirichlet and Neumann BCs are considered in the present work. Linear systems (30) are typically overdetermined, because the dimension of the basis (23) is smaller than the number of nodes in the grid boundary γ. Therefore, they need to be solved in the weak sense, i.e., by minimizing the appropriately chosen norm of the residual, see [36–39]. Choosing the norm to be minimized

as the Euclidean norm leads to the solution in the sense of least squares.[2] Solving either (30a) or (30b) by least squares at each time step yields the Fourier coefficients of $\left(u^{n+1}, \frac{\partial u}{\partial n}^{n+1}\right)\Big|_\Gamma$, from which we compute u_γ^{n+1} by (25). We then obtain the numerical solution on $\mathbb{N}^+$ by constructing the difference potential with density u_γ^{n+1}, see (15).

3.7. Overview of the high-order MDP algorithm for the wave equation and its cost analysis

In the algorithm below, components (a)–(c) represent the pre-processing steps. Not only are these steps computed just once for a given problem, but they may, in fact, be reused. In other words, they can be computed only once for a range of related problems with a shared geometry. For example, if we wish to solve several problems on the same domain Ω but with different boundary conditions (as done in Section 4) or different source terms, then steps (a)–(b) are performed only once provided that the same set of basis functions is suitable. In particular, the algorithm requires only very simple (and negligible from the standpoint of cost) changes for treating Dirichlet or Neumann boundary conditions. This has also been extended to Robin boundary conditions and in fact, the algorithm will remain valid for any smooth, periodic boundary conditions without additional analysis, just by adding the boundary conditions to the BEP system solved by least squares. An even broader class of boundary conditions has been analyzed in [1].

The component (c) below is unique to each problem but is computationally trivial, while step (d) and its subparts represent the time marching algorithm that is required for each problem. We emphasize that while in this paper we have chosen Ω to be a circle, this choice was made only for simplicity, because our primary focus is on time-dependent computations rather than the geometry. The MDP can, in fact, easily handle boundaries of irregular shape, as demonstrated, e.g., in our recent time-harmonic paper [55].

(a) We choose the number N so that a Fourier expansion w.r.t. $2N+1$ basis functions on the circle Γ (23) sufficiently accurately approximates the given Dirichlet or Neumann boundary condition (1d). Then, there are $4N+2$ basis functions total, with $2N+1$ of the form $(e^{ij\theta}, 0)$ and $2N+1$ of the form $(0, e^{ij\theta})$, with $-N \le j \le N$. The homogeneous extension $\boldsymbol{Ex}_H$ of (22a) is applied to each of the $4N+2$ basis functions, resulting in $4N+2$ extended basis vectors with values on the grid boundary γ.

(b) The extended basis vectors are substituted into the boundary equation with projection (14), which requires the evaluation of the difference potential with density given by each extended basis vector. Altogether, this step involves $4N+2$ numerical solutions of the auxiliary problem (8) by finite differences on the square with different right-hand sides g. The latter are specified by the expression in square brackets on the right-hand side of (12) with each extended basis vector as a density.

(c) The first time step u^1 is computed by the initial conditions (1b)–(1c) via the Taylor expansion (5). Using a Taylor expansion of u^0 and u^1 from Γ to the grid nodes $\mathbb{N}^+ \setminus \mathbb{M}^+$, we compute f_B^0 and f_B^1 on $\mathbb{M}^+$ by applying the compact FD operator (9) to the values of u^0 and u^1 on $\mathbb{N}^+$ (see Section 3.3). On Γ, we also compute u^{-1} and $\frac{\partial u^0}{\partial r}$, $\frac{\partial u^1}{\partial r}$ from the initial conditions.

(d) At time step t_{n+1},

(i) Compute $\boldsymbol{Ex}_I^{n+1}$ by (22b) and compute $\boldsymbol{Q}_I^{n+1} = (\boldsymbol{P}_\gamma - \boldsymbol{I}_\gamma)\boldsymbol{Ex}_I^{n+1}$. Applying the difference projection $\boldsymbol{P}_\gamma$ requires computing a difference potential of density $\boldsymbol{Ex}_I^{n+1}$, which involves the solution of the AP (8) on the square Ω_0 by FD with the right-hand side g given by the expression in square brackets in (12) with density $\boldsymbol{Ex}_I^{n+1}$.

(ii) Compute $\tilde{f}_B^{n+1}$. This requires two matrix-vector multiplications and storing two vectors, see Section 3.3.

(iii) Compute $\left(\boldsymbol{G}^{(h)}\tilde{f}_B^{n+1}\right)\Big|_\gamma$ on the right-hand side of the linear system (28). This requires one solution of the AP by FD with $g = \tilde{f}_B^{n+1}$.

(iv) Compute the Fourier coefficients of the given boundary data from (7b).

(v) Solve the least squares system (30a) for a Dirichlet BC ($\boldsymbol{c}^{(0)}$ is given and fixed):

$$\boldsymbol{c}^{(1)} = \arg\min_{\tilde{\boldsymbol{c}}^{(1)}} \left\| \boldsymbol{Q}_1\tilde{\boldsymbol{c}}^{(1)} + \left(\boldsymbol{G}^{(h)}\tilde{f}_B^{n+1}\right)\Big|_\gamma + \boldsymbol{Q}_I^{n+1} + \boldsymbol{Q}_0\boldsymbol{c}^{(0)} \right\|_2,$$

or (30b) for a Neumann BC ($\boldsymbol{c}^{(1)}$ is given and fixed):

$$\boldsymbol{c}^{(0)} = \arg\min_{\tilde{\boldsymbol{c}}^{(0)}} \left\| \boldsymbol{Q}_0\tilde{\boldsymbol{c}}^{(0)} + \left(\boldsymbol{G}^{(h)}\tilde{f}_B^{n+1}\right)\Big|_\gamma + \boldsymbol{Q}_I^{n+1} + \boldsymbol{Q}_1\boldsymbol{c}^{(1)} \right\|_2.$$

This yields the approximate Fourier coefficients $\boldsymbol{c}$ for the full boundary data $\left(u^{n+1}, \frac{\partial u}{\partial r}^{n+1}\right)\Big|_\Gamma$. Numerically, the least squares minimization is implemented by QR decomposition.

[2] As long as the original elliptic BVP (7) has a unique solution, the weak solution of the corresponding system (30a) or (30b) is expected to be "almost" classical in the sense that the minimum norm of the residual will be achieved at zero, or, more precisely, it will be a quantity of order $\mathcal{O}(h^p)$, where $p = 4$ is the order of accuracy of the finite difference scheme.

(vi) Compute u_γ^{n+1} by the basis expansion (25). Then, the numerical solution u^{n+1} on $\mathbb{N}^+$ is given by (15), which is found by computing the difference potential with density u_γ^{n+1} and adding the inhomogeneous term. This requires one more solution of the AP by finite differences with right-hand side g given by the expression in square brackets in (12) with density u_γ^{n+1}.

We emphasize that while the treatment of the boundary conditions in our algorithm consists of a number of stages, the overall procedure remains perfectly linear and eventually reduces to solving an overdetermined system of linear algebraic equations in the sense of the least squares.

The largest computational costs of the algorithm are the steps that require solving the AP (8) by FD with a new right-hand side. Therefore, the most expensive single step of the above algorithm is step (b), which requires $4N+2$ solutions of the AP as a pre-processing stage. The time marching of (d) requires altogether 3 solutions of the AP by FD with new right-hand sides per time step, and this number can be reduced to 2 as explained at the end of this section.

We emphasize that the pre-processing can be reused for any problem for which the same basis functions can be employed, so that only the cost of time marching is required for solving additional problems within a suitable class. In particular, this applies to problems with different boundary or initial conditions on the same geometry. If a direct linear solver is used for the AP (8), which is the case for the current 2D implementation, then the LU factorization needs to be performed only once at the pre-processing stage. Each subsequent solution of the AP, including those 2 solutions per time step that are needed for time marching, involves only the new right-hand side. Hence, it requires only backward substitution and incurs only a quadratic computational cost with respect to the grid dimension. Moreover, in the case of a constant propagation speed, which is not the most general one but nonetheless important, the AP on the Cartesian grid that is not tied to the geometry of Ω can be solved by FFT. As the complexity of the latter is essentially linear (neglecting the log factor), the overall cost of time marching is equivalent to that of an explicit scheme. At the same time, the compact stencil of the scheme (2)–(3) avoids having to set the additional non-physical initial/boundary conditions that are always required for explicit/non-compact high-order accurate finite difference schemes. In 3D, the LU decomposition is likely to become prohibitively expensive. However, multigrid iterations that are also characterized by linear complexity have shown superior efficiency for the modified Helmholtz equation, as did conjugate gradient iterations, see our recent work [16].

Additionally, we observed that the final difference potential computation in step (vi) of the foregoing algorithm can be accomplished without solving a new AP, but at the expense of additional memory, bringing the total computational effort of each time step to only 2 FD solves rather than 3. After solving for the coefficients of u_γ^{n+1} in step (v), the difference potential with density u_γ^{n+1} can be written in either of the following ways by linear superposition:

$$\begin{aligned} u^{n+1}|_{\mathbb{N}^+} &= \boldsymbol{P}_{\mathbb{N}^+} u_\gamma^{n+1} + \left(\boldsymbol{G}^{(h)} \tilde{f}_B^{n+1}\right)\Big|_{\mathbb{N}^+} \\ &= \sum_{j=-N}^{N} \left[c_j^{(0)} \boldsymbol{P}_{\mathbb{N}^+} \boldsymbol{E}\boldsymbol{x}_H\left(e^{ij\theta}, 0\right) + c_j^{(1)} \boldsymbol{P}_{\mathbb{N}^+} \boldsymbol{E}\boldsymbol{x}_H\left(0, e^{ij\theta}\right)\right] \\ &\quad + \boldsymbol{P}_{\mathbb{N}^+} \boldsymbol{E}\boldsymbol{x}_I^{n+1} + \left(\boldsymbol{G}^{(h)} \tilde{f}_B^{n+1}\right)\Big|_{\mathbb{N}^+}. \end{aligned} \tag{31}$$

A difference potential for each extended basis function is computed in the pre-processing step (b) and also for $\boldsymbol{E}\boldsymbol{x}_I^{n+1}$ at step (i), but only their projections on γ are stored. If instead we store the full difference potential for each basis function and for $\boldsymbol{E}\boldsymbol{x}_I^{n+1}$, each of which is a vector of size $|\mathbb{N}|$ rather than $|\gamma|$, then (31) can be computed by simply summing the difference potentials with the coefficients $\boldsymbol{c}$. Therefore, for the cost of storing $4N+3$ vectors of size $|\mathbb{N}|$, we can avoid the computational cost of solving a FD problem on the auxiliary domain $\mathbb{N}$. To our knowledge, this observation has not been made in any previous publications and represents a substantial gain in computational efficiency. The numerical tests of Section 4 use the algorithm as presented above and so do not reflect these savings in computation time, but we have tested and verified the accuracy of this alternative approach.

An alternative to the method proposed in the current paper, for the type of problems we are targeting, is to solve the problem by high-order finite elements in space combined with a high-order explicit time-marching scheme such as the Runge–Kutta method. High-order finite elements will require grid generation, which is a separate non-trivial task for domains of general shape. Moreover, a fourth order accurate Runge–Kutta method takes four evaluations of the spatial operator per time step; each will entail linear complexity with respect to the dimension of the spatial discretization. Our full-fledged fourth order accurate method (with the boundary conditions taken into account), while implicit in time, needs only two linear or log-linear solves per time step. Given that high-order finite elements typically require many additional degrees of freedom per grid node, and our scheme maintains a single degree of freedom per node, one may expect a better overall performance from the currently proposed method.

Yet another approach is to map the general domain into a square and then use a high-order scheme coupled with a high-order one-sided approximation near the boundary. This can be constructed in a stable manner using summation by parts operators and imposing the boundary conditions weakly via simultaneous approximation terms (SATs), see e.g., [60]. Both these approaches suffer the disadvantage of the need to construct a grid that matches the general domain or

equivalently a mapping of the domain to a rectangle. In contrast, the methodology described in the current paper uses only Cartesian or polar grids.

4. Numerical tests

4.1. Computational parameters

All of the tests were carried out on a disk of radius 1 centered at the origin, $\Omega = \{(x, y) : x^2 + y^2 \leq 1\}$, embedded in a square auxiliary domain of side length $s = 2.2$, $\Omega_0 = \{(x, y) : -1.1 \leq x, y \leq 1.1\}$. We consider a test solution on the disk Ω of the form

$$u = \cos(Ax)\cos(By)\cos(\omega t), \tag{32}$$

and the source term F is found by substituting the test solution (32) into the wave equation (1a) as follows:

$$F = u_{tt} - c^2\Delta u = \left[c^2(A^2 + B^2) - \omega^2\right]u. \tag{33}$$

In each numerical example, Dirichlet or Neumann conditions on the circle $x^2 + y^2 = 1$ are specified by the test solution (32).

The initial time steps u^1 and u^{-1} are computed from the given initial conditions (1b) by (5), and for the test solution (32) this reduces to

$$u^1 = u^{-1} \approx u^0 - \frac{\omega^2}{2}h_t^2 + \frac{\omega^4}{24}h_t^4.$$

All of the following results use the implicit time discretization (3) with $\theta = \frac{1}{12}$, so that the time discretization is 4th order and the scheme is conditionally stable. The finite difference scheme (9) is used for solving the auxiliary problem (8) in the MDP algorithm, so that the stability condition is given by $\lambda^2 = \frac{c^2h_t^2}{h_x^2} \leq \frac{3}{8}$. Thus for a grid of step size h_x and maximum wave speed $c_{\max}$ on Ω_0, we take the largest time step such that $h_t \leq \frac{3}{8c_{\max}}h_x$. We take a final computation time of $t_F = 1$ for all tests. The error in Section 4.2 is quantified using the maximum norm.

For each test case, the wave speed $c(x, y)$ is given analytically as a C^∞ function on Ω which is also defined on all of Ω_0. We note that this may cause $c_{\max}$ on Ω_0 to be larger than the maximum value of c on the physical domain Ω, and thus a smaller CFL number may be needed. In practice, one may eliminate this inefficiency by using a continuously differentiable polynomial extension of c from Ω to Ω_0 which does not continue to grow on Ω_0 outside of a small neighborhood of Ω.

The Fourier basis functions were used for the series expansion of the boundary data with $N = 20$ on all grids. With 41 basis functions each for the Dirichlet and Neumann data on Γ, there are altogether 82 basis functions involved, so that the pre-processing step (b) requires 82 solutions of the AP by FD. This is sufficient to exceed the overall accuracy of the algorithm even on the finest grid, and therefore we conclude that for coarser grids the same accuracy can be achieved with fewer basis functions.

A direct LU solver is used for the solution of the AP by FD using MATLAB's built-in `lu` command, with the initial factorization being done once in the pre-processing stage of the algorithm so that subsequent backsolves of the AP for time-marching are calculated efficiently (see Section 3.7). All computations were performed on a Mac Pro with 64 GB of RAM and a 12-core Intel Xeon Processor E5-v2 at 2.7 GHz.

4.2. Numerical results

The pre-processing steps (a)–(b) of the MDP algorithm (see Section 3.7) are performed only once for each new test case even though two different boundary conditions are solved for. Even though all test cases use the same geometry, their wave speeds are different and thus pre-processing is necessary. Since we use a direct LU solver, the pre-processing includes the LU factorization of the FD matrix. Table 1 displays the pre-processing time on each grid corresponding to the first test case below; however, it is representative of the cost for subsequent test cases as well since the amount of work done is affected only by the grid size and number of basis functions, which are the same across all test cases. The times listed in subsequent tables correspond only to treating the initial conditions in (c) and the time-marching steps (d).

In the numerical examples of Tables 2–4, the values $A = 2$ and $B = 5$ were used in all tests. In Tables 2–4, the errors and convergence rates for Dirichlet and Neumann BCs are provided along with the CPU time that is unique to each boundary condition and is added on top of the pre-processing times of Table 1 (recall, the pre-processing is done only once for each test case). Because there is virtually no change in the MDP algorithm between the Dirichlet and Neumann BCs, the same amount of computational work is done for each. Therefore only the time required for the Neumann problems are displayed in each of Tables 2–4 as a representative case. These running times are dominated by 3 solutions of the AP by FD per time marching step; however, as discussed in the final comment of Section 3.7, this may be reduced to only 2 solutions of the AP by FD at the expense of storing $2(2N_0 + 1) + 1$ vectors on the full computational grid $\mathbb{N}$.

Table 1
The time required for pre-processing steps (a)–(b) of the MDP algorithm (see Section 3.7), which includes the LU factorization of the FD matrix and constructing the difference potential (i.e., solving an AP) for each of the 82 basis functions. The pre-processing is performed only once for each of the ensuing computational examples even though two problems with different BCs are solved. Additional problems with different BCs or different inhomogeneous source terms can make use of the same pre-processing.

Grid	Setup time (s)
32	0.55
64	1.07
128	2.47
256	6.30
512	26.6
1024	117

Table 2
Homogeneous test solution $u = \cos 2x \cos 5y \cos \sqrt{29}t$ with constant wave speed $c = 1$ and the CFL restriction $h_t = 0.6h_x$.

Grid	Dirichlet BC		Neumann BC		# Time steps	Time (s)
	Error	Conv. rate	Error	Conv. rate		
32	1.59×10^{-3}	–	1.23×10^{-3}	–	24	0.088
64	8.61×10^{-5}	4.21	3.24×10^{-4}	1.92	48	0.37
128	5.74×10^{-6}	3.91	2.74×10^{-5}	3.56	96	2.38
256	3.37×10^{-7}	4.09	2.79×10^{-6}	3.30	191	20.8
512	2.17×10^{-8}	3.96	5.53×10^{-7}	2.33	382	210
1024	1.37×10^{-9}	3.99	6.69×10^{-8}	3.05	764	2220

Table 3
Inhomogeneous test solution $u = \cos 2x \cos 5y \cos 6.46t$ with wave speed $c = \frac{x^2+y^2}{4} + 1$ and CFL condition $h_t = 0.38h_x$.

Grid	Dirichlet BC		Neumann BC		# Time steps	Time (s)
	Error	Conv. rate	Error	Conv. rate		
32	1.17×10^{-3}	–	8.47×10^{-4}	–	39	0.14
64	6.83×10^{-5}	4.10	1.86×10^{-4}	2.19	77	0.58
128	4.18×10^{-6}	4.03	1.60×10^{-5}	3.54	153	3.74
256	2.60×10^{-7}	4.01	1.62×10^{-6}	3.30	305	33.2
512	1.65×10^{-8}	3.98	3.09×10^{-7}	2.40	612	330
1024	1.07×10^{-9}	3.95	3.73×10^{-8}	3.04	1224	3508

One may see that for coarse grids the pre-processing times of Table 1 are smaller than the time-marching steps of the algorithm. This is due to the arbitrary choice of the final time $t_F = 1$ and the fact that the number of time steps needed is inversely proportional to the grid step size h_x.

For the first example, we choose a constant wave speed $c = 1$ with the parameter $\omega = c\sqrt{A^2 + B^2} = \sqrt{29} \approx 5.39$, so that the wave equation (1a) is homogeneous, $F = 0$. The stability condition then becomes $h_t^2 \leq \frac{3}{8}h_x^2$, and we note that $\sqrt{\frac{3}{8}} \approx 0.612$. The results for Dirichlet and Neumann boundary conditions with a constant wave speed with a final time of $t_F = 1$ are recorded in Table 2 along with the CPU time that is unique to each problem.

Next, we consider the variable wave speed $c^2(x, y) = \frac{x^2+y^2}{4} + 1$. We take $\omega = 1.2\sqrt{29} \approx 6.46$ in the test solution (32), so that the inhomogeneous term (33) of the wave equation becomes $F = -(1.2 + c^2(x, y))\sqrt{29}u$. We have $c^2_{\max} \approx 2.57$ on Ω_0, so that the stability condition becomes $h_t^2 \leq \frac{3}{8c^2_{\max}}h_x^2 \approx 0.381h_x^2$. Therefore we take the time step $h_t = 0.38h_x$. Table 3 displays the results for the Dirichlet and Neumann problems with a final time of $t_F = 1$.

For yet one more example, we take the variable wave speed $c^2(x, y) = e^x$. We again let $\omega = 1.2\sqrt{29} \approx 6.46$ for the test solution (32), and the inhomogeneous term (33) is $F = -(1.2 + c^2(x, y))\sqrt{29}u$. The maximum value of c on the domain is $c^2_{\max} = e^{1.1} \approx 3.004$ and the CFL condition becomes $h_t^2 \leq \frac{3}{8c^2_{\max}}h_x^2 \approx 0.0415h_x^2$ or $h_t \leq 0.2038h_x$. We thus take $h_t = 0.2h_x$. Results for the Dirichlet and Neumann problems are summarized in Table 4.

Our final numerical example is designed to demonstrate the performance of the proposed method in the case of an oscillatory wave speed: $c(x, y) = 1 + \sin^2 5\pi x \cos^2 2\pi y$. The parameters of the test solution (32) are $A = 5$, $B = 3$, and $\omega = 4$, and the right-hand side of the wave equation $F = F(x, y, t)$ is taken in accordance with (33). All other parameters, such as the geometry, the dimension of the Fourier basis, etc., are the same as in the previous examples. The results for the oscillatory wave speed are shown in Table 5.

The difference in running times for Tables 2–5 is directly proportional to the number of time steps performed, which is determined by the accompanying CFL restrictions of each problem according to the maximum wave speed on Ω_0.

Table 4
Inhomogeneous test solution $u = \cos 2x \cos 5y \cos 6.46t$ with wave speed $c = e^x$ and CFL condition $h_t = 0.2h_x$.

Grid	Dirichlet BC		Neumann BC		# Time steps	Time (s)
	Error	Conv. rate	Error	Conv. rate		
32	2.65×10^{-3}	–	1.44×10^{-3}	–	67	0.24
64	1.35×10^{-4}	4.30	1.86×10^{-4}	2.95	139	1.08
128	8.47×10^{-6}	3.98	1.44×10^{-5}	3.69	281	7.00
256	4.93×10^{-7}	4.10	1.51×10^{-6}	3.25	567	62.5
512	2.75×10^{-8}	4.16	3.97×10^{-7}	1.93	1035	620
1024	1.81×10^{-9}	3.93	4.03×10^{-8}	3.23	2281	6582

Table 5
Inhomogeneous test solution $u = \cos 5x \cos 3y \cos 4t$ with wave speed $c = 1 + \sin^2 5\pi x \cos^2 2\pi y$ and CFL condition $h_t = 0.3h_x$.

Grid	Dirichlet BC		Neumann BC		# Time steps	Time (s)
	Error	Conv. rate	Error	Conv. rate		
32	2.03×10^{-3}	–	1.70×10^{-3}	–	49	0.28
64	1.64×10^{-4}	3.63	3.95×10^{-4}	2.11	97	0.75
128	1.18×10^{-5}	3.80	4.75×10^{-5}	3.06	194	4.83
256	6.98×10^{-7}	4.08	6.45×10^{-6}	2.88	388	42.9
512	4.52×10^{-8}	3.95	8.31×10^{-7}	2.96	776	448
1024	3.11×10^{-9}	3.86	1.02×10^{-7}	3.03	1552	4358

The theoretical 4th order convergence rate in space and time is achieved for the Dirichlet problem in Tables 2–5 while the convergence rate for the Neumann problem is diminished by about half an order. We note that in other applications, e.g., spectral methods, it is well known that there is a loss of one order of accuracy for Neumann boundary conditions [61].

In general, one may suspect that the deterioration of the Neumann convergence rate in our simulations is due to the insufficiently high order of the extension (16), only $\mathcal{O}(\rho^4)$, while $\mathcal{O}(\rho^5)$ may be needed. On one hand, given that this extension involves backward differencing in time, improving the $\mathcal{O}(\rho^4)$ order may not be very straightforward, see equation (19) and the discussion that follows. One the other hand, for our current approach the difficulty may, in fact, be elsewhere. By substituting the exact solution, we have been able to verify unambiguously that the term in the discretization that needs to be computed more accurately in order to improve the overall Neumann convergence is $\frac{1}{\theta}\boldsymbol{B}^{(h)}\Delta u^n$ on $\mathbb{M}^+$. At the moment, it is approximated by $\frac{1}{\theta}\boldsymbol{L}_0^{(h)}u^n$ on the right-hand side of formula (11). This approximation will be improved in the future.

5. Discussion

A high-order difference potentials algorithm for the wave equation is developed on domains that do not conform to a Cartesian mesh. It uses a conditionally stable 4th order accurate implicit time discretization of the wave equation coupled with an auxiliary problem for the elliptic equation at each time step. Wave speeds that vary in space and a range of boundary conditions are easily handled. The time discretization requires only 3 time levels, while the AP is solved by a fourth order compact finite difference scheme at each time step. Thus, the overall scheme is fourth order accurate and compact both in space and time. The equation-based extension relies on the continuous wave equation (1a) and employs backwards differences in time.

For time-harmonic problems that we analyzed previously [1], an equation-based extension of at least $\mathcal{O}(\rho^5)$ accuracy is required to maintain overall 4th order convergence of the algorithm. In this paper, the design accuracy of the algorithm is achieved for Dirichlet boundary conditions, which is fourth order in space and time, even though the equation-based extension is only $\mathcal{O}(\rho^4)$. Results for the Neumann boundary conditions fall slightly short of the design rate but are still high-order.

The algorithm is efficient, requiring 3 solutions of the AP on a square by finite differences per time step to obtain a high-order accurate solution on the nonconforming domain of interest, which can be reduced to only 2 FD solves per time step by a small modification requiring additional storage. A direct LU solver is used and performs well for problems at this scale in 2D, especially since the factorization of the matrix is done only once in preprocessing. For 3D, iterative solvers such as conjugate gradients (CG) or multigrid (MG) can be used, see [16], since direct solvers become prohibitively expensive. In our simulations [16], we took the solution from the previous time level as the initial guess, and both CG and MG solvers converged very rapidly, driving the residual down almost to machine precision in only a few iterations. The corresponding accuracy of the solution to the linear system will exceed that of the finite difference approximation on the grid. We therefore expect that if an iterative solver were used for the AP (8), its accuracy can easily be made sufficiently high so that the discrete projection operators will not be noticeably affected.

In the current paper, we have considered only smoothly varying wave speeds $c = c(x, y)$. However, wave speeds that lack differentiability and may even become discontinuous can also be included. The corresponding modification of the algorithm

will be very similar to that described in our recent work [55] which treats the material discontinuities in the time-harmonic framework.

Allowing for the variation of the wave speed in time will be more challenging. While the formal constructs of the proposed algorithm will stay the same, the elliptic equation (7a) will be changing from one time level to another because the quantity k^2 depends on c. Therefore, one will no longer be able to pre-compute the boundary operators once and for all, as suggested in step (b) of the algorithm (see page 36). Instead, one will have to re-compute the projections on every time step, which will obviously cause a substantial reduction in efficiency.

References

[1] D.S. Britt, S.V. Tsynkov, E. Turkel, A high-order numerical method for the Helmholtz equation with nonstandard boundary conditions, SIAM J. Sci. Comput. 35 (5) (2013) A2255–A2292.
[2] S. Magura, S. Petropavlovsky, S. Tsynkov, E. Turkel, High-order numerical solution of the Helmholtz equation for domains with reentrant corners, Appl. Numer. Math. 118 (2017) 87–116.
[3] A. Bayliss, C.I. Goldstein, E. Turkel, On accuracy conditions for the numerical computation of waves, J. Comput. Phys. 59 (3) (1985) 396–404.
[4] Ivo M. Babuska, Stefan A. Sauter, Is the pollution effect of the FEM avoidable for the Helmholtz equation considering high wave numbers?, SIAM J. Numer. Anal. 34 (6) (1997) 2392–2423.
[5] Frank Ihlenburg, Finite Element Analysis of Acoustic Scattering, Appl. Math. Sci., vol. 132, Springer, 1998.
[6] Heinz-Otto Kreiss, Joseph Oliger, Comparison of accurate methods for the integration of hyperbolic equations, Tellus 24 (1972) 199–215.
[7] Christopher K.W. Tam, Jay C. Webb, Dispersion-relation-preserving finite difference schemes for computational acoustics, J. Comput. Phys. 107 (2) (1993) 262–281.
[8] Christopher K.W. Tam, Jay C. Webb, Radiation boundary condition and anisotropy correction for finite difference solutions of the Helmholtz equation, J. Comput. Phys. 113 (1) (1994) 122–133.
[9] S. Britt, S. Tsynkov, E. Turkel, A compact fourth order scheme for the Helmholtz equation in polar coordinates, J. Sci. Comput. 45 (1–3) (2010) 26–47.
[10] Steven Britt, Semyon Tsynkov, Eli Turkel, Numerical simulation of time-harmonic waves in inhomogeneous media using compact high order schemes, Commun. Comput. Phys. 9 (3) (March 2011) 520–541.
[11] Eli Turkel, Dan Gordon, Rachel Gordon, Semyon Tsynkov, Compact 2D and 3D sixth order schemes for the Helmholtz equation with variable wave number, J. Comput. Phys. 232 (1) (2013) 272–287.
[12] S.K. Lele, Compact finite difference schemes with spectral like resolution, J. Comput. Phys. 103 (1992) 16–42.
[13] I. Harari, E. Turkel, Accurate finite difference methods for time-harmonic wave propagation, J. Comput. Phys. 119 (2) (1995) 252–270.
[14] I. Singer, E. Turkel, High-order finite difference methods for the Helmholtz equation, Comput. Methods Appl. Mech. Eng. 163 (1–4) (1998) 343–358.
[15] I. Singer, E. Turkel, Sixth-order accurate finite difference schemes for the Helmholtz equation, J. Comput. Acoust. 14 (3) (2006) 339–351.
[16] S. Britt, S. Tsynkov, E. Turkel, A high order compact time/space finite difference scheme for the wave equation in variable media, J. Sci. Comput. (2017), Submitted for publication.
[17] B.M. Irons, O.C. Zienkiewicz, The Isoparametric Finite Element System: A New Concept in Finite Element Analysis, Royal Aeronautical Society, London, 1969.
[18] R.E. Kleinman, G.F. Roach, Boundary integral equations for the three-dimensional Helmholtz equation, SIAM Rev. 16 (1974) 214–236.
[19] George Biros, Lexing Ying, Denis Zorin, A fast solver for the Stokes equations with distributed forces in complex geometries, J. Comput. Phys. 193 (1) (2004) 317–348.
[20] Shidong Jiang, Shravan Veerapaneni, Leslie Greengard, Integral equation methods for unsteady Stokes flow in two dimensions, SIAM J. Sci. Comput. 34 (4) (2012) A2197–A2219.
[21] Mark D. Preston, Peter G. Chamberlain, Simon N. Chandler-Wilde, An integral equation method for a boundary value problem arising in unsteady water wave problems, J. Integral Equ. Appl. 20 (1) (2008) 121–152.
[22] Ha-Duong Tuong, On retarded potential boundary integral equations and their discretisation, in: Topics in Computational Wave Propagation, in: Lect. Notes Comput. Sci. Eng., vol. 31, Springer, Berlin, 2003, pp. 301–336.
[23] Francisco-Javier Sayas, Retarded Potentials and Time Domain Boundary Integral Equations: A Road-Map, University of Delaware, March 2013.
[24] Daniel S. Weile, G. Pisharody, Nan-Wei Chen, B. Shanker, E. Michielssen, A novel scheme for the solution of the time-domain integral equations of electromagnetics, IEEE Trans. Antennas Propag. 52 (1) (Jan 2004) 283–295.
[25] G. Kobidze, Jun Gao, B. Shanker, E. Michielssen, A fast time domain integral equation based scheme for analyzing scattering from dispersive objects, IEEE Trans. Antennas Propag. 53 (3) (March 2005) 1215–1226.
[26] Toufic Abboud, Patrick Joly, Jerónimo Rodríguez, Isabelle Terrasse, Coupling discontinuous Galerkin methods and retarded potentials for transient wave propagation on unbounded domains, J. Comput. Phys. 230 (15) (2011) 5877–5907.
[27] Víctor Domínguez, Francisco-Javier Sayas, Some properties of layer potentials and boundary integral operators for the wave equation, J. Integral Equ. Appl. 25 (2) (2013) 253–294.
[28] Charles L. Epstein, Leslie Greengard, Thomas Hagstrom, On the stability of time-domain integral equations for acoustic wave propagation, Discrete Contin. Dyn. Syst. 36 (8) (2016) 4367–4382.
[29] C. Lubich, Convolution quadrature and discretized operational calculus. I, Numer. Math. 52 (2) (1988) 129–145.
[30] C. Lubich, Convolution quadrature and discretized operational calculus. II, Numer. Math. 52 (4) (1988) 413–425.
[31] Silvia Falletta, Giovanni Monegato, An exact non-reflecting boundary condition for 2D time-dependent wave equation problems, Wave Motion 51 (1) (2014) 168–192.
[32] Silvia Falletta, Giovanni Monegato, Exact non-reflecting boundary condition for 3D time-dependent multiple scattering–multiple source problems, Wave Motion 58 (2015) 281–302.
[33] M. Medvinsky, S. Tsynkov, E. Turkel, The method of difference potentials for the Helmholtz equation using compact high order schemes, J. Sci. Comput. 53 (1) (2012) 150–193.
[34] M. Medvinsky, S. Tsynkov, E. Turkel, High order numerical simulation of the transmission and scattering of waves using the method of difference potentials, J. Comput. Phys. 243 (2013) 305–322.
[35] S. Britt, S. Petropavlovsky, S. Tsynkov, E. Turkel, Computation of singular solutions to the Helmholtz equation with high order accuracy, Appl. Numer. Math. 93 (July 2015) 215–241.
[36] V.S. Ryaben'kii, Boundary equations with projections, Russ. Math. Surv. 40 (2) (1985) 147–183.
[37] V.S. Ryaben'kii, Method of Difference Potentials and Its Applications, Springer Ser. Comput. Math., vol. 30, Springer-Verlag, Berlin, 2002.
[38] V.S. Ryaben'kii, Difference potentials method and its applications, Math. Nachr. 177 (1996) 251–264.
[39] V.S. Ryaben'kii, On the method of difference potentials, J. Sci. Comput. 28 (2–3) (2006) 467–478.

[40] A.P. Calderon, Boundary-value problems for elliptic equations, in: Proceedings of the Soviet–American Conference on Partial Differential Equations in Novosibirsk, Fizmatgiz, Moscow, 1963, pp. 303–304.
[41] R.T. Seeley, Singular integrals and boundary value problems, Am. J. Math. 88 (1966) 781–809.
[42] I. Petrowsky, On the diffusion of waves and the lacunas for hyperbolic equations, Mat. Sb. (Recueil Mathématique) 17(59) (3) (1945) 289–370.
[43] S. Petropavlovsky, S. Tsynkov, E. Turkel, An efficient numerical algorithm for the 3D wave equation in domains of complex shape, in: Mathematical and Numerical Aspects of Wave Propagation WAVES 2017, The 13th International Conference, Minneapolis, MN, USA, May 15–19, 2017, Minneapolis, MN, USA, in: Book of Abstracts, 2017, pp. 365–366.
[44] V.S. Ryaben'kii, S.V. Tsynkov, V.I. Turchaninov, Long-time numerical computation of wave-type solutions driven by moving sources, Appl. Numer. Math. 38 (2001) 187–222.
[45] V.S. Ryaben'kii, S.V. Tsynkov, V.I. Turchaninov, Global discrete artificial boundary conditions for time-dependent wave propagation, J. Comput. Phys. 174 (2) (2001) 712–758.
[46] S.V. Tsynkov, Artificial boundary conditions for the numerical simulation of unsteady acoustic waves, J. Comput. Phys. 189 (2) (August 2003) 626–650.
[47] S.V. Tsynkov, On the application of lacunae-based methods to Maxwell's equations, J. Comput. Phys. 199 (1) (September 2004) 126–149.
[48] H. Qasimov, S. Tsynkov, Lacunae based stabilization of PMLs, J. Comput. Phys. 227 (2008) 7322–7345.
[49] S.V. Petropavlovsky, S.V. Tsynkov, A non-deteriorating algorithm for computational electromagnetism based on quasi-lacunae of Maxwell's equations, J. Comput. Phys. 231 (2) (2012) 558–585.
[50] S.V. Petropavlovsky, S.V. Tsynkov, Quasi-lacunae of Maxwell's equations, SIAM J. Appl. Math. 71 (4) (2011) 1109–1122.
[51] S.V. Petropavlovsky, S.V. Tsynkov, Non-deteriorating time domain numerical algorithms for Maxwell's electrodynamics, J. Comput. Phys. 336 (May 2017) 1–35.
[52] Avy Soffer, Chris Stucchio, Stable and accurate outgoing wave filters for anisotropic and nonlocal waves, in: Frontiers of Applied and Computational Mathematics, World Sci. Publ., Hackensack, NJ, 2008, pp. 240–247.
[53] Yekaterina Epshteyn, Algorithms composition approach based on difference potentials method for parabolic problems, Commun. Math. Sci. 12 (4) (2014) 723–755.
[54] Jason Albright, Yekaterina Epshteyn, Kyle R. Steffen, High-order accurate difference potentials methods for parabolic problems, Appl. Numer. Math. 93 (2015) 87–106.
[55] M. Medvinsky, S. Tsynkov, E. Turkel, Solving the Helmholtz equation for general smooth geometry using simple grids, Wave Motion 62 (2016) 75–97.
[56] Juliette Chabassier, Sébastien Imperiale, Introduction and study of fourth order theta schemes for linear wave equations, J. Comput. Appl. Math. 245 (2013) 194–212.
[57] Hui Liang, M.Z. Liu, Wanjin Lv, Stability of θ-schemes in the numerical solution of a partial differential equation with piecewise continuous arguments, Appl. Math. Lett. 23 (2) (2010) 198–206.
[58] A.A. Reznik, Approximation of surface potentials of elliptic operators by difference potentials, Dokl. Akad. Nauk SSSR 263 (6) (1982) 1318–1321.
[59] A.A. Reznik, Approximation of the Surface Potentials of Elliptic Operators by Difference Potentials and Solution of Boundary-Value Problems, PhD thesis, Moscow Institute for Physics and Technology, Moscow, USSR, 1983 (in Russian).
[60] Samira Nikkar, Jan Nordström, Fully discrete energy stable high order finite difference methods for hyperbolic problems in deforming domains, J. Comput. Phys. 291 (2015) 82–98.
[61] Claudio Canuto, Alfio Quarteroni, Variational methods in the theoretical analysis of spectral approximations, in: Spectral Methods for Partial Differential Equations, Hampton, Va., 1982, SIAM, Philadelphia, PA, 1984, pp. 55–78.