

Local Compatibility Boundary Conditions for High-Order Accurate Finite-Difference Approximations of PDEs

Nour G. Al Hassanieh^{a,1}, Jeffrey W. Banks^a, William D. Henshaw^{a,1,*},
Donald W. Schwendeman^{a,1}

^a*Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, NY 12180, USA*

Abstract

We describe a new approach to derive numerical approximations of boundary conditions for high-order accurate finite-difference approximations. The approach, called the Local Compatibility Boundary Condition (LCBC) method, uses boundary conditions and compatibility boundary conditions derived from the governing equations, as well as interior and boundary grid values, to construct a local polynomial, whose degree matches the order of accuracy of the interior scheme, centered at each boundary point. The local polynomial is then used to derive a discrete formula for each ghost point in terms of the data. This approach leads to centered approximations that are generally more accurate and stable than one-sided approximations. Moreover, the stencil approximations are local since they do not couple to neighboring ghost-point values which can occur with traditional compatibility conditions. The local polynomial is derived using continuous operators and derivatives which enables the automatic construction of stencil approximations at different orders of accuracy. The LCBC method is developed here for problems governed by second-order partial differential equations, and it is verified for a wide range of sample problems, both time-dependent and time-independent, in two space dimensions and for schemes up to sixth-order accuracy.

Keywords: compatibility conditions, boundary conditions, heat equation, wave equation, high-order finite-differences

Contents

1	Introduction	2
2	Two sample problems	5
2.1	Heat equation in one-dimension	5
2.2	The wave equation in two dimensions	7
3	Second-order PDE initial-boundary-value problems and discretizations	11
3.1	Second-order PDE initial-boundary-value problems	12
3.2	Semi-discrete approximations	12
3.3	Compatibility boundary conditions	14

*Corresponding author

Email addresses: alhasn@rpi.edu (Nour G. Al Hassanieh), banksj3@rpi.edu (Jeffrey W. Banks), henshw@rpi.edu (William D. Henshaw), schwed@rpi.edu (Donald W. Schwendeman)

¹Research supported by the National Science Foundation under grants DMS-1519934 and DMS-1818926.

²This work was partially funded by the NSF Research Training Group Grant DMS-1344962.

4	LCBC method	14
4.1	Dirichlet boundary	15
4.1.1	LCBC method: Direct approach	17
4.1.2	LCBC method: Stencil approach	19
4.2	Neumann boundary	20
4.3	LCBC conditions at a corner	22
4.4	Conditioning of LCBC matrices	24
5	Analysis of the LCBC approach	25
5.1	Solvability of the LCBC matrix systems	25
5.2	Symmetry properties of the LCBC conditions	28
5.3	Stability of LCBC approximations for the wave equation	31
6	Numerical results	34
6.1	Fully discrete schemes	34
6.2	Convergence tests using the method of manufactured solutions	36
6.2.1	Test 1. Square domain and constant-coefficient functions	37
6.2.2	Test 2. Square domain and variable-coefficient functions	37
6.2.3	Test 3. Curvilinear domain and constant-coefficient functions	39
6.3	Scattering of a plane wave from a cylinder	40
6.4	Heat flow in a wavy channel	43
6.5	Pulse propagation in curved channel	44
7	Conclusion	47
Appendix A	LCBC corner conditions	48
Appendix A.1	Neumann-Neumann Corner	48
Appendix A.2	Dirichlet-Neumann Corner	48
Appendix B	Analytical proof of the stability of fourth-order LCBC approximations of the wave equation	49
Appendix C	Time-step restrictions	50
Appendix C.1	Parabolic time-step restrictions for method of lines schemes	51
Appendix C.2	Time-step restriction for hyperbolic problems	53

1. Introduction

We describe a new approach for constructing discrete boundary conditions for high-order accurate numerical approximations to partial differential equations (PDEs). The approach, called the Local Compatibility Boundary Condition (LCBC) method, combines the given physical boundary conditions (BCs) with additional compatibility boundary conditions (CBCs) formed from the PDE and its derivatives. Our focus here is on finite-difference (and finite-volume) methods for both time-dependent and steady PDEs in second-order form with physical BCs of Dirichlet or Neumann type. A high-order accurate centered finite-difference approximation of the spatial operator of the PDE involves a wide stencil which then requires some special treatment to handle the approximation at grid points near the boundary. Unlike a typical approach involving one-sided approximations of the PDE near the boundary and one-sided approximations of Neumann-type BCs, the LCBC approach results in *fully centered* approximations. These centered approximations are generally

more accurate than one-sided approximations, and for the case of time-dependent PDEs they are more stable and less stiff (i.e. do not decrease the stable explicit time-step). Furthermore, the new LCBC approach improves upon a more traditional derivation of discrete CBCs by defining local conditions that are not coupled to neighboring grids along the boundary in tangential directions. As a result, there is no need to solve a system of equations along the boundary which is a significant advantage for explicit time-stepping schemes. In the case of implicit time-stepping methods, and for approximations of steady (elliptic) PDEs, where the solution of large linear systems is required, this tangential decoupling can also be useful for iterative schemes, such as multigrid and Krylov methods.

The development of LCBCs is motivated by our interest in high-order accurate approximations of PDEs in complex domains using overset grids, although the applicability of LCBCs is broader. As shown in Figure 1, an overset grid consists of multiple overlapping structured component grids used to cover a complex, and perhaps moving, problem domain. A mapping is defined for each component grid from physical space to a unit square (or cube) in a computational (index) space, and the mapped PDE is discretized in the computational space. Solution values on the grid are interpolated at internal boundaries where two component grids overlap, and the given physical BCs are applied on external grid boundaries. We have developed second-order accurate schemes for the equations of linear and nonlinear elasticity [1, 2], and up to fourth-order accurate schemes for the incompressible Navier-Stokes equations [3, 4] and Maxwell’s equations [5–7] using overset grids, among other applications. We generally use the physical BCs, along with CBCs, to define discrete centered boundary conditions at external boundaries (with the aid of ghost points), but this approach would become increasingly difficult as the order of the approximation increases. The difficulty stems from the algebraic complexity associated with taking higher and higher derivatives of the spatial operator of the mapped PDE and working out its attendant discrete approximations (with tangential couplings). An associated difficulty involves the special treatments required at corners of the problem domain where separate BCs along sides meet. The LCBC approach overcomes these difficulties by introducing a polynomial interpolant of the solution about each point on the boundary. The polynomial degree is determined by the desired order of accuracy of the approximation, and the coefficients of the polynomial are specified by imposing constraints involving known solution values at grid points interior to the boundary, the physical BCs and CBCs. This approach only requires CBCs defined at a continuous level, and these conditions can be applied to the polynomial interpolant recursively thus easing the aforementioned algebraic complexity. Once defined, the polynomial interpolant can be used to specify solution values at ghost points normal to the boundary (or in corner ghost points for the case of a domain corner) without tangential couplings.

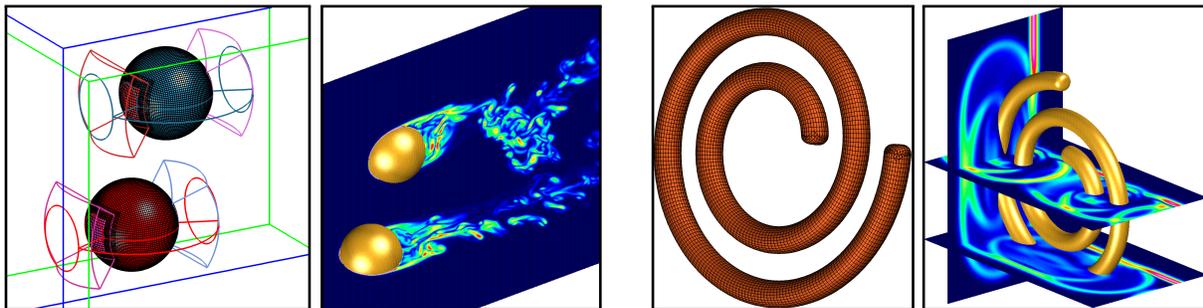


Figure 1: Some target applications for the new LCBC approach. Left: overset grid for two spherical bodies and computed incompressible flow (vorticity). Right: overset grid for a spiral wire and computed electromagnetic scattering.

The aim of the present paper is to describe the LCBC approach in detail for a general class of PDEs in second-order form and to investigate the properties of the resulting discretizations. For example, in the case of a straight boundary and where the spatial operator is the Laplacian, it is well known that for Dirichlet (Neumann) boundary conditions the solution has odd (even) symmetry at the boundary. This leads to simple numerical reflection conditions, and we show that the LCBC approach naturally results in these same reflection conditions (while one-sided approximations would not in general). Beyond this special case, we show that the LCBC approach leads to accurate discretizations of the PDEs, and their BCs, for all orders of accuracy tested (up to sixth order). Further, we show that there is no additional time-step restriction for stability for the case of explicit time-stepping schemes. We focus here on linear PDEs, but the approach should be extendible to nonlinear problems as well. In this article we focus on scalar PDEs, but the approach is also applicable to problems with vector PDEs (e.g. the equations of linear elasticity and Maxwell’s equations) and to problems with material interfaces. Our ultimate goal is to automate the construction of CBC conditions for any order of accuracy and for a wide range of PDEs. We believe that by using the LCBC approach that this goal is achievable. This construction includes the development of LCBC conditions at grid faces as well as at grid corners for two-dimensional domains and at grid edges and vertices for three-dimensional domains.

Compatibility boundary conditions have been used with finite-difference methods for many years³, although it appears that the approach is not widely known. In our work, we have used CBCs for second-order and fourth-order accurate approximations of the heat equation [8] and the incompressible Navier-Stokes equations [3, 4]. For wave equations, we have described the use of CBCs for the compressible Euler equations [9] and linear elasticity [1], and for high-order accurate approximations to Maxwell’s equations [5, 6]. CBCs are also useful for problems involving material interfaces, such as conjugate heat transfer [10] and electromagnetics [5, 7]. In recent work, we have developed Added-Mass Partitioned (AMP) schemes for a wide range of fluid-structure interaction (FSI) problems, including schemes for incompressible flows coupled to rigid bodies [11–13] and elastic solids [14, 15]. These strongly-partitioned schemes incorporate AMP interface conditions derived using CBCs and the physical matching conditions at fluid-solid interfaces in order to overcome added-mass instabilities that can occur for the case of light bodies [16, 17]. In related work, we have also used CBCs in the CHAMP scheme [18] to form discrete interface conditions for a partitioned approach to the solution of conjugate heat transfer problems.

In other work, CBCs are used in the book by Gustafsson on high-order difference methods [19]. CBCs have also been used to derive stable and accurate embedded boundary⁴ approximations [20–22]. CBCs have been incorporated into summation-by-parts schemes by Sjögreen and Petersson for the equations of elasticity [23]. CBCs have been used by LeVeque and Li with their immersed interface method to develop accurate approximations at embedded interfaces [24–26]. Shu and collaborators have used CBCs in their inverse-Lax-Wendroff approach for hyperbolic equations and conservation laws [27–30] as well as for parabolic and advection-diffusion equations [31–33].

In this article we focus on high-order accurate finite-difference schemes. We note, however, that CBCs could also be useful for Galerkin schemes. Typical high-order accurate FEM or DG schemes that use polynomial approximations over an element effectively use one-sided approximations near boundaries. This can result in time-step restrictions that force the time-step to decrease rather significantly as the order-of-accuracy increases [34–36]. Similarly for B-Spline FEM, as commonly

³For example, CBCs were known to Professor H.-O. Kreiss and his students at least since the 1980’s.

⁴By embedded boundary we mean a boundary curve (or boundary surface in three dimensions) that passes through a grid in an irregular fashion (as opposed to a boundary-conforming grid).

used in isogeometric analysis, one-sided operators occurring near boundaries results in spurious large eigenvalues, so-called outlier eigenvalues [37]. Banks et al. [38–40], however, have shown that when CBCs are used with their Galerkin-Difference method, a class of FEM schemes, the spectrum of the operator is near-optimal, and the time-step restriction for explicit integration gives approximately the maximal CFL-one stability.

The remaining sections of the paper are organized as follows. In Section 2, we consider two sample problems to introduce the LCBC method and to compare it to the standard CBC approach. A second-order PDE initial-boundary-problem is introduced in Section 3, and this model problem is used along with its physical boundary conditions and derived CBCs as a basis to describe the LCBC approach for discretizations of the equations up to sixth-order accuracy. In Section 4, we detail the steps of the LCBC approach for boundary conditions of Dirichlet and Neumann type, as well as the implementation of the method at domain corners where boundary conditions of various types meet. We provide algorithms for the efficient application of the LCBC method for both domain sides and corners, and we discuss the conditioning of the matrices generated by the LCBC procedure. Various elements of the LCBC method, including solvability, symmetry properties, and stability for wave equations, are analyzed in Section 5. The accuracy and stability of the method is illustrated in Section 6 by considering the numerical results for a variety of initial-boundary-value problems. Concluding remarks and a discussion of future directions of the LCBC approach are given in Section 7.

2. Two sample problems

We begin by discussing CBCs and the LCBC approach for two relatively simple sample problems, the first involving the heat (diffusion) equation and the second involving the wave equation. This is done to establish basic ideas and to introduce useful notation.

2.1. Heat equation in one-dimension

Let $u(x, t)$ solve the following initial-boundary-value problem (IBVP) for the heat equation on the interval $x \in [a, b]$, with a Dirichlet boundary condition on the left and a Neumann boundary condition on the right,

$$\begin{cases} \partial_t u = \kappa \partial_x^2 u, & x \in (a, b), \quad t > 0, \\ u(a, t) = g_a(t), & t > 0, \\ \partial_x u(b, t) = g_b(t), & t > 0, \\ u(x, 0) = u_0(x) & x \in [a, b], \end{cases} \quad (1)$$

where $\kappa > 0$ is a constant diffusivity, and $g_a(t)$, $g_b(t)$ and $u_0(x)$ are given smooth functions. The CBCs we use for this problem are found by taking ν time derivatives of the boundary conditions in (1), and then using the PDE to replace time derivatives with space derivatives. This leads to a sequence of conditions given by

$$\kappa^\nu \partial_x^{2\nu} u(a, t) = \partial_t^\nu g_a(t), \quad \kappa^\nu \partial_x^{2\nu+1} u(b, t) = \partial_t^\nu g_b(t), \quad \nu = 1, 2, \dots \quad (2)$$

The conditions in (2), denoted by $\text{CBC}_s[\nu]$, $s = a$ and b , can be used in a finite difference scheme for the IBVP in (1) as additional *numerical boundary conditions*, in place of extrapolation conditions or one-sided approximations. For example, suppose the PDE and the Neumann boundary condition are discretized to order of accuracy $2p$ using central difference approximations with a stencil width $2p + 1$, and suppose CBCs are used to generate additional numerical boundary conditions. The

number of derivatives in $\text{CBC}_s[\nu]$ increases with ν , but it can be shown that the CBCs can be approximated to lower-order accuracy as ν increases so that the stencil width remains $2p + 1$.

To illustrate the use of CBCs for a specific case, consider a fourth-order accurate ($p = 2$) approximation of (1). Our focus is on the spatial approximations, especially near the boundaries, and so we consider a method-of-lines approach. Let $U_j(t) \approx u(x_j, t)$, $j = -2, -1, 0, \dots, N_x + 2$, where $x_j = a + jh$ are points on a uniform grid with grid spacing $h = (b - a)/N_x$ as shown in Figure 2. Note that the grid includes *ghost points* at each boundary, and these are appended to the grid covering the problem domain $x \in [a, b]$ to facilitate the approximations of the spatial derivatives in the Neumann boundary condition at $x = b$ and the CBCs applied at both boundaries.

The fourth-order accurate approximation to (1), using a CBC at each boundary, is given by

$$\begin{cases} \partial_t U_j(t) = \kappa D_{4xx} U_j, & j = 1, 2, \dots, N_x, \quad t > 0, \\ U_0(t) = g_a(t), & t > 0, \\ \kappa D_{4xx} U_0(t) = \partial_t g_a(t), & t > 0, \quad \text{CBC}_a[1], \\ \kappa^2 (D_{+x} D_{-x})^2 U_0(t) = \partial_t^2 g_a(t), & t > 0, \quad \text{CBC}_a[2], \\ D_{4x} U_{N_x}(t) = g_b(t), & t > 0, \\ \kappa D_{0x} D_{+x} D_{-x} U_{N_x}(t) = \partial_t g_b(t), & t > 0, \quad \text{CBC}_b[1]. \end{cases} \quad (3)$$

Here, D_{4x} and D_{4xx} are fourth-order accurate approximations to ∂_x and ∂_x^2 , respectively, defined by

$$D_{4x} \stackrel{\text{def}}{=} D_{0x} \left(I - \frac{h^2}{6} D_{+x} D_{-x} \right), \quad D_{4xx} \stackrel{\text{def}}{=} D_{+x} D_{-x} \left(I - \frac{h^2}{12} D_{+x} D_{-x} \right), \quad (4)$$

where D_{+x} , D_{-x} and D_{0x} are the usual divided difference operators defined by

$$D_{+x} U_j \stackrel{\text{def}}{=} \frac{U_{j+1} - U_j}{h}, \quad D_{-x} U_j \stackrel{\text{def}}{=} \frac{U_j - U_{j-1}}{h}, \quad D_{0x} U_j \stackrel{\text{def}}{=} \frac{U_{j+1} - U_{j-1}}{2h}. \quad (5)$$

Note that the discretization of the PDE is applied on all interior grid points and on the right boundary ($j = N_x$) as a CBC for the Neumann boundary condition. Note also that $\text{CBC}_a[2]$ on the left and $\text{CBC}_b[1]$ on the right are approximated only to second-order accuracy. Despite this, the scheme is fourth-order accurate in space. Initial conditions have been omitted in (3) as these are not essential for the present discussion.

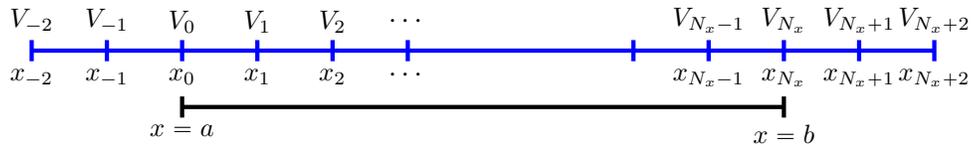


Figure 2: Grid in one space dimension, with ghost points, for the fourth-order accurate approximation of the heat equation in (3).

We now describe the LCBC method as an alternative approach to derive discrete boundary conditions in (3). Let us first consider boundary conditions at $x = a$. In the LCBC approach we approximate the solution near the boundary at $x = a$ as a polynomial $\tilde{u}(x)$ of degree $2p = 4$. The polynomial approximation changes in time as the solution evolves, but this dependence is

suppressed here for notational convenience. In terms of a monomial basis⁵, for example, we have

$$\tilde{u}(x) = \sum_{n=0}^4 c_n (x-a)^n. \quad (6)$$

The five coefficients c_n , $n = 0, 1, \dots, 4$, in (6) are found by requiring \tilde{u} to match U_j on the boundary and two interior points, as well as to satisfy the two compatibility conditions $\text{CBC}_a[1]$ and $\text{CBC}_a[2]$, i.e.

$$\tilde{u}(a) = g_a(t), \quad (7a)$$

$$\tilde{u}(x_j) = U_j(t), \quad j = 1, 2, \quad (7b)$$

$$\kappa^\nu \partial_x^{2\nu} \tilde{u}(a) = \partial_t^\nu g_a(t), \quad \nu = 1, 2, \quad (7c)$$

for a fixed time t . The five constraints in (7) imply a local system of linear equations for the coefficients $\{c_n\}$ in (6). Once the system is solved, ghost values of U_j are then determined by setting

$$U_j = \tilde{u}(x_j), \quad j = -2, -1. \quad (8)$$

For the right boundary, a similar polynomial approximation to that in (6) is defined about $x = b$, and its coefficients are found using the constraints

$$\partial_x \tilde{u}(b) = g_b(t), \quad (9a)$$

$$\tilde{u}(x_j) = U_j(t), \quad j = N_x - 2, N_x - 1, N_x, \quad (9b)$$

$$\kappa^\nu \partial_x^{2\nu+1} \tilde{u}(b) = \partial_t^\nu g_b(t), \quad \nu = 1. \quad (9c)$$

The polynomial approximation about $x = b$ is then used to determine ghost values at x_{N_x+1} and x_{N_x+2} . We note that the LCBC approach uses CBCs at a continuous level to define local polynomial approximations instead of their discrete approximations as in (3). However, for this simple one-dimensional problem, it can be shown that the LCBC approach yields an equivalent approximation to that in (3).

2.2. The wave equation in two dimensions

We now consider an IBVP for the wave equation (with unit wave speed) on a unit square domain, $\mathbf{x} = (x, y) \in \Omega = (0, 1)^2$. For this problem, we let $u(\mathbf{x}, t)$ solve

$$\begin{cases} \partial_t^2 u = \Delta u, & \mathbf{x} \in \Omega, \quad t > 0, \\ u = g_\ell(y, t) & \mathbf{x} \in \partial\Omega_\ell, \quad t > 0, \\ u = g_r(y, t) & \mathbf{x} \in \partial\Omega_r, \quad t > 0, \\ \partial_y u = g_b(x, t) & \mathbf{x} \in \partial\Omega_b, \quad t > 0, \\ \partial_y u = g_\tau(x, t) & \mathbf{x} \in \partial\Omega_\tau, \quad t > 0, \\ u = u_0(\mathbf{x}), \quad \partial_t u = u_1(\mathbf{x}), & \mathbf{x} \in \bar{\Omega}, \quad t = 0, \end{cases} \quad (10)$$

⁵In practice we use a Lagrange polynomial basis as it leads to better conditioned equations.

where Δ is the Laplacian operator. Dirichlet conditions are imposed on the left ($x = 0$) and right ($x = 1$) boundaries, denoted by $\partial\Omega_\ell$ and $\partial\Omega_r$, respectively, with given smooth functions $g_\ell(y, t)$ and $g_r(y, t)$. Similarly, Neumann conditions are specified on the bottom ($y = 0$) and top ($y = 1$) boundaries, denoted by $\partial\Omega_b$ and $\partial\Omega_\tau$, respectively, with given smooth functions $g_b(y, t)$ and $g_\tau(y, t)$. Initial conditions are determined by the given functions $u_0(\mathbf{x})$ and $u_1(\mathbf{x})$ defined for $\bar{\Omega} = \Omega \cup \partial\Omega_s$, $s = \ell, r, b, \tau$.

As before, we consider CBCs for the problem by taking time derivatives of the boundary conditions and then replacing even time derivatives of $u(\mathbf{x}, t)$ in favor of spatial derivatives using the PDE. For example, 2ν time derivatives of the Dirichlet conditions at $s = \ell$ gives

$$\Delta^\nu u(0, y, t) = \partial_t^{2\nu} g_\ell(y, t), \quad \nu = 1, 2, \dots, \quad (11)$$

which implies a sequence of CBCs, denoted by $\text{CBC}_\ell[\nu]$, $\nu = 1, 2, \dots$, whose first two conditions are given by

$$\partial_x^2 u(\mathbf{x}, t) = \partial_t^2 g_\ell(y, t) - \partial_y^2 g_\ell(y, t), \quad \mathbf{x} \in \partial\Omega_\ell, \quad (12a)$$

$$\partial_x^4 u(\mathbf{x}, t) = \partial_t^4 g_\ell(y, t) - 2\partial_t^2 \partial_y^2 g_\ell(y, t) + \partial_y^4 g_\ell(y, t), \quad \mathbf{x} \in \partial\Omega_\ell. \quad (12b)$$

We can also write down CBCs corresponding to the Neumann boundary conditions by taking time derivatives. At the bottom boundary, for example, this leads to $\text{CBC}_b[\nu]$, $\nu = 1, 2, \dots$, whose first two conditions are given by

$$\partial_y^3 u(\mathbf{x}, t) = \partial_t^2 g_b(x, t) - \partial_x^2 g_b(x, t), \quad \mathbf{x} \in \partial\Omega_b, \quad (13a)$$

$$\partial_y^5 u(\mathbf{x}, t) = \partial_t^4 g_b(x, t) - 2\partial_t^2 \partial_x^2 g_b(x, t) + \partial_x^4 g_b(x, t), \quad \mathbf{x} \in \partial\Omega_b. \quad (13b)$$

The CBCs along the boundaries with $s = r$ and $s = \tau$ are similar to those given in (12) and (13), respectively.

A standard use of CBCs to define numerical boundary conditions for a high-order accurate discretization of the IBVP in (10) for the wave equation in two space dimensions follows similar steps as were used above for the heat equation in one space dimension. We first introduce a Cartesian grid for $\bar{\Omega}$ defined by

$$\bar{\Omega}_h \stackrel{\text{def}}{=} \{\mathbf{x}_i = (x_i, y_j) = (i\Delta x, j\Delta y), \quad i = 0, \dots, N_x, \quad j = 0, \dots, N_y\}, \quad (14)$$

where N_x and N_y determine the number of grid lines in the x and y directions, respectively, $\Delta x = 1/N_x$ and $\Delta y = 1/N_y$ are grid spacings, and $\mathbf{i} = (i, j)$ is a multi-index, see the left plot of Figure 3. Let $\partial\Omega_{s,h}$, $s = \ell, r, b, \tau$, represent the grid points on the left, right, bottom and top boundaries of $\bar{\Omega}_h$ respectively. The set $\partial\Omega_h = \bigcup_{s=\ell, r, \tau, b} \partial\Omega_{s,h}$ represents all the boundary grid points of $\bar{\Omega}_h$ and $\Omega_h = \bar{\Omega}_h \setminus \partial\Omega_h$ denotes the interior grid points. The grid includes ghost points, as shown in the figure, and these are used to facilitate the approximation of the Neumann boundary conditions and the CBCs. Let $U_i(t) \approx u(x_i, y_j, t)$, and consider a fourth-order accurate semi-discrete

scheme for (10) given by

$$\begin{cases} \partial_t^2 U_i(t) = \Delta_{4h} U_i, & \mathbf{x}_i \in \Omega_h \cup \partial\Omega_{b,h} \cup \partial\Omega_{\tau,h}, \quad t > 0, \\ U_i(t) = g_\ell(y_j, t), & \mathbf{x}_i \in \partial\Omega_{\ell,h}, \quad t > 0, \\ U_i(t) = g_r(y_j, t), & \mathbf{x}_i \in \partial\Omega_{r,h}, \quad t > 0, \\ D_{4y} U_i(t) = g_b(x_i, t), & \mathbf{x}_i \in \partial\Omega_{b,h}, \quad t > 0, \\ D_{4y} U_i(t) = g_\tau(x_i, t), & \mathbf{x}_i \in \partial\Omega_{\tau,h}, \quad t > 0, \end{cases} \quad (15)$$

where Δ_{4h} and D_{4y} are fourth-order centered approximations of Δ and ∂_y , respectively, following the definitions in (4) and (5). Discrete approximations of $\text{CBC}_\ell[1]$ and $\text{CBC}_\ell[2]$ in (12) given by

$$D_{4xx} U_i(t) = \partial_t^2 g_\ell(y_j, t) - \partial_y^2 g_\ell(y_j, t), \quad \mathbf{x}_i \in \partial\Omega_{\ell,h}, \quad (16a)$$

$$(D_{+x} D_{-x})^2 U_i(t) = \partial_t^4 g_\ell(y_j, t) - 2\partial_y^2 \partial_t^2 g_\ell(y_j, t) + \partial_y^4 g_\ell(y_j, t), \quad \mathbf{x}_i \in \partial\Omega_{\ell,h}, \quad (16b)$$

respectively, are added to (15). Note that the approximation of $\text{CBC}_\ell[1]$ is fourth-order accurate, while a second-order accurate approximation is used for $\text{CBC}_\ell[2]$, following the scheme in (3) for the CBCs about its Dirichlet boundary. Similar approximations of $\text{CBC}_r[1]$ and $\text{CBC}_r[2]$ are also added to (15). For the Neumann conditions, we use second-order accurate approximations of $\text{CBC}_b[1]$ and $\text{CBC}_\tau[1]$. For example, the approximation of $\text{CBC}_b[1]$ in (13) given by

$$D_{0y} D_{+y} D_{-y} U_i(t) = \partial_t^2 g_b(x_i, t) - \partial_x^2 g_b(x_j, t), \quad \mathbf{x}_i \in \partial\Omega_{b,h}, \quad (17)$$

is added to (15). As before, the specifications of the initial conditions have been omitted in (15).

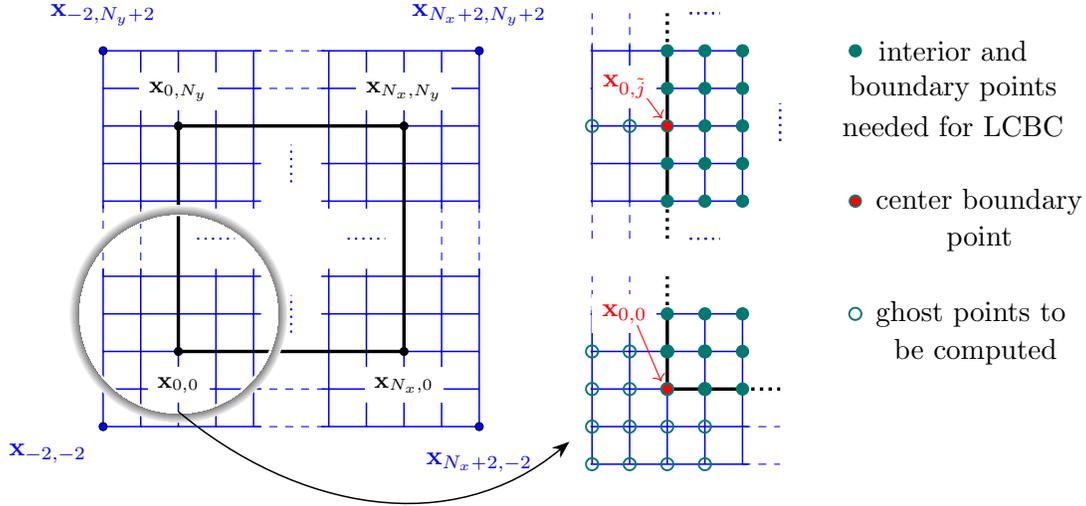


Figure 3: Grid in two space dimension, with ghost points, for the fourth-order accurate approximation of the wave equation in (15).

We now consider the LCBC approach for the fourth-order accurate approximation of the wave equation in (15). For this two-dimensional problem, it is convenient to define an interpolating

polynomial $\tilde{u}(x, y)$, centered about (\tilde{x}, \tilde{y}) , as

$$\tilde{u}(x, y) \stackrel{\text{def}}{=} \sum_{\hat{n}=-p}^p \sum_{\hat{m}=-p}^p d_{\hat{m}, \hat{n}} L_{\hat{m}} \left(\frac{x - \tilde{x}}{\Delta x} \right) L_{\hat{n}} \left(\frac{y - \tilde{y}}{\Delta y} \right), \quad p \in \mathbb{N}, \quad (18)$$

where $d_{\hat{m}, \hat{n}}$ are coefficients and $L_k(z)$ is a Lagrange basis function given by

$$L_k(z) = \prod_{\substack{l=-p \\ l \neq k}}^p \frac{(z - l)}{(k - l)}, \quad k = -p, \dots, p. \quad (19)$$

Note that \tilde{u} has the property

$$\tilde{u}(\tilde{x} + \hat{i}\Delta x, \tilde{y} + \hat{j}\Delta y) = d_{\hat{i}, \hat{j}}, \quad \hat{i}, \hat{j} = -p, \dots, p. \quad (20)$$

We set $p = 2$ in (18) for the fourth-order accurate scheme, and use known data at grid points in the interior and on the boundary, along with the physical boundary conditions and CBCs, to determine the coefficients of $\tilde{u}(x, y)$ centered at grid points along the boundaries. For example, along the left Dirichlet boundary at a fixed time t , let $(\tilde{x}, \tilde{y}) = \mathbf{x}_{0, \tilde{j}}$, $\tilde{j} \in \{2, 3, \dots, N_y - 2\}$, and impose the constraints

$$\tilde{u}(\tilde{x}, \tilde{y} + \hat{j}\Delta y) = g_\ell(\tilde{y} + \hat{j}\Delta y, t), \quad \hat{j} = -2, \dots, 2, \quad (21a)$$

$$\tilde{u}(\tilde{x} + \hat{i}\Delta x, \tilde{y} + \hat{j}\Delta y) = U_{\hat{i}, \tilde{j} + \hat{j}}(t), \quad \hat{i} = 1, 2, \quad \hat{j} = -2, \dots, 2, \quad (21b)$$

$$\partial_y^\mu \Delta^\nu \tilde{u}(\tilde{x}, \tilde{y}) = \partial_y^\mu \partial_t^{2\nu} g_\ell(\tilde{y}, t), \quad \nu = 1, 2, \quad \mu = 0, \dots, 4. \quad (21c)$$

The 25 constraints in (21) lead to a linear system for the 25 coefficients $d_{\hat{m}, \hat{n}}$, $\hat{m}, \hat{n} = -2, \dots, 2$, in (18) for each point $\mathbf{x}_{0, \tilde{j}}$ along the boundary. Once solved, the coefficients determine the ghost points as $U_{\hat{i}, \tilde{j}} = d_{\hat{i}, 0}$, $\hat{i} = -2, -1$. Similarly, along the bottom Neumann boundary, let $(\tilde{x}, \tilde{y}) = \mathbf{x}_{\tilde{i}, 0}$, $\tilde{i} \in \{2, 3, \dots, N_x - 2\}$, and impose the constraints

$$\tilde{u}(\tilde{x} + \hat{i}\Delta x, \tilde{y} + \hat{j}\Delta y) = U_{\tilde{i} + \hat{i}, \hat{j}}(t), \quad \hat{j} = 0, 1, 2, \quad \hat{i} = -2, \dots, 2, \quad (22a)$$

$$\partial_x^\mu \Delta^\nu \partial_y \tilde{u}(\tilde{x}, \tilde{y}) = \partial_x^\mu \partial_t^{2\nu} g_b(\tilde{x}, t), \quad \nu = 0, 1, \quad \mu = 0, \dots, 4. \quad (22b)$$

Note that the Neumann boundary condition appears in (22b) for the case $\nu = \mu = 0$, and tangential derivatives of this boundary condition are applied for $\mu > 0$. As before, the 25 constraints in (22) determine the 25 coefficients in (18), which then specify ghost points as $U_{\tilde{i}, \hat{j}} = d_{0, \hat{j}}$, $\hat{j} = -2, -1$, for each point $\mathbf{x}_{0, \tilde{j}}$ along the boundary.

The application of the LCBC approach at corners requires special treatment. For example, let us consider the corner at $\tilde{\mathbf{x}} = (0, 0)$ where the Dirichlet boundary on the left meets the Neumann boundary along the bottom (see the right plot of Figure 3). Here, we choose a linearly independent set of constraints from those implied by (21) and (22). We begin with the data available in the interior and on the Neumann boundary,

$$\tilde{u}(\hat{i}\Delta x, \hat{j}\Delta y) = U_{\tilde{i}, \hat{j}}(t), \quad \hat{i} = 1, 2, \quad \hat{j} = 0, 1, 2. \quad (23a)$$

Next, we impose tangential derivatives of the primary boundary conditions

$$\partial_y^{\mu_\ell} \tilde{u}(0, 0) = \partial_y^{\mu_\ell} g_\ell(0, t), \quad \mu_\ell = 0, 2, 3, 4, \quad (23b)$$

$$\partial_x^{\mu_b} \partial_y \tilde{u}(0, 0) = \partial_x^{\mu_b} g_b(0, t), \quad \mu_b = 1, 2, 3, 4, \quad (23c)$$

along with

$$\partial_y \tilde{u}(0, 0) = \frac{1}{2} (\partial_y g_\ell(0, t) + g_b(0, t)). \quad (23d)$$

The latter is an average of the linearly dependent constraints in (23b) and (23c) corresponding to $\mu_\ell = 1$ and $\mu_b = 0$. The remaining constraints are taken from compatibility conditions, and they are

$$\partial_y^{\mu_\ell} \Delta^\nu \tilde{u}(0, 0) = \partial_y^{\mu_\ell} \partial_t^{2\nu} g_\ell(0, t), \quad \nu = 1, 2, \quad \mu_\ell = 0, 2, 4, \quad (23e)$$

$$\partial_x^{\mu_b} \Delta \partial_y \tilde{u}(0, 0) = \partial_x^{\mu_b} \partial_t^2 g_b(0, t), \quad \mu_b = 1, 3, 4, \quad (23f)$$

and

$$\partial_x^2 \partial_y^3 \tilde{u}(0, 0) = \frac{\Delta y^2 \partial_y^3 \partial_t^2 g_\ell(0, t) + \Delta x^2 \partial_x^2 \partial_t^2 g_b(0, t)}{\Delta y^2 + \Delta x^2}. \quad (23g)$$

Again, an average is used when the conditions obtained from the two boundaries are linearly dependent. This occurs for the constraints in (23e) and (23f) corresponding to $(\nu, \mu_\ell) = (1, 3)$ and $\mu_b = 2$, respectively. Note that we use a weighted average in (23g) to balance the tangential derivatives taken in the x and y directions, and that certain high derivatives of \tilde{u} implied by the compatibility conditions have been omitted since they vanish for the chosen degree of the interpolating polynomial in (18). There are 25 constraints in (23) that determine the 25 coefficients of $\tilde{u}(\mathbf{x})$ about $\tilde{\mathbf{x}} = (0, 0)$, and this interpolating polynomial is evaluated at \mathbf{x}_i , for $\mathbf{i} = [-2, -1] \times [-2, 1]$ and $\mathbf{i} = [0, 1] \times [-2, -1]$, to determine the ghost values there.

Note that we have chosen to use the corner LCBC polynomial to assign solution values in corner ghost points, e.g. U_i for $\mathbf{i} = [-2, 1] \times [-2, -1]$, as well as values in nearby ghost points along the adjacent faces, e.g. U_i for $\mathbf{i} = [-2, -1] \times [0, 1]$ and $\mathbf{i} = [0, 1] \times [-2, -1]$, see Figure 3. These latter values could instead have been determined by special LCBC polynomials that used appropriate compatibility conditions from the primary face and the secondary adjacent face. For higher-order approximations this would result in many more special cases as the corner is approached. To avoid these complications, we have chosen to employ the simpler approach of using the corner LCBC polynomial at all adjacent face ghost points where the standard LCBC polynomial for the face does not apply.

The application of the LCBC method just outlined for the two sample problems describes the basic approach. In the subsequent sections, we elaborate on the LCBC method by considering initial-boundary-value problems involving a broader class of linear PDEs and corresponding spatial discretizations of higher orders of accuracy.

3. Second-order PDE initial-boundary-value problems and discretizations

In this section, we consider initial-boundary-value problems for a general scalar second-order PDE and corresponding high-order accurate finite-difference approximations as a basis for a full description of the LCBC approach which follows in the next section.

3.1. Second-order PDE initial-boundary-value problems

Consider the initial-boundary-value problem on $[0, T] \times \Omega$, $T > 0$, given by

$$\begin{cases} \mathcal{L}_q u = Qu + f(\mathbf{x}, t), & \mathbf{x} \in \Omega, \quad t \in (0, T], \quad q = 0, 1, 2, \\ \mathcal{B}u(\mathbf{x}, t) = g(\mathbf{x}, t), & \mathbf{x} \in \partial\Omega, \quad t \in [0, T], \\ \partial_t^{\alpha-1} u(\mathbf{x}, 0) = u_{\alpha-1}(\mathbf{x}), & \mathbf{x} \in \bar{\Omega}, \quad \alpha = 1, \dots, q, \quad q = 1, 2. \end{cases} \quad (24)$$

Here, $\Omega \subset \mathbb{R}^2$ is a general domain, $\partial\Omega$ denotes the boundary of Ω , and $\bar{\Omega} = \Omega \cup \partial\Omega$ as before. We define

$$\mathcal{L}_q u \stackrel{\text{def}}{=} \begin{cases} 0, & q = 0, \\ \partial_t^q u, & q = 1, 2, \end{cases} \quad (25)$$

and

$$Qu \stackrel{\text{def}}{=} c_{11}(\mathbf{x})\partial_x^2 u + 2c_{12}(\mathbf{x})\partial_x\partial_y u + c_{22}(\mathbf{x})\partial_y^2 u + c_1(\mathbf{x})\partial_x u + c_2(\mathbf{x})\partial_y u + c_0(\mathbf{x})u, \quad \mathbf{x} \in \Omega. \quad (26)$$

We assume that the coefficient functions $c_{11}(\mathbf{x})$, $c_{12}(\mathbf{x})$, etc., are smooth, and they are chosen, together with the boundary and initial conditions, so that the problem is well posed. [wdh: For example, for \$q = 1, 2\$](#) , necessary conditions are that $c_{11}(\mathbf{x}) > 0$, $c_{22}(\mathbf{x}) > 0$ and

$$c_{11}(\mathbf{x})c_{22}(\mathbf{x}) - c_{12}^2(\mathbf{x}) \geq \delta > 0, \quad \forall \mathbf{x} \in \Omega. \quad (27)$$

We note that (26) is taken in non-conservative form for the purposes of this article; LCBC methods for problems in conservative form are left to future work.

The governing equation in (24), with given forcing function $f(\mathbf{x}, t)$, takes the form of a general elliptic ($q = 0$), parabolic ($q = 1$) or hyperbolic ($q = 2$) PDE in second-order form depending on the choice of the index q . The boundary conditions in (24), with given forcing function $g(\mathbf{x}, t)$, are written in terms of the boundary operator given by

$$\mathcal{B}u \stackrel{\text{def}}{=} b_1(\mathbf{x})u + b_2(\mathbf{x})\partial_n u, \quad \mathbf{x} \in \partial\Omega, \quad (28)$$

where ∂_n is the outward normal derivative and the coefficient functions satisfy $|b_1(\mathbf{x})| + |b_2(\mathbf{x})| \neq 0$, $\forall \mathbf{x} \in \partial\Omega$. Initial conditions are specified for the cases, $q = 1$ and 2.

We are motivated by the application of the LCBC method for high-order accurate discretizations of the model problem in (24) on mapped grids. For such discretizations, we consider a smooth mapping from the unit square to Ω . The form of the model problem remains unchanged in the mapped domain, so it suffices to study the governing equations in (24) over the domain $\Omega = (0, 1)^2$ as a model problem.

3.2. Semi-discrete approximations

Let

$$U_{\mathbf{i}} \approx \begin{cases} u(\mathbf{x}_{\mathbf{i}}), & q = 0, \\ u(\mathbf{x}_{\mathbf{i}}, t), & q = 1, 2, \end{cases}$$

represent the numerical approximation of the exact solution of (24) at discrete points $\mathbf{x}_{\mathbf{i}}$ on the Cartesian grid $\bar{\Omega}_h$ given in (14) and at a fixed time $t \in [0, T]$. Our principal focus is on discretizations of (24) to fourth and sixth-order accuracy, although we also consider second-order accurate approximations as a baseline. A second-order accurate discretization of (24) employs standard

centered differences given by

$$D_{2,\zeta} \stackrel{\text{def}}{=} D_{0\zeta}, \quad D_{2,\zeta\zeta} \stackrel{\text{def}}{=} D_{+\zeta}D_{-\zeta}, \quad \zeta = x, y. \quad (29)$$

Fourth-order accurate operators, $D_{4,\zeta}$ and $D_{4,\zeta\zeta}$, $\zeta = x, y$, were introduced in the previous section, and we define sixth-order accurate centered operators as

$$\left. \begin{aligned} D_{6,\zeta} &\stackrel{\text{def}}{=} D_{0\zeta} \left(I - \frac{\Delta\zeta^2}{6} D_{+\zeta}D_{-\zeta} + \frac{\Delta\zeta^4}{30} (D_{+\zeta}D_{-\zeta})^2 \right), \\ D_{6,\zeta\zeta} &\stackrel{\text{def}}{=} D_{+\zeta}D_{-\zeta} \left(I - \frac{\Delta\zeta^2}{12} D_{+\zeta}D_{-\zeta} + \frac{\Delta\zeta^4}{90} (D_{+\zeta}D_{-\zeta})^2 \right), \end{aligned} \right\} \quad \zeta = x, y. \quad (30)$$

Thus,

$$Q_{d,h} \stackrel{\text{def}}{=} c_{11}(\mathbf{x}_i)D_{d,xx} + 2c_{12}(\mathbf{x}_i)D_{d,x}D_{d,y} + c_{22}(\mathbf{x}_i)D_{d,yy} + c_1(\mathbf{x}_i)D_{d,x} + c_2(\mathbf{x}_i)D_{d,y} + c_0(\mathbf{x}_i)I, \quad (31)$$

for $\mathbf{x}_i \in \Omega_h$. Set $\mathcal{B}_{d,h}$ to be the d^{th} -order accurate centered discretization of the boundary operator \mathcal{B} .

In view of the finite difference operators defined above, we note that (31) can be written as

$$\begin{aligned} Q_{d,h}V_i &= c_{11}(\mathbf{x}_i)D_{+x}D_{-x} \left\{ V_i + \sum_{n=1}^{k-1} a_n \Delta x^{2n} (D_{+x}D_{-x})^n V_i \right\} \\ &+ 2c_{12}(\mathbf{x}_i)D_{0x}D_{0y} \left\{ V_i + \sum_{n=1}^{k-1} \sum_{l=0}^n b_l b_{n-l} \Delta x^{2l} \Delta y^{2(n-l)} (D_{+x}D_{-x})^l (D_{+y}D_{-y})^{n-l} V_i \right\} \\ &+ c_{22}(\mathbf{x}_i)D_{+y}D_{-y} \left\{ V_i + \sum_{n=1}^{k-1} a_n \Delta y^{2n} (D_{+y}D_{-y})^n V_i \right\} \\ &+ c_1(\mathbf{x}_i)D_{0x} \left\{ V_i + \sum_{n=1}^{k-1} b_n \Delta x^{2n} (D_{+x}D_{-x})^n V_i \right\} \\ &+ c_2(\mathbf{x}_i)D_{0y} \left\{ V_i + \sum_{n=1}^{k-1} b_n \Delta y^{2n} (D_{+y}D_{-y})^n V_i \right\} + c_0(\mathbf{x}_i)V_i, \end{aligned} \quad (32)$$

for some grid function V_i and for $d = 2k$, $k = 1, 2, 3$. The higher-order correction terms given by the sums in (32) are omitted for the second-order accurate case with $k = 1$. The coefficients a_n and b_n of the correction terms involving powers of $D_{+x}D_{-x}$ and $D_{+y}D_{-y}$ are given by

$$a_1 = -\frac{1}{12}, \quad a_2 = \frac{1}{90}, \quad b_0 = 1, \quad b_1 = -\frac{1}{6}, \quad b_2 = \frac{1}{30}. \quad (33)$$

The form of the discrete operator in (32) is particularly useful for the calculation of $(Q_{d,h})^\nu V_i$, $\nu = 1, 2, \dots$, which is required for the approximation of compatibility boundary conditions as is discussed in the next section.

The semi-discrete model problem takes the form

$$\begin{cases} \mathcal{L}_q U_i(t) = Q_{d,h} U_i(t) + f(\mathbf{x}_i, t), & \mathbf{x}_i \in \Omega_h, \quad t \in (0, T], \quad q = 0, 1, 2, \\ \mathcal{B}_{d,h} U_i(t) = g(\mathbf{x}_i, t), & \mathbf{x}_i \in \partial\Omega_h, \quad t \in [0, T], \\ \partial_t^{\alpha-1} U_i(0) = u_{\alpha-1}(\mathbf{x}_i), & \mathbf{x}_i \in \bar{\Omega}_h, \quad \alpha = 1, \dots, q, \quad q = 1, 2. \end{cases} \quad (34)$$

Grid points along ghost lines at each boundary of Ω_h are introduced to accommodate the stencil of the discrete spatial operators near the boundaries, and these are included in the extended grid defined by

$$\Omega_h^e \stackrel{\text{def}}{=} \{\mathbf{x}_i \mid \mathbf{i} = (i, j), i = -p, \dots, N_x + p, j = -p, \dots, N_y + p\}, \quad (35)$$

where $p = d/2$. We evaluate the solution at the ghost points using the LCBC method.

3.3. Compatibility boundary conditions

The LCBC method uses compatibility boundary conditions obtained from the primary boundary conditions and the governing PDE (and its derivatives) applied on the boundary. The steps taken to obtain the CBCs depend on whether the PDE is time-dependent or not. We begin with the time-dependent cases for which $q = 1$ or 2 . Here, we take q time derivatives of the primary boundary condition in (24) to give

$$\mathcal{B}\partial_t^q u(\mathbf{x}, t) = \partial_t^q g(\mathbf{x}, t), \quad \mathbf{x} \in \partial\Omega, \quad (36)$$

at a fixed time $t \in [0, T]$. Now apply the PDE from (24) to obtain

$$\mathcal{B}Qu(\mathbf{x}, t) = \partial_t^q g(\mathbf{x}, t) - \mathcal{B}f(\mathbf{x}, t), \quad \mathbf{x} \in \partial\Omega. \quad (37)$$

Repeat the process ν times to find the ν^{th} compatibility condition, denoted by $\text{CBC}_{\mathcal{B},q}[\nu]$, as

$$\mathcal{B}Q^\nu u(\mathbf{x}, t) = \partial_t^{q\nu} g(\mathbf{x}, t) - \mathcal{B}\Psi_\nu f(\mathbf{x}, t), \quad \mathbf{x} \in \partial\Omega, \quad \nu = 1, 2, \dots, \quad (38)$$

where Ψ_ν is a differential operator defined by

$$\Psi_\nu f(\mathbf{x}, t) \stackrel{\text{def}}{=} \sum_{k=1}^{\nu} Q^{k-1} \partial_t^{q(\nu-k)} f(\mathbf{x}, t), \quad \mathbf{x} \in \partial\Omega, \quad \nu = 1, 2, \dots \quad (39)$$

For the time-independent case with $q = 0$, we derive the ν^{th} compatibility condition by first applying the elliptic operator Q to the governing PDE $\nu - 1$ times to obtain

$$Q^\nu u(\mathbf{x}) = -Q^{\nu-1} f(\mathbf{x}), \quad \mathbf{x} \in \bar{\Omega}, \quad \nu = 1, 2, \dots \quad (40)$$

The boundary operator in (24) is then applied to (40) to give

$$\mathcal{B}Q^\nu u(\mathbf{x}) = -\mathcal{B}Q^{\nu-1} f(\mathbf{x}), \quad \mathbf{x} \in \partial\Omega, \quad \nu = 1, 2, \dots, \quad (41)$$

which we denote by $\text{CBC}_{\mathcal{B},0}[\nu]$. We observe that the form of the CBCs in (38) for the cases $q = 1$ and 2 and in (41) for the case $q = 0$ are similar, and these conditions provide the additional constraints needed to complete the specification of the local interpolants of the solution on the boundary for the LCBC method.

4. LCBC method

We now provide a full description of the LCBC method for the semi-discrete model problem in (34). We first consider a coordinate boundary away from corners where two coordinate boundaries meet. We choose a Dirichlet-type boundary condition and introduce the LCBC method using a *direct approach*. For a more efficient implementation, we improve upon this direct approach by adopting a stencil representation of the solution at the ghost points; we call this improved method

the *stencil approach*. After this is done, we present an example where the boundary condition is of the Neumann type. Finally, we describe the treatment near the corner and discuss the conditioning of linear systems that arise in the LCBC method.

4.1. Dirichlet boundary

As an example of the LCBC method for Dirichlet boundary conditions, let us consider the left boundary, $x = 0$ with $y \in [0, 1]$, and assume that the boundary operator in (24) becomes

$$u(\mathbf{x}, t) = g_\ell(y, t), \quad \mathbf{x} \in \partial\Omega_\ell, \quad (42)$$

for a fixed time $t \in [0, T]$. Following the approach described in Section 2.2 for the wave equation defined on a unit square domain, we consider the interpolating polynomial $\tilde{u}(\mathbf{x})$ in (18) centered about a point $\tilde{\mathbf{x}} \in \partial\Omega_\ell$, and then specify its $\tilde{m} = (2p + 1)^2$ coefficients $d_{\tilde{m}, \hat{n}}$, $\hat{m}, \hat{n} = -p, \dots, p$, by enforcing the constraints

$$\tilde{u}(0, \tilde{y} + \hat{j}\Delta y) = g_\ell(\tilde{y} + \hat{j}\Delta y, t), \quad \hat{j} = -p, \dots, p, \quad (43a)$$

$$\tilde{u}(\hat{i}\Delta x, \tilde{y} + \hat{j}\Delta y) = U_{\hat{i}, \hat{j} + \hat{j}}(t), \quad \hat{i} = 1, \dots, p, \quad \hat{j} = -p, \dots, p, \quad (43b)$$

$$\partial_y^\mu Q^\nu \tilde{u}(0, \tilde{y}) = \partial_y^\mu R_{\ell, \nu}(\tilde{y}, t), \quad \nu = 1, \dots, p, \quad \mu = 0, \dots, 2p, \quad (43c)$$

where

$$R_{\ell, \nu}(y, t) \stackrel{\text{def}}{=} \begin{cases} -Q^{\nu-1} f(0, y), & q = 0, \\ \partial_t^{q\nu} g_\ell(y, t) - \Psi_\nu f(0, y, t), & q = 1, 2. \end{cases} \quad (44)$$

The constraints in (43a) are the Dirichlet boundary condition applied at $2p + 1$ grid points about the boundary point $(0, \tilde{y})$, while (43b) sets \tilde{u} equal to $U_{\mathbf{i}}$ at $p(2p + 1)$ grid points interior to the boundary point. The last constraints in (43c) require that \tilde{u} satisfy $(2p + 1)$ tangential derivatives of the compatibility boundary conditions, $\text{CBC}_{\ell, q}[\nu]$, $\nu = 1, \dots, p$, evaluated at the boundary point $(0, \tilde{y})$. Together, the constraints in (43) imply $\tilde{m} = (2p + 1)^2$ linear equations for the \tilde{m} coefficients in \tilde{u} for each point $(0, \tilde{y}) \in \partial\tilde{\Omega}_{\ell, h}$, where

$$\partial\tilde{\Omega}_{\ell, h} \stackrel{\text{def}}{=} \{ \mathbf{x}_{\mathbf{i}} \mid i = 0, j = p, p + 1, \dots, N_y - p \}, \quad (45)$$

is the set of grid points along the left boundary $x = 0$ sufficiently separated from the corners at $y = 0$ and 1.

The $\tilde{m} \times \tilde{m}$ linear system implied by (43) has the form

$$\mathbf{A} \mathbf{d} = \mathbf{b}, \quad (46)$$

where $A \in \mathbb{R}^{\tilde{m} \times \tilde{m}}$ is a coefficient matrix, $\mathbf{b} \in \mathbb{R}^{\tilde{m}}$ is a right-hand side vector and $\mathbf{d} \in \mathbb{R}^{\tilde{m}}$ is a vector containing the coefficients of the interpolating polynomial in (18) organized as

$$\mathbf{d} = [d_{-p, -p}, \dots, d_{-p, p} \mid d_{-p+1, -p}, \dots, d_{-p+1, p} \mid \dots \mid d_{p, -p}, \dots, d_{p, p}]^T. \quad (47)$$

The matrix A , as constructed in Algorithm 1 for a point $\tilde{\mathbf{x}}$ on the boundary, has the 2×2 block structure

$$A = \begin{bmatrix} A_{11} & A_{12} \\ 0 & I \end{bmatrix}. \quad (48)$$

The elements in the matrices $A_{11} \in \mathbb{R}^{\tilde{m}_1 \times \tilde{m}_1}$ and $A_{12} \in \mathbb{R}^{\tilde{m}_1 \times \tilde{m}_2}$, with $\tilde{m}_1 = p(2p + 1)$ and

$\tilde{m}_2 = (p + 1)(2p + 1)$, are obtained from derivatives of the interpolating polynomial \tilde{u} implied by the conditions in (43c). The $\tilde{m}_2 \times \tilde{m}_2$ identity in the lower-right block of A is implied by the conditions in (43a) and (43b). The matrix A is nonsingular provided that the coefficient function $c_{11}(\mathbf{x})$ associated with the highest x -derivative in the differential operator Q does not vanish (see Theorem 1 discussed later in Section 5.1). Algorithm 2 shows the construction of the right-hand side vector \mathbf{b} which follows similar steps to that used to build A . The solution of (46) yields the coefficients $d_{\hat{m},\hat{n}}$ of the interpolating polynomial, and in particular

$$U_{\hat{i},\hat{j}} = d_{\hat{i},0}, \quad \hat{i} = -p, \dots, -1, \quad (49)$$

which sets the values of U_i in the p ghost points corresponding to the boundary point $\tilde{\mathbf{x}}$.

Algorithm 1 Construct the coefficient matrix A for a Dirichlet boundary.

```

1:  $r = 0$ ;
2: for  $\nu = 1, \dots, p$  do
3:   for  $\mu = 0, \dots, 2p$  do
4:      $r = r + 1$ ;
5:     for  $\hat{m} = -p, \dots, p$  do
6:       for  $\hat{n} = -p, \dots, p$  do
7:          $c = (2p + 1)(\hat{m} + p) + \hat{n} + p + 1$ ;
8:          $A(r, c) = \partial_y^\mu Q^\nu L_{\hat{m}}((x - \tilde{x})/\Delta x) L_{\hat{n}}((y - \tilde{y})/\Delta y)|_{\mathbf{x}=\tilde{\mathbf{x}}}$ ; ▷ Elements of  $A$  from (43c)
9:       end for
10:    end for
11:  end for
12: end for
13: for  $\hat{i} = 0, \dots, p$  do
14:   for  $\hat{j} = -p, \dots, p$  do
15:      $r = r + 1$ ;
16:      $A(r, r) = 1$ ; ▷ Elements of  $A$  from (43a) and (43b)
17:   end for
18: end for

```

Algorithm 2 Construct the right-hand side vector \mathbf{b} for a Dirichlet boundary.

```

1:  $r = 0$ ;
2: for  $\nu = 1, \dots, p$  do
3:   for  $\mu = 0, \dots, 2p$  do
4:      $r = r + 1$ ;
5:      $b(r) = \partial_y^\mu R_{\ell,\nu}(\tilde{y}, t)$ ; ▷ Elements of  $\mathbf{b}$  from (43c)
6:   end for
7: end for
8: for  $\hat{j} = -p, \dots, p$  do
9:    $r = r + 1$ ;
10:   $b(r) = g_\ell(\tilde{y} + \hat{j}\Delta y, t)$ ; ▷ Elements of  $\mathbf{b}$  from (43a)
11: end for
12: for  $\hat{i} = 1, \dots, p$  do
13:   for  $\hat{j} = -p, \dots, p$  do
14:      $r = r + 1$ ;
15:      $b(r) = U_{\hat{i},\hat{j}+\hat{j}}(t)$ ; ▷ Elements of  $\mathbf{b}$  from (43b)
16:   end for
17: end for

```

4.1.1. LCBC method: Direct approach

In the direct approach to the LCBC method, the matrix A and vector \mathbf{b} in (46) are constructed for each point on the boundary, and then the system is solved to determine ghost points following the assignments in (49) for example. Algorithms 1 and 2 provide the steps for a point $\tilde{\mathbf{x}} = (\tilde{x}, \tilde{y})$ along the boundary $\tilde{x} = 0$ for the case of a Dirichlet boundary condition, while the case of a Neumann boundary condition is described in Section 4.2 below. Points on the boundary near corners require special treatment, and this is discussed in Section 4.3.

An important element of the direct approach, and the stencil approach discussed next, is an efficient calculation of the matrix A . The main step in this calculation appears in line 8 of Algorithm 1, which is independent of time t and need only be performed once for a given problem. This step involves applying repeated y -derivatives and powers of the operator Q on the product of Lagrange basis functions $L_{\hat{m}}$ and $L_{\hat{n}}$, and then evaluating the result at a point $\tilde{\mathbf{x}}$ on the boundary. While this calculation can be carried out analytically, the form of Q in (26) involving general coefficient functions, $c_{11}(\mathbf{x})$, $c_{12}(\mathbf{x})$, etc., makes this calculation increasingly messy as the order of accuracy determined by p increases. Also, it is desirable to avoid having to specify derivatives of the coefficient functions. With these issues in mind, a more practical approach is described in Algorithm 3 which computes suitable approximations of these elements, denoted by $Z_{\hat{m}, \hat{n}}[\mu, \nu]$, in a particular column of A determined by given values of $\hat{m}, \hat{n} \in \{-p, \dots, p\}$ defining the basis functions. The row entries are determined by the integers μ and ν , and we note in advance that the algorithm only requires evaluations of the coefficient functions at points on the grid.

Algorithm 3 Compute $Z_{\hat{m}, \hat{n}}[\mu, \nu] \approx \partial_y^\mu Q^\nu L_{\hat{m}}((x - \tilde{x})/\Delta x) L_{\hat{n}}((y - \tilde{y})/\Delta y)|_{\mathbf{x}=\tilde{\mathbf{x}}}$.

```

1: for  $k = 1, \dots, p$  do
2:   for  $\hat{\mathbf{i}} \in \hat{\Omega}_h[0, k]$  do ▷ Initialize  $V_{\hat{\mathbf{i}}}[0, k] = L_{\hat{m}}(\hat{i})L_{\hat{n}}(\hat{j})$ 
3:      $V_{\hat{\mathbf{i}}}[0, k] = L_{\hat{m}}(\hat{i})L_{\hat{n}}(\hat{j});$ 
4:   end for
5: end for
6: for  $\nu = 0, \dots, p - 1$  do
7:   for  $k = 1, \dots, p - \nu$  do
8:     for  $l = 1, \dots, k - 1$  do
9:       for  $\hat{\mathbf{i}} \in \hat{\Omega}_h[\nu, k]$  do ▷ Compute corrections  $W_{\hat{\mathbf{i}}}^{(m,n)}[\nu, l]$  involving  $V_{\hat{\mathbf{i}}}[\nu, l]$ 
10:        for  $m = 0, \dots, k - l$  do
11:           $W_{\hat{\mathbf{i}}}^{(m, (k-l)-m)}[\nu, l] = (D_{+x}D_{-x})^m (D_{+y}D_{-y})^{(k-l)-m} V_{\hat{\mathbf{i}}}[\nu, l];$ 
12:        end for
13:      end for
14:    end for
15:    for  $\hat{\mathbf{i}} \in \hat{\Omega}_h[\nu + 1, k]$  do ▷ Compute  $V_{\hat{\mathbf{i}}}[\nu + 1, k] = (Q_{2k,h})V_{\hat{\mathbf{i}}}[\nu, k]$ 
16:       $V_{\hat{\mathbf{i}}}[\nu + 1, k] = \text{applyQh}\{V_{\hat{\mathbf{i}}}[\nu, k], W_{\hat{\mathbf{i}}}^{(m,n)}[\nu, k - 1], \dots, W_{\hat{\mathbf{i}}}^{(m,n)}[\nu, 1]\};$ 
17:    end for
18:  end for
19: end for
20: for  $\nu = 1, \dots, p$  do ▷ Compute  $Z_{i,j}[\mu, \nu]$  using  $V_{\hat{\mathbf{i}}}[\nu, k]$ ,  $k = 1, 2, \dots, p + 1 - \nu$ 
21:    $k = p + 1 - \nu;$ 
22:    $Z_{\hat{m}, \hat{n}}[0, \nu] = V_{0,0}[\nu, k]$ 
23:   for  $l = 1, \dots, p$  do
24:      $\mu = 2l;$ 
25:      $\{Z_{\hat{m}, \hat{n}}[\mu - 1, \nu], Z_{\hat{m}, \hat{n}}[\mu, \nu]\} = \text{applyDy}\{V_{\hat{\mathbf{i}}}[\nu, 1], \dots, V_{\hat{\mathbf{i}}}[\nu, k]\};$ 
26:   end for
27: end for

```

The first collection of steps in the algorithm results in the calculation of the grid function

$V_{\hat{\mathbf{i}}}[\nu + 1, k]$ in line 16 defined by

$$V_{\hat{\mathbf{i}}}[\nu, k] \stackrel{\text{def}}{=} (Q_{d,h})^\nu L_{\hat{m}}(\hat{i}) L_{\hat{n}}(\hat{j}), \quad \nu = 1, \dots, p, \quad (50)$$

where the indices (\hat{m}, \hat{n}) are fixed and the order of accuracy of the approximation is $d = 2k$, $k = 1, \dots, p + 1 - \nu$. Note that the highest order of accuracy, given by $2(p + 1 - \nu)$, decreases as ν increases. The calculation of $V_{\hat{\mathbf{i}}}[\nu + 1, k]$, determined by the function `applyQh`, follows from the form of the discrete operator $Q_{d,h}$ given in (32). Using the correction terms denoted by $W_{\hat{\mathbf{i}}}^{(m,n)}$ in line 11, the function sets

$$\begin{aligned} V_{\hat{\mathbf{i}}}[\nu + 1, k] = & c_{11}(\mathbf{x}_{\hat{\mathbf{i}}}) D_{+x} D_{-x} \left\{ V_{\hat{\mathbf{i}}}[\nu, k] + \sum_{n=1}^{k-1} a_n \Delta x^{2n} W_{\hat{\mathbf{i}}}^{(n,0)}[\nu, k - n] \right\} \\ & + 2c_{12}(\mathbf{x}_{\hat{\mathbf{i}}}) D_{0x} D_{0y} \left\{ V_{\hat{\mathbf{i}}}[\nu, k] + \sum_{n=1}^{k-1} \sum_{l=0}^n b_l b_{n-l} \Delta x^{2l} \Delta y^{2(n-l)} W_{\hat{\mathbf{i}}}^{(l,n-l)}[\nu, k - n] \right\} \\ & + c_{22}(\mathbf{x}_{\hat{\mathbf{i}}}) D_{+y} D_{-y} \left\{ V_{\hat{\mathbf{i}}}[\nu, k] + \sum_{n=1}^{k-1} a_n \Delta y^{2n} W_{\hat{\mathbf{i}}}^{(0,n)}[\nu, k - n] \right\} \\ & + c_1(\mathbf{x}_{\hat{\mathbf{i}}}) D_{0x} \left\{ V_{\hat{\mathbf{i}}}[\nu, k] + \sum_{n=1}^{k-1} b_n \Delta x^{2n} W_{\hat{\mathbf{i}}}^{(n,0)}[\nu, k - n] \right\} \\ & + c_2(\mathbf{x}_{\hat{\mathbf{i}}}) D_{0y} \left\{ V_{\hat{\mathbf{i}}}[\nu, k] + \sum_{n=1}^{k-1} b_n \Delta y^{2n} W_{\hat{\mathbf{i}}}^{(0,n)}[\nu, k - n] \right\} + c_0(\mathbf{x}_{\hat{\mathbf{i}}}) V_{\hat{\mathbf{i}}}[\nu, k], \end{aligned} \quad (51)$$

where the coefficients (a_n, b_n) are given in (33). The domain for the local index $\hat{\mathbf{i}}$, denoted by $\hat{\Omega}_h[\nu, k]$, for each calculation is defined by

$$\hat{\Omega}_h[\nu, k] \stackrel{\text{def}}{=} [-w_x, w_x] \times [-w_y, w_y], \quad w_x = p - (\nu + k - 1), \quad w_y = w_x + p, \quad (52)$$

and this gives the minimum stencil width required for the subsequent calculation of the discrete y -derivatives of $V_{\hat{\mathbf{i}}}[\nu, k]$ performed in the second collection of steps starting at line 20. Here, the main step involves the function `applyDy` in line 25 which computes the odd/even derivative pair $Z_{\hat{m}, \hat{n}}[\mu - 1, \nu]$ and $Z_{\hat{m}, \hat{n}}[\mu, \nu]$ using standard centered finite differences in the y -direction to order of accuracy $d = 2k = 2(p + 1 - \nu)$.

The elements of the right-hand side vector \mathbf{b} in (46) are specified by Algorithm 2 for the case of a Dirichlet boundary along $\tilde{x} = 0$. The difficult step appears in line 5 and it involves the calculation of successive y -derivatives of $R_{\ell, \nu}(\tilde{y}, t)$ defined in (44). The calculation of $R_{\ell, \nu}(y, t)$, in turn, requires powers of the operator Q applied to the forcing function $f(\mathbf{x}, t)$. As before, we use a practical approach in which the various derivatives, both in space and time, are performed approximately to appropriate orders of accuracy. At present we have considered only a spatial discretization in the semi-discrete model in (34) and so we assume the time derivatives in $R_{\ell, \nu}(\tilde{y}, t)$ are exact for now. In terms of the spatial approximations, a key step involves applying powers of the discrete operator $Q_{d,h}$ onto $f(\mathbf{x}, t)$ evaluated at grid points about $\tilde{\mathbf{x}}$, and this can be done efficiently following steps similar to those described in Algorithm 3. Discrete y -derivatives are then applied to the result, again following the previous algorithm. The principal details involve the approximations of $R_{\ell, \nu}(\tilde{y}, t)$ and these are given in Algorithm 4 for the time-dependent cases $q = 1, 2$ (with straightforward simplifications for the steady case $q = 0$). For Algorithm 4, we redefine the domain for the local index $\hat{\mathbf{i}}$ in (52) such that $w_x = p - (\nu + k)$.

Algorithm 4 Compute $R_{\ell, \hat{j}}[\nu, t] \approx R_{\ell, \nu}(\tilde{y} + \hat{j}\Delta y, t)$ for $q > 0$

```

1: for  $\nu = 1, \dots, p$  do
2:   for  $\hat{j} \in [-p, p]$  do ▷ Initialize  $R_{\ell, \hat{j}}[\nu, t] = \partial_t^{q\nu} g_\ell(\tilde{y} + \hat{j}\Delta y, t)$ 
3:      $R_{\ell, \hat{j}}[\nu, t] = \text{applyDt}\{g_{\ell, \hat{j}}(t), q\nu\}$ ;
4:   end for
5: end for
6: for  $n = 0, \dots, p - 1$  do
7:   for  $k = 1, \dots, p$  do
8:     for  $\hat{i} \in \hat{\Omega}_h[0, k]$  do ▷ Initialize  $F_{\hat{i}}[0, k, t] = \partial_t^{qn} f(\tilde{\mathbf{x}} + \mathbf{x}_{\hat{i}}, t)$ 
9:        $F_{\hat{i}}[0, k, t] = \text{applyDt}\{f_{\hat{i}}(t), qn\}$ ;
10:    end for
11:  end for
12:  for  $\bar{\nu} = 0, \dots, p - n - 2$  do
13:    for  $k = 1, \dots, p - \bar{\nu} - 1$  do
14:      for  $l = 1, \dots, k - 1$  do
15:        for  $\hat{i} \in \hat{\Omega}_h[\bar{\nu}, k]$  do ▷ Compute corrections  $W_{\hat{i}}^{(m, n)}[\bar{\nu}, l, t]$  involving  $F_{\hat{i}}[\bar{\nu}, l, t]$ 
16:          for  $m = 0, \dots, k - l$  do
17:             $W_{\hat{i}}^{(m, (k-l)-m)}[\bar{\nu}, l, t] = (D_{+x}D_{-x})^m (D_{+y}D_{-y})^{(k-l)-m} F_{\hat{i}}[\bar{\nu}, l, t]$ ;
18:          end for
19:        end for
20:      end for
21:      for  $\hat{i} \in \hat{\Omega}_h[\bar{\nu} + 1, k]$  do ▷ Compute  $F_{\hat{i}}[\bar{\nu} + 1, k, t] = (Q_{2k, h})F_{\hat{i}}[\bar{\nu}, k, t]$ 
22:         $F_{\hat{i}}[\bar{\nu} + 1, k, t] = \text{applyQh}\{F_{\hat{i}}[\bar{\nu}, k, t], W_{\hat{i}}^{(m, n)}[\bar{\nu}, k - 1, t], \dots, W_{\hat{i}}^{(m, n)}[\bar{\nu}, 1, t]\}$ ;
23:      end for
24:    end for
25:  end for
26:  for  $\nu = n + 1, \dots, p$  do
27:     $\bar{\nu} = \nu - n - 1$ ;
28:     $k = \min\{p - \bar{\nu}, p\}$ ;
29:    for  $\hat{j} \in [-p, p]$  do ▷ Update  $R_{\ell, \hat{j}}[\bar{\nu}, t]$ 
30:       $R_{\ell, \hat{j}}[\nu, t] = R_{\ell, \hat{j}}[\bar{\nu}, t] - F_{0, \hat{j}}[\bar{\nu}, k, t]$ ;
31:    end for
32:  end for
33: end for

```

It is worth noting that for the time-dependent cases, the elements of \mathbf{b} must be calculated at each time step. Also, the approximation of $\partial_y^\mu R_{\ell, \nu}(\tilde{y}, t)$ uses values of $R_{\ell, \hat{j}}[\nu, t]$ about \tilde{y} , computed in Algorithm 4, and these can be used by the approximations at neighboring values along the boundary. This observation suggests a possible savings in computational cost that is explored with the stencil approach discussed next.

4.1.2. LCBC method: Stencil approach

The aim of the stencil approach is to manipulate the linear system in (46) so that the values in the ghost points in (49) corresponding to a point $\tilde{\mathbf{x}}$ on the boundary can be computed using the stencil formula

$$U_{\hat{i}, \tilde{j}} = \sum_{\nu=1}^p \sum_{j=\tilde{j}-p}^{\tilde{j}+p} \alpha_{\hat{i}, \tilde{j}}^{(\nu, j)} R_{\ell, j}[\nu, t] + \sum_{i=0}^p \sum_{j=\tilde{j}-p}^{\tilde{j}+p} \beta_{\hat{i}, \tilde{j}}^{(i, j)} U_{i, j}(t), \quad \hat{i} = -p, \dots, -1, \quad (53)$$

where $\alpha_{\hat{i}, \tilde{j}}^{(\nu, j)}$ and $\beta_{\hat{i}, \tilde{j}}^{(i, j)}$ are coefficients belonging to the left boundary centered at $\mathbf{x}_{0, \tilde{j}}$. A central point is that the coefficients in (53) do not depend on time t and can be computed from the matrix A in (48). Thus, the values in the ghost points can be computed efficiently via a fixed linear

combination of the relevant time-dependent data, assuming $q = 1, 2$, given by $R_{\ell,j}[\nu, t]$ and the grid data given by $U_{i,j}(t)$. This grid data includes values at interior points close to the boundary for $i = 1, \dots, p$ and Dirichlet boundary data, $U_{0,j}(t) = g_\ell(y_j, t)$. Note that Algorithm 4 computes $R_{\ell,\hat{j}}[\nu, t]$ for values of the local index \hat{j} about j , but the range of the y -index can be extended readily to cover the whole left boundary (sufficiently separated from the corners).

To compute the coefficients in (53), we consider the linear system in (46) in the form

$$\begin{bmatrix} A_{11} & A_{12} \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \end{bmatrix} = \begin{bmatrix} D_y \mathbf{R}(t) \\ \mathbf{U}(t) \end{bmatrix}, \quad (54)$$

where $\mathbf{d} = [\mathbf{d}_1, \mathbf{d}_2]^T$ holds the coefficients of the interpolating polynomial, $\mathbf{R}(t) \in \mathbb{R}^{\tilde{m}_1}$ is a vector containing $R_{\ell,j}[\nu, t]$, $\mathbf{U}(t) \in \mathbb{R}^{\tilde{m}_2}$ is a vector containing $U_{i,j}(t)$, and $D_y \in \mathbb{R}^{\tilde{m}_1 \times \tilde{m}_1}$ is the matrix operator representing the discrete y -derivatives of $R_{\ell,j}[\nu, t]$. We are mainly interested in the elements of \mathbf{d}_1 which give the ghost values in (49). The lower set of \tilde{m}_2 equations in (54) implies $\mathbf{d}_2 = \mathbf{U}(t)$ so that the upper set of \tilde{m}_1 equations becomes

$$A_{11} \mathbf{d}_1 = D_y \mathbf{R}(t) - A_{12} \mathbf{U}(t). \quad (55)$$

Let $C_\alpha \in \mathbb{R}^{\tilde{m}_1 \times \tilde{m}_1}$ and $C_\beta \in \mathbb{R}^{\tilde{m}_1 \times \tilde{m}_2}$ solve the matrix systems

$$A_{11} C_\alpha = D_y, \quad A_{11} C_\beta = -A_{12}, \quad (56)$$

so that (55) reduces to

$$\mathbf{d}_1 = C_\alpha \mathbf{R}(t) + C_\beta \mathbf{U}(t). \quad (57)$$

The sets of coefficients, $\{\alpha_{i,\hat{j}}^{(\nu,j)}\}$ and $\{\beta_{i,\hat{j}}^{(i,j)}\}$, in the stencil formula in (53) are given by the elements along selected rows of C_α and C_β , respectively, corresponding to the desired ghost values in \mathbf{d}_1 . We note also that the linear systems in (56) are dense but not very large, e.g. A_{11} is 21×21 for $p = 3$, and they can be solved readily using standard linear algebra software.

4.2. Neumann boundary

We now turn our attention to the case of a Neumann boundary. Let us again treat the left boundary, $x = 0$ with $y \in [0, 1]$, and assume a primary boundary condition given by

$$\partial_x u(\mathbf{x}, t) = g_\ell(y, t), \quad \mathbf{x} \in \partial\Omega_\ell, \quad (58)$$

for a fixed time $t \in [0, T]$. We consider the interpolating polynomial $\tilde{u}(\mathbf{x})$ in (18) centered about a point $\tilde{\mathbf{x}} \in \partial\Omega_\ell$, and then specify its coefficients by enforcing the constraints

$$\tilde{u}(\hat{i}\Delta x, \tilde{y} + \hat{j}\Delta y) = U_{\hat{i}, \hat{j} + \hat{j}}(t), \quad \hat{i} = 0, \dots, p, \quad \hat{j} = -p, \dots, p, \quad (59a)$$

$$\partial_y^\mu \partial_x Q^\nu \tilde{u}(0, \tilde{y}) = \partial_y^\mu S_{\ell,\nu}(\tilde{y}, t), \quad \nu = 0, \dots, p-1, \quad \mu = 0, \dots, 2p, \quad (59b)$$

where

$$S_{\ell,\nu}(y, t) \stackrel{\text{def}}{=} \begin{cases} -\partial_x Q^{\nu-1} f(0, y), & q = 0, \\ \partial_t^{q\nu} g_\ell(y, t) - \partial_x \Psi_\nu f(0, y, t), & q = 1, 2. \end{cases} \quad (60)$$

The interpolating polynomial \tilde{u} is set equal to the data $U_i(t)$ in (59a) at $2p + 1$ points on the boundary and at $p(2p + 1)$ points in the interior adjacent to the boundary. The compatibility

conditions in (59b) include tangential derivatives of the Neumann boundary condition in (58) for the case $\nu = 0$. Note that we use only $p - 1$ CBCs here since the boundary condition involves a normal derivative, whereas p CBCs are used for the case of a Dirichlet boundary. As before, the constraints in (59) imply $\tilde{m} = (2p + 1)^2$ linear equations for the \tilde{m} coefficients in \tilde{u} for each point $(0, \tilde{y}) \in \partial\tilde{\Omega}_{\ell, h}$ given in (45), and these equations can be written in a matrix form following (46) with a coefficient matrix A and right-hand side vector \mathbf{b} generated in Algorithms 5 and 6, respectively. The solution of the linear system gives \mathbf{d} , the coefficients of $\tilde{u}(\mathbf{x})$, and thus the ghost point values in (49) associated with the point $\tilde{\mathbf{x}}$ on the Neumann boundary.

Algorithm 5 Construct the coefficient matrix A for a Neumann boundary.

```

1:  $r = 0$ ;
2: for  $\nu = 0, \dots, p - 1$  do
3:   for  $\mu = 0, \dots, 2p$  do
4:      $r = r + 1$ ;
5:     for  $\hat{m} = -p, \dots, p$  do
6:       for  $\hat{n} = -p, \dots, p$  do
7:          $c = (2p + 1)(\hat{m} + p) + \hat{n} + p + 1$ ;
8:          $A(r, c) = \partial_y^\mu \partial_x Q^\nu L_{\hat{m}}((x - \tilde{x})/\Delta x) L_{\hat{n}}((y - \tilde{y})/\Delta y)|_{\mathbf{x}=\tilde{\mathbf{x}}}$ ; ▷ Elements of  $A$  from (59b)
9:       end for
10:    end for
11:  end for
12: end for
13: for  $i = 0, \dots, p$  do
14:   for  $j = -p, \dots, p$  do
15:      $r = r + 1$ ;
16:      $A(r, r) = 1$ ; ▷ Elements of  $A$  from (59a)
17:   end for
18: end for

```

Algorithm 6 Construct the right-hand side vector \mathbf{b} for a Neumann boundary.

```

1:  $r = 0$ ;
2: for  $\nu = 0, \dots, p - 1$  do
3:   for  $\mu = 0, \dots, 2p$  do
4:      $r = r + 1$ ;
5:      $b(r) = \partial_y^\mu S_{\ell, \nu}(\tilde{y}, t)$ ; ▷ Elements of  $\mathbf{b}$  from (59b)
6:   end for
7: end for
8: for  $\hat{i} = 0, \dots, p$  do
9:   for  $\hat{j} = -p, \dots, p$  do
10:     $r = r + 1$ ;
11:     $b(r) = U_{\hat{i}, \hat{j} + \hat{j}}(t)$ ; ▷ Elements of  $\mathbf{b}$  from (59a)
12:   end for
13: end for

```

The elements in line 8 of Algorithm 5, involving repeated applications of the operator Q on the x -derivative of the Lagrange basis functions followed by successive y -derivatives, are obtained approximately. The details of these approximations are similar to those given in Algorithm 3 for the Dirichlet case. The main difference for the case of a Neumann boundary is the addition of a discrete x -derivative applied to $V_i[\nu, k]$, to an appropriate order of accuracy, prior to applications of the discrete y -derivatives. The y -derivatives of $S_{\ell, \nu}(\tilde{y}, t)$ in line 5 of Algorithm 6 are also computed approximately.

Instead of solving the linear system in (46) at each time step (for $q = 1, 2$), we prefer the stencil

approach. Here, the stencil formula for the ghost points associated with a point $\mathbf{x}_{0,\tilde{j}}$ along the left boundary becomes

$$U_{\hat{i},\tilde{j}} = \sum_{\nu=0}^{p-1} \sum_{j=\tilde{j}-p}^{\tilde{j}+p} \bar{\alpha}_{\hat{i},\tilde{j}}^{(\nu,j)} S_{\ell,j}[\nu, t] + \sum_{i=0}^p \sum_{j=\tilde{j}-p}^{\tilde{j}+p} \bar{\beta}_{\hat{i},\tilde{j}}^{(i,j)} U_{i,j}(t), \quad \hat{i} = -p, \dots, -1, \quad (61)$$

where $S_{\ell,j}[\nu, t] \approx S_{\ell,\nu}(y_j, t)$ for $q > 0$ and $\nu = 0, \dots, p-1$. The coefficients in (61) for each boundary point are obtained by manipulating the system

$$\begin{bmatrix} \bar{A}_{11} & \bar{A}_{12} \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \end{bmatrix} = \begin{bmatrix} \bar{D}_y \mathbf{S}(t) \\ \mathbf{U}(t) \end{bmatrix}. \quad (62)$$

For the Neumann case, the block matrices $\bar{A}_{11} \in \mathbb{R}^{\tilde{m}_1 \times \tilde{m}_1}$ and $\bar{A}_{12} \in \mathbb{R}^{\tilde{m}_1 \times \tilde{m}_2}$ are obtained from the approximations of the elements in line 8 of Algorithm 5. The vectors $\mathbf{S}(t) \in \mathbb{R}^{\tilde{m}_1}$ and $\mathbf{U}(t) \in \mathbb{R}^{\tilde{m}_2}$ contain $S_{\ell,j}[\nu, t]$ and $U_{i,j}(t)$, respectively, and $\bar{D}_y \in \mathbb{R}^{\tilde{m}_1 \times \tilde{m}_1}$ contains the coefficients associated with the discrete y -derivatives of $S_{\ell,j}[\nu, t]$, $\nu = 0, \dots, p-1$. Let $\bar{C}_{\bar{\alpha}} \in \mathbb{R}^{\tilde{m}_1 \times \tilde{m}_1}$ and $\bar{C}_{\bar{\beta}} \in \mathbb{R}^{\tilde{m}_1 \times \tilde{m}_2}$ solve the systems

$$\bar{A}_{11} \bar{C}_{\bar{\alpha}} = \bar{D}_y, \quad \bar{A}_{11} \bar{C}_{\bar{\beta}} = -\bar{A}_{12}, \quad (63)$$

so that

$$\mathbf{d}_1 = \bar{C}_{\bar{\alpha}} \mathbf{S}(t) + \bar{C}_{\bar{\beta}} \mathbf{U}(t). \quad (64)$$

As before, the sets of coefficients of the stencil formula in (61) are given by the elements along selected rows of $\bar{C}_{\bar{\alpha}}$ and $\bar{C}_{\bar{\beta}}$ corresponding to the desired ghost values in \mathbf{d}_1 .

4.3. LCBC conditions at a corner

As a representative case involving the conditions at a corner, let us consider the bottom-left corner, $\tilde{x} = (0, 0)$, where two Dirichlet boundaries meet. The cases of a Neumann-Neumann corner and a Dirichlet-Neumann corner are discussed in Appendix A. The physical (primary) boundary are taken to be

$$u(\mathbf{x}, t) = g_\ell(y, t), \quad \mathbf{x} \in \partial\Omega_\ell, \quad (65a)$$

$$u(\mathbf{x}, t) = g_b(x, t), \quad \mathbf{x} \in \partial\Omega_b, \quad (65b)$$

for a some fixed time t . We start by specifying the interpolating polynomial $\tilde{u}(\mathbf{x})$ at known interior data given by

$$\tilde{u}(\hat{i}\Delta x, \hat{j}\Delta y) = U_{\hat{i},\hat{j}}(t), \quad \hat{i} = 1, \dots, p, \quad \hat{j} = 1, \dots, p. \quad (66a)$$

Next, we apply tangential derivatives of the primary boundary conditions and compatibility conditions given by

$$\left. \begin{aligned} \partial_y^\mu \tilde{u}(0, 0) &= \partial_y^\mu g_\ell(0, t) \\ \partial_x^\mu \tilde{u}(0, 0) &= \partial_x^\mu g_b(0, t) \end{aligned} \right\} \quad \mu \in \mathcal{M}_0, \quad (66b)$$

and

$$\left. \begin{aligned} \partial_y^\mu Q^\nu \tilde{u}(0, 0) &= \partial_y^\mu R_{\ell,\nu}(0, t) \\ \partial_x^\mu Q^\nu \tilde{u}(0, 0) &= \partial_x^\mu R_{b,\nu}(0, t) \end{aligned} \right\} \quad \nu = 1, \dots, p, \quad \mu \in \mathcal{M}_\nu, \quad (66c)$$

respectively, where $R_{\ell,\nu}(y, t)$ is defined in (44) and $R_{b,\nu}(x, t)$ is defined by

$$R_{b,\nu}(x, t) \stackrel{\text{def}}{=} \begin{cases} -Q^{\nu-1} f(x, 0), & q = 0, \\ \partial_t^{q\nu} g_b(x, t) - \Psi_\nu f(x, 0, t), & q = 1, 2. \end{cases} \quad (67)$$

The sets \mathcal{M}_ν , $\nu = 0, \dots, p$, chosen to eliminate redundant constraints, are given by

$$\mathcal{M}_\nu = \begin{cases} 0, 1, 2, 3, \dots, 2p-1, 2p, & \text{if } \nu = 0, \text{ with an average for } \mu = 0, \\ 1, 2, 3, 4, \dots, 2p-1, 2p, & \text{if } \nu = 1, \text{ with an average for } \mu = 2, \\ 1, 3, 4, 5, \dots, 2p-1, 2p, & \text{if } \nu = 2, \text{ with an average for } \mu = 4, \\ \vdots & \vdots \\ 1, 3, 5, \dots, 2p-1, 2p, & \text{if } \nu = p, \text{ with an average for } \mu = 2p. \end{cases} \quad (68)$$

Note that there is one value for μ in each set \mathcal{M}_ν where the pairs in (66b) and (66c) are averaged to resolve linearly dependent constraints (and to balance the constraints on the left and bottom boundaries). The weights for the averages are Δy^μ and Δx^μ for the CBCs arising from the left and bottom boundaries, respectively, to balance the tangential derivatives taken in the y and x directions (following the previous discussion in Section 2.2). The $\tilde{m} = (2p+1)^2$ constraints in (66) for the \tilde{m} coefficients $d_{\hat{m}, \hat{n}}$ of $\tilde{u}(\mathbf{x})$ are a generalization of the approach for the bottom-left corner given in (23) for the wave equation. Ghost points near the corner can be obtained from the solution of the linear system implied by (66) following a direct approach, or these ghost points can be written in terms of the stencil formula

$$U_{\hat{i}, \hat{j}} = \sum_{\nu=0}^p \sum_{j=-p}^p \tilde{\alpha}_{\hat{i}, \hat{j}}^{(\nu, j)} R_{\ell, j}[\nu, t] + \sum_{\nu=0}^p \sum_{i=-p}^p \tilde{\beta}_{\hat{i}, \hat{j}}^{(\nu, i)} R_{b, i}[\nu, t] + \sum_{i=1}^p \sum_{j=1}^p \tilde{\gamma}_{\hat{i}, \hat{j}}^{(i, j)} U_{i, j}(t), \quad \hat{\mathbf{i}} \in \hat{\Omega}_c, \quad (69)$$

where

$$\hat{\Omega}_c \stackrel{\text{def}}{=} \{ \hat{\mathbf{i}} = (\hat{i}, \hat{j}) \mid -p \leq (\hat{i}, \hat{j}) < p \setminus 1 \leq (\hat{i}, \hat{j}) < p \} \quad (70)$$

defines the set of local indices for the ghost-point values in (69). The time-dependent data $R_{\ell, j}[\nu, t]$ and $R_{b, i}[\nu, t]$ in (69), assuming $q = 1, 2$, are discrete approximations of $R_{\ell, \nu}(j\Delta y, t)$ and $R_{b, \nu}(i\Delta x, t)$, respectively, for $\nu = 1, \dots, p$. The boundary conditions are specified in (69) by setting

$$R_{\ell, j}[0, t] = g_\ell(j\Delta y, t), \quad j = -p, \dots, p, \quad (71a)$$

$$R_{b, i}[0, t] = g_b(i\Delta x, t), \quad i = -p, \dots, p, \quad (71b)$$

similar to previous specifications. The coefficients in the stencil formula are derived from the $\tilde{m} \times \tilde{m}$ linear system implied by (66) following the analysis described for the Dirichlet and Neumann boundaries.

Our choice for the constraints in (66) is guided by the case when Q in (26) is the Laplacian operator. For this case, the constraints are linearly independent. For the more general operator Q with variable coefficients, the constraints remain linearly independent provided $c_{11}(\mathbf{x}) > 0$, $c_{22}(\mathbf{x}) > 0$ and $\max \left\{ |c_{12}(\mathbf{x})| / \sqrt{c_{11}(\mathbf{x})c_{22}(\mathbf{x})} \right\}$ is small, with \mathbf{x} evaluated at the corner. Should these conditions be violated, the $\tilde{m} \times \tilde{m}$ matrix A implied by (66) may become singular or badly conditioned. For example, if $c_{11} > 0$, $c_{22} > 0$ and c_{12} are constants, and if $c_1 = c_2 = c_0 = 0$, then

the determinant of A for the case $p = 1$ ($d = 2$) has the form

$$\det(A) = -D\Delta x\Delta y(c_{11} + c_{22})(c_{11}c_{22} - 4c_{12}^2), \quad D = \text{constant} > 0.$$

Thus, A becomes singular when $|c_{12}| = \sqrt{c_{11}c_{22}}/2$. Another case for which A is rank deficient occurs when $c_{11} = c_{22} = 1$, $c_{12} = 1/2$, $c_1 = c_2 = c_0 = 0$ and $\Delta x = \Delta y$, and for any value of p .

As noted earlier, we are motivated by high-order accurate discretizations of the model problem in (3.1). For many problems of interest, this model problem is obtained by an orthogonal, or near-orthogonal, mapping of a PDE in physical space involving the Laplacian operator. The resulting mapped problem, of the form in section (3.1), would have $|c_{12}(\mathbf{x})|$ small relative to $c_{11}(\mathbf{x})$ and $c_{22}(\mathbf{x})$ resulting in a nonsingular matrix A implied by the constraints in (66) for a Dirichlet-Dirichlet corner. The matrices for the Neumann-Neumann and Dirichlet-Neumann corners, discussed in the appendix, are also nonsingular under these conditions.

4.4. Conditioning of LCBC matrices

We are interested in the sensitivity of the computational procedures involving LCBC matrices to data perturbations and round-off errors. We first note that the rows of the LCBC matrices for both sides and corners vary in magnitude significantly. Therefore, we choose to use the `equilibrate` function in MATLAB to optimize the conditioning of the LCBC matrices by permuting the rows and then applying row and column scalings. The diagonal entries of each permuted and scaled matrix are all one in magnitude while the off-diagonal elements are less than or equal to one. This decreases the number of pivots needed for numerical stability in Gaussian elimination (see [41]). We shall then adopt the scaling proposed by `equilibrate` to generate the condition numbers for the numerical results presented below.

Let $\kappa(A) = \|A\| \|A^{-1}\|$ be the 2-norm condition number of an LCBC matrix A generated for a point on a boundary or a corner, and let κ_{\max} be the maximum of $\kappa(A)$ for all points along the four boundaries or for the four corners. We compute κ_{\max} for the following test cases:

1. We set $\Omega = (0, 1)^2$, $Q = \Delta$ and take Dirichlet boundary conditions at $x = 0$ and 1 , $y \in [0, 1]$, and Neumann boundary conditions at $y = 0$ and 1 , $x \in [0, 1]$. This case corresponds to the numerical example presented later in Section 6.2.1.
2. We set $\Omega = \{\mathbf{x} = (\rho \cos \theta, \rho \sin \theta) \mid 1 < \rho < 2, 0 < \theta < \pi/2\}$ and $Q = \Delta$, and take Dirichlet boundary conditions at $\rho = 1$, $\theta \in [0, \pi/2]$ and at $\theta = 0$, $\rho \in [1, 2]$ and Neumann boundary conditions at $\rho = 2$, $\theta \in [0, \pi/2]$ and at $\theta = \pi/2$, $\rho \in [1, 2]$. This case corresponds to the numerical example presented in Section 6.2.3.
3. We set Ω to be the wavy channel domain defined in (130) for the problem described in Section 6.4. The spacial operator is $Qu = \mathcal{D}\Delta u - \mathbf{v} \cdot \nabla u + \gamma u$, where $\mathcal{D} = 0.2$, $\mathbf{v} = (0.5, 0.3)$ and $\gamma = 1$, and we use Dirichlet conditions for $\mathbf{x} \in \partial\Omega$.

Figure 4 shows the computed values of κ_{\max} for the side and corner LCBC matrices as functions of the grid spacing h used for each test case and for orders of accuracy $d = 2, 4$ and 6 . For the first two test cases, we observe that κ_{\max} for the side and corner LCBC matrices are approximately equal, and they are nearly independent of the grid spacing h , although κ_{\max} shows a moderate increase with increasing values of the order of approximation d for $Q_{d,h}$. The behavior of κ_{\max} for the third test case shows more variation with h for the corner matrices when $d = 4$ and 6 . The plots in Figure 5 show more details of the behavior of κ_{\max} as a function of h for $d = 4$. There is more variation for coarser grids, which we attribute to the affects of the lower-order terms in Q , while the behavior of κ_{\max} appears to settle to a constant value as the grid spacing gets smaller (and affects of the lower-order terms become less significant).

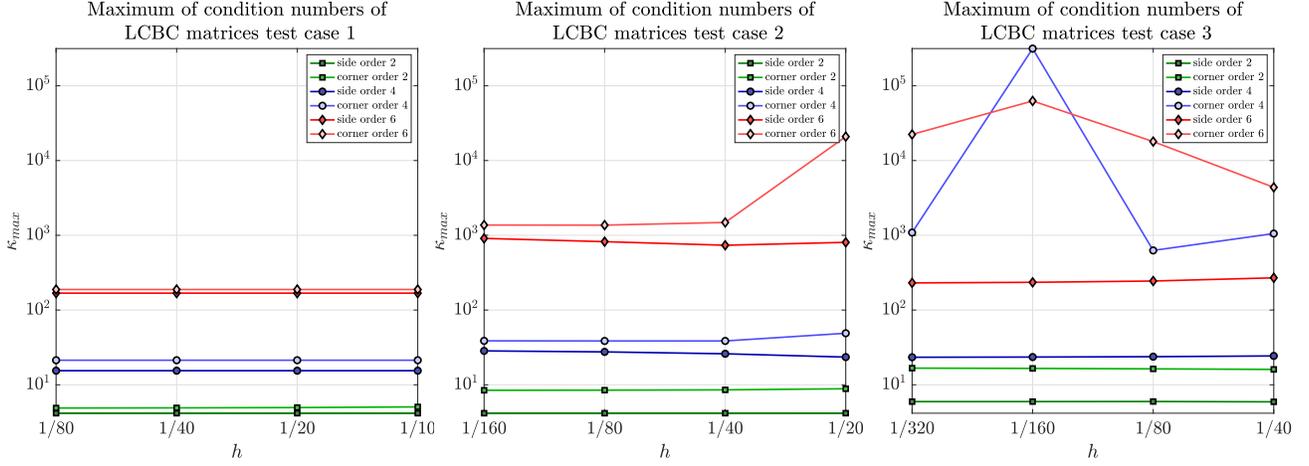


Figure 4: Condition numbers of LCBC matrices as functions of the grid spacing h for test cases 1, 2 and 3.

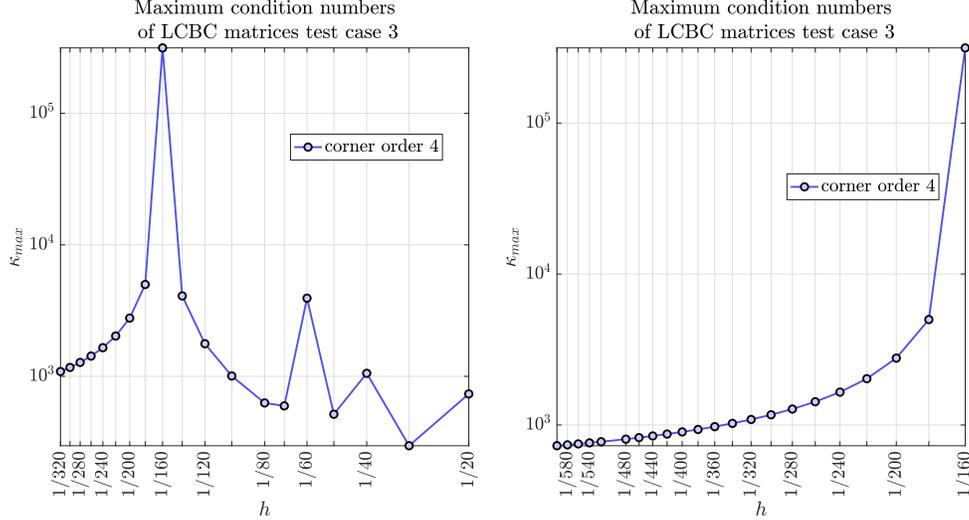


Figure 5: Test case 3: κ_{\max} versus h for corner matrices at order $d = 4$.

5. Analysis of the LCBC approach

In this section, we provide some results of an analysis of the LCBC approach. In particular, we consider the solvability of the matrix systems associated with the constraints implied by the LCBC method for points along a grid side and at a grid corner. We then consider symmetry properties of the discrete approximations generated by the LCBC method for the case when the PDE involves the Laplacian operator. Finally, we examine the stability of explicit time-stepping schemes for the wave equation with numerical boundary conditions given by the LCBC approach.

5.1. Solvability of the LCBC matrix systems

We first consider conditions required for the LCBC matrix systems to be non-singular. This is done for the case of a constant-coefficient operator Q given by

$$Q = c_{11}\partial_x^2 + 2c_{12}\partial_x\partial_y + c_{22}\partial_y^2 + c_1\partial_x + c_2\partial_y + c_0. \quad (72)$$

For this operator, we have the following result:

Theorem 1 (Solvability on a face). *The matrix resulting from the order $2p = 2, 4, 6$ LCBC constraints for the constant-coefficient operator Q in (72) with a Dirichlet or Neumann boundary condition on a grid face is non-singular provided $c_{11} > 0$ and Δx is sufficiently small (left or right face) or $c_{22} > 0$ and Δy is sufficiently small (bottom or top face). If $c_1 = 0$ (left face) or $c_2 = 0$ (right face), then the matrix is non-singular for any Δx and Δy .*

Proof. Let us focus on the left boundary, while similar arguments hold for the other boundaries. The form of the LCBC matrix system is given in (46), where the coefficient matrix A is obtained from the constraints in (43) for a Dirichlet boundary and in (59) for a Neumann boundary. For either a Dirichlet or Neumann boundary, the determinant of A , for order of accuracy $2p = 2, 4, 6$, has the form

$$\det(A) = K_p G_p(\xi), \quad \xi = \frac{c_1 \Delta x}{c_{11}}, \quad p = 1, 2, 3, \quad (73)$$

where K_p is a non-zero constant depending on Δx , Δy and c_{11} , and where $G_p(\xi)$ is a polynomial satisfying $G_p(0) = 1$. For the Dirichlet case, the polynomials are given by

$$\begin{aligned} G_1(\xi) &= \left(1 - \frac{\xi}{2}\right)^3, \\ G_2(\xi) &= \left(1 - \frac{3\xi}{2} + \frac{\xi^2}{2} - \frac{\xi^3}{18}\right)^5, \\ G_3(\xi) &= \left(1 - 3\xi + \frac{11\xi^2}{4} - \frac{1691\xi^3}{1440} + \frac{121\xi^4}{480} - \frac{11\xi^5}{400} + \frac{\xi^6}{800}\right)^7, \end{aligned}$$

and for the Neumann case, we have

$$\begin{aligned} G_1(\xi) &= 1, \\ G_2(\xi) &= \left(1 - \frac{2\xi}{9}\right)^5, \\ G_3(\xi) &= \left(1 - \frac{23\xi}{30} + \frac{11\xi^2}{75} - \frac{\xi^3}{100}\right)^7. \end{aligned}$$

The result of the theorem follows from the form of the determinant of A in (73). As expected, the lower order terms in (72) become less important for the solvability of the system as the grid spacings tend to zero.

□

The solvability conditions at a corner are more complicated. For this case, we focus on the constant-coefficient operator in (72) with the coefficients of the lower-order terms set to zero, i.e. $c_0 = c_1 = c_2 = 0$, and define the dimensionless parameters

$$\gamma = \frac{c_{12}}{\sqrt{c_{11}c_{22}}}, \quad \sigma = \sqrt{\frac{c_{11}/\Delta x^2}{c_{22}/\Delta y^2}},$$

assuming $c_{11} > 0$ and $c_{22} > 0$. Recall that when choosing the corner compatibility conditions we assumed that $|c_{12}|$ is small compared to c_{11} and c_{22} , and this now corresponds to $|\gamma|$ small. The

following theorem describes the solvability of the LCBC matrix systems for the Dirichlet-Dirichlet (D-D), Neumann-Neumann (N-N) and Dirichlet-Neumann (D-N) corners.

Theorem 2 (Solvability at a corner). *The matrices resulting from the LCBC constraints at D-D, N-N and D-N corners for the constant-coefficient operator Q in (72) with $c_{11} > 0$, $c_{22} > 0$, and $c_0 = c_1 = c_2 = 0$ are nonsingular provided any of the following conditions hold:*

1. $\gamma = 0$ ($c_{12} = 0$), for orders $2p = 2, 4, 6$.
2. $|\gamma|$ is sufficiently small, for orders $2p = 2, 4$.
3. $\gamma < 0$ and $|\gamma|$ is sufficiently small, for order $2p = 6$.
4. $\gamma > 0$ and $(\sigma + 1/\sigma)\gamma$ is sufficiently small, for order $2p = 6$.

Proof. We consider the corner where the left and bottom boundaries meet, while similar arguments hold for the other corners. The form of the LCBC matrix system is given in (46), where the coefficient matrix A is obtained from the constraints in (66), (A.2) and (A.6) for D-D, N-N and D-N corners, respectively. For all three cases, the determinant of A , for order of accuracy $2p = 2, 4, 6$, has the form

$$\det(A) = K_p H_p(\gamma) F_p(\gamma, \sigma), \quad p = 1, 2, 3, \quad (74)$$

where K_p is a non-zero constant depending on Δx , Δy , c_{11} and c_{22} , $H_p(\gamma)$ is a polynomial satisfying $H_p(0) = 1$, and $F_p(\gamma, \sigma)$ is a polynomial in γ with coefficients that depend on σ . For a D-D corner, we have

$$\begin{aligned} H_1(\gamma) &= 1 - 4\gamma^2, \\ H_2(\gamma) &= (1 - 4\gamma^2)^2 (1 - 28\gamma^2 + 208\gamma^4 - 256\gamma^6), \\ H_3(\gamma) &= (1 - 4\gamma^2)^4 (1 - 12\gamma^2 + 16\gamma^4)^2 (1 - 104\gamma^2 + 3984\gamma^4 - 68480\gamma^6 \\ &\quad + 509440\gamma^8 - 1278976\gamma^{10} + 921600\gamma^{12}), \end{aligned}$$

and

$$\begin{aligned} F_1(\gamma, \sigma) &= 1, \\ F_2(\gamma, \sigma) &= 3 \left(\sigma + \frac{1}{\sigma} \right) - 4\gamma, \\ F_3(\gamma, \sigma) &= 7200 \left(\sigma^3 + \sigma + \frac{1}{\sigma} + \frac{1}{\sigma^3} \right) - \gamma \left[3960 \left(\sigma^4 + \frac{1}{\sigma^4} \right) + 28070 \left(\sigma^2 + \frac{1}{\sigma^2} \right) + 26620 \right] \\ &\quad + \gamma^2 \left[13423 \left(\sigma^3 + \frac{1}{\sigma^3} \right) + 39483 \left(\sigma + \frac{1}{\sigma} \right) \right] - \gamma^3 \left[14399 \left(\sigma^2 + \frac{1}{\sigma^2} \right) + 28798 \right] \\ &\quad + \gamma^4 \left[5940 \left(\sigma + \frac{1}{\sigma} \right) \right]. \end{aligned}$$

For a N-N corner, we have

$$\begin{aligned} H_1(\gamma) &= 1, \\ H_2(\gamma) &= 1 - 4\gamma^2, \\ H_3(\gamma) &= (1 - 4\gamma^2)^2 (1 - 28\gamma^2 + 208\gamma^4 - 256\gamma^6), \end{aligned}$$

and

$$\begin{aligned}
F_1(\gamma, \sigma) &= 1, \\
F_2(\gamma, \sigma) &= 27 \left(\sigma + \frac{1}{\sigma} \right) - 32\gamma, \\
F_3(\gamma, \sigma) &= 56250 \left(\sigma^3 + \sigma + \frac{1}{\sigma} + \frac{1}{\sigma^3} \right) - \gamma \left[24750 \left(\sigma^4 + \frac{1}{\sigma^4} \right) + 194625 \left(\sigma^2 + \frac{1}{\sigma^2} \right) + 189750 \right] \\
&\quad + \gamma^2 \left[240310 \left(\sigma + \frac{1}{\sigma} \right) + 25450 \left(\sigma^3 + \frac{1}{\sigma^3} \right) \right] - \gamma^3 \left[71564 \left(\sigma^2 + \frac{1}{\sigma^2} \right) + 150040 \right] \\
&\quad + \gamma^4 \left[25344 \left(\sigma + \frac{1}{\sigma} \right) \right].
\end{aligned}$$

Finally, for a D-N corner, we have

$$\begin{aligned}
H_1(\gamma) &= 1, \\
H_2(\gamma) &= 1 - 12\gamma^2 + 32\gamma^4, \\
H_3(\gamma) &= (1 - 4\gamma^2)^2 (1 - 64\gamma^2 + 1504\gamma^4 - 16128\gamma^6 + 80640\gamma^8 - 171008\gamma^{10} + 122880\gamma^{12}),
\end{aligned}$$

and

$$\begin{aligned}
F_1(\gamma, \sigma) &= 1, \\
F_2(\gamma, \sigma) &= 3 \left(\sigma + \frac{1}{\sigma} \right) - 4\gamma, \\
F_3(\gamma, \sigma) &= 135000 \left(\sigma^3 + \sigma + \frac{1}{\sigma} + \frac{1}{\sigma^3} \right) - 75\gamma \left(660\sigma^4 + 7353\sigma^2 + 6523 + 6418\frac{1}{\sigma^2} + 1188\frac{1}{\sigma^4} \right) \\
&\quad + 5\gamma^2 \left(35985\sigma^3 + 158555\sigma + 126287\frac{1}{\sigma} + 54738\frac{1}{\sigma^3} \right) \\
&\quad - \gamma^3 \left(215985\sigma^2 + 520784 + 239532\frac{1}{\sigma^2} \right) + 35640\gamma^4 \left(3\sigma + 2\frac{1}{\sigma} \right).
\end{aligned}$$

Note that when $\gamma < 0$, the functions F_p are always positive and bounded away from zero. The result of the theorem follows from the form of the determinant of A in (74).

□

We note that a good quality grid usually aims to have $\sigma \approx 1$. One way to see this is to note that if $c_{11} \ll c_{22}$ then there could be boundary layers near $x = 0$ or $x = 1$, which would require a small value for Δx to resolve the solution there. We also note that for order $2p = 6$ when $\gamma > 0$ ($c_{12} > 0$), we require not just γ to be small but also $\gamma\sigma$ and γ/σ to be small. Thus the corner LCBC matrix could be poorly conditioned if σ becomes large or small when $c_{12} > 0$. This could occur, for example, if one only refined the grid in the x -direction.

5.2. Symmetry properties of the LCBC conditions

The next two theorems concern symmetry properties of the numerical boundary conditions generated by the LCBC method for a boundary face and corner. These symmetry conditions pertain to the case when Q is the Laplacian operator and the domain is represented by a Cartesian grid. The first theorem considers the symmetry for a boundary face.

Theorem 3 (Symmetry on a face). *When applied to the operator $Q = \Delta$ on a Cartesian grid, the LCBC approach on a face, at any order $2p = 2, 4, 6, \dots$, results in numerical boundary conditions with odd symmetry for the case of homogeneous Dirichlet boundary conditions and with even symmetry for the case of homogeneous Neumann boundary conditions, for example,*

$$U_{i-\hat{i},j} = -U_{i+\hat{i},j}, \quad \hat{i} = 1, \dots, p, \quad \text{Dirichlet BC at } i = 0 \text{ or } i = N_x, \quad (75a)$$

$$U_{i-\hat{i},j} = U_{i+\hat{i},j}, \quad \hat{i} = 1, \dots, p, \quad \text{Neumann BC at } i = 0 \text{ or } i = N_x. \quad (75b)$$

Proof. First consider the case of a homogeneous Dirichlet boundary condition on the left side, $i = 0$, away from the corner. Without loss of generality we may take $\tilde{x} = 0$ and $\tilde{y} = 0$, and then the polynomial interpolant \tilde{u} can be written as

$$\tilde{u}(x, y) = \sum_{n=0}^{2p} \sum_{m=0}^{2p} a_{n,m} x^n y^m, \quad p = 1, 2, \dots \quad (76)$$

We wish to show that $\tilde{u}(x, 0)$ is an odd function in x , so that

$$\tilde{u}(x, 0) = a_{1,0}x + a_{3,0}x^3 + \dots + a_{2p-1,0}x^{2p-1}, \quad (77)$$

for then we have $\tilde{u}(-x, 0) = -\tilde{u}(x, 0)$ and the desired result follows. The CBCs in (43c) reduce to

$$\partial_y^\mu \Delta^\nu \tilde{u}(\mathbf{0}) = 0, \quad \nu = 0, \dots, p, \quad \mu = 0, \dots, 2p, \quad \text{CBC}[\mu, \nu], \quad (78)$$

where the case $\nu = 0$ follows since $U_{0,j} = 0$ from the homogeneous boundary condition. For the purposes of the proof, we have labeled the conditions in (78) as $\text{CBC}[\mu, \nu]$. We will show that (78) implies that all even x -derivatives of \tilde{u} at $\mathbf{x} = \mathbf{0}$ are zero,

$$\partial_x^{2\nu} \tilde{u}(\mathbf{0}) = 0, \quad \nu = 0, \dots, p, \quad (79)$$

which implies (77). The conditions in (79) can be shown as follows. We have $\partial_y^\mu \tilde{u}(\mathbf{0}) = 0$, for $\mu = 0, 1, \dots$, since the Dirichlet conditions are homogeneous and since \tilde{u} is a polynomial of finite degree. Then, from $\text{CBC}[0, 1]$, we see that (79) holds for $\nu = 1$ since

$$\partial_x^2 \tilde{u}(\mathbf{0}) = -\partial_y^2 \tilde{u}(\mathbf{0}) = 0, \quad (80)$$

and from $\text{CBC}[\mu, 1]$ we also find

$$\partial_y^\mu \partial_x^2 \tilde{u}(\mathbf{0}) = -\partial_y^{\mu+2} \tilde{u}(\mathbf{0}) = 0, \quad \mu = 0, 1, \dots \quad (81)$$

Now from $\text{CBC}[0, 2]$, we find that (79) holds for $\nu = 2$, since

$$\partial_x^4 \tilde{u}(\mathbf{0}) = (-2\partial_x^2 \partial_y^2 - \partial_y^4) \tilde{u}(\mathbf{0}) = 0, \quad (82)$$

and from $\text{CBC}[\mu, 2]$ we also find

$$\partial_y^\mu \partial_x^4 \tilde{u}(\mathbf{0}) = 0, \quad \mu = 0, 1, \dots \quad (83)$$

The process can be repeated to show (79).

The argument is similar for the case of a homogeneous Neumann boundary condition except that

in this case it can be shown that all odd x -derivatives are zero,

$$\partial_x^{2\nu+1}\tilde{u}(\mathbf{0}) = 0, \quad \nu = 0, \dots, p, \quad (84)$$

so that $\tilde{u}(-x, 0) = \tilde{u}(x, 0)$.

□

We now consider the symmetry at a corner. For this case, note that the LCBC conditions are used to obtain values in ghost points in the corner of the extended grid and also at nearby ghost points belonging to the adjacent faces, see Figure 3 for the case $p = 2$ for example.

Theorem 4 (Symmetry at a corner). *When applied to the operator $Q = \Delta$ on a Cartesian grid, the LCBC approach applied at any corner and at any order $2p = 2, 4, 6, \dots$, results in numerical boundary conditions on the adjacent faces with odd symmetry for the case of homogeneous Dirichlet boundary conditions and with even symmetry for the case of homogeneous Neumann boundary conditions. At a left boundary, for example, the symmetries are given in (75). Values at the corner ghost points have even symmetry for Dirichlet-Dirichlet (D-D) or Neumann-Neumann (N-N) corners and odd symmetry for Dirichlet-Neumann (D-N) corners. At a bottom-left corner, for example, the values satisfy*

$$U_{i-\hat{i}, j-\hat{j}} = U_{i+\hat{i}, j+\hat{j}}, \quad \hat{i}, \hat{j} = 1, \dots, p, \quad D-D \text{ or } N-N \text{ corners}, \quad (85a)$$

$$U_{i-\hat{i}, j-\hat{j}} = -U_{i+\hat{i}, j+\hat{j}}, \quad \hat{i}, \hat{j} = 1, \dots, p, \quad D-N \text{ corner}. \quad (85b)$$

Proof. Consider the case of homogeneous Dirichlet boundary conditions on the left side, $i = 0$, and the bottom side, $j = 0$, so that we have a D-D corner at $\mathbf{x} = (0, 0)$ and grid index $\mathbf{i} = (0, 0)$. With \tilde{u} given in (76) we show that

$$\tilde{u}(-x, y) = -\tilde{u}(x, y), \quad \tilde{u}(x, -y) = -\tilde{u}(x, y), \quad (86)$$

and thus

$$\tilde{u}(-x, -y) = \tilde{u}(x, y). \quad (87)$$

To show (86), we show

$$\partial_x^{m_1} \partial_y^{m_2} \tilde{u}(\mathbf{0}) = 0, \quad m_1 = 2k, \quad m_2 = 0, 1, \dots, 2p, \quad (88a)$$

$$\partial_x^{m_1} \partial_y^{m_2} \tilde{u}(\mathbf{0}) = 0, \quad m_1 = 0, 1, \dots, 2p, \quad m_2 = 2k, \quad (88b)$$

where $k = 0, 1, \dots, p$.

Recall that \tilde{u} satisfies the boundary conditions in (66b) and the compatibility conditions in (66c) with homogeneous boundary data, so that

$$\left. \begin{aligned} \partial_y^\mu \Delta^\nu \tilde{u}(\mathbf{0}) &= 0 \\ \partial_x^\mu \Delta^\nu \tilde{u}(\mathbf{0}) &= 0 \end{aligned} \right\} \quad \nu = 0, 1, \dots, p, \quad \mu \in \mathcal{M}_\nu, \quad (89)$$

where \mathcal{M}_ν is defined in (68). Using mathematical induction, we find that (89) implies

$$\left. \begin{aligned} \partial_y^\mu \partial_x^{2\nu} \tilde{u}(\mathbf{0}) &= 0 \\ \partial_x^\mu \partial_y^{2\nu} \tilde{u}(\mathbf{0}) &= 0 \end{aligned} \right\} \quad \nu = 0, 1, \dots, p, \quad \mu \in \mathcal{M}_\nu. \quad (90)$$

Set $m_1 = 2k$ for $k = 0, 1, \dots, p$. The first set of conditions in (90) implies that

$$\partial_x^{m_1} \partial_y^{m_2} \tilde{u}(\mathbf{0}) = 0, \quad \text{for } m_2 = 1, 3, 5, \dots, 2k-1, 2k, 2k+1, \dots, 2p, \quad (91)$$

while the second set of conditions in (90) gives

$$\partial_x^{m_1} \partial_y^{m_2} \tilde{u}(\mathbf{0}) = 0, \quad \text{for } m_2 = 0, 2, 4, \dots, 2k. \quad (92)$$

Hence, for $m_1 = 2k$, we have

$$\partial_x^{m_1} \partial_y^{m_2} \tilde{u}(\mathbf{0}) = 0, \quad \text{for } m_2 = 0, 1, 2, 3, \dots, 2p, \quad (93)$$

for any $k = 0, 1, \dots, p$. The result in (88b) follows using a symmetric argument. Therefore, we have odd symmetry on the Dirichlet sides near the corner and even symmetry at the D-D corner. The results for N-N and D-N corners follow using similar arguments.

□

5.3. Stability of LCBC approximations for the wave equation

We now consider the stability of an explicit modified equation (ME) time-stepping algorithm for the wave equation on a Cartesian grid using the LCBC approach at the boundary. For the present model problem in (24), the standard wave equation corresponds to the case $q = 2$ and $Q = c^2 \Delta$ for a wave speed $c > 0$, and the ME time-stepping schemes are given in (113), (114) and (115) for orders of accuracy $2p = 2, 4$ and 6 , respectively. In [42] it was shown that an ME scheme for the wave equation in one space dimension is stable at any order of accuracy, $2p = 2, 4, 6, \dots$, under the condition $c\Delta t/\Delta x < 1$, where Δt is the time-step. In two dimensions (or three dimensions), the time-step condition depends on whether selected terms are dropped to retain a stencil width of $2p+1$ or not. For example, at sixth-order, the term $\Delta_{4,h}^2 U_{\mathbf{i}}^n$ appears, and it has a term proportional to $\Delta x^4 (D_{+x} D_{-x})^4 U_{\mathbf{i}}^n$ which can be dropped (since it is also multiplied by Δt^2). If appropriate terms are dropped so that the stencil width of the ME scheme is $2p+1$, then the time-step restriction for two-dimensional problems is

$$c^2 \Delta t^2 \left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} \right) < 1, \quad (94)$$

for orders of accuracy $2p = 2, 4, 6$, as given by Theorem 5 discussed below. We call this version the *compact* ME scheme, and we conjecture that the condition in (94) holds at any even order $2p = 2, 4, 6, \dots$ (with a similar result holding for three-dimensional problems).

The compact ME scheme with LCBC conditions thus has some nice properties. It achieves high-order accuracy in space and time in a single step. In addition, the time-step restriction does not change as the order of accuracy increases, in contrast to some other high-order accurate schemes (e.g. explicit multi-step methods) where the stable time-step decreases significantly as the order of accuracy increases.

Theorem 5 (Stability of approximations for the wave equation). *The IBVP in (24) for the wave equation with $q = 2$ and $Q = c^2 \Delta$ discretized to orders $2p = 2, 4, 6$ with the compact ME time-stepping scheme and the LCBC method on a Cartesian grid with Dirichlet or Neumann boundary conditions is stable under the time-step restriction given in (94).*

Proof. Let the domain be $\Omega = [0, L_x] \times [0, L_y]$, i.e. a physical domain with lengths L_x and L_y . We consider the case of Dirichlet boundary conditions on the left and right faces and Neumann bound-

ary conditions on the top and bottom. The proof for other combinations of boundary conditions follow in a similar way. Let us look for normal mode solutions of the form

$$W_{\mathbf{i}}^n = A^n \kappa_x^i \kappa_y^j, \quad (95)$$

where A is an amplification factor, (κ_x, κ_y) are constants and $\mathbf{i} = (i, j)$. Since the LCBC approach leads to discrete boundary conditions that enforce even and odd symmetry, we can look for normal-mode solutions in space that satisfy these symmetry conditions. In this case we find that the normal modes are

$$W_{\mathbf{i}}^n = A_{\pm}^n \sin\left(\frac{\pi k_x}{L_x} x_i\right) \cos\left(\frac{\pi k_y}{L_y} y_j\right), \quad k_x = 1, 2, \dots, N_x - 1, \quad k_y = 0, 1, 2, \dots, N_y, \quad (96)$$

where A_{\pm} are two possible values for the amplification factor (see below). Any grid function $V_{\mathbf{i}}$ satisfying the boundary conditions can then be represented as a sum of normal modes,

$$V_{\mathbf{i}}^n = \sum_{k_x=1}^{N_x-1} \sum_{k_y=0}^{N_y} \hat{V}_{\mathbf{k}} \sin\left(\frac{\pi k_x}{L_x} x_i\right) \cos\left(\frac{\pi k_y}{L_y} y_j\right), \quad (97)$$

for some coefficients $\hat{V}_{\mathbf{k}}$, where $\mathbf{k} = (k_x, k_y)$. The general solution to the IBVP takes the form,

$$U_{\mathbf{i}}^n = \sum_{k_x=1}^{N_x-1} \sum_{k_y=0}^{N_y} \left(A_{+, \mathbf{k}}^n \hat{V}_{+, \mathbf{k}} + A_{-, \mathbf{k}}^n \hat{V}_{-, \mathbf{k}} \right) \sin\left(\frac{\pi k_x}{L_x} x_i\right) \cos\left(\frac{\pi k_y}{L_y} y_j\right), \quad (98)$$

where the coefficients $\hat{V}_{\pm, \mathbf{k}}$ are determined from the two initial conditions.

For stability we choose Δt so that $|A_{\pm, \mathbf{k}}| \leq 1$ for all valid k_x and k_y . It is straightforward to find the symbols of $D_{+x}D_{-x}$ and $D_{+y}D_{-y}$,

$$D_{+x}D_{-x} \sin\left(\frac{\pi k_x}{L_x} x_i\right) = -\hat{k}_x^2 \sin\left(\frac{\pi k_x}{L_x} x_i\right), \quad (99a)$$

$$D_{+y}D_{-y} \cos\left(\frac{\pi k_y}{L_y} y_j\right) = -\hat{k}_y^2 \cos\left(\frac{\pi k_y}{L_y} y_j\right), \quad (99b)$$

where

$$\hat{k}_x \stackrel{\text{def}}{=} \frac{\sin(\xi_x/2)}{\Delta x/2}, \quad \hat{k}_y \stackrel{\text{def}}{=} \frac{\sin(\xi_y/2)}{\Delta y/2}, \quad \xi_x \stackrel{\text{def}}{=} \frac{\pi k_x}{L_x} \Delta x, \quad \xi_y \stackrel{\text{def}}{=} \frac{\pi k_y}{L_y} \Delta y. \quad (100)$$

Substituting (95) into the ME time-stepping schemes for the different orders of accuracy, determined by p , leads to a quadratic equation for A ,

$$A^2 - 2b_p A + 1 = 0, \quad p = 1, 2, 3, \quad (101)$$

where b depends on the various parameters of the discretization. Stability requires $b_p \in \mathbb{R}$ and $|b_p| < 1$. Note that when $b_p = \pm 1$ there is a double root for A which leads to algebraic growth which we exclude.

For $p = 1$,

$$b_1 = 1 - 2(\hat{\lambda}_x^2 + \hat{\lambda}_y^2) \quad (102)$$

where

$$\hat{\lambda}_x \stackrel{\text{def}}{=} c\Delta t \frac{\hat{k}_x}{2}, \quad \hat{\lambda}_y \stackrel{\text{def}}{=} c\Delta t \frac{\hat{k}_y}{2}, \quad (103)$$

with

$$|\hat{\lambda}_x| \leq \frac{c\Delta t}{\Delta x}, \quad |\hat{\lambda}_y| \leq \frac{c\Delta t}{\Delta y}. \quad (104)$$

Note that $b_1 < 1$ is clearly satisfied, while the condition $b_1 > -1$ implies

$$\max_{\{k_x, k_y\}} (\hat{\lambda}_x^2 + \hat{\lambda}_y^2) < 1, \quad (105)$$

and this implies the time-step restriction in (94).

For $p = 2$,

$$b_2 = 1 - 2 \left(\hat{\lambda}_x^2 + \hat{\lambda}_y^2 + \frac{\Delta x^2}{12} \hat{\lambda}_x^2 \hat{k}_x^2 + \frac{\Delta y^2}{12} \hat{\lambda}_y^2 \hat{k}_y^2 \right) + \frac{2}{3} \left(\hat{\lambda}_x^2 + \hat{\lambda}_y^2 \right)^2. \quad (106)$$

From (101), we find $A_{\pm} = b_2 \pm \sqrt{b_2^2 - 1}$ which we express in terms of the four variables $(c\Delta t/\Delta x, c\Delta t/\Delta y, \Delta x \hat{k}_x, \Delta y \hat{k}_y)$. For each $(c\Delta t/\Delta x, c\Delta t/\Delta y)$, we define

$$A_{\max} = \max \left\{ \begin{array}{cc} \max_{\substack{-2 \leq \Delta x \hat{k}_x \leq 2 \\ -2 \leq \Delta y \hat{k}_y \leq 2}} |A_+|, & \max_{\substack{-2 \leq \Delta x \hat{k}_x \leq 2 \\ -2 \leq \Delta y \hat{k}_y \leq 2}} |A_-| \end{array} \right\}, \quad (107)$$

and find the region in the $(c\Delta t/\Delta x, c\Delta t/\Delta y)$ plane such that $A_{\max} \leq 1$. We run a similar experiment for $p = 3$, where b_3 takes the form

$$\begin{aligned} b_3 = & 1 - 2 \left[\hat{\lambda}_x^2 \left(1 + \frac{\Delta x^2}{12} \hat{k}_x^2 + \frac{\Delta x^4}{90} \hat{k}_x^4 \right) + \hat{\lambda}_y^2 \left(1 + \frac{\Delta y^2}{12} \hat{k}_y^2 + \frac{\Delta y^4}{90} \hat{k}_y^4 \right) \right] \\ & + \frac{2}{3} \left[\hat{\lambda}_x^4 \left(1 + \frac{\Delta x^2}{6} \hat{k}_x^2 \right) + \hat{\lambda}_y^4 \left(1 + \frac{\Delta y^2}{6} \hat{k}_y^2 \right) + 2\hat{\lambda}_x^2 \hat{\lambda}_y^2 \left(1 + \frac{\Delta x^2}{12} \hat{k}_x^2 \right) \left(1 + \frac{\Delta y^2}{12} \hat{k}_y^2 \right) \right] \\ & - \frac{4}{45} \left(\hat{\lambda}_x^2 + \hat{\lambda}_y^2 \right)^3. \end{aligned} \quad (108)$$

Figure 6 shows that the stability region, $A_{\max} \leq 1$, for both the fourth-order ($p = 2$) and sixth-order ($p = 3$) accurate time-stepping schemes. The stability region for both schemes is found to lie within the unit circle, and thus Δt satisfies the condition in (94) when $p = 2$ and 3.

□

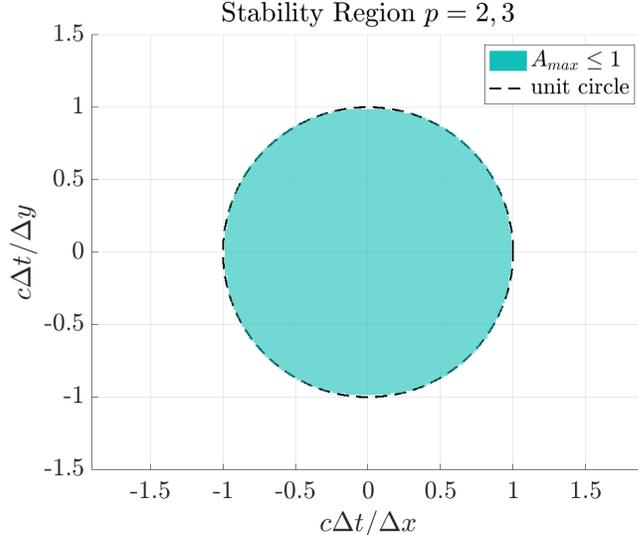


Figure 6: Stability region of the fourth-order and sixth-order accurate ME time-stepping schemes for the wave equation on a Cartesian grid using the LCBC approach.

In [Appendix B](#), we provide an analytical proof for the stability results observed in [Figure 6](#) when $p = 2$.

6. Numerical results

The aim of this section is to demonstrate the accuracy of the LCBC method by considering several examples involving second-order and higher-order accurate discretizations of the model problem in [\(24\)](#). A semi-discrete scheme for this problem is given in [\(34\)](#), and our first task is to describe suitable time-stepping schemes for the cases when $q = 1$ and 2 . We then utilize the method of manufactured solutions to verify the accuracy of the fully discrete schemes with the LCBC method for problems with constant and variable coefficients defined on rectangular and curvilinear domains. For the latter case, a mapping is used to bring the problem to a unit-square computational domain. Finally, we consider physically-motivated examples involving heat flow and wave propagation as further tests of the LCBC method.

We note that when solving the wave equation, which has no inherent dissipation, some upwind dissipation is often required to retain a stable time-stepping scheme for problems with variable coefficients or for discretizations on curvilinear or overset grids. For wave equations in second-order form an effective upwind dissipation is described in [\[43, 44\]](#). For the results presented here, however, no upwind dissipation has been added.

6.1. Fully discrete schemes

For the case of elliptic problems ($q = 0$), the discretization in [\(34\)](#) becomes

$$\begin{cases} Q_{d,h}U_{\mathbf{i}} = -f(\mathbf{x}_{\mathbf{i}}), & \mathbf{x}_{\mathbf{i}} \in \Omega_h, \\ \mathcal{B}_{d,h}U_{\mathbf{i}} = g(\mathbf{x}_{\mathbf{i}}), & \mathbf{x}_{\mathbf{i}} \in \partial\Omega_h, \end{cases} \quad (109)$$

where $Q_{d,h}$ is the d -th order accurate approximation of the differential operator Q given in [\(26\)](#) applied for points $\mathbf{x}_{\mathbf{i}}$ on the grid Ω_h . The discrete operator $\mathcal{B}_{d,h}$ is a d -th order accurate approx-

imation of the boundary operator \mathcal{B} given in (28) applied on the boundary $\partial\Omega_h$, and the LCBC method is used to obtain ghost values on the extended domain.

For the time-dependent cases ($q = 1, 2$), we require methods of time-stepping. For these cases, let $t^n = n\Delta t$, $n = 0, 1, \dots, N_t$, for a fixed time step $\Delta t = T/N_t$, and let $U_{\mathbf{i}}^n \approx u(\mathbf{x}_{\mathbf{i}}, t^n)$. For the parabolic case ($q = 1$), we consider an explicit forward-Euler (FE) time-stepping scheme as well as implicit Backward Differentiation Formula (BDF) time-stepping schemes. For the explicit FE scheme, we choose a stable time step Δt given by ⁶

$$\Delta t = \frac{\text{cfl}}{\sqrt{\left(\frac{\alpha_d}{2}\right)^2 + \beta_d^2}}, \quad (110)$$

where $\text{cfl} \leq 1$ is a safety factor, set to $\text{cfl} = 0.9$ for all calculations, and α_d and β_d are estimates for the real and imaginary parts of the time-stepping eigenvalue corresponding to $Q_{d,h}$ with frozen coefficient functions. These values are given by

$$\begin{aligned} \alpha_d &= r_d \left(\frac{c_{11,\max}}{\Delta x^2} + \frac{c_{22,\max}}{\Delta y^2} \right) + s_d \frac{|c_{12}|_{\max}}{\Delta x \Delta y} + |c_0|_{\max} \\ \beta_d &= q_d \left(\frac{|c_1|_{\max}}{\Delta x} + \frac{|c_2|_{\max}}{\Delta y} \right) \end{aligned} \quad (111)$$

where the maximum values of the coefficient functions (or their absolute values) are taken for $\mathbf{x} \in \Omega$, and the parameter values are taken to be $(r_d, s_d, q_d) = (4, 2, 1)$, $(16/3, 9/2, 3/2)$ and $(272/45, 131/261, 1199/756)$, for $d = 2, 4$ and 6 , respectively. Details of the stability analyses leading to these conditions can be found in [Appendix C.1](#). For the implicit BDF schemes, we take

$$\Delta t = \min\{\Delta x, \Delta y\}, \quad (112)$$

and match the temporal accuracy of the BDF scheme with the chosen spatial order of accuracy. The fully discrete schemes for the hyperbolic case ($q = 2$) are based on a modified equation (ME) approach as discussed, for example, in [5] for the case of Maxwell's equations in second-order form. For the model problem here, the baseline second-order accurate ($d = 2$) scheme is given by

$$\begin{cases} D_{+t}D_{-t}U_{\mathbf{i}}^n = Q_{2,h}U_{\mathbf{i}}^n + f(\mathbf{x}_{\mathbf{i}}, t^n), & \mathbf{x}_{\mathbf{i}} \in \Omega_h, \\ \mathcal{B}_{2,h}U_{\mathbf{i}}^{n+1} = g(\mathbf{x}_{\mathbf{i}}, t^{n+1}), & \mathbf{x}_{\mathbf{i}} \in \partial\Omega_h, \\ U_{\mathbf{i}}^0 = u_0(\mathbf{x}_{\mathbf{i}}), & \mathbf{x}_{\mathbf{i}} \in \bar{\Omega}_h, \\ U_{\mathbf{i}}^1 = u_0(\mathbf{x}_{\mathbf{i}}) + \Delta t u_1(\mathbf{x}_{\mathbf{i}}) + \frac{\Delta t^2}{2} [Qu_0(\mathbf{x}_{\mathbf{i}}) + f(\mathbf{x}_{\mathbf{i}}, 0)], & \mathbf{x}_{\mathbf{i}} \in \bar{\Omega}_h, \end{cases} \quad (113)$$

⁶This condition arises from approximating the region of absolute stability of the time-stepping scheme by an ellipse with semi-axes of lengths α_d and β_d .

while the fourth and sixth-order accurate schemes ($d = 4$ and 6) are

$$\left\{ \begin{array}{ll} D_{+t}D_{-t}U_{\mathbf{i}}^n = Q_{4,h}U_{\mathbf{i}}^n + f(\mathbf{x}_{\mathbf{i}}, t^n) + \frac{\Delta t^2}{12} [Q_{2,h}^2 U_{\mathbf{i}}^n + \Psi_2 f(\mathbf{x}_{\mathbf{i}}, t^n)], & \mathbf{x}_{\mathbf{i}} \in \Omega_h, \\ \mathcal{B}_{4,h}U_{\mathbf{i}}^{n+1} = g(\mathbf{x}_{\mathbf{i}}, t^{n+1}), & \mathbf{x}_{\mathbf{i}} \in \partial\Omega_h, \\ U_{\mathbf{i}}^0 = u_0(\mathbf{x}_{\mathbf{i}}), & \mathbf{x}_{\mathbf{i}} \in \bar{\Omega}_h, \\ U_{\mathbf{i}}^1 = u_0(\mathbf{x}_{\mathbf{i}}) + \Delta t u_1(\mathbf{x}_{\mathbf{i}}) \\ \quad + \frac{\Delta t^2}{2} [Qu_0(\mathbf{x}_{\mathbf{i}}) + f(\mathbf{x}_{\mathbf{i}}, 0)] + \frac{\Delta t^3}{6} [Qu_1(\mathbf{x}_{\mathbf{i}}) + \partial_t f(\mathbf{x}_{\mathbf{i}}, 0)] \\ \quad + \frac{\Delta t^4}{24} [Q^2 u_0(\mathbf{x}_{\mathbf{i}}) + \Psi_2 f(\mathbf{x}_{\mathbf{i}}, 0)], & \mathbf{x}_{\mathbf{i}} \in \bar{\Omega}_h, \end{array} \right. \quad (114)$$

and

$$\left\{ \begin{array}{ll} D_{+t}D_{-t}U_{\mathbf{i}}^n = Q_{6,h}U_{\mathbf{i}}^n + f(\mathbf{x}_{\mathbf{i}}, t^n) \\ \quad + \frac{\Delta t^2}{12} [\tilde{Q}_{4,h}^2 U_{\mathbf{i}}^n + \Psi_2 f(\mathbf{x}_{\mathbf{i}}, t^n)] + \frac{\Delta t^4}{360} [Q_{2,h}^3 U_{\mathbf{i}}^n + \Psi_3 f(\mathbf{x}_{\mathbf{i}}, t^n)], & \mathbf{x}_{\mathbf{i}} \in \Omega_h, \\ \mathcal{B}_{6,h}U_{\mathbf{i}}^{n+1} = g(\mathbf{x}_{\mathbf{i}}, t^{n+1}), & \mathbf{x}_{\mathbf{i}} \in \partial\Omega_h, \\ U_{\mathbf{i}}^0 = u_0(\mathbf{x}_{\mathbf{i}}), & \mathbf{x}_{\mathbf{i}} \in \bar{\Omega}_h, \\ U_{\mathbf{i}}^1 = u_0(\mathbf{x}_{\mathbf{i}}) + \Delta t u_1(\mathbf{x}_{\mathbf{i}}) \\ \quad + \frac{\Delta t^2}{2} [Qu_0(\mathbf{x}_{\mathbf{i}}) + f(\mathbf{x}_{\mathbf{i}}, 0)] + \frac{\Delta t^3}{6} [Qu_1(\mathbf{x}_{\mathbf{i}}) + \partial_t f(\mathbf{x}_{\mathbf{i}}, 0)] \\ \quad + \frac{\Delta t^4}{24} [Q^2 u_0(\mathbf{x}_{\mathbf{i}}) + \Psi_2 f(\mathbf{x}_{\mathbf{i}}, 0)] + \frac{\Delta t^5}{120} [Q^2 u_1(\mathbf{x}_{\mathbf{i}}) + \partial_t \Psi_2 f(\mathbf{x}_{\mathbf{i}}, 0)] \\ \quad + \frac{\Delta t^6}{720} [Q^3 u_0(\mathbf{x}_{\mathbf{i}}) + \Psi_3 f(\mathbf{x}_{\mathbf{i}}, 0)], & \mathbf{x}_{\mathbf{i}} \in \bar{\Omega}_h, \end{array} \right. \quad (115)$$

respectively, where $\tilde{Q}_{4,h}^2$ directly approximates the continuous operator Q^2 to fourth-order accuracy rather than forming the square of the discrete operator $Q_{4,h}$ which would have a stencil greater than $2p + 1$. This approach leads to the compact ME scheme noted previously in Section 5.3.

For all three schemes, an estimate for a stable time step is given by

$$\Delta t = \frac{\text{cfl}}{\sqrt{\frac{c_{11,\max}}{\Delta x^2} + \frac{c_{22,\max}}{\Delta y^2}}}, \quad (116)$$

where $\text{cfl} < 1$. We use $\text{cfl} = 0.9$ for all numerical examples. Details of the stability analysis leading to (116) can be found in Appendix C.2.

6.2. Convergence tests using the method of manufactured solutions

We first verify the accuracy of the fully discrete schemes with the LCBC method for the problem in (24) using manufactured solutions. In this approach, we set the forcing functions of the PDE and boundary conditions, and the initial conditions, to be

$$\left\{ \begin{array}{ll} f(\mathbf{x}, t) = \mathcal{L}_q u_e(\mathbf{x}, t) - Qu_e(\mathbf{x}, t), & \mathbf{x} \in \Omega, \quad t \in (0, T], \quad q = 0, 1, 2, \\ g(\mathbf{x}, t) = \mathcal{B}u_e(\mathbf{x}, t), & \mathbf{x} \in \partial\Omega, \quad t \in [0, T], \\ u_{\alpha-1}(\mathbf{x}) = \partial_t^{\alpha-1} u_e(\mathbf{x}, 0), & \mathbf{x} \in \bar{\Omega}, \quad \alpha = 1, \dots, q, \quad q = 1, 2, \end{array} \right. \quad (117)$$

so that $u = u_e(\mathbf{x}, t)$ is an exact solution of the problem in (24). For all tests, we select an exact solution of the form

$$u_e(\mathbf{x}, t) = \cos(k_x x) \cos(k_y y) \phi(t), \quad \mathbf{x} \in \bar{\Omega}, \quad t \in [0, T], \quad (118)$$

where $(k_x, k_y) = (2\pi, \sqrt{2}\pi)$ are wave numbers, and $\phi(t)$ is taken to be

$$\phi(t) = \begin{cases} 1, & \text{if } q = 0, \\ t + 1, & \text{if } q = 1 \text{ and explicit FE time-stepping,} \\ \cos(\pi t), & \text{otherwise.} \end{cases} \quad (119)$$

When $q \neq 0$, we choose the time-stepping schemes and $\phi(t)$ such that the temporal accuracy is at least equal to the spatial accuracy of the discretized problem. For example, the FE scheme, used for parabolic problems, is first-order accurate in time. For this case, we choose $\phi(t)$ to be a linear polynomial since the FE time-stepping scheme is exact for polynomials of degree one or less and thus the convergence rates of the fully discrete scheme reflect the spatial accuracy. On the other hand, the implicit BDF methods for problems with $q = 1$ and the explicit schemes for problems with $q = 2$ have temporal orders of accuracy balanced with the spatial accuracy. For such high-order accurate schemes, $\phi(t)$ may be any smooth function. Note that we use exact values taken from the manufactured solution to start the multi-step BDF schemes.

We now consider a sequence of three tests of the fully discrete schemes.

6.2.1. Test 1. Square domain and constant-coefficient functions

The first test considers the set-up discussed previously in Section 2.2. The domain for the problem is the unit square, $\Omega = (0, 1)^2$, with $Q = \Delta$, the usual Laplacian operator. The previous discussion focused on the wave equation, $q = 2$, but here we also consider the cases when $q = 0$ and 1. The boundary conditions on the left ($x = 0$) and right ($x = 1$) are taken to be Dirichlet conditions, while Neumann conditions are assumed on the bottom ($y = 0$) and top ($y = 1$), as given in (10). Numerical solutions are computed on uniform grids with $\Delta x = \Delta y = h = h_0/2^j$, $h_0 = 1/10$, $j = 0, 1, 2, 3$, and the maximum error between the numerical solution and the exact (manufactured) solution is determined for each value of h , and at a time $T = 1$ for the time-dependent cases, $q = 1, 2$.

Figure 7 shows maximum errors versus grid spacing for the elliptic case ($q = 0$), the parabolic case ($q = 1$) for both FE and BDF time-stepping, and the hyperbolic case ($q = 2$). Calculations are performed using schemes with order of accuracy $d = 2, 4$ and 6. In each case, the computed errors can be compared with reference lines indicating the design order of accuracy. We observe that the computed errors show the expected convergence rates.

6.2.2. Test 2. Square domain and variable-coefficient functions

The second test also uses a unit-square domain and boundary conditions as in the previous test, but now the coefficient functions in the operator Q defined in (26) are taken to be

$$\begin{aligned} c_{11}(\mathbf{x}) &= \frac{e^x}{2 + x + y}, & c_{12}(\mathbf{x}) &= xy e^{-x-y-1}, & c_{22}(\mathbf{x}) &= \frac{e^{x+y}}{2 + y}, \\ c_1(\mathbf{x}) &= 1 + xy, & c_2(\mathbf{x}) &= 1 + \frac{x}{1 + y}, & c_0(\mathbf{x}) &= 2 + x - y. \end{aligned}$$

Note that for this choice, we have

$$\max_{\mathbf{x} \in \Omega} \frac{|c_{12}(\mathbf{x})|}{\sqrt{c_{11}(\mathbf{x})c_{22}(\mathbf{x})}} \approx 0.3772,$$

so that the cross term is significant, but small enough so that the corner conditions are sufficiently well conditioned. Numerical solutions are computed on uniform grids with $\Delta x = \Delta y = h = h_0/2^j$,

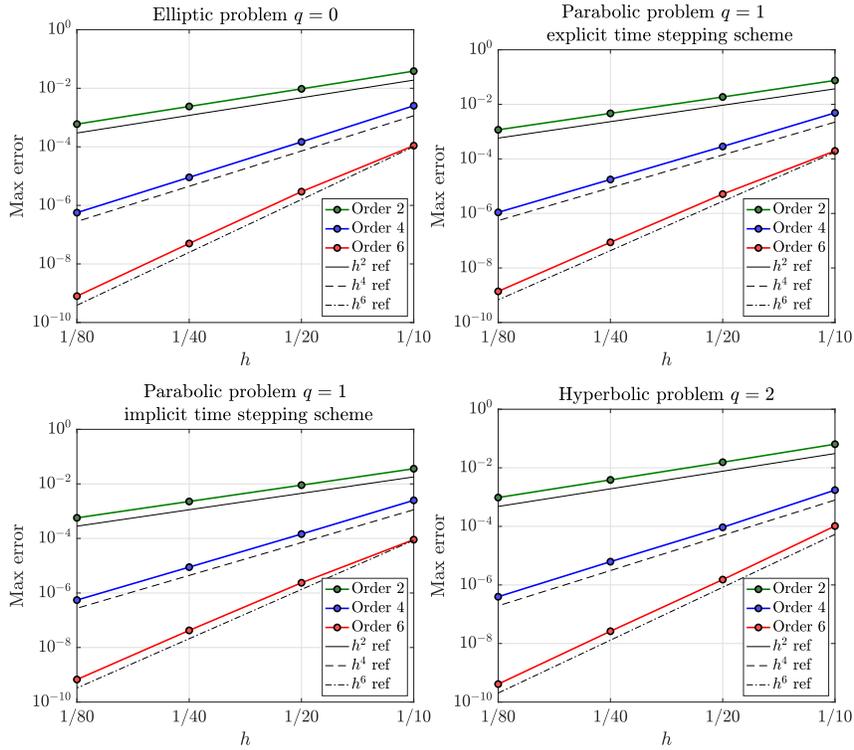


Figure 7: Maximum errors versus grid spacing h for Test 1 using manufactured solutions. Fully discrete schemes for $q = 0$ (upper left), $q = 1$ and FE time-stepping (upper right), $q = 1$ and BDF time-stepping (lower left), and $q = 2$ (lower right).

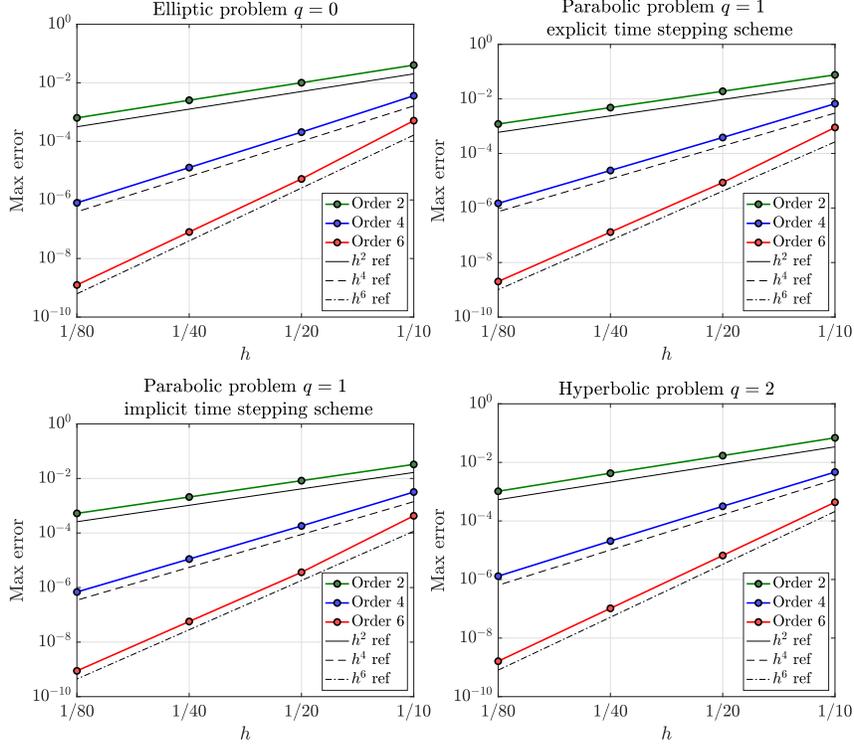


Figure 8: Maximum errors versus grid spacing h for Test 2 using manufactured solutions. Fully discrete schemes for $q = 0$ (upper left), $q = 1$ and FE time-stepping (upper right), $q = 1$ and BDF time-stepping (lower left), and $q = 2$ (lower right).

$h_0 = 1/10$, $j = 0, 1, 2, 3$, as before, and maximum errors are determined at $T = 1$ for the time-dependent cases.

Figure 8 shows maximum errors versus grid spacing for the elliptic case ($q = 0$), the parabolic case ($q = 1$) for both FE and BDF time-stepping, and the hyperbolic case ($q = 2$). Comparing the computed errors with the reference lines, we again observe the expected rate of convergence of the fully discrete schemes for $d = 2, 4$ and 6 .

6.2.3. Test 3. Curvilinear domain and constant-coefficient functions

As a final test using the method of manufactured solutions, we consider a problem defined on an annular domain, $\Omega^P = \{\mathbf{x} = (\rho \cos \theta, \rho \sin \theta) \mid \rho_1 < \rho < \rho_2, 0 < \theta < \theta_2\}$, where (ρ, θ) are polar coordinates. The parameters (ρ_1, ρ_2) define the inner and outer radii of the problem domain and θ_2 defines its angular extent. In the annular domain, the problem is taken to be

$$\begin{cases} \mathcal{L}_q u = \Delta u + f(\mathbf{x}, t), & \mathbf{x} \in \Omega^P, \quad q = 0, 1, 2, \\ u(\mathbf{x}, t) = g_\ell(\mathbf{x}, t), & \mathbf{x} \in \partial\Omega_\ell^P, \\ u(\mathbf{x}, t) = g_b(\mathbf{x}, t), & \mathbf{x} \in \partial\Omega_b^P, \\ \partial_n u(\mathbf{x}, t) = g_r(\mathbf{x}, t), & \mathbf{x} \in \partial\Omega_r^P, \\ \partial_n u(\mathbf{x}, t) = g_\tau(\mathbf{x}, t), & \mathbf{x} \in \partial\Omega_\tau^P, \\ \partial_t^{\alpha-1} u(\mathbf{x}, 0) = u_{\alpha-1}(\mathbf{x}), & \mathbf{x} \in \bar{\Omega}^P, \quad \alpha = 1, \dots, q, \quad q = 1, 2, \end{cases} \quad (120)$$

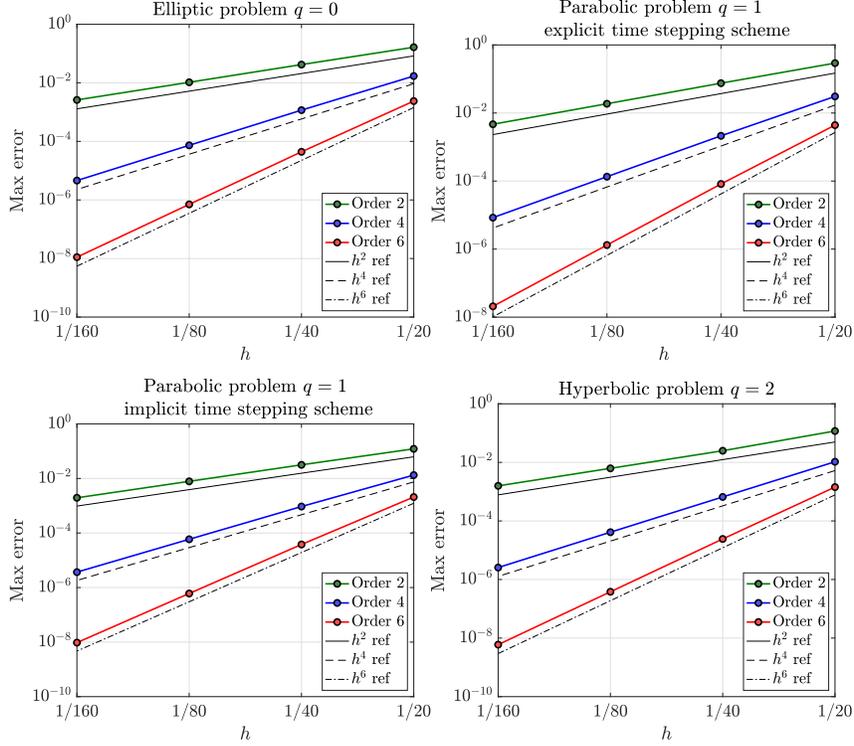


Figure 9: Maximum errors versus an approximate grid spacing h for Test 3 using manufactured solutions. Fully discrete schemes for $q = 0$ (upper left), $q = 1$ and FE time-stepping (upper right), $q = 1$ and BDF time-stepping (lower left), and $q = 2$ (lower right).

where $\partial\Omega_s^P$ for $s = \ell, r, b, \tau$ correspond to the boundaries at $\rho = \rho_1$, $\rho = \rho_2$, $\theta = 0$ and $\theta = \theta_2$ respectively. The problem is transformed from the “physical” domain $\mathbf{x} \in \Omega^P$ to the computational domain $\mathbf{r} = (r, s) \in \Omega = (0, 1)^2$ using the mapping

$$\mathbf{x} = \mathbf{G}(\mathbf{r}) = \begin{bmatrix} (\rho_1 + r(\rho_2 - \rho_1)) \cos(s\theta_2) \\ (\rho_1 + r(\rho_2 - \rho_1)) \sin(s\theta_2) \end{bmatrix}, \quad \mathbf{r} = (r, s) \in \Omega. \quad (121)$$

In terms of the mapped coordinates, the spatial operator of the PDE becomes

$$Qu = \partial_r^2 u + \frac{4}{\pi^2(1+r)^2} \partial_s^2 u + \frac{1}{(1+r)} \partial_r u, \quad (122)$$

for the choice $\rho_1 = 1$, $\rho_2 = 2$ and $\theta_2 = \pi/2$. The mapped problem is discretized in the computational domain using $\Delta r = \Delta r_0/2^j$, $\Delta r_0 = 1/10$, $j = 0, 1, 2, 3$, $\Delta s = \Delta r/2$, and $h = \min(\Delta r, \Delta s) = \Delta s$. Note that for each grid resolution j the corresponding grid spacings in physical space are approximately equal.

Figure 9 shows maximum errors at $T = 1$ versus grid spacing. The calculations are performed using various schemes for $q = 0, 1, 2$ with order of accuracy $d = 2, 4$ and 6 . In each case, we observe that the computed errors agree with the expected convergence rates.

6.3. Scattering of a plane wave from a cylinder

Moving on from tests involving the method of manufactured solutions, we now consider three physically-motivated problems. The first of these problems involves the scattering of an incident

plane wave from a cylinder of radius equal to one. The incident wave, traveling from left to right, is given by

$$u_{\text{inc}}(\mathbf{x}, t) = \cos[k(x - ct)], \quad (123)$$

where k is the wave number and c is the wave velocity. An exact solution of the wave equation for the scattered field $u(\mathbf{x}, t)$ from a hard cylinder with $u = -u_{\text{inc}}(\mathbf{x}, t)$ for $|\mathbf{x}| = 1$ is given by

$$u_e(\mathbf{x}, t) = -\text{Re} \left[e^{-ikct} \left(\frac{J_0(k)}{H_0^{(1)}(k)} H_0^{(1)}(k\rho) + 2 \sum_{n=1}^{\infty} i^n \frac{J_n(k)}{H_n^{(1)}(k)} H_n^{(1)}(k\rho) \cos(n\theta) \right) \right], \quad |\mathbf{x}| > 1, \quad (124)$$

where J_n and $H_n^{(1)}$ are Bessel and Hankel functions of the first kind, respectively, and (ρ, θ) are polar coordinates (see [45]). Our aim is to compute the scattered field for this problem numerically by solving a corresponding half-plane problem with a symmetry condition applied along the mid-line, $\theta = 0$ and π for $\rho > 1$. For this half-plane problem, we consider a finite domain given by

$$\Omega^P = \{\mathbf{x} = (\rho \cos \theta, \rho \sin \theta) \mid 1 < \rho < \rho_\infty, 0 < \theta < \pi\},$$

and let $u(\mathbf{x}, t)$ satisfy

$$\begin{cases} \partial_t^2 u = c^2 \Delta u, & \mathbf{x} \in \Omega^P, \quad t \in (0, T], \\ u(\mathbf{x}, t) = -u_{\text{inc}}(\mathbf{x}, t), & \rho = 1, \quad \theta \in [0, \pi], \\ u(\mathbf{x}, t) = u_e(\mathbf{x}, t), & \rho = \rho_\infty, \quad \theta \in [0, \pi], \\ \partial_n u(\mathbf{x}, t) = 0, & \theta = 0, \theta = \pi, \quad \rho > 1, \\ u(\mathbf{x}, 0) = u_e(\mathbf{x}, 0), & \mathbf{x} \in \bar{\Omega}^P, \\ \partial_t u(\mathbf{x}, 0) = \partial_t u_e(\mathbf{x}, 0), & \mathbf{x} \in \bar{\Omega}^P. \end{cases} \quad (125)$$

The exact solution of the problem in (125) is given by (124) for any choice of the outer radius $\rho_\infty > 1$ of the domain Ω^P .

We solve (125) numerically to a final time $T = 1$ for the case $k = 30$, $c = 1$ and $\rho_\infty = 2$. This is done by first mapping the problem to a unit-square computational domain using (121) with $\rho_1 = 1$, $\rho_2 = \rho_\infty$ and $\theta_2 = \pi$. On the computational domain $\mathbf{r} = (r, s) \in \Omega$, we use the explicit time-stepping schemes in (113), (114) and (115) corresponding to orders of accuracy equal to $d = 2, 4$ and 6 , respectively, in both time and space. The LCBC method is used to handle the boundary and corner conditions for each scheme. Numerical solutions are computed using grids with $\Delta r = \Delta r_0 / 2^j$, $\Delta r_0 = 1/20$, $j = 0, 1, 2, 3$, and with $\Delta s = \Delta r / 5$ so that the grid spacings in physical space are approximately equal. Figure 10 illustrates the results of the calculations. Maximum errors are computed for each time-stepping scheme and for each grid resolution, and the results are shown in the upper-left plot in the figure (using $h = \min\{\Delta r, \Delta s\} = \Delta s$). Here, we observe that the errors verify the expected convergence rates of the three schemes. The coarsest grid used for the calculations is shown in the lower-left plot for reference. The plots in the right column of the figure show the scattered field (top), the error in the scattered field (middle), and the total field (bottom), all at $T = 1$ computed using the sixth-order ($d = 6$) accurate scheme on the finest grid. We note that the error in the computed scattered field is smooth on the grid as expected based on the convergence behavior of the solutions.

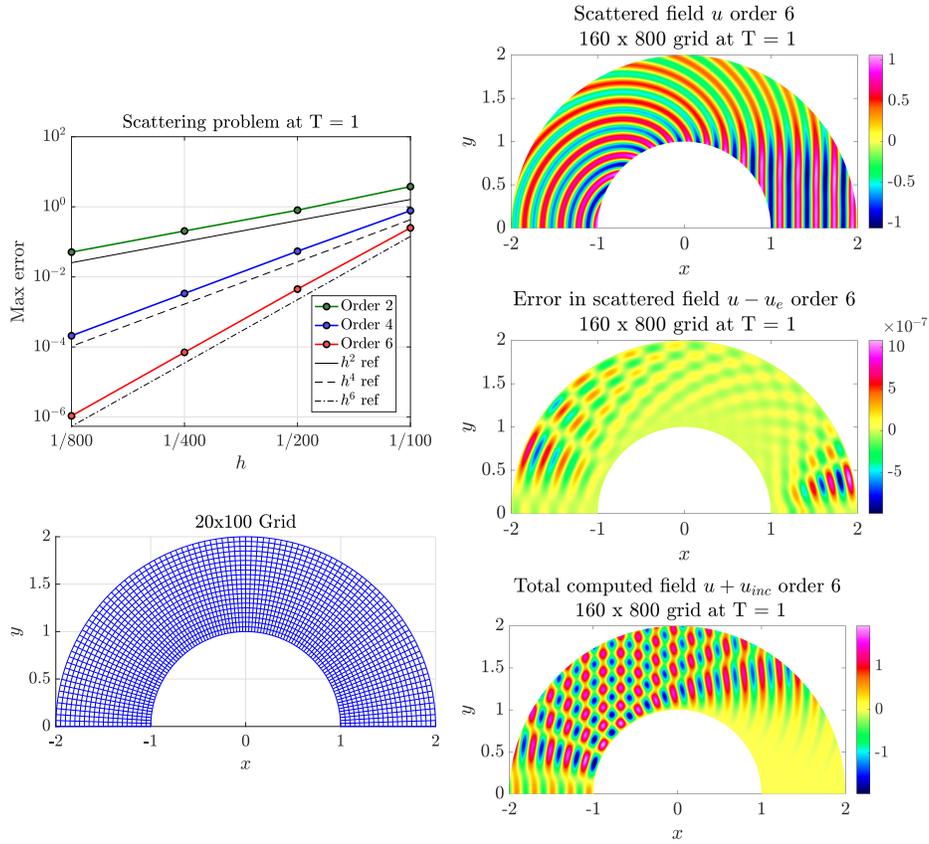


Figure 10: Plane-wave scattering from a cylinder. Maximum errors at $T = 1$ for solutions computed using the explicit time-stepping schemes with $d = 2, 4$ and 6 (upper left) and the coarsest grid for $h = 1/100$ (lower left). Right column shows the scattered field (top), error in the scattered field (middle) and the total field (bottom) at $T = 1$ computed using the sixth-order accurate scheme on the finest grid.

6.4. Heat flow in a wavy channel

For the next case, we consider a heat flow problem in channel domain denoted by $\mathbf{x} \in \Omega^P$. Assuming the temperature $u(\mathbf{x}, t)$ is specified on the boundary of the domain, the problem is given by

$$\begin{cases} u_t = \mathcal{D}\Delta u - \mathbf{v} \cdot \nabla u + \gamma u, & \mathbf{x} \in \Omega^P, \quad t \in (0, T], \\ u(\mathbf{x}, t) = g(\mathbf{x}, t), & \mathbf{x} \in \partial\Omega^P, \\ u(\mathbf{x}, 0) = u_0(\mathbf{x}), & \mathbf{x} \in \bar{\Omega}^P, \end{cases} \quad (126)$$

where \mathcal{D} is a diffusivity, \mathbf{v} is a convection velocity and γ is a reaction rate, all taken to be constants, and $u_0(\mathbf{x})$ is the initial temperature. An exact solution for this problem can be constructed by first considering the free-space solution of the PDE given by

$$u_e(\mathbf{x}, t) = \frac{e^{\gamma t}}{4\pi\mathcal{D}t} \iint_{\mathbb{R}^2} u_0(\mathbf{x}') \exp\left(\frac{-|\mathbf{x} - \mathbf{x}' - \mathbf{v}t|^2}{4\mathcal{D}t}\right) d\mathbf{x}'. \quad (127)$$

Assuming a Gaussian initial condition of the form

$$u_0(\mathbf{x}) = \exp(-\sigma|\mathbf{x}|^2), \quad \sigma = \text{constant} > 0, \quad (128)$$

the solution in (127) reduces to

$$u_e(\mathbf{x}, t) = \frac{e^{\gamma t}}{1 + 4\sigma\mathcal{D}t} \exp\left(\frac{-\sigma|\mathbf{x} - \mathbf{v}t|^2}{1 + 4\sigma\mathcal{D}t}\right). \quad (129)$$

We now set the Dirichlet boundary forcing $g(\mathbf{x}, t)$ in (126) to equal the free-space solution so that (129) becomes the exact solution of the heat flow problem for any channel domain Ω^P .

We let $\mathbf{x} \in \Omega^P$ be a wavy channel domain given by

$$\left. \begin{aligned} x &= G_1(r, s) = (x_2 - x_1)r + x_1 \\ y &= G_2(r, s) = (y_2 - y_1)s + y_1 + h(r) \end{aligned} \right\} \quad (r, s) \in \Omega = [0, 1]^2, \quad (130)$$

where $(x_1, x_2) = (y_1, y_2) = (-1, 1)$. The shape of the bottom and top walls of the wavy channel is specified by

$$h(r) = A_c \sin(2\pi r),$$

where A_c is an amplitude. The heat flow problem in (126) is mapped to the unit-square Ω , and numerical solutions are computed using the implicit BDF schemes with $d = 2, 4, 6$ for the parameter values $\mathcal{D} = 0.2$, $\mathbf{v} = (0.5, 0.3)$, $\gamma = 1$, $\sigma = 6$ and $A_c = 0.1$. Note that the mapping in (130) is not orthogonal unlike the annular mappings considered in previous problems.

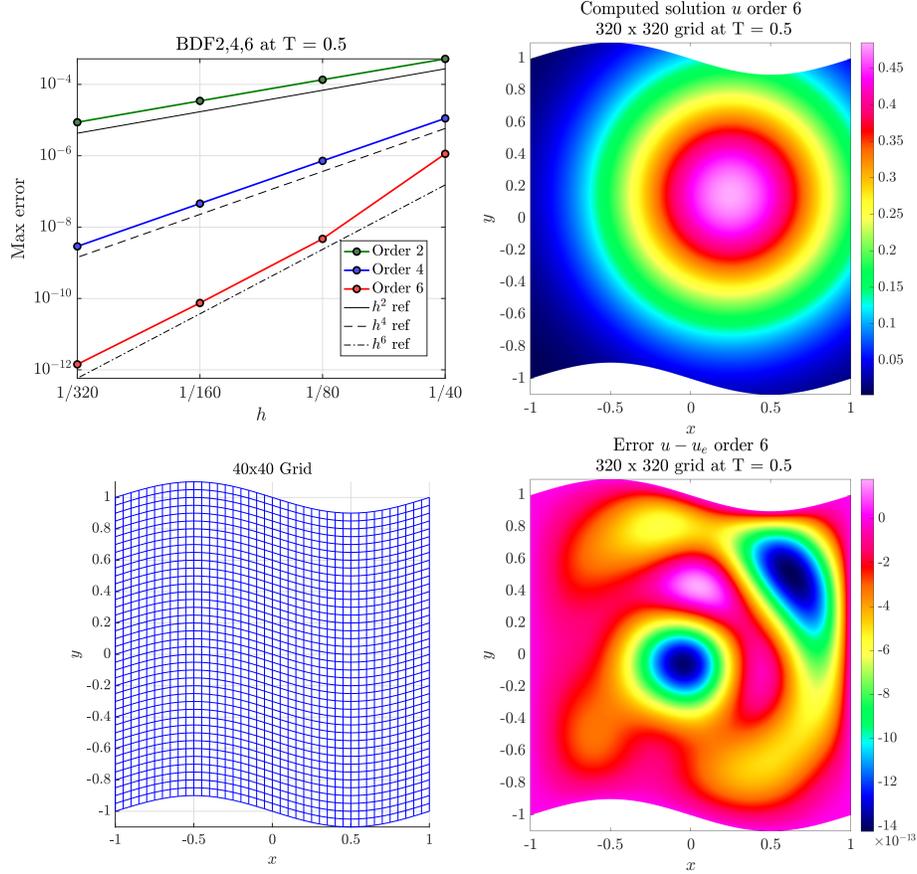


Figure 11: Heat flow in a wavy channel. Maximum errors at $T = 0.5$ for solutions computed using the BDF time-stepping schemes with $d = 2, 4$ and 6 (upper left) and the coarsest grid for $h = 1/40$ (lower left). Right column shows the temperature (top) and its error (bottom) at $T = 0.5$ computed using the sixth-order accurate scheme on the finest grid.

Figure 11 shows the results of the calculations at the final time $T = 0.5$. The upper-left plot shows the maximum error in the solutions with $d = 2, 4, 6$ for grids with $h = \Delta r = \Delta s = h_0/2^j$, $h_0 = 1/40$, $j = 0, 1, 2, 3$. These errors verify the expected convergence rates. The lower-left plot illustrates the grid used for the calculations, here for $h = 1/40$, and the plots in the right column show the sixth-order accurate solution and its error at $T = 0.5$, both for the finest grid with $h = 1/320$. In particular, we observe that the error is smooth throughout the domain, including the boundaries and corners.

6.5. Pulse propagation in curved channel

As a final test, we consider a wave propagation problem for $u(\mathbf{x}, t)$ in curved channel with closed ends. The domain of the channel Ω^P is taken to be the quarter annulus described previously by the mapping in (121) with $\rho_1 = 1$, $\rho_2 = 2$ and $\theta_2 = \pi/2$. We assume that $u = \partial_t u = 0$ initially, and that a pulse is generated in the channel by setting $u(\mathbf{x}, t) = g_p(\mathbf{x}, t)$ on the vertical boundary $x = 0$, $y \in [1, 2]$ denoted by $\mathbf{x} \in \partial\Omega_{\text{pulse}}^P$. The form of this boundary forcing is taken to be

$$g_p(\mathbf{x}, t) = e^{-\beta(t-t_0)^2} \sin(\omega t) \left[\frac{1}{2} \cos \left(2\pi \left(y - \frac{3}{2} \right) \right) - \frac{1}{2} \right]^{z+1}, \quad (131)$$

where β , t_0 and ω are constants, and $z = d/2$ for a calculation using a scheme with order of accuracy equal to d . Assuming zero Neumann conditions on the remaining boundaries, the problem to solve is

$$\begin{cases} \partial_t^2 u = c^2 \Delta u, & \mathbf{x} \in \Omega^P, \quad t \in (0, T], \\ u(\mathbf{x}, t) = g_p(\mathbf{x}, t), & \mathbf{x} \in \partial\Omega_{\text{pulse}}^P, \\ \partial_n u(\mathbf{x}, t) = 0, & \mathbf{x} \in \partial\Omega^P \setminus \partial\Omega_{\text{pulse}}^P, \\ u(\mathbf{x}, 0) = \partial_t u(\mathbf{x}, 0) = 0, & \mathbf{x} \in \overline{\Omega}^P. \end{cases} \quad (132)$$

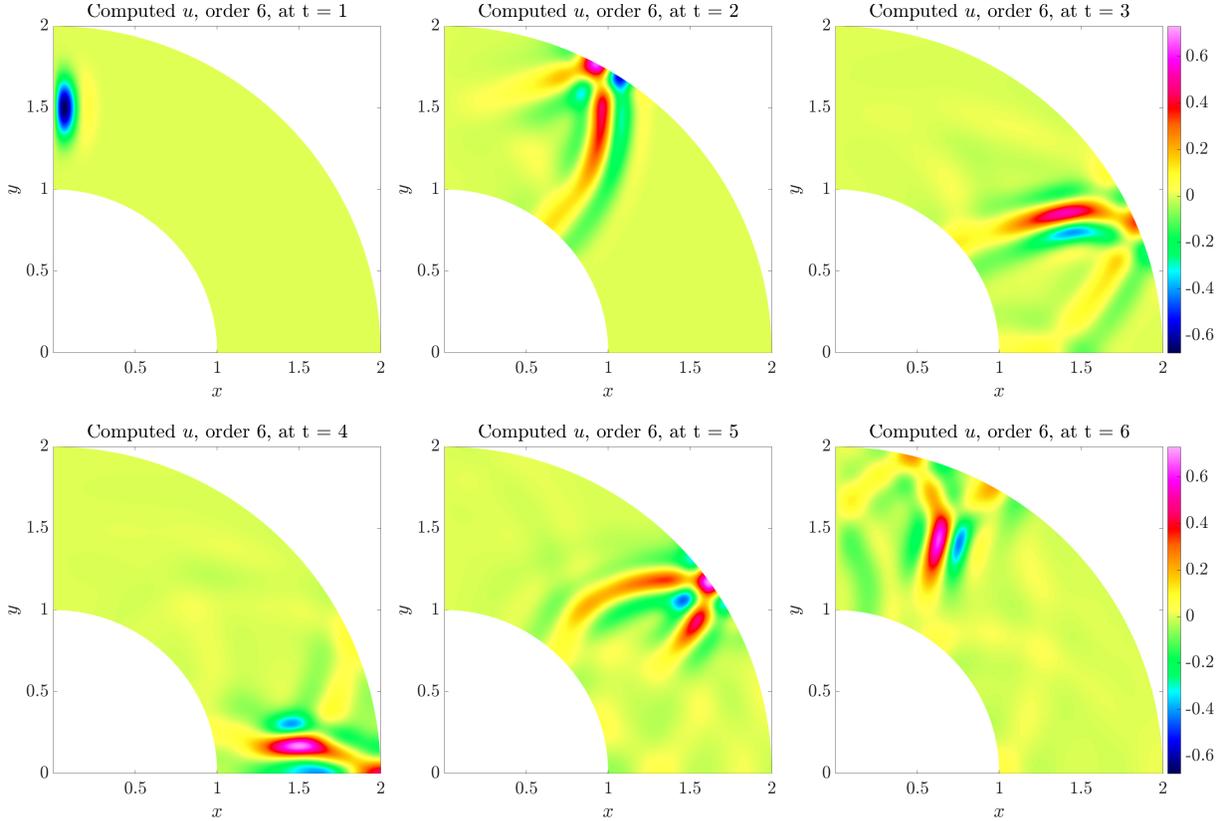


Figure 12: Evolution of a pulse in a curved channel with closed ends. Solutions at times $t = 1, 2, \dots, 6$ are computed using the sixth-order accurate scheme in (115) for a grid with $h = 1/640$, grid resolution $j = 3$.

The pulse propagation problem in (132) is solved numerically for $c = 1$, $\beta = 50$, $t_0 = 1$ and $\omega = 6\pi$. This is done by mapping the problem to the unit-square Ω , and then solutions are computed using the second, fourth and sixth-order accurate time-stepping schemes given by (113), (114) and (115), respectively. The grid spacings in the unit-square coordinates (r, s) are taken to be $\Delta r = \Delta r_0/2^j$, $\Delta r_0 = 1/40$, $j = 0, 1, 2, 3$, $\Delta s = \Delta r/2$, and $h = \min(\Delta r, \Delta s) = \Delta s$. Figure 12 shows the evolution of the pulse for the times $t = 1, 2, \dots, 6$ as determined by the sixth-order accurate scheme on the finest grid. Here we observe the forward propagation of the pulse at an early time ($t = 1$) and the subsequent interaction with the curved walls of the channel at later times ($t = 2, 3$). By $t = 4$ the pulse has reached the lower flat wall of the channel, and then reflects backwards into the channel at the final times ($t = 5, 6$).

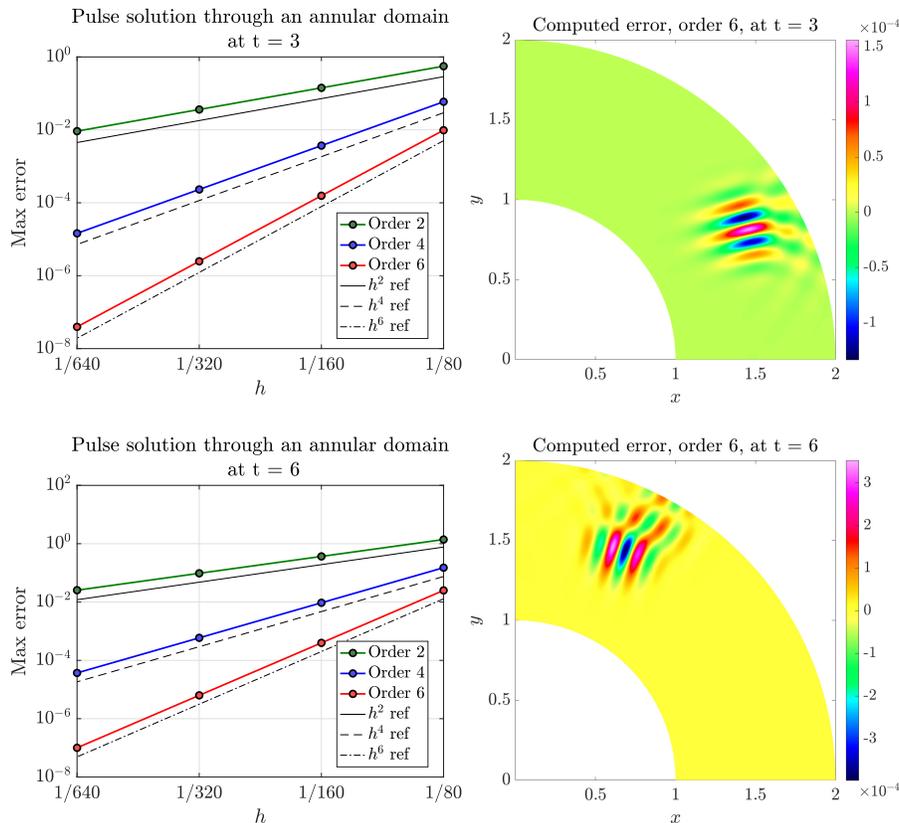


Figure 13: Evolution of a pulse in a curved channel with closed ends. Behavior of the computed maximum-norm errors at times $t = 3$ (top) and $t = 6$ (bottom). Estimated errors versus h (left) and errors in the sixth-order accurate solution for the grid with $h = 1/160$, grid resolution $j = 1$.

Figure 13 shows the behavior of the computed error in the solutions at $t = 3$ and 6. Since an exact solution for the problem is not available, we compute the error approximately using a Richardson extrapolation approach [9, 46]. Let $U_{\mathbf{i}}^{(j)}$ denote the solution computed on a grid with mesh spacing h_j at a time t^n . The error is computed at a fixed time $T = t^n$, and so the dependence on t^n is suppressed for notational convenience. Assuming convergent schemes, we have

$$U_{\mathbf{i}}^{(j)} \approx u_e(\mathbf{x}_{\mathbf{i}}^{(j)}) + C_{\mathbf{i}}^{(j)} h_j^\sigma,$$

where $u_e(\mathbf{x})$ is the exact solution (at $t^n = T$), σ is the rate of convergence (for $h \rightarrow 0$), and $C_{\mathbf{i}}^{(j)}$ is a grid function (whose values are independent of h_j and σ). Subtracting solutions at grid resolutions j and $j + 1$, assuming $h_{j+1} = \frac{1}{2}h_j$, gives

$$U_{\mathbf{i}}^{(j)} - R_j U_{\mathbf{i}}^{(j+1)} \approx C_{\mathbf{i}}^{(j)} (1 - 2^{-\sigma}) h_j^\sigma,$$

where R_j is an operator that restricts a grid function at resolution $j + 1$ to one at j . Taking the maximum norm of both sides gives

$$\Delta U_j \stackrel{\text{def}}{=} \|U_{\mathbf{i}}^{(j)} - R_j U_{\mathbf{i}}^{(j+1)}\|_\infty \approx C(1 - 2^{-\sigma}) h_j^\sigma,$$

where C is a positive constant. A similar expression involving solutions at grid resolutions $j + 1$

and $j + 2$ can be used to give

$$\sigma \approx \log_2 \left(\frac{\Delta U_j}{\Delta U_{j+1}} \right), \quad C \approx \frac{\Delta U_j}{(1 - 2^{-\sigma})h_j^\sigma}. \quad (133)$$

The maximum error in the numerical solution at grid resolution j is then estimated as

$$\|U_{\mathbf{i}}^{(j)} - u_e(\mathbf{x}_{\mathbf{i}}^{(j)})\|_\infty \approx Ch_j^\sigma, \quad (134)$$

where the convergence rate σ and constant C are given in (133). The convergence plots at $t^n = 3$ and $t^n = 6$ in the left column of Figure 13 are given by (134) using values for σ and C obtained from solutions at grid resolutions $j = 1, 2, 3$. The two plots show convergence rates that agree with the expected order of accuracy of the explicit time-stepping schemes for $d = 2, 4$ and 6 . The plots in the right column show representative behaviors of the computed error.

7. Conclusion

We have described a new approach for developing numerical approximations to boundary conditions for high-order accurate finite difference approximations. In contrast to traditional one-sided approximations, this new local compatibility boundary condition (LCBC) approach results in centered approximations that are generally more accurate and stable than one-sided approximations. The LCBC approach uses a local polynomial representation of the solution whose coefficients are defined using the given boundary conditions and additional compatibility boundary conditions derived from the governing equations, as well as using (known) solution data at grid points in the interior and on the boundary. The LCBC approach uses the boundary conditions and CBCs defined at continuous level, which enables automatic construction of the local polynomial, an important consideration when developing arbitrarily high-order accurate schemes. Values of the discrete solution at ghost points are then determined by evaluation of the local polynomial. Algorithms have been given for computing the local polynomial as well as for forming the discrete stencil approximations for the ghost point values which can be used to efficiently assign the ghost point values. The LCBC approach at corners has also been described. In this case the boundary conditions and compatibility conditions from the two sides adjacent to the corner are used to define the local polynomial representation.

The LCBC approach was developed for general boundary-value problems (BVPs) and initial-boundary-value problems (IBVPs) for second-order scalar PDEs, including elliptic, parabolic and hyperbolic equations. The approach was formulated for two-dimensional domains that are described by a smooth mapping from the reference parameter space unit square. Properties of the LCBC approximations were analyzed. Conditions for the solvability of the LCBC equations were derived. It was also shown that the LCBC approach results in even and odd symmetry conditions in some important special cases. The stability of high-order approximations to the wave equation were also analyzed.

Numerical results were presented that demonstrate the accuracy and stability of the approach. These numerical tests included manufactured solutions, problems with known solutions, and problems where the errors were estimated using a grid refinement self-convergence approach. To test the LCBC conditions, high-order accurate schemes were presented for discretizing the BVPs and IBVPs. For time-dependent problems, formulae for choosing the maximal stable time-step were given. Results were presented for orders of accuracy equal to two, four and six. In all cases the numerical solutions converged at close to the expected rates.

In future work we will consider extensions of the LCBC approach to BVPs and IBVPs in three dimensions, problems with interfaces, problems involving vector PDEs such as those that appear in electromagnetics or elasticity, and nonlinear problems.

Appendix A. LCBC corner conditions

Appendix A.1. Neumann-Neumann Corner

Consider the bottom-left corner, $\tilde{x} = (0, 0)$, where two Neumann boundaries meet. The physical (primary) boundary are taken to be

$$\partial_x u(\mathbf{x}, t) = g_\ell(y, t), \quad \mathbf{x} \in \partial\Omega_\ell, \quad (\text{A.1a})$$

$$\partial_y u(\mathbf{x}, t) = g_b(x, t), \quad \mathbf{x} \in \partial\Omega_b, \quad (\text{A.1b})$$

for some fixed time t . We start by specifying the interpolating polynomial $\tilde{u}(\mathbf{x})$ at known interior data given by

$$\tilde{u}(\hat{i}\Delta x, \hat{j}\Delta y) = U_{\hat{i}, \hat{j}}(t), \quad \hat{i} = 1, \dots, p, \quad \hat{j} = 1, \dots, p. \quad (\text{A.2a})$$

Next, we apply tangential derivatives of the primary boundary conditions and compatibility conditions given by

$$\left. \begin{aligned} \partial_y^\mu \partial_x Q^\nu \tilde{u}(0, 0) &= \partial_y^\mu S_{\ell, \nu}(0, t) \\ \partial_x^\mu \partial_y Q^\nu \tilde{u}(0, 0) &= \partial_x^\mu S_{b, \nu}(0, t) \end{aligned} \right\} \quad \nu = 0, \dots, p-1, \quad \mu \in \mathcal{M}_\nu, \quad (\text{A.2b})$$

respectively, where $S_{\ell, \nu}(y, t)$ is defined in (60) and

$$S_{b, \nu}(x, t) \stackrel{\text{def}}{=} \begin{cases} -\partial_x Q^{\nu-1} f(x, 0), & q = 0, \\ \partial_x^{q\nu} g_b(x, t) - \partial_x \Psi_\nu f(x, 0, t), & q = 1, 2. \end{cases} \quad (\text{A.3})$$

The sets \mathcal{M}_ν , $\nu = 0, \dots, p-1$, chosen to eliminate redundant constraints, are given by

$$\mathcal{M}_\nu = \begin{cases} 0, 1, 2, 3, \dots, 2p-1, 2p, & \text{if } \nu = 0, \text{ with an average for } \mu = 1, \\ 0, 2, 3, 4, \dots, 2p-1, 2p, & \text{if } \nu = 1, \text{ with an average for } \mu = 3, \\ 0, 2, 4, 5, \dots, 2p-1, 2p, & \text{if } \nu = 2, \text{ with an average for } \mu = 5, \\ \vdots & \vdots \\ 0, 2, 4, 6, \dots, 2p-1, 2p, & \text{if } \nu = p-1, \text{ with an average for } \mu = 2p-1. \end{cases} \quad (\text{A.4})$$

Appendix A.2. Dirichlet-Neumann Corner

Consider the bottom-left corner, $\tilde{x} = (0, 0)$, where a Dirichlet boundary on the left meets a Neumann boundary at the bottom. The physical (primary) boundary are taken to be

$$u(\mathbf{x}, t) = g_\ell(y, t), \quad \mathbf{x} \in \partial\Omega_\ell, \quad (\text{A.5a})$$

$$\partial_y u(\mathbf{x}, t) = g_b(x, t), \quad \mathbf{x} \in \partial\Omega_b, \quad (\text{A.5b})$$

for some fixed time t . We specify the interpolating polynomial $\tilde{u}(\mathbf{x})$ at known interior data:

$$\tilde{u}(\hat{i}\Delta x, \hat{j}\Delta y) = U_{\hat{i}, \hat{j}}(t), \quad \hat{i} = 1, \dots, p, \quad \hat{j} = 1, \dots, p. \quad (\text{A.6a})$$

Next, we apply tangential derivatives of the primary boundary conditions and compatibility conditions given by

$$\left. \begin{aligned} \partial_y^{\mu_1} Q^\nu \tilde{u}(0, 0) &= \partial_y^{\mu_1} R_{\ell, \nu}(0, t) \\ \partial_x^{\mu_2} \partial_y Q^\nu \tilde{u}(0, 0) &= \partial_x^{\mu_2} S_{b, \nu}(0, t) \end{aligned} \right\} \quad \nu = 0, \dots, p-1, \quad (\mu_1, \mu_2) \in \mathcal{M}_\nu, \quad (\text{A.6b})$$

and

$$\left. \partial_y^{\mu_1} Q^\nu \tilde{u}(0, 0) = \partial_y^{\mu_1} R_{\ell, \nu}(0, t) \right\} \quad \nu = p, \quad \mu_1 = 0, 2, 4, \dots, 2(p-1), 2p, \quad (\text{A.6c})$$

respectively, where $R_{\ell, \nu}(y, t)$ is defined in (44) and $S_{b, \nu}(x, t)$ is defined in (A.3).

The sets \mathcal{M}_ν , $\nu = 0, \dots, p-1$, chosen to eliminate redundant constraints, are given by

$$\mathcal{M}_\nu = \begin{cases} (\mu_1, \mu_2) \in \mathcal{M}_0^{(1)} \times \mathcal{M}_0^{(2)}, & \text{if } \nu = 0, \text{ with an average for } (\mu_1, \mu_2) = (1, 0), \\ (\mu_1, \mu_2) \in \mathcal{M}_1^{(1)} \times \mathcal{M}_1^{(2)}, & \text{if } \nu = 1, \text{ with an average for } (\mu_1, \mu_2) = (3, 2), \\ (\mu_1, \mu_2) \in \mathcal{M}_2^{(1)} \times \mathcal{M}_2^{(2)}, & \text{if } \nu = 2, \text{ with an average for } (\mu_1, \mu_2) = (5, 4), \\ \vdots & \vdots \\ (\mu_1, \mu_2) \in \mathcal{M}_{p-1}^{(1)} \times \mathcal{M}_{p-1}^{(2)}, & \text{if } \nu = p-1, \text{ with an average for } (\mu_1, \mu_2) = (2p-1, 2p-2), \end{cases} \quad (\text{A.7})$$

where $\mathcal{M}_\nu^{(1)}$, $\mathcal{M}_\nu^{(2)}$, $\nu = 0, \dots, p-1$ are defined as

$$\mathcal{M}_\nu^{(1)} = \begin{cases} 0, 1, 2, 3, \dots, 2p-1, 2p, & \text{if } \nu = 0, \\ 0, 2, 3, 4, \dots, 2p-1, 2p, & \text{if } \nu = 1, \\ 0, 2, 4, 5, \dots, 2p-1, 2p, & \text{if } \nu = 2, \\ \vdots & \vdots \\ 0, 2, 4, 6, \dots, 2p-1, 2p, & \text{if } \nu = p-1, \end{cases} \quad (\text{A.8})$$

and

$$\mathcal{M}_\nu^{(2)} = \begin{cases} 0, 1, 2, 3, \dots, 2p-2, 2p-1, 2p, & \text{if } \nu = 0, \\ 1, 2, 3, 4, \dots, 2p-2, 2p-1, 2p, & \text{if } \nu = 1, \\ 1, 3, 4, 5, \dots, 2p-2, 2p-1, 2p, & \text{if } \nu = 2, \\ \vdots & \vdots \\ 1, 3, 5, \dots, 2p-2, 2p-1, 2p, & \text{if } \nu = p-1. \end{cases} \quad (\text{A.9})$$

Appendix B. Analytical proof of the stability of fourth-order LCBC approximations of the wave equation

In this appendix, we provide an analytical proof of the stability of the fourth-order accurate LCBC approximations of the wave equation. We have already demonstrated the stability of such approximations in section 5.3 via a numerical approach. In the proof of theorem 5, we arrived at the quadratic equation (101) for the amplification factor A which involved a parameter b_2 defined in (106). We have also defined the parameters \hat{k}_x , \hat{k}_y in (100) and required that $|b_2| < 1$ for all values of \hat{k}_x , \hat{k}_y for the approximations to be stable.

We now set $\Lambda_x = c\Delta t/\Delta x$, $\Lambda_y = c\Delta t/\Delta y$, $X = (\Delta x \hat{k}_x)^2$ and $Y = (\Delta y \hat{k}_y)^2$, $X, Y \in [0, 4]$. We

write b_2 as

$$b_2 = 1 - \frac{1}{2} \left[\Lambda_x^2 X \left(1 + \frac{X}{12} \right) + \Lambda_y^2 Y \left(1 + \frac{Y}{12} \right) \right] + \frac{1}{24} (\Lambda_x^2 X + \Lambda_y^2 Y)^2,$$

and define the variable $z = \Lambda_x^2 + \Lambda_y^2$.

From the numerical results in figure 6, we observe that

$$\text{Given } \Delta t \text{ such that } z < 1, |b_2| < 1 \text{ for all } X, Y \in [0, 4].$$

Proof. Assuming $z < 1$, we aim to show $|b_2| < 1$ or $b_2^2 < 1$. For this, we write $b_2^2 - 1 = F(F + 2)$ where

$$\begin{aligned} F(X, Y) &= -\frac{1}{2} \left[\Lambda_x^2 X \left(1 + \frac{X}{12} \right) + \Lambda_y^2 Y \left(1 + \frac{Y}{12} \right) \right] + \frac{1}{24} (\Lambda_x^2 X + \Lambda_y^2 Y)^2, \\ &= \underbrace{\frac{1}{24} \Lambda_x^2 (\Lambda_x^2 - 1) X^2}_{\mathcal{A}} + \underbrace{\frac{1}{12} \Lambda_x^2 \Lambda_y^2 XY}_{\mathcal{B}} + \underbrace{\frac{1}{24} \Lambda_y^2 (\Lambda_y^2 - 1) Y^2}_{\mathcal{C}} - \frac{1}{2} \Lambda_x^2 X - \frac{1}{2} \Lambda_y^2 Y. \end{aligned} \quad (\text{B.1})$$

For stability we require $-2 < F(X, Y) < 0$ for all values of X, Y . The function $F(X, Y)$ in (B.1) is a bi-variate quadratic function in X and Y . We find the discriminant of F to be

$$\begin{aligned} \mathcal{B}^2 - 4\mathcal{A}\mathcal{C} &= \frac{1}{144} \Lambda_x^4 \Lambda_y^4 - \frac{1}{144} \Lambda_x^2 \Lambda_y^2 (\Lambda_x^2 - 1)(\Lambda_y^2 - 1), \\ &= \frac{1}{144} \Lambda_x^2 \Lambda_y^2 [\Lambda_x^2 \Lambda_y^2 - (\Lambda_x^2 \Lambda_y^2 - \Lambda_x^2 - \Lambda_y^2 + 1)], \\ &= \frac{1}{144} \Lambda_x^2 \Lambda_y^2 (z - 1) < 0, \end{aligned}$$

which shows that F is an elliptic paraboloid. Also, with $\Lambda_x^2, \Lambda_y^2 < z < 1$, we find that $\mathcal{A}, \mathcal{B} < 1$ and F is concave downwards. The maximum of F occurs at $X = Y = 0$ where $F(0, 0) = 0$. The minimum of F occurs at $X = Y = 4$ where

$$f(z) \stackrel{\text{def}}{=} F(4, 4) = -\frac{8}{3}z + \frac{2}{3}z^2.$$

The function $f(z)$ is a monotonically decreasing function with increasing $z \in (0, 1)$. We find $\lim_{z \rightarrow 1} f(z) = -2$ which is within the bound needed for stability. Hence, when $z < 1$, $b_2^2 < 1$ and the fourth-order accurate LCBC approximation of the wave equation is stable. \square

Appendix C. Time-step restrictions

In this section we provide details of the determination of the time-step for the schemes used in Section 6. For variable coefficient problems we freeze coefficients and consider the constant coefficient operator

$$Q = c_{11} \partial_x^2 + 2c_{12} \partial_x \partial_y + c_{22} \partial_y^2 + c_1 \partial_x + c_2 \partial_y + c_0, \quad (\text{C.1})$$

where $c_{11}, c_{12}, c_{22}, c_1, c_2, c_0$ are constants. We then perform a von-Neumann stability analysis to arrive at a time-step restriction for the frozen coefficients. The final time-step is chosen to enforce

stability over the range of possible coefficients in the problem at hand.

Appendix C.1. Parabolic time-step restrictions for method of lines schemes

We consider the parabolic problem

$$u_t = Qu, \quad \mathbf{x} \in \Omega = [0, 2\pi]^2, \quad (\text{C.2})$$

with Q given by (C.1) together with periodic boundary conditions in both x, y directions and an initial condition. We consider a method of lines approximation. Let

$$\mathbf{x}_j = (j_1 \Delta x, j_2 \Delta y), \quad j_1 = 0, 1, \dots, N_x, \quad j_2 = 0, 1, \dots, N_y, \quad (\text{C.3})$$

denote the grid points for Ω with $N_x + 1$ and $N_y + 1$ points in the x and y directions, where $\Delta x = 2\pi/N_x$ and $\Delta y = 2\pi/N_y$. Let $U_j(t) \approx u(\mathbf{x}_j, t)$ denote the semi-discrete grid function.

Second-order accurate discretization. We discretize (C.2) to second-order accuracy in space

$$\frac{d}{dt} U_j(t) = c_{11} D_{+x} D_{-x} U_j + 2c_{12} D_{0x} D_{0y} U_j + c_{22} D_{+y} D_{-y} U_j + c_1 D_{0x} U_j + c_2 D_{0y} U_j + c_0 U_j. \quad (\text{C.4})$$

We perform a von-Neumann analysis to find the time-step restriction. The semi-discrete solution is expanded in a discrete Fourier series in space. We thus make the ansatz

$$U_j(t) = \hat{V}_{\mathbf{k}}(t) e^{i\mathbf{k} \cdot \mathbf{x}_j}, \quad k_x = -N_x/2, \dots, (N_x/2) - 1, \quad k_y = -N_y/2, \dots, (N_y/2) - 1,$$

where $\mathbf{k} = (k_x, k_y)$ and where we have assumed N_x and N_y are even for convenience. Substitute this ansatz into (C.4) and simplify to arrive at the standard test equation

$$\frac{d\hat{V}_{\mathbf{k}}}{dt} = \lambda_{\mathbf{k}}^{(2)} \hat{V}_{\mathbf{k}}, \quad (\text{C.5})$$

where the time-stepping eigenvalue is

$$\begin{aligned} \lambda_{\mathbf{k}}^{(2)} &= \alpha + i\beta, \\ \alpha &\stackrel{\text{def}}{=} -c_{11} \hat{k}_x^2 - 2c_{12} \hat{k}_{0x} \hat{k}_{0y} - c_{22} \hat{k}_y^2 + c_0, \\ \beta &\stackrel{\text{def}}{=} c_1 \hat{k}_{0x} + c_2 \hat{k}_{0y}, \end{aligned}$$

and where

$$\hat{k}_x \stackrel{\text{def}}{=} \frac{\sin(\xi_x/2)}{\Delta x/2}, \quad \hat{k}_y \stackrel{\text{def}}{=} \frac{\sin(\xi_y/2)}{\Delta y/2}, \quad (\text{C.6})$$

$$\hat{k}_{0x} \stackrel{\text{def}}{=} \frac{\sin(\xi_x)}{\Delta x}, \quad \hat{k}_{0y} \stackrel{\text{def}}{=} \frac{\sin(\xi_y)}{\Delta y}, \quad (\text{C.7})$$

$$\xi_x \stackrel{\text{def}}{=} k_x \Delta x, \quad \xi_y \stackrel{\text{def}}{=} k_y \Delta y. \quad (\text{C.8})$$

Note that we have the bounds,

$$-\pi \leq \xi_x \leq \pi, \quad -\pi \leq \xi_y \leq \pi, \quad (\text{C.9})$$

$$|\hat{k}_x| \leq \frac{2}{\Delta x}, \quad |\hat{k}_y| \leq \frac{2}{\Delta y}, \quad (\text{C.10})$$

$$|\hat{k}_{0x}| \leq \frac{1}{\Delta x}, \quad |\hat{k}_{0y}| \leq \frac{1}{\Delta y}. \quad (\text{C.11})$$

We now un-freeze the coefficients to obtain an estimate of the worst case time-stepping eigenvalue $\lambda_{\max} = \alpha_{\min} + i\beta_{\max}$ as

$$\alpha_{\min} = \min_{\mathbf{x} \in \Omega^P} \left[-\frac{4|c_{11}(\mathbf{x})|}{\Delta x^2} - \frac{2|c_{12}(\mathbf{x})|}{\Delta x \Delta y} - \frac{4|c_{22}(\mathbf{x})|}{\Delta y^2} + c_0(\mathbf{x}) \right],$$

$$\beta_{\max} = \max_{\mathbf{x} \in \Omega^P} \left[\frac{|c_1(\mathbf{x})|}{\Delta x} + \frac{|c_2(\mathbf{x})|}{\Delta y} \right],$$

where Ω^P is the physical domain under consideration. We approximate the region of absolute stability of the time-stepping scheme as an ellipse in the complex plane for $z = x + iy$, (for schemes not stable on the imaginary axis such as forward-Euler this assumes we are not too close to the imaginary axis)

$$\left(\frac{x}{\alpha_0} \right)^2 + \left(\frac{y}{\beta_0} \right)^2 \leq 1, \quad (\text{C.12})$$

and ensure that $z = \lambda_{\max} \Delta t$ lies within the approximate stability region. This leads to the condition,

$$\left(\frac{\alpha_{\min} \Delta t}{\alpha_0} \right)^2 + \left(\frac{\beta_{\max} \Delta t}{\beta_0} \right)^2 \leq 1, \quad (\text{C.13})$$

which can be re-arranged to give an inequality for Δt .

For forward-Euler we approximate the region of absolute stability using ellipse parameters $\alpha_0 = 2$ and $\beta_0 = 1$.

Fourth-order accurate discretization. We now use fourth-order accurate centered differences in (C.4). The time stepping eigenvalue is

$$\lambda_{\mathbf{k}}^{(4)} = -c_{11} \hat{k}_x^2 \left[1 + \frac{\Delta x^2}{12} \hat{k}_x^2 \right] - 2c_{12} \hat{k}_{0x} \left[1 + \frac{\Delta x^2}{6} \hat{k}_x^2 \right] \hat{k}_{0y} \left[1 + \frac{\Delta y^2}{6} \hat{k}_y^2 \right]$$

$$- c_{22} \hat{k}_y^2 \left[1 + \frac{\Delta y^2}{12} \hat{k}_y^2 \right] + c_1 i \hat{k}_{0x} \left[1 + \frac{\Delta x^2}{6} \hat{k}_x^2 \right] + c_2 i \hat{k}_{0y} \left[1 + \frac{\Delta y^2}{6} \hat{k}_y^2 \right] + c_0.$$

We shall use the bounds

$$\hat{k}_x^2 \left[1 + \frac{\Delta x^2}{12} \hat{k}_x^2 \right] \leq \frac{16}{3\Delta x^2}, \quad (\text{C.14})$$

$$|\hat{k}_{0x}| \left[1 + \frac{\Delta x^2}{6} \hat{k}_x^2 \right] \leq \frac{3}{2\Delta x}, \quad (\text{C.15})$$

and similar ones for the y variable to obtain an estimate for the worst case time-stepping eigenvalue with real and imaginary parts given by

$$\alpha_{\min} = \min_{\mathbf{x} \in \Omega^P} \left[-\frac{16|c_{11}(\mathbf{x})|}{3\Delta x^2} - \frac{9|c_{12}(\mathbf{x})|}{2\Delta x \Delta y} - \frac{16|c_{22}(\mathbf{x})|}{3\Delta y^2} + c_0(\mathbf{x}) \right],$$

$$\beta_{\max} = \max_{\mathbf{x} \in \Omega^P} \left[\frac{3|c_1(\mathbf{x})|}{2\Delta x} + \frac{3|c_2(\mathbf{x})|}{2\Delta y} \right].$$

Sixth-order accurate discretization. For sixth-order accurate centered differences in (C.4), the time stepping eigenvalue is

$$\begin{aligned}\lambda_{\mathbf{k}}^{(6)} = & -c_{11} \hat{k}_x^2 \left[1 + \frac{\Delta x^2}{12} \hat{k}_x^2 + \frac{\Delta x^4}{90} \hat{k}_x^4 \right] - c_{22} \hat{k}_y^2 \left[1 + \frac{\Delta y^2}{12} \hat{k}_y^2 + \frac{\Delta y^4}{90} \hat{k}_y^4 \right] \\ & - 2c_{12} \hat{k}_{0x} \left[1 + \frac{\Delta x^2}{6} \hat{k}_x^2 + \frac{\Delta x^4}{30} \hat{k}_x^4 \right] \hat{k}_{0y} \left[1 + \frac{\Delta y^2}{6} \hat{k}_y^2 + \frac{\Delta y^4}{30} \hat{k}_y^4 \right] \\ & + c_1 i \hat{k}_{0x} \left[1 + \frac{\Delta x^2}{6} \hat{k}_x^2 + \frac{\Delta x^4}{30} \hat{k}_x^4 \right] + c_2 i \hat{k}_{0y} \left[1 + \frac{\Delta y^2}{6} \hat{k}_y^2 + \frac{\Delta y^4}{30} \hat{k}_y^4 \right] + c_0.\end{aligned}$$

We use the bounds

$$\hat{k}_x^2 \left[1 + \frac{\Delta x^2}{12} \hat{k}_x^2 + \frac{\Delta x^4}{90} \hat{k}_x^4 \right] \leq \frac{272}{45\Delta x^2}, \quad (\text{C.16})$$

$$|\hat{k}_{0x}| \left[1 + \frac{\Delta x^2}{6} \hat{k}_x^2 + \frac{\Delta x^4}{30} \hat{k}_x^4 \right] \leq \frac{1199}{756\Delta x}, \quad (\text{C.17})$$

and similar ones for the y variable to estimate the worst case time-stepping eigenvalue with real and imaginary parts as follows

$$\begin{aligned}\alpha_{\min} = & \min_{\mathbf{x} \in \Omega^P} \left[-\frac{272 |c_{11}(\mathbf{x})|}{45\Delta x^2} - \frac{1313 |c_{12}(\mathbf{x})|}{261\Delta x \Delta y} - \frac{272 |c_{22}(\mathbf{x})|}{45\Delta y^2} + c_0(\mathbf{x}) \right], \\ \beta_{\max} = & \max_{\mathbf{x} \in \Omega^P} \left[\frac{1199 |c_1(\mathbf{x})|}{756\Delta x} + \frac{1199 |c_2(\mathbf{x})|}{756\Delta y} \right].\end{aligned}$$

Appendix C.2. Time-step restriction for hyperbolic problems

We consider the hyperbolic problem

$$u_{tt} = Qu, \quad \mathbf{x} \in [0, 2\pi]^2, \quad (\text{C.18})$$

with Q given in (C.1) together with periodic boundary conditions in both x, y directions and initial conditions. We set $U_j^n \approx u(\mathbf{x}_j, t^n)$ where \mathbf{x}_j is defined in (C.3).

Second-order accurate discretization. We discretize (C.18) to second-order accuracy in space and time

$$D_{+t}D_{-t}U_j^n = c_{11}D_{+x}D_{-x}U_j^n + 2c_{12}D_{0x}D_{0y}U_j^n + c_{22}D_{+y}D_{-y}U_j^n + c_1D_{0x}U_j^n + c_2D_{0y}U_j^n + c_0U_j^n, \quad \mathbf{x}_j \in \Omega_h. \quad (\text{C.19})$$

We perform a Von-Neumann analysis to find the time-step restriction where we make the ansatz

$$U_j^n = A^n e^{i\mathbf{k} \cdot \mathbf{x}_j}, \quad k_x = -N_x/2, \dots, (N_x/2) - 1, \quad k_y = -N_y/2, \dots, (N_y/2) - 1,$$

with $\mathbf{k} = (k_x, k_y)$ and N_x, N_y assumed even. We plug the ansatz in (C.19) and simplify to arrive at a quadratic equation for A

$$A^2 - 2bA + 1 = 0, \quad (\text{C.20})$$

where

$$b = 1 + \frac{\Delta t^2}{2} \hat{Q}_h, \quad (\text{C.21})$$

and

$$\hat{Q}_h = -c_{11}\hat{k}_x^2 - 2c_{12}\hat{k}_{0x}\hat{k}_{0y} - c_{22}\hat{k}_y^2 + ic_1\hat{k}_{0x} + ic_2\hat{k}_{0y} + c_0. \quad (\text{C.22})$$

The parameters $\hat{k}_x, \hat{k}_y, \hat{k}_{0x}, \hat{k}_{0y}$ are defined in (C.6) and (C.7).

For stability, we wish to find a bound on Δt such that the roots of (C.20), A_{\pm} , satisfy $|A_+|, |A_-| \leq 1$, and that there be no double roots with $|A| = 1$, to avoid algebraic growth in time. From equation (C.20), we see that $A_+A_- = 1$. Therefore, $A_+ = e^{i\theta}$ and $A_- = e^{-i\theta}$, for $\theta \in \mathbb{R}$. Furthermore, $b = (A_+ + A_-)/2 = \cos \theta$. Since there are double roots when $b = \pm 1$, we require Δt such that $b \in \mathbb{R}$ and $|b| < 1$.

For this problem, b is complex if $c_1 \neq 0$ or $c_2 \neq 0$ and then we cannot satisfy $|A| \leq 1$. In this case some dissipation should therefore be added to the scheme so that there is no growth in time. We note that we could generalize our definition of stability to allow bounded growth in time in which case the scheme could formally be stable using this extended definition of stability. This is because as we refine the mesh, and assuming $\Delta t \leq (\text{const.}) \min\{\Delta x, \Delta y\}$ then

$$b \sim 1 - \underbrace{\frac{\Delta t^2}{2} \left(c_{11}\hat{k}_x^2 + 2c_{12}\hat{k}_{0x}\hat{k}_{0y} + c_{22}\hat{k}_y^2 \right)}_{\tilde{b}} + \mathcal{O}(\Delta t), \quad \text{as } \Delta t \rightarrow 0, \quad (\text{C.23})$$

since $\Delta t^2|\hat{k}_{0x}| \leq (\text{const.})\Delta t$ and $\Delta t^2|\hat{k}_{0y}| \leq (\text{const.})\Delta t$. The contribution of the complex part of the symbol to b in (C.23), which comes from the lower order terms, goes to zero like Δt as $\Delta t \rightarrow 0$. Therefore, provided $|\tilde{b}| < 1$, we have $|A_{\pm}| = 1 + \mathcal{O}(\Delta t)$ and the scheme will thus be stable, with bounded growth. Even in this case it is still recommended that dissipation be added to the scheme.

Consider then

$$\begin{aligned} \tilde{b}^2 &= 1 - \Delta t^2 \left(c_{11}\hat{k}_x^2 + 2c_{12}\hat{k}_{0x}\hat{k}_{0y} + c_{22}\hat{k}_y^2 \right) \\ &\quad + \frac{\Delta t^4}{4} \left(c_{11}\hat{k}_x^2 + 2c_{12}\hat{k}_{0x}\hat{k}_{0y} + c_{22}\hat{k}_y^2 \right)^2 < 1. \end{aligned}$$

We look for a time-step Δt such that

$$\frac{\Delta t^2}{4} \left(c_{11}\hat{k}_x^2 + 2c_{12}\hat{k}_{0x}\hat{k}_{0y} + c_{22}\hat{k}_y^2 \right) < 1. \quad (\text{C.24})$$

With the necessary condition that $c_{11}c_{22} > c_{12}^2$, we find that

$$\max_{-\pi \leq \xi_x, \xi_y < \pi} \left[c_{11}\hat{k}_x^2 + 2c_{12}\hat{k}_{0x}\hat{k}_{0y} + c_{22}\hat{k}_y^2 \right] = 4 \left(\frac{c_{11}}{\Delta x^2} + \frac{c_{22}}{\Delta y^2} \right), \quad (\text{C.25})$$

occurs when $(\xi_x, \xi_y) = (-\pi, -\pi)$. Hence,

$$\Delta t < \frac{1}{\sqrt{\frac{c_{11}}{\Delta x^2} + \frac{c_{22}}{\Delta y^2}}}. \quad (\text{C.26})$$

We obtain the bound in (C.25) by considering the maximum of the function

$$\chi(\xi_x, \xi_y) = c_{11}\hat{k}_x^2(\xi_x) + 2c_{12}\hat{k}_{0x}(\xi_x)\hat{k}_{0y}(\xi_y) + c_{22}\hat{k}_y^2(\xi_y). \quad (\text{C.27})$$

We find the critical points of χ using the equations

$$\chi_{\xi_x} = \frac{c_{11}}{\Delta x^2} \sin \xi_x + \frac{c_{12}}{\Delta x \Delta y} \cos \xi_x \sin \xi_y = 0, \quad (\text{C.28})$$

$$\chi_{\xi_y} = \frac{c_{22}}{\Delta y^2} \sin \xi_y + \frac{c_{12}}{\Delta x \Delta y} \sin \xi_x \cos \xi_y = 0. \quad (\text{C.29})$$

Seeing that $\cos(\xi_y) \neq 0$ (otherwise we arrive at a contradiction), equation (C.29) gives

$$\sin \xi_x = -\frac{\Delta x}{\Delta y} \frac{c_{22}}{c_{12}} \tan \xi_y. \quad (\text{C.30})$$

We use (C.30) to rewrite (C.28) in terms of ξ_y ,

$$\sin \xi_y \left(-\frac{c_{11}c_{22}}{c_{12}} \frac{1}{\cos \xi_y} \pm c_{12} \sqrt{1 - \frac{\Delta x^2}{\Delta y^2} \frac{c_{22}^2}{c_{12}^2} \tan^2 \xi_y} \right) = 0. \quad (\text{C.31})$$

If we suppose that

$$\pm c_{12} \sqrt{1 - \frac{\Delta x^2}{\Delta y^2} \frac{c_{22}^2}{c_{12}^2} \tan^2 \xi_y} = \frac{c_{11}c_{22}}{c_{12}} \frac{1}{\cos \xi_y},$$

square both sides and simplify to obtain

$$\begin{aligned} \cos^2 \xi_y - \frac{\Delta x^2}{\Delta y^2} \frac{c_{22}^2}{c_{12}^2} \sin^2 \xi_y &= \frac{c_{11}^2 c_{22}^2}{c_{12}^4}, \\ 1 - \sin^2 \xi_y - \frac{\Delta x^2}{\Delta y^2} \frac{c_{22}^2}{c_{12}^2} \sin^2 \xi_y &= \frac{c_{11}^2 c_{22}^2}{c_{12}^4}, \\ \sin^2 \xi_y \left(-c_{12}^4 - \frac{\Delta x^2}{\Delta y^2} c_{22}^2 c_{12}^2 \right) &= c_{11}^2 c_{22}^2 - c_{12}^4, \end{aligned}$$

we find that $c_{11}^2 c_{22}^2 - c_{12}^4 < 0 \implies c_{11} c_{22} < c_{12}^2$ which violates the necessary condition $c_{11} c_{22} > c_{12}^2$.

Therefore, the function $\chi(\xi_x, \xi_y)$ has critical points when $\sin(\xi_y) = \sin(\xi_x) = 0$, or at $(-\pi, -\pi), (-\pi, 0), (0, \pi), (0, 0)$ and is maximum at $(-\pi, -\pi)$.

References

- [1] D. Appellö, J. W. Banks, W. D. Henshaw, D. W. Schwendeman, Numerical methods for solid mechanics on overlapping grids: Linear elasticity, *J. Comput. Phys.* 231 (18) (2012) 6012–6050.
- [2] J. W. Banks, W. D. Henshaw, A. Kapila, D. W. Schwendeman, An added-mass partitioned algorithm for fluid-structure interactions of compressible fluids and nonlinear solids, *J. Comput. Phys.* 305 (2016) 1037–1064.
- [3] W. D. Henshaw, A fourth-order accurate method for the incompressible Navier-Stokes equations on overlapping grids, *J. Comput. Phys.* 113 (1) (1994) 13–25.
- [4] F. Meng, J. W. Banks, W. D. Henshaw, D. W. Schwendeman, Fourth-order accurate fractional-step IMEX schemes for the incompressible Navier-Stokes equations on moving overlapping grids, *Computer Methods in Applied Mechanics and Engineering* 366 (2020) 113040.

- [5] W. D. Henshaw, A high-order accurate parallel solver for Maxwell's equations on overlapping grids, *SIAM J. Sci. Comput.* 28 (5) (2006) 1730–1765.
- [6] J. Angel, J. W. Banks, W. D. Henshaw, M. J. Jenkinson, A. V. Kildishev, G. Kovačič, L. J. Prokopeva, D. W. Schwendeman, A high-order accurate scheme for Maxwell's equations with a generalized dispersion model, *J. Comput. Phys.* 378 (2019) 411–444.
- [7] J. W. Banks, B. Buckner, W. D. Henshaw, M. J. Jenkinson, , A. V. Kildishev, G. Kovačič, L. J. Prokopeva, D. W. Schwendeman, A high-order accurate scheme for Maxwell's equations with a generalized dispersive material (GDM) model and material interfaces, *J. Comput. Phys.* 412 (2020) 109424.
- [8] W. D. Henshaw, H.-O. Kreiss, L. G. M. Reyna, A fourth-order accurate difference approximation for the incompressible Navier-Stokes equations, *Comput. Fluids* 23 (4) (1994) 575–593.
- [9] W. D. Henshaw, D. W. Schwendeman, Moving overlapping grids with adaptive mesh refinement for high-speed reactive and non-reactive flow, *J. Comput. Phys.* 216 (2) (2006) 744–779.
- [10] W. D. Henshaw, K. K. Chand, A composite grid solver for conjugate heat transfer in fluid-structure systems, *J. Comput. Phys.* 228 (2009) 3708–3741.
- [11] J. W. Banks, W. D. Henshaw, D. W. Schwendeman, Q. Tang, A stable partitioned FSI algorithm for rigid bodies and incompressible flow. Part II: General formulation, *J. Comput. Phys.* 343 (2017) 469–500.
- [12] J. W. Banks, W. D. Henshaw, D. W. Schwendeman, Q. Tang, A stable partitioned FSI algorithm for rigid bodies and incompressible flow. Part I: Model problem analysis, *J. Comput. Phys.* 343 (2017) 432–468.
- [13] J. W. Banks, W. D. Henshaw, D. W. Schwendeman, Q. Tang, A stable partitioned FSI algorithm for rigid bodies and incompressible flow in three dimensions, *J. Comput. Phys.* 373 (2018) 455–492.
- [14] D. A. Serino, J. W. Banks, W. D. Henshaw, D. W. Schwendeman, A stable added-mass partitioned (AMP) algorithm for elastic solids and incompressible flow, *J. Comput. Phys.* 399 (2019) 1–30.
- [15] D. A. Serino, J. W. Banks, W. D. Henshaw, D. W. Schwendeman, A stable added-mass partitioned (AMP) algorithm for elastic solids and incompressible flow: Model problem analysis, *SIAM J. Sci. Comput.* 41 (4) (2019) A2464–A2484.
- [16] J. W. Banks, W. D. Henshaw, D. W. Schwendeman, An analysis of a new stable partitioned algorithm for FSI problems. Part I: Incompressible flow and elastic solids, *J. Comput. Phys.* 269 (2014) 108–137.
- [17] J. W. Banks, W. D. Henshaw, D. W. Schwendeman, An analysis of a new stable partitioned algorithm for FSI problems. Part II: Incompressible flow and structural shells, *J. Comput. Phys.* 268 (2014) 399–416.
- [18] F. Meng, J. W. Banks, W. D. Henshaw, D. W. Schwendeman, A stable and accurate partitioned algorithm for conjugate heat transfer, *J. Comput. Phys.* 344 (2017) 51–85.

- [19] B. Gutfafsson, High Order Difference Methods for Time Dependent PDE, Springer Series in Computational Mathematics, Springer, 2008.
- [20] H.-O. Kreiss, N. A. Petersson, A second order accurate embedded boundary method for the wave equation with Dirichlet data, *SIAM Journal on Scientific Computing* 27 (4) (2006) 1141–1167.
- [21] S. Nilsson, N. A. Petersson, B. Sjögreen, H.-O. Kreiss, Stable difference approximations for the elastic wave equation in second order formulation, *SIAM Journal on Numerical Analysis* 45 (5) (2007) 1902–1936.
- [22] D. Appelö, N. A. Petersson, A fourth-order accurate embedded boundary method for the wave equation, *SIAM Journal on Scientific Computing* 34 (6) (2012) A2982–A3008.
- [23] B. Sjögreen, N. A. Petersson, A fourth order finite difference scheme for the elastic wave equation in second order formulation, *Scient. Comput.* 52 (2012) 17–48.
- [24] R. J. LeVeque, Z. Li, The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, *SIAM Journal on Numerical Analysis* 31 (4) (1994) 1019–1044.
- [25] R. J. LeVeque, Z. Li, Immersed interface methods for stokes flow with elastic boundaries or surface tension, *SIAM Journal on Scientific Computing* 18 (3) (1997) 709–735.
- [26] R. J. LeVeque, D. Calhoun, Cartesian grid methods for fluid flow in complex geometries, in: L. J. Fauci, S. Gueron (Eds.), *Computational Modeling in Biological Fluid Dynamics*, Vol. 124 of IMA Volumes in Mathematics and its Applications, Springer-Verlag, 2001, pp. 117–143.
- [27] S. Tan, C.-W. Shu, Inverse lax-wendroff procedure for numerical boundary conditions of conservation laws, *Journal of Computational Physics* 229 (21) (2010) 8144–8166.
- [28] R. B. d. R. Borges, N. D. P. da Silva, F. A. A. Gomes, C.-W. Shu, S. Tan, A Sequel of Inverse Lax–Wendroff High Order Wall Boundary Treatment for Conservation Laws, *Archives of Computational Methods in Engineering*.
- [29] J. Lu, C.-W. Shu, S. Tan, M. Zhang, An inverse Lax-Wendroff procedure for hyperbolic conservation laws with changing wind direction on the boundary, *Journal of Computational Physics* 426 (2021) 109940.
- [30] C.-W. Shu, S. Tan, Chapter 2 - inverse Lax–Wendroff procedure for numerical boundary treatment of hyperbolic equations, in: R. Abgrall, C.-W. Shu (Eds.), *Handbook of Numerical Methods for Hyperbolic Problems*, Vol. 18 of Handbook of Numerical Analysis, Elsevier, 2017, pp. 23–52.
- [31] T. Li, C.-W. Shu, M. Zhang, Stability Analysis of the Inverse Lax–Wendroff Boundary Treatment for High Order Central Difference Schemes for Diffusion Equations, *Journal of Scientific Computing* 70 (2) (2017) 576–607.
- [32] T. Li, C.-W. Shu, M. Zhang, Stability Analysis of the Inverse Lax–Wendroff Boundary Treatment for High Order Central Difference Schemes for Diffusion Equations, *Journal of Scientific Computing* 70 (2) (2017) 576–607.

- [33] J. Lu, J. Fang, S. Tan, C.-W. Shu, M. Zhang, Inverse Lax–Wendroff procedure for numerical boundary conditions of convection–diffusion equations, *Journal of Computational Physics* 317 (2016) 276–300.
- [34] T. R. Hughes, *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*, Dover Publications, New York, 2000.
- [35] G. Strang, G. Fix, *An Analysis of the Finite Element Method 2nd Edition*, 2nd Edition, Wellesley-Cambridge, 2008.
- [36] J. S. Hesthaven, T. Warburton, *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications.*, Springer Verlag, New York, 2008.
- [37] T. J. Hughes, J. A. Evans, A. Reali, Finite element and NURBS approximations of eigenvalue, boundary-value, and initial-value problems, *Comput. Methods Appl. Mech. Engrg.* 272 (2014) 290–320.
- [38] J. W. Banks, T. Hagstrom, On Galerkin difference methods, *J. Comput. Phys.* 313 (2016) 310–327.
- [39] J. W. Banks, T. Hagstrom, J. Jacangelo, Galerkin differences for acoustic and elastic wave equations in two space dimensions, *J. Comput. Phys.* 372 (2018) 864–892.
- [40] J. J. Jacangelo, J. W. Banks, T. Hagstrom, Galerkin differences for high-order partial differential equations, *SIAM J. Sci. Comput.* 412 (2020) B447–B471.
- [41] I. S. Duff, J. Koster, On algorithms for permuting large entries to the diagonal of a sparse matrix, *SIAM Journal on Matrix Analysis and Applications* 22 (4) (2001) 973–996.
- [42] L. Anné, P. Joly, Q. H. Tran, Construction and analysis of higher order finite difference schemes for the 1d wave equation, *Computational Geosciences* 4 (2000) 207–249.
- [43] J. W. Banks, W. D. Henshaw, Upwind schemes for the wave equation in second-order form, *J. Comput. Phys.* 231 (17) (2012) 5854–5889.
- [44] J. Angel, J. W. Banks, W. D. Henshaw, High-order upwind schemes for the wave equation on overlapping grids: Maxwell’s equations in second-order form, *J. Comput. Phys.* 352 (2018) 534–567.
- [45] P. A. Martin, *Multiple Scattering: Interaction of Time-Harmonic Waves with N Obstacles*, *Encyclopedia of Mathematics and its Applications*, Cambridge University Press, 2006.
- [46] J. W. Banks, T. Aslam, W. J. Rider, On sub-linear convergence for linearly degenerate waves in capturing schemes, *J. Comput. Phys.* 227 (14) (2008) 6985–7002, .