LOCAL COMPATIBILITY BOUNDARY CONDITIONS FOR HIGH-ORDER ACCURATE FINITE-DIFFERENCE APPROXIMATIONS OF PDES*

NOUR G. AL HASSANIEH†, JEFFREY W. BANKS†, WILLIAM D. HENSHAW†, AND DONALD W. SCHWENDEMAN†

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 in two space dimensions for schemes up to sixth-order accuracy.

Key word. compatibility conditions, boundary conditions, heat equation, wave equation, high-order finite-differences

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 grid points 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.

*Submitted to the editors May 17, 2022.

Funding: Research supported by the National Science Foundation under grant DMS-1818926.

†Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, NY 12180, USA (alhasn@rpi.edu, banksj3@rpi.edu, henshw@rpi.edu, schwed@rpi.edu).

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 1.1, 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. We have developed second-order accurate schemes for the equations of linear and nonlinear elasticity [4, 9], and up to fourth-order accurate schemes for the incompressible Navier-Stokes equations [17, 35] and Maxwell's equations [18, 2, 6] 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 becomes 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.

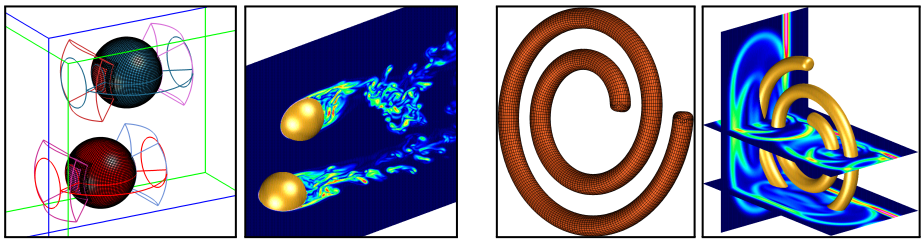

FIGURE 1.1. *Some target applications for the new LCBC approach. Left: overset grid for two spherical bodies and computed incompressible flow (vorticity). Right: overset grid for a spiral wire and computed electromagnetic scattering.*

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 years[1], 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 [20] and the incompressible Navier-Stokes equations [17, 35]. For wave equations, we have described the use of CBCs for the compressible Euler equations [21] and linear elasticity [4], and for high-order accurate approximations to Maxwell's equations [18, 2]. CBCs are also useful for problems involving material interfaces, such as conjugate heat transfer [19] and electromagnetics [18, 6]. 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 [13, 12, 14] and elastic solids [37, 38]. 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 [10, 11]. In related work, we have also used CBCs in the CHAMP scheme [34] 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 [16]. CBCs have also been used to derive stable and accurate embedded boundary[2] approximations [26, 36, 5]. CBCs have been incorporated into summation-by-parts schemes by Sjögreen and Petersson for the equations of elasticity [40]. CBCs have been used by LeVeque and Li with their immersed interface method to develop accurate approximations at embedded interfaces [28, 29, 27]. Shu and collaborators have used CBCs in their inverse-Lax-Wendroff approach for hyperbolic equations and conservation laws [42, 15, 33, 39] as well as for parabolic and advection-diffusion equations [30, 31, 32].

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 [24, 41, 22]. Similarly for B-Spline FEM, as commonly used in isogeometric analysis, one-sided operators occurring near boundaries result in spurious large eigenvalues, so-called outlier eigenvalues [23]. Banks et al. [7, 8, 25], 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.

[1] For example, CBCs were known to Professor H.-O. Kreiss and his students at least since the 1980's.

[2] By 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).

2. 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. Consider the initial-boundary-value problem[3] on $[0,T]\times\Omega$, $T>0$, given by

$$\begin{cases} \partial_t^q u = Qu + f(\mathbf{x},t), & \mathbf{x}\in\Omega, \quad t\in(0,T], \quad q=1,2, \\ \mathcal{B}u(\mathbf{x},t) = g(\mathbf{x},t), & \mathbf{x}\in\partial\Omega, \quad t\in[0,T], \\ \partial_t^{\alpha-1}u(\mathbf{x},0) = u_{\alpha-1}(\mathbf{x}), & \mathbf{x}\in\overline{\Omega}, \quad \alpha=1,\ldots,q, \quad q=1,2. \end{cases} \tag{2.1}$$

Here, $\Omega\subset\mathbb{R}^2$ is a general domain, $\partial\Omega$ denotes the boundary of Ω, and $\overline{\Omega}=\Omega\cup\partial\Omega$. We define the variable coefficient elliptic operator Q as

$$Qu \stackrel{\text{def}}{=} c_{11}(\mathbf{x})\partial_x^2 u + 2c_{12}(\mathbf{x})\partial_x\partial_y u + c_{22}(\mathbf{x})\partial_y^2 u + c_1(\mathbf{x})\partial_x u + c_2(\mathbf{x})\partial_y u + c_0(\mathbf{x})u. \tag{2.2}$$

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. For example, necessary conditions are that $c_{11}(\mathbf{x})>0$, $c_{22}(\mathbf{x})>0$ and $c_{11}(\mathbf{x})c_{22}(\mathbf{x})-c_{12}^2(\mathbf{x})\geqslant\delta>0$, for all $\mathbf{x}\in\Omega$. We note that (2.2) 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 (2.1), with given forcing function $f(\mathbf{x},t)$, takes the form of a parabolic ($q=1$) or hyperbolic ($q=2$) PDE in second-order form depending on the choice of the index q. The boundary conditions in (2.1), with given forcing function $g(\mathbf{x},t)$, are written in terms of the boundary operator given by

$$\mathcal{B}u \stackrel{\text{def}}{=} b_1(\mathbf{x})u + b_2(\mathbf{x})\partial_n u, \qquad \mathbf{x}\in\partial\Omega, \tag{2.3}$$

where ∂_n is the outward normal derivative and the coefficient functions satisfy $|b_1(\mathbf{x})|+|b_2(\mathbf{x})|\neq 0$, $\forall\mathbf{x}\in\partial\Omega$.

We are motivated by the application of the LCBC method for high-order accurate discretizations of the problem in (2.1) on mapped grids. For such discretizations, we consider a smooth mapping from the unit square to Ω. The form of the problem remains unchanged in the mapped domain, so it suffices to study the governing equations in (2.1) over the domain $\Omega=(0,1)^2$.

Let $U_{\mathbf{i}}\approx u(\mathbf{x}_{\mathbf{i}},t)$ represent the numerical approximation of the exact solution of (2.1) at discrete points $\mathbf{x}_{\mathbf{i}}$ on the Cartesian grid $\overline{\Omega}_h$,

$$\overline{\Omega}_h \stackrel{\text{def}}{=} \{\mathbf{x}_{\mathbf{i}}=(x_i,y_j)=(i\Delta x, j\Delta y),\ i=0,\ldots,N_x,\ j=0,\ldots,N_y\}, \tag{2.4}$$

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 3.1. Let $\partial\Omega_h$ denote the set of grid points on the boundary and $\Omega_h=\overline{\Omega}_h\backslash\partial\Omega_h$ the interior grid points.

Our principal focus is on discretizations of (2.1) to fourth and sixth-order accuracy, although we also consider second-order accurate approximations as a baseline. A second-order accurate discretization of (2.1) employs standard centered differences for the first and second derivatives given by

$$D_{2,\zeta} \stackrel{\text{def}}{=} D_{0\zeta}, \qquad D_{2,\zeta\zeta} \stackrel{\text{def}}{=} D_{+\zeta}D_{-\zeta}, \qquad \zeta=x,y. \tag{2.5}$$

[3]The solution of second-order elliptic boundary value problems can also be treated with the LCBC approach, see [1].

Compact fourth and sixth-order accurate centered approximations, $D_{2p,\zeta}$ and $D_{2p,\zeta\zeta}$, $p = 2, 3$, are defined in the usual way, for example,

$$D_{6,\zeta\zeta} \stackrel{\text{def}}{=} D_{+\zeta}D_{-\zeta}\big(I - \frac{\Delta\zeta^2}{12}D_{+\zeta}D_{-\zeta} + \frac{\Delta\zeta^4}{90}\left(D_{+\zeta}D_{-\zeta}\right)^2\big). \tag{2.6}$$

Using these approximations, the d^{th}-order accurate approximation to Q is given by

$$\begin{aligned} Q_{d,h} \stackrel{\text{def}}{=} & c_{11}(\mathbf{x_i})D_{d,xx} + 2c_{12}(\mathbf{x_i})D_{d,x}D_{d,y} + c_{22}(\mathbf{x_i})D_{d,yy} \\ & + c_1(\mathbf{x_i})D_{d,x} + c_2(\mathbf{x_i})D_{d,y} + c_0(\mathbf{x_i})I. \end{aligned} \tag{2.7}$$

Similarly let $\mathcal{B}_{d,h}$ be the d^{th}-order accurate centered discretization of the boundary operator $\mathcal{B}$. The semi-discrete model problem now takes the form

$$\begin{cases} \partial_t^q U_{\mathbf{i}}(t) = Q_{d,h}U_{\mathbf{i}}(t) + f(\mathbf{x_i}, t), & \mathbf{x_i} \in \Omega_h, \quad t \in (0, T], \\ \mathcal{B}_{d,h}U_{\mathbf{i}}(t) = g(\mathbf{x_i}, t), & \mathbf{x_i} \in \partial\Omega_h, \quad t \in [0, T], \\ \partial_t^{\alpha-1}U_{\mathbf{i}}(0) = u_{\alpha-1}(\mathbf{x_i}), & \mathbf{x_i} \in \overline{\Omega}_h, \quad \alpha = 1, \ldots, q. \end{cases} \tag{2.8}$$

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

$$\Omega_h^e \stackrel{\text{def}}{=} \{\mathbf{x_i} \mid \mathbf{i} = (i, j),\ i = -p, \ldots, N_x + p,\ j = -p, \ldots, N_y + p\}, \tag{2.9}$$

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

The LCBC method uses compatibility boundary conditions obtained from the primary boundary conditions and the governing PDE (and its derivatives) applied on the boundary. Taking q time derivatives of the primary boundary condition in (2.1) gives

$$\mathcal{B}\partial_t^q u(\mathbf{x}, t) = \partial_t^q g(\mathbf{x}, t), \qquad \mathbf{x} \in \partial\Omega, \tag{2.10}$$

at a fixed time $t \in [0, T]$. Applying the PDE from (2.1) yields

$$\mathcal{B}Qu(\mathbf{x}, t) = \partial_t^q g(\mathbf{x}, t) - \mathcal{B}f(\mathbf{x}, t), \qquad \mathbf{x} \in \partial\Omega. \tag{2.11}$$

Repeating the process ν times gives the ν^{th} compatibility condition

$$\mathcal{B}Q^\nu u(\mathbf{x}, t) = \partial_t^{q\nu} g(\mathbf{x}, t) - \mathcal{B}\Psi_\nu f(\mathbf{x}, t), \quad \mathbf{x} \in \partial\Omega, \quad \nu = 1, 2, \ldots, \quad \text{CBC}_{\mathcal{B},q}[\nu] \tag{2.12}$$

denoted by $\text{CBC}_{\mathcal{B},q}[\nu]$, where Ψ_ν is a differential operator defined by

$$\Psi_\nu f(\mathbf{x}, t) \stackrel{\text{def}}{=} \sum_{k=1}^{\nu} Q^{k-1}\partial_t^{q(\nu-k)} f(\mathbf{x}, t), \qquad \mathbf{x} \in \partial\Omega, \quad \nu = 1, 2, \ldots. \tag{2.13}$$

3. LCBC method. We now provide a description of the LCBC method for the IBVP in (2.8). The goal is to specify solution values at ghost points adjacent to grid faces and grid corners; these are shown in Figure 3.1 for the case of a fourth-order accurate scheme that requires two ghost points. 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*. The process is similar for a Neumann or Robin boundary condition, see [1]. Finally, we describe the treatment near the corner.

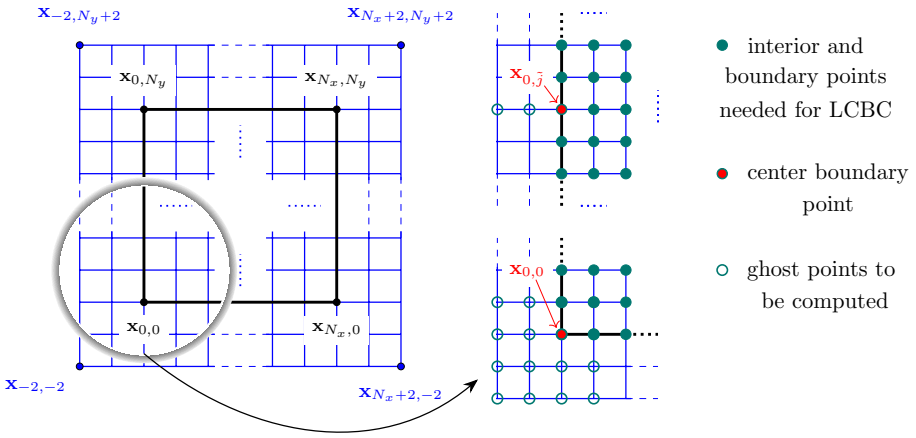

FIGURE 3.1. *Grid, with ghost points, for a fourth-order accurate approximation.*

3.1. 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 (2.1) becomes

$$u(\mathbf{x},t) = g_\ell(y,t), \qquad \mathbf{x} \in \partial\Omega_\ell, \tag{3.1}$$

for a fixed time $t \in [0,T]$. Define an interpolating polynomial $\tilde{u}(x,y)$, centered about $(\tilde{x},\tilde{y})$, as

$$\tilde{u}(x,y) \stackrel{\text{def}}{=} \sum_{\hat{n}=-p}^{p} \sum_{\hat{m}=-p}^{p} d_{\hat{m},\hat{n}} L_{\hat{m}}\left(\frac{x-\tilde{x}}{\Delta x}\right) L_{\hat{n}}\left(\frac{y-\tilde{y}}{\Delta y}\right), \qquad p \in \mathbb{N}, \tag{3.2}$$

where $L_k(z)$ is a Lagrange basis function $L_k(z) = \prod_{\substack{l=-p \\ l\neq k}}^{p} \frac{(z-l)}{(k-l)}$. Note that $\tilde{u}$ has the property $\tilde{u}\big(\tilde{x}+\hat{i}\Delta x, \tilde{y}+\hat{j}\Delta y\big) = d_{\hat{i},\hat{j}}$, for $\hat{i},\hat{j} = -p,\ldots,p$. The $\tilde{m} = (2p+1)^2$ coefficients $d_{\hat{m},\hat{n}}$, $\hat{m},\hat{n} = -p,\ldots,p$ in (3.2) are found by enforcing the constraints

$$\tilde{u}\big(0, \tilde{y}+\hat{j}\Delta y\big) = g_\ell\big(\tilde{y}+\hat{j}\Delta y, t\big), \qquad \hat{j} = -p,\ldots,p, \tag{3.3a}$$

$$\tilde{u}\big(\hat{i}\Delta x, \tilde{y}+\hat{j}\Delta y\big) = U_{\hat{i},\tilde{j}+\hat{j}}(t), \qquad \hat{i} = 1,\ldots,p, \quad \hat{j} = -p,\ldots,p, \tag{3.3b}$$

$$\partial_y^\mu Q^\nu \tilde{u}(0,\tilde{y}) = \partial_y^\mu R_{\ell,\nu}(\tilde{y},t), \qquad \nu = 1,\ldots,p, \quad \mu = 0,\ldots,2p, \tag{3.3c}$$

where

$$R_{\ell,\nu}(y,t) \stackrel{\text{def}}{=} \partial_t^{q\nu} g_\ell(y,t) - \Psi_\nu f(0,y,t). \tag{3.4}$$

The constraints in (3.3a) are the Dirichlet boundary condition applied at $2p+1$ grid points about the boundary point $(0,\tilde{y})$, while (3.3b) sets $\tilde{u}$ equal to $U_{\mathbf{i}}$ at $p(2p+1)$ grid points interior to the boundary point. The last constraints in (3.3c) require that $\tilde{u}$ satisfy $2p+1$ tangential derivatives of the compatibility boundary conditions,

$\text{CBC}_{\ell,q}[\nu]$, $\nu = 1, \ldots, p$, evaluated at the boundary point $(0, \tilde{y})$. Together, the constraints in (3.3) 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

$$\partial\tilde{\Omega}_{\ell,h} \stackrel{\text{def}}{=} \left\{ \mathbf{x_i} \mid i = 0,\ j = p, p+1, \ldots, N_y - p \right\}, \tag{3.5}$$

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 (3.3) has the form

$$A\mathbf{d} = \mathbf{b}, \tag{3.6}$$

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 (3.2) organized as

$$\mathbf{d} = [d_{-p,-p}, \ldots, d_{-p,p} \,|\, d_{-p+1,-p}, \ldots, d_{-p+1,p} \,|\, \cdots \,|\, d_{p,-p}, \ldots, d_{p,p}]^T. \tag{3.7}$$

The matrix A, as constructed in Algorithm 3.1 for a point $\tilde{\mathbf{x}}$ on the boundary, has the 2×2 block structure

$$A = \begin{bmatrix} A_{11} & A_{12} \\ 0 & I \end{bmatrix}. \tag{3.8}$$

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 (3.3c). The $\tilde{m}_2 \times \tilde{m}_2$ identity in the lower-right block of A is implied by the conditions in (3.3a) and (3.3b). 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 4.1 discussed later in Section 4.1). Algorithm 3.2 shows the construction of the right-hand side vector $\mathbf{b}$ which follows similar steps to that used to build A. The solution of (3.6) yields the coefficients $d_{\hat{m},\hat{n}}$ of the interpolating polynomial, and in particular

$$U_{\hat{i},\tilde{j}} = d_{\hat{i},0}, \qquad \hat{i} = -p, \ldots, -1, \tag{3.9}$$

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

3.1.1. LCBC method: Direct approach. In the direct approach to the LCBC method, the matrix A and vector $\mathbf{b}$ in (3.6) are constructed for each point on the boundary, and then the system is solved to determine ghost points following the assignments in (3.9) for example. Points on the boundary near corners require special treatment, and this is discussed in Section 3.2.

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 3.1, 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 (2.2) involving general coefficient functions, $c_{11}(\mathbf{x})$, $c_{12}(\mathbf{x})$, etc., makes this calculation increasingly messy as the order

Algorithm 3.1 Construct the coefficient matrix A for a Dirichlet boundary.

1: $r = 0$;
2: **for** $\nu = 1, \ldots, p$ **do**
3: **for** $\mu = 0, \ldots, 2p$ **do**
4: $r = r + 1$;
5: **for** $\hat{m} = -p, \ldots, p$ **do**
6: **for** $\hat{n} = -p, \ldots, p$ **do**
7: $c = (2p+1)(\hat{m}+p) + \hat{n} + p + 1$;
8: $A(r,c) = \partial_y^\mu Q^\nu L_{\hat{m}}\big((x-\tilde{x})/\Delta x\big) L_{\hat{n}}\big((y-\tilde{y})/\Delta y\big)\big|_{\mathbf{x}=\tilde{\mathbf{x}}}$; ▷ Elements of A from (3.3c)
9: **end for**
10: **end for**
11: **end for**
12: **end for**
13: **for** $\hat{i} = 0, \ldots, p$ **do**
14: **for** $\hat{j} = -p, \ldots, p$ **do**
15: $r = r + 1$;
16: $A(r,r) = 1$; ▷ Elements of A from (3.3a) and (3.3b)
17: **end for**
18: **end for**

Algorithm 3.2 Construct the right-hand side vector $\mathbf{b}$ for a Dirichlet boundary.

1: $r = 0$;
2: **for** $\nu = 1, \ldots, p$ **do**
3: **for** $\mu = 0, \ldots, 2p$ **do**
4: $r = r + 1$;
5: $b(r) = \partial_y^\mu R_{\ell,\nu}(\tilde{y}, t)$; ▷ Elements of $\mathbf{b}$ from (3.3c)
6: **end for**
7: **end for**
8: **for** $\hat{j} = -p, \ldots, p$ **do**
9: $r = r + 1$;
10: $b(r) = g_\ell(\tilde{y} + \hat{j}\Delta y, t)$; ▷ Elements of $\mathbf{b}$ from (3.3a)
11: **end for**
12: **for** $\hat{i} = 1, \ldots, p$ **do**
13: **for** $\hat{j} = -p, \ldots, p$ **do**
14: $r = r + 1$;
15: $b(r) = U_{\hat{i},\tilde{j}+\hat{j}}(t)$; ▷ Elements of $\mathbf{b}$ from (3.3b)
16: **end for**
17: **end for**

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 3.3 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 $\hat{m}, \hat{n} \in \{-p, \ldots, p\}$ 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.

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

$$V_{\hat{\mathbf{i}}}[\nu,k] \stackrel{\text{def}}{=} (Q_{d,h})^\nu L_{\hat{m}}\big(\hat{i}\big) L_{\hat{n}}\big(\hat{j}\big), \qquad \nu = 1, \ldots, p, \tag{3.10}$$

where the indices $(\hat{m}, \hat{n})$ are fixed and the order of accuracy of the approximation is $d = 2k$, $k = 1, \ldots, p+1-\nu$. 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 $Q_{d,h}$. The

Algorithm 3.3 Compute $Z_{\hat{m},\hat{n}}[\mu,\nu] \approx \partial_y^\mu Q^\nu L_{\hat{m}}\big((x-\tilde{x})/\Delta x\big) L_{\hat{n}}\big((y-\tilde{y})/\Delta y\big)\big|_{\mathbf{x}=\tilde{\mathbf{x}}}$.

1: **for** $k = 1,\ldots,p$ **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** $\nu = 0,\ldots,p-1$ **do**
7: **for** $k = 1,\ldots,p-\nu$ **do**
8: **for** $l = 1,\ldots,k-1$ **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** $m = 0,\ldots,k-l$ **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: $V_{\hat{\mathbf{i}}}[\nu+1,k] = \text{applyQh}\{V_{\hat{\mathbf{i}}}[\nu,k], W_{\hat{\mathbf{i}}}^{(m,n)}[\nu,k-1],\ldots,W_{\hat{\mathbf{i}}}^{(m,n)}[\nu,1]\}$;
17: **end for**
18: **end for**
19: **end for**
20: **for** $\nu = 1,\ldots,p$ **do** ▷ Compute $Z_{i,j}[\mu,\nu]$ using $V_{\hat{\mathbf{i}}}[\nu,k]$, $k = 1,2,\ldots,p+1-\nu$
21: $k = p+1-\nu$;
22: $Z_{\hat{m},\hat{n}}[0,\nu] = V_{0,0}[\nu,k]$
23: **for** $l = 1\ldots,p$ **do**
24: $\mu = 2l$;
25: $\{Z_{\hat{m},\hat{n}}[\mu-1,\nu], Z_{\hat{m},\hat{n}}[\mu,\nu]\} = \text{applyDy}\{V_{\hat{\mathbf{i}}}[\nu,1],\ldots,V_{\hat{\mathbf{i}}}[\nu,k]\}$;
26: **end for**
27: **end for**

domain for the local index $\hat{\mathbf{i}}$, denoted by $\hat{\Omega}_h[\nu,k]$, for each calculation is defined by

$$\hat{\Omega}_h[\nu,k] \stackrel{\text{def}}{=} [-w_x,w_x]\times[-w_y,w_y], \qquad w_x = p-(\nu+k-1), \quad w_y = w_x+p, \tag{3.11}$$

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 (3.6) are specified by Algorithm 3.2 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 (3.4). 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 (2.8) 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 3.3. 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 3.4.

It is worth noting that 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 3.4 Compute $R_{\ell,\hat{j}}[\nu,t] \approx R_{\ell,\nu}\big(\tilde{y}+\hat{j}\Delta y,t\big)$ for $q>0$

for $\nu = 1,\ldots,p$ **do**
  for $\hat{j} \in [-p,p]$ **do** ▷ Initialize $R_{\ell,\hat{j}}[\nu,t] = \partial_t^{q\nu} g_\ell(\tilde{y}+\hat{j}\Delta y,t)$
    $R_{\ell,\hat{j}}[\nu,t] = \text{applyDt}\{g_{\ell,\hat{j}}(t), q\nu\}$;
  end for
end for
for $n = 0,\ldots,p-1$ **do**
  for $k = 1,\ldots,p$ **do**
    for $\hat{\mathbf{i}} \in \hat{\Omega}_h[0,k]$ **do** ▷ Initialize $F_{\hat{\mathbf{i}}}[0,k,t] = \partial_t^{qn} f(\tilde{\mathbf{x}}+\mathbf{x}_{\hat{\mathbf{i}}},t)$
      $F_{\hat{\mathbf{i}}}[0,k,t] = \text{applyDt}\{f_{\hat{\mathbf{i}}}(t), qn\}$;
    end for
  end for
  for $\bar{\nu} = 0,\ldots,p-n-2$ **do**
    for $k = 1,\ldots,p-\bar{\nu}$ **do**
      for $l = 1,\ldots,k-1$ **do**
        for $\hat{\mathbf{i}} \in \hat{\Omega}_h[\bar{\nu},k]$ **do** ▷ Compute corrections $W_{\hat{\mathbf{i}}}^{(m,n)}[\bar{\nu},l,t]$ involving $F_{\hat{\mathbf{i}}}[\bar{\nu},l,t]$
          for $m = 0,\ldots,k-l$ **do**
            $W_{\hat{\mathbf{i}}}^{(m,(k-l)-m)}[\bar{\nu},l,t] = (D_{+x}D_{-x})^m (D_{+y}D_{-y})^{(k-l)-m} F_{\hat{\mathbf{i}}}[\bar{\nu},l,t]$;
          end for
        end for
      end for
      for $\hat{\mathbf{i}} \in \hat{\Omega}_h[\bar{\nu}+1,k]$ **do** ▷ Compute $F_{\hat{\mathbf{i}}}[\bar{\nu}+1,k,t] = \big(Q_{2k,h}\big)F_{\hat{\mathbf{i}}}[\bar{\nu},k,t]$
        $F_{\hat{\mathbf{i}}}[\bar{\nu}+1,k,t] = \text{applyQh}\{F_{\hat{\mathbf{i}}}[\bar{\nu},k,t], W_{\hat{\mathbf{i}}}^{(m,n)}[\bar{\nu},k-1,t],\ldots,W_{\hat{\mathbf{i}}}^{(m,n)}[\bar{\nu},1,t]\}$;
      end for
    end for
  end for
  for $\nu = n+1,\ldots,p$ **do**
    $\bar{\nu} = \nu-n-1$;
    $k = \min\{p+1-\bar{\nu}\,,\,p\}$;
    for $\hat{j} \in [-p,p]$ **do** ▷ Update $R_{\ell,\hat{j}}[\tilde{\nu},t]$
      $R_{\ell,\hat{j}}[\nu,t] = R_{\ell,\hat{j}}[\nu,t] - F_{0,\hat{j}}[\bar{\nu},k,t]$;
    end for
  end for
end for

Algorithm 3.4, 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.

3.1.2. LCBC method: Stencil approach. The aim of the stencil approach is to manipulate the linear system in (3.6) so that the values in the ghost points in (3.9) corresponding to a point $\tilde{\mathbf{x}}$ on the boundary can be computed using the stencil formula

$$U_{\hat{i},\tilde{j}} = \sum_{\nu=1}^{p} \sum_{j=\tilde{j}-p}^{\tilde{j}+p} \alpha_{\hat{i},\tilde{j}}^{(\nu,j)} R_{\ell,j}[\nu,t] + \sum_{i=0}^{p} \sum_{j=\tilde{j}-p}^{\tilde{j}+p} \beta_{\hat{i},\tilde{j}}^{(i,j)} U_{i,j}(t), \quad \hat{i} = -p,\ldots,-1, \tag{3.12}$$

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 (3.12) do not depend on time t and can be computed from the matrix A in (3.8). Thus, the values in the ghost points can be computed efficiently via a fixed linear combination of the relevant time-dependent data 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 $i = 1,\ldots,p$ and Dirichlet boundary data, $U_{0,j}(t) = g_\ell(y_j,t)$. Note that Algorithm 3.4 computes $R_{\ell,\hat{j}}[\nu,t]$ for values of the

local index $\hat{j}$ about $\tilde{j}$, 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 (3.12), we consider the linear system in (3.6) in the form

$$\begin{bmatrix} A_{11} & A_{12} \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \end{bmatrix} = \begin{bmatrix} D_y \mathbf{R}(t) \\ \mathbf{U}(t) \end{bmatrix}, \tag{3.13}$$

where $\mathbf{d} = \big[\mathbf{d}_1, \mathbf{d}_2\big]^T$ 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 (3.9). The lower set of $\tilde{m}_2$ equations in (3.13) implies $\mathbf{d}_2 = \mathbf{U}(t)$ so that the upper set of $\tilde{m}_1$ equations becomes

$$A_{11}\mathbf{d}_1 = D_y \mathbf{R}(t) - A_{12}\mathbf{U}(t). \tag{3.14}$$

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

$$A_{11}C_\alpha = D_y, \qquad A_{11}C_\beta = -A_{12}, \tag{3.15}$$

so that (3.14) reduces to

$$\mathbf{d}_1 = C_\alpha \mathbf{R}(t) + C_\beta \mathbf{U}(t). \tag{3.16}$$

The sets of coefficients, $\big\{\alpha_{\hat{i},\tilde{j}}^{(\nu,j)}\big\}$ and $\big\{\beta_{\hat{i},\tilde{j}}^{(i,j)}\big\}$, in the stencil formula in (3.12) 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 (3.15) are dense but not very large, e.g. A_{11} is 21×21 for $p = 3$.

3.2. 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 [1]. The physical (primary) boundary conditions are taken to be

$$u(\mathbf{x}, t) = g_\ell(y, t), \qquad \mathbf{x} \in \partial\Omega_\ell, \tag{3.17a}$$

$$u(\mathbf{x}, t) = g_b(x, t), \qquad \mathbf{x} \in \partial\Omega_b, \tag{3.17b}$$

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

$$\tilde{u}\big(\hat{i}\Delta x, \hat{j}\Delta y\big) = U_{\hat{i},\hat{j}}(t), \qquad \hat{i} = 1, \ldots, p, \quad \hat{j} = 1, \ldots, p. \tag{3.18a}$$

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

$$\left.\begin{aligned} \partial_y^\mu \tilde{u}(0, 0) &= \partial_y^\mu g_\ell(0, t) \\ \partial_x^\mu \tilde{u}(0, 0) &= \partial_x^\mu g_b(0, t) \end{aligned}\right\} \qquad \mu \in \mathcal{M}_0, \tag{3.18b}$$

and

$$\left.\begin{aligned} \partial_y^\mu Q^\nu \tilde{u}(0, 0) &= \partial_y^\mu R_{\ell,\nu}(0, t) \\ \partial_x^\mu Q^\nu \tilde{u}(0, 0) &= \partial_x^\mu R_{b,\nu}(0, t) \end{aligned}\right\} \qquad \nu = 1, \ldots, p, \quad \mu \in \mathcal{M}_\nu, \tag{3.18c}$$

respectively, where $R_{\ell,\nu}(y,t)$ is defined in (3.4) and $R_{b,\nu}(x,t)$ is defined by

$$R_{b,\nu}(x,t) \stackrel{\text{def}}{=} \partial_t^{q\nu} g_b(x,t) - \Psi_\nu f(x,0,t). \tag{3.19}$$

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

(3.20)
$$\mathcal{M}_\nu = \begin{cases} 0,1,2,3,\ldots,2p-1,2p, & \text{if } \nu=0, \text{ with an average for } \mu=0,\\ 1,2,3,4,\ldots,2p-1,2p, & \text{if } \nu=1, \text{ with an average for } \mu=2,\\ 1,\quad 3,4,5,\ldots,2p-1,2p, & \text{if } \nu=2, \text{ with an average for } \mu=4,\\ \quad\vdots & \quad\vdots\\ 1,\quad 3,\quad 5,\ldots,2p-1,2p, & \text{if } \nu=p, \text{ with an average for } \mu=2p. \end{cases}$$

Note that there is one value for μ in each set $\mathcal{M}_\nu$ where the pairs in (3.18b) and (3.18c) 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. Ghost points near the corner can be obtained from the solution of the linear system implied by (3.18) following a direct approach, or these ghost points can be written in terms of the stencil formula

$$U_{\hat{i},\hat{j}} = \sum_{\nu=0}^{p}\sum_{j=-p}^{p} \tilde{\alpha}_{\hat{i},\hat{j}}^{(\nu,j)} R_{\ell,j}[\nu,t] + \sum_{\nu=0}^{p}\sum_{i=-p}^{p} \tilde{\beta}_{\hat{i},\hat{j}}^{(\nu,i)} R_{b,i}[\nu,t] + \sum_{i=1}^{p}\sum_{j=1}^{p} \tilde{\gamma}_{\hat{i},\hat{j}}^{(i,j)} U_{i,j}(t), \tag{3.21}$$

where $\hat{\Omega}_c \stackrel{\text{def}}{=} \left\{ \hat{\mathbf{i}} = (\hat{i},\hat{j}) \,|\, -p \leqslant (\hat{i},\hat{j}) < p \setminus 1 \leqslant (\hat{i},\hat{j}) < p \right\}$ defines the set of local indices for the ghost-point values in (3.21). The time-dependent data $R_{\ell,j}[\nu,t]$ and $R_{b,i}[\nu,t]$ in (3.21) are discrete approximations of $R_{\ell,\nu}(j\Delta y,t)$ and $R_{b,\nu}(i\Delta x,t)$, respectively, for $\nu = 1,\ldots,p$. The boundary conditions are specified in (3.21) by setting

$$R_{\ell,j}[0,t] = g_\ell(j\Delta y,t), \qquad j=-p,\ldots,p, \tag{3.22a}$$
$$R_{b,i}[0,t] = g_b(i\Delta x,t), \qquad i=-p,\ldots,p, \tag{3.22b}$$

similar to previous specifications. The coefficients in the stencil formula are derived from the $\tilde{m}\times\tilde{m}$ linear system implied by (3.18) following the analysis described for the Dirichlet boundary.

Our choice for the constraints in (3.18) is guided by the case when Q in (2.2) 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 $|c_{12}(\mathbf{x})|/\sqrt{c_{11}(\mathbf{x})c_{22}(\mathbf{x})}$ is small, for $\mathbf{x}$ in a neighborhood of the corner. Should these conditions be violated, the $\tilde{m}\times\tilde{m}$ matrix A implied by (3.18) 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

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

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 IBVP in (2.1). For many problems of interest, this problem is obtained by an orthogonal, or near-orthogonal, mapping of a PDE in physical space involving the Laplacian operator. The resulting mapped problem 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 (3.18) for a Dirichlet-Dirichlet corner. The matrices for the Neumann-Neumann and Dirichlet-Neumann corners are also nonsingular under these conditions, see Theorem 4.2.

4. 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.

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

$$Q = c_{11}\partial_x^2 + 2c_{12}\partial_x\partial_y + c_{22}\partial_y^2 + c_1\partial_x + c_2\partial_y + c_0. \tag{4.1}$$

For this operator, we have the following result:

THEOREM 4.1 (Solvability on a face). *The matrix resulting from the order $2p = 2, 4, 6$ LCBC constraints for the constant-coefficient operator Q in (4.1) with a Dirichlet or Neumann boundary condition on a grid face is nonsingular 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 nonsingular for any Δx and Δy.*

Proof. Let us focus on the left boundary, while similar arguments hold for the other boundaries. For either a Dirichlet or Neumann boundary, the determinant of A, for order of accuracy $2p = 2, 4, 6$, has the form

$$\det(A) = K_p G_p(\xi), \qquad \xi = \frac{c_1 \Delta x}{c_{11}}, \qquad p = 1, 2, 3, \tag{4.2}$$

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

$$G_1(\xi) = \left(1 - \frac{\xi}{2}\right)^3, \quad G_2(\xi) = \left(1 - \frac{3\xi}{2} + \frac{\xi^2}{2} - \frac{\xi^3}{18}\right)^5,$$

$$G_3(\xi) = \left(1 - 3\xi + \frac{11\xi^2}{4} - \frac{1691\xi^3}{1440} + \frac{121\xi^4}{480} - \frac{11\xi^5}{400} + \frac{\xi^6}{800}\right)^7.$$

The forms of G_p for the Neumann case can be found in [1]. The result of the theorem follows from the form of the determinant of A in (4.2). As expected, the lower order terms in (4.1) 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 (4.1) 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

$\gamma = \frac{c_{12}}{\sqrt{c_{11}c_{22}}}$, and $\sigma = \sqrt{\frac{c_{11}/\Delta x^2}{c_{22}/\Delta y^2}}$, 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 4.2 (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 (4.1) 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. For D-D, N-N and D-N corners, the determinant of A has the form

$$\det(A) = K_p H_p(\gamma) F_p(\gamma, \sigma), \qquad p = 1, 2, 3, \tag{4.3}$$

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

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

and

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

The corresponding formulae for the N-N corner and D-N corner are given in [1]. 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 (4.3). ∎

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.

4.2. 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 4.3 (Symmetry on a face). *When applied to the operator $Q = \Delta$ on a Cartesian grid, the LCBC approach on a face, at any order $2p = 2, 4, 6, \ldots$, 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,*

$$U_{i-\hat{i},j} = -U_{i+\hat{i},j}, \quad \hat{i} = 1, \ldots, p, \qquad \textit{Dirichlet BC at } i = 0 \textit{ or } i = N_x, \tag{4.4a}$$

$$U_{i-\hat{i},j} = U_{i+\hat{i},j}, \quad \hat{i} = 1, \ldots, p, \qquad \textit{Neumann BC at } i = 0 \textit{ or } i = N_x. \tag{4.4b}$$

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

$$\tilde{u}(x, y) = \sum_{n=0}^{2p} \sum_{m=0}^{2p} a_{n,m}\, x^n\, y^m, \qquad p = 1, 2, \ldots. \tag{4.5}$$

We wish to show that $\tilde{u}(x, 0)$ is an odd function in x, so that $\tilde{u}(x, 0) = a_{1,0}x + a_{3,0}x^3 + \ldots a_{2p-1,0}x^{2p-1}$, for then we have $\tilde{u}(-x, 0) = -\tilde{u}(x, 0)$ and the desired result follows. The CBCs in (3.3c) reduce to

$$\partial_y^\mu \Delta^\nu \tilde{u}(\mathbf{0}) = 0, \qquad \nu = 0, \ldots, p, \quad \mu = 0, \ldots, 2p, \quad \text{CBC}[\mu, \nu], \tag{4.6}$$

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 (4.6) as CBC$[\mu, \nu]$. We will show that (4.6) implies that all even x-derivatives of $\tilde{u}$ at $\mathbf{x} = \mathbf{0}$ are zero,

$$\partial_x^{2\nu} \tilde{u}(\mathbf{0}) = 0, \qquad \nu = 0, \ldots, p, \tag{4.7}$$

which implies that $\tilde{u}(x, 0)$ is an odd function in x. The conditions in (4.7) can be shown as follows. We have $\partial_y^\mu \tilde{u}(\mathbf{0}) = 0$, for $\mu = 0, 1, \ldots$, since the Dirichlet conditions are homogeneous and since $\tilde{u}$ is a polynomial of finite degree. Then, from CBC$[0, 1]$, we see that (4.7) holds for $\nu = 1$ since $\partial_x^2 \tilde{u}(\mathbf{0}) = -\partial_y^2 \tilde{u}(\mathbf{0}) = 0$, and from CBC$[\mu, 1]$ we also find $\partial_y^\mu \partial_x^2 \tilde{u}(\mathbf{0}) = -\partial_y^{\mu+2} \tilde{u}(\mathbf{0}) = 0$, for $\mu = 0, 1, \ldots$. Now from CBC$[0, 2]$, we find that (4.7) holds for $\nu = 2$, since $\partial_x^4 \tilde{u}(\mathbf{0}) = (-2\partial_x^2 \partial_y^2 - \partial_y^4)\tilde{u}(\mathbf{0}) = 0$, and from CBC$[\mu, 2]$ we also find $\partial_y^\mu \partial_x^4 \tilde{u}(\mathbf{0}) = 0$, for $\mu = 0, 1, \ldots$. The process can be repeated to show (4.7).

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, $\partial_x^{2\nu+1} \tilde{u}(\mathbf{0}) = 0$, for $\nu = 0, \ldots, p$, 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 3.1 for the case $p = 2$ for example.

THEOREM 4.4 (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 $2p = 2, 4, 6, \ldots$, 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 (4.4). 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*

$$U_{i-\hat{\imath},j-\hat{\jmath}} = U_{i+\hat{\imath},j+\hat{\jmath}}, \quad \hat{\imath},\hat{\jmath} = 1,\ldots,p, \qquad \textit{D-D or N-N corners}, \tag{4.8a}$$

$$U_{i-\hat{\imath},j-\hat{\jmath}} = -U_{i+\hat{\imath},j+\hat{\jmath}}, \quad \hat{\imath},\hat{\jmath} = 1,\ldots,p, \qquad \textit{D-N corner}. \tag{4.8b}$$

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 (4.5) we show that

$$\tilde{u}(-x,y) = -\tilde{u}(x,y), \qquad \tilde{u}(x,-y) = -\tilde{u}(x,y), \tag{4.9}$$

and thus $\tilde{u}(-x,-y) = \tilde{u}(x,y)$. To show (4.9), we show

$$\partial_x^{m_1}\partial_y^{m_2}\tilde{u}(\mathbf{0}) = 0, \qquad m_1 = 2k, \quad m_2 = 0,1,\ldots,2p, \tag{4.10a}$$

$$\partial_x^{m_1}\partial_y^{m_2}\tilde{u}(\mathbf{0}) = 0, \qquad m_1 = 0,1,\ldots,2p, \quad m_2 = 2k, \tag{4.10b}$$

where $k = 0,1,\ldots,p$. Recall that $\tilde{u}$ satisfies the boundary conditions in (3.18b) and the compatibility conditions in (3.18c) with homogeneous boundary data, so that

$$\left.\begin{aligned} \partial_y^{\mu}\Delta^{\nu}\tilde{u}(\mathbf{0}) &= 0 \\ \partial_x^{\mu}\Delta^{\nu}\tilde{u}(\mathbf{0}) &= 0 \end{aligned}\right\} \qquad \nu = 0,1,\ldots,p, \quad \mu \in \mathcal{M}_{\nu}, \tag{4.11}$$

where $\mathcal{M}_{\nu}$ is defined in (3.20). Using mathematical induction, we find that (4.11) implies

$$\left.\begin{aligned} \partial_y^{\mu}\partial_x^{2\nu}\tilde{u}(\mathbf{0}) &= 0 \\ \partial_x^{\mu}\partial_y^{2\nu}\tilde{u}(\mathbf{0}) &= 0 \end{aligned}\right\} \qquad \nu = 0,1,\ldots,p, \quad \mu \in \mathcal{M}_{\nu}. \tag{4.12}$$

Set $m_1 = 2k$ for $k = 0,1,\ldots,p$. The first set of conditions in (4.12) implies that

$$\partial_x^{m_1}\partial_y^{m_2}\tilde{u}(\mathbf{0}) = 0, \qquad \text{for } m_2 = 1,3,5,\ldots,2k-1,2k,2k+1,\ldots,2p, \tag{4.13}$$

while the second set of conditions in (4.12) gives

$$\partial_x^{m_1}\partial_y^{m_2}\tilde{u}(\mathbf{0}) = 0, \qquad \text{for } m_2 = 0,2,4,\ldots,2k. \tag{4.14}$$

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

$$\partial_x^{m_1}\partial_y^{m_2}\tilde{u}(\mathbf{0}) = 0, \qquad \text{for } m_2 = 0,1,2,3,\ldots,2p, \tag{4.15}$$

for any $k = 0,1,\ldots,p$. The result in (4.10b) follows using a symmetric argument. Therefore, we have odd symmetry on the Dirichlet side near the corner and even symmetry at the D-D corner. The results for N-N and D-N corners follow using similar arguments. ∎

4.3. 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 $\partial_t^2 u = c^2 \Delta u$ on a Cartesian grid using the LCBC approach at the boundary. The ME time-stepping schemes are given in [1]. In [3] it was shown that an ME scheme for the wave equation in one space dimension is stable at any order of accuracy, $2p = 2, 4, 6, \ldots$, 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

$$c^2 \Delta t^2 \Big(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} \Big) < 1, \tag{4.16}$$

for orders of accuracy $2p = 2, 4, 6$, as given by Theorem 4.5 discussed below. We call this version the *compact* ME scheme, and we conjecture that the condition in (4.16) holds at any even order $2p = 2, 4, 6, \ldots$ (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 4.5 (Stability of approximations for the wave equation). *The IBVP in (2.1) 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 (4.16).*

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 boundary 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

$$W_{\mathbf{i}}^n = A^n \, \kappa_x^i \, \kappa_y^j, \tag{4.17}$$

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

$$W_{\mathbf{i}}^n = A_{\pm,\mathbf{k}}^n \sin\Big(\frac{\pi k_x}{L_x} x_i \Big) \cos\Big(\frac{\pi k_y}{L_y} y_j \Big), \quad k_x = 1, \ldots, N_x - 1, \quad k_y = 0, 1, \ldots, N_y,$$

where $A_{\pm,\mathbf{k}}$ are two possible values for the amplification factor (see below) and $\mathbf{k} = (k_x, k_y)$. For stability we choose Δt so that $|A_{\pm,\mathbf{k}}| \leqslant 1$ for all valid k_x and k_y. It is straightforward to find the symbols of $D_{+x} D_{-x}$ and $D_{+y} D_{-y}$, given by $D_{+x} D_{-x} \sin\big(\frac{\pi k_x x_i}{L_x}\big) = -\hat{k}_x^2 \sin\big(\frac{\pi k_x x_i}{L_x}\big)$, and $D_{+y} D_{-y} \cos\big(\frac{\pi k_y y_j}{L_y}\big) = -\hat{k}_y^2 \cos\big(\frac{\pi k_y y_j}{L_y}\big)$,

where $\hat{k}_x \stackrel{\text{def}}{=} \frac{\sin(\xi_x/2)}{\Delta x/2}$, $\hat{k}_y \stackrel{\text{def}}{=} \frac{\sin(\xi_y/2)}{\Delta y/2}$, $\xi_x \stackrel{\text{def}}{=} \frac{\pi k_x}{L_x}\Delta x$, and $\xi_y \stackrel{\text{def}}{=} \frac{\pi k_y}{L_y}\Delta y$. Substituting (4.17) into the ME time-stepping schemes for the different orders of accuracy, determined by p, leads to a quadratic equation for A,

$$A^2 - 2b_p A + 1 = 0, \qquad p = 1, 2, 3, \tag{4.18}$$

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 $p = 1$, $b_1 = 1 - 2\big(\hat{\lambda}_x^2 + \hat{\lambda}_y^2\big)$ where $\hat{\lambda}_x \stackrel{\text{def}}{=} c\Delta t \frac{\hat{k}_x}{2}$, $\hat{\lambda}_y \stackrel{\text{def}}{=} c\Delta t \frac{\hat{k}_y}{2}$, with $|\hat{\lambda}_x| \leqslant \frac{c\Delta t}{\Delta x}$, $|\hat{\lambda}_y| \leqslant \frac{c\Delta t}{\Delta y}$. Note that $b_1 < 1$ is clearly satisfied, while the condition $b_1 > -1$ implies $\max_{\{k_x,k_y\}} \big(\hat{\lambda}_x^2 + \hat{\lambda}_y^2\big) < 1$, and this implies the time-step restriction in (4.16).

For $p = 2$, $b_2 = 1 - 2\Big(\hat{\lambda}_x^2 + \hat{\lambda}_y^2 + \frac{\Delta x^2}{12}\hat{\lambda}_x^2\,\hat{k}_x^2 + \frac{\Delta y^2}{12}\hat{\lambda}_y^2\,\hat{k}_y^2\Big) + \frac{2}{3}\Big(\hat{\lambda}_x^2 + \hat{\lambda}_y^2\Big)^2$. From (4.18), we find $A_\pm = b_2 \pm \sqrt{b_2^2 - 1}$. For each $\lambda_x \stackrel{\text{def}}{=} c\Delta t/\Delta x$ and $\lambda_y \stackrel{\text{def}}{=} c\Delta t/\Delta y$, we define $A_{\max}(\lambda_x, \lambda_y) = \max_{k_x,k_y}\{|A_+|, |A_-|\}$, and find the region in the (λ_x, λ_y) plane where $A_{\max} \leqslant 1$. We repeat this procedure for the sixth-order accurate scheme $2p = 6$. Figure 4.1 shows that the stability region, $A_{\max} \leqslant 1$, 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 (4.16) when $p = 2$ and 3. In [1] we provide an analytical proof for the stability results observed in Figure 4.1 when $p = 2$. ∎

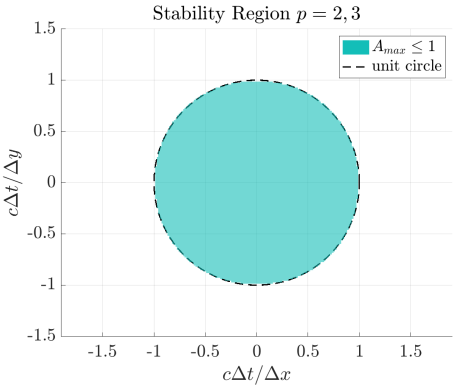

FIGURE 4.1. *Stability region of the fourth-order and sixth-order accurate ME time-stepping schemes for the wave equation on a Cartesian grid using the LCBC approach.*

5. Numerical results. We restrict our numerical results to two representative examples; more extensive numerical results are found in the ArXiv version of this article [1].

We first consider the scattering of a plane incident wave $u_{\text{inc}}(\mathbf{x}, t) = \cos[k(x - ct)]$ from a cylinder of radius one. We solve the wave equation to orders $2p = 2, 4, 6$ using the modified equation (ME) approach. For testing the LCBC method for a problem with corners, we solve on a domain covering one half the cylinder and use Neumann boundary conditions on the axis of symmetry and Dirichlet boundary conditions on the other boundaries. Results are shown in Figure 5.1 where it is seen that the schemes all achieve their design order of accuracy. The error is seen to be smooth up

to the boundaries which is a good way to assess the quality of the numerical boundary conditions.

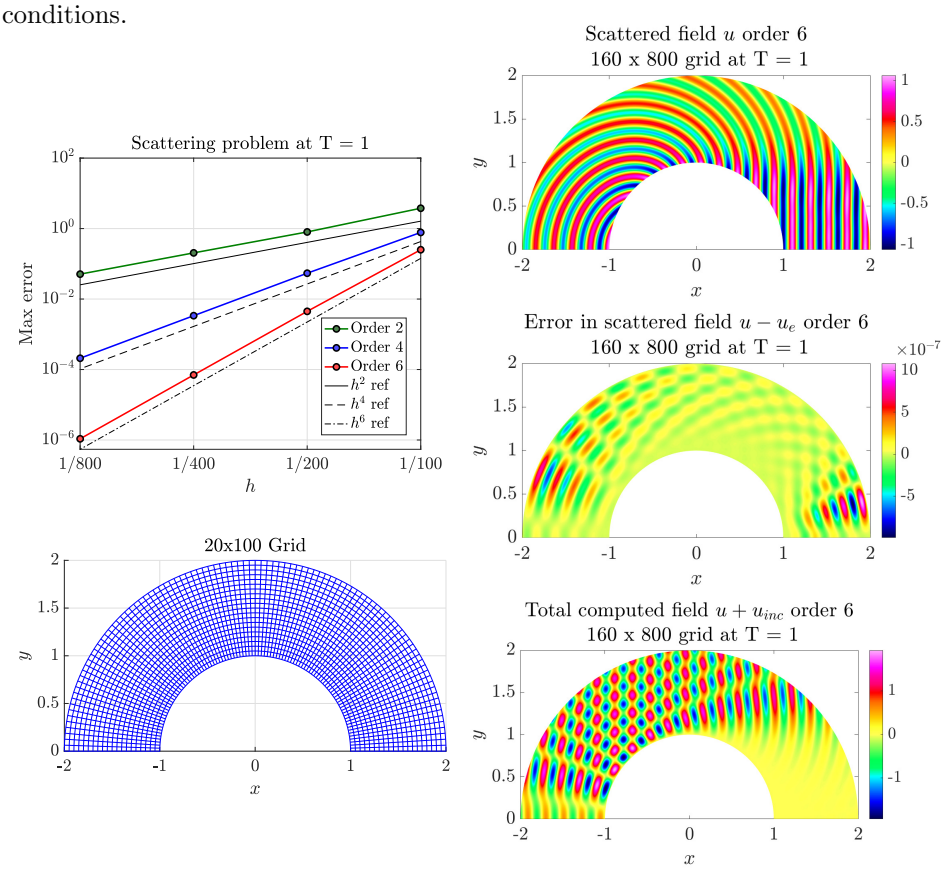

FIGURE 5.1. *Plane-wave scattering from a cylinder. Maximum errors at $T = 1$ for solutions computed using explicit ME time-stepping schemes with $d = 2$, 4 and 6 (upper left) and the coarsest grid for $h = 1/100$ (lower left). Right column shows the scattered field (top), error in the scattered field (middle) and the total field (bottom) at $T = 1$ computed using the sixth-order accurate scheme on the finest grid.*

For the second example, we solve the IBVP for an advection-diffusion problem given by

$$\begin{cases} u_t = \mathcal{D}\Delta u - \mathbf{v}\cdot\nabla u + \gamma u, & \mathbf{x}\in\Omega^P, \quad t\in(0,T], \\ u(\mathbf{x},t) = g(\mathbf{x},t), & \mathbf{x}\in\partial\Omega^P, \\ u(\mathbf{x},0) = u_0(\mathbf{x}), & \mathbf{x}\in\bar{\Omega}^P, \end{cases} \tag{5.1}$$

where $\mathcal{D}$ is a diffusivity, $\mathbf{v}$ is a convection velocity and γ is a reaction rate, all taken to be constants. The domain and non-orthogonal grid are shown in Figure 5.2. This problem is solved with the LCBC approach using Backward Differentiation Formula (BDF) time-stepping. The boundary values are set according to an exact solution given in [1]. Results, given in Figure 5.2, show that the LCBC-based schemes give the design order of accuracy and the errors are again smooth up to the boundaries.

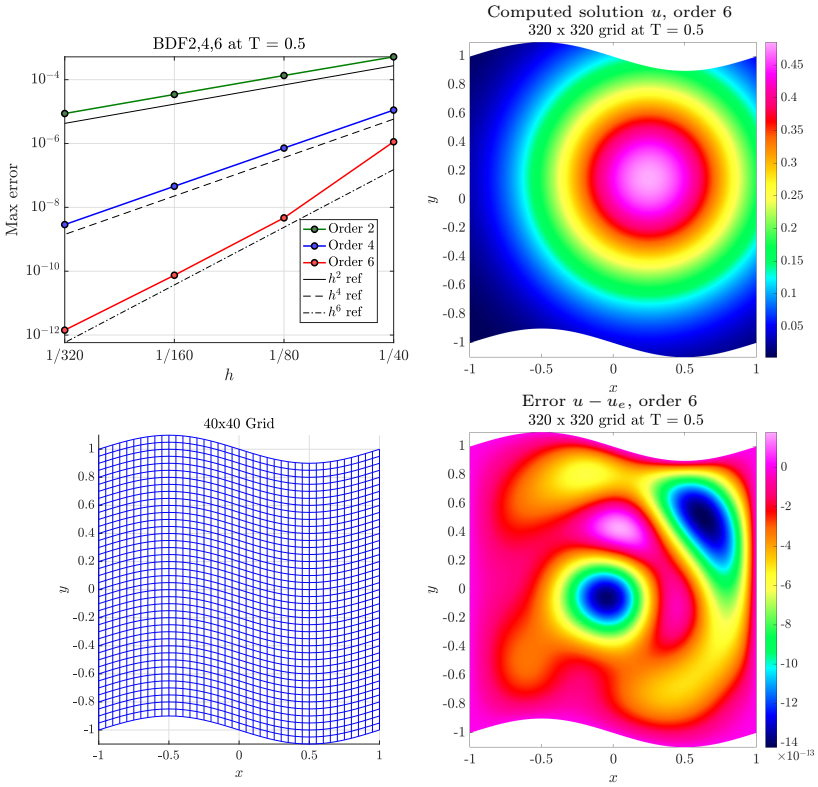

FIGURE 5.2. *Heat flow in a wavy channel. Maximum errors at $T = 0.5$ for solutions computed using BDF time-stepping schemes with $d = 2$, 4 and 6 (upper left) and the coarsest grid for $h = 1/40$ (lower left). Right column shows the temperature (top) and its error (bottom) at $T = 0.5$ computed using the sixth-order accurate scheme on the finest grid.*

6. Conclusions. We have described a new approach for constructing numerical approximations to boundary conditions for high-order accurate finite difference approximations. The local compatibility boundary condition (LCBC) approach was developed for general initial-boundary-value problems for second-order scalar PDEs. The LCBC approach uses compatibility boundary conditions and a local polynomial approximation on the boundary. Algorithms have been given for computing the local LCBC polynomial as well as for forming the discrete stencil approximations that can be used to efficiently assign ghost point values. The LCBC approach at corners has also been described. Numerical results were presented in two dimensions that demonstrate the accuracy and stability of the approach.

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.

REFERENCES

[1] N. G. Al Hassanieh, J. W. Banks, W. D. Henshaw, and D. W. Schwendeman, *Local compatibility boundary conditions for high-order accurate finite-difference approximations of PDEs*, preprint arXiv:2111.02915, 2021.

[2] J. Angel, J. W. Banks, W. D. Henshaw, M. J. Jenkinson, A. V. Kildishev, G. Kovačič, L. J. Prokopeva, and D. W. Schwendeman, *A high-order accurate scheme for Maxwell's equations with a generalized dispersion model*, J. Comput. Phys., 378 (2019), pp. 411–444.

[3] L. Anné, P. Joly, and Q. H. Tran, *Construction and analysis of higher order finite difference schemes for the 1D wave equation*, Computational Geosciences, 4 (2000), pp. 207–249.

[4] D. Appelö, J. W. Banks, W. D. Henshaw, and D. W. Schwendeman, *Numerical methods for solid mechanics on overlapping grids: Linear elasticity*, J. Comput. Phys., 231 (2012), pp. 6012–6050.

[5] D. Appelö and N. A. Petersson, *A fourth-order accurate embedded boundary method for the wave equation*, SIAM Journal on Scientific Computing, 34 (2012), pp. A2982–A3008.

[6] J. W. Banks, B. Buckner, W. D. Henshaw, M. J. Jenkinson, A. V. Kildishev, G. Kovačič, L. J. Prokopeva, and 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), p. 109424.

[7] J. W. Banks and T. Hagstrom, *On Galerkin difference methods*, J. Comput. Phys., 313 (2016), pp. 310–327.

[8] J. W. Banks, T. Hagstrom, and J. Jacangelo, *Galerkin differences for acoustic and elastic wave equations in two space dimensions*, J. Comput. Phys., 372 (2018), pp. 864–892.

[9] J. W. Banks, W. D. Henshaw, A. Kapila, and D. W. Schwendeman, *An added-mass partitioned algorithm for fluid-structure interactions of compressible fluids and nonlinear solids*, J. Comput. Phys., 305 (2016), pp. 1037–1064.

[10] J. W. Banks, W. D. Henshaw, and 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), pp. 108–137.

[11] J. W. Banks, W. D. Henshaw, and 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), pp. 399–416.

[12] J. W. Banks, W. D. Henshaw, D. W. Schwendeman, and Q. Tang, *A stable partitioned FSI algorithm for rigid bodies and incompressible flow. Part I: Model problem analysis*, J. Comput. Phys., 343 (2017), pp. 432–468.

[13] J. W. Banks, W. D. Henshaw, D. W. Schwendeman, and Q. Tang, *A stable partitioned FSI algorithm for rigid bodies and incompressible flow. Part II: General formulation*, J. Comput. Phys., 343 (2017), pp. 469–500.

[14] J. W. Banks, W. D. Henshaw, D. W. Schwendeman, and Q. Tang, *A stable partitioned FSI algorithm for rigid bodies and incompressible flow in three dimensions*, J. Comput. Phys., 373 (2018), pp. 455–492.

[15] R. B. d. R. Borges, N. D. P. da Silva, F. A. A. Gomes, C.-W. Shu, and S. Tan, *A Sequel of Inverse Lax–Wendroff High Order Wall Boundary Treatment for Conservation Laws*, Archives of Computational Methods in Engineering, (2020).

[16] B. Gutsafsson, *High Order Difference Methods for Time Dependent PDE*, Springer Series in Computational Mathematics, Springer, 2008.

[17] W. D. Henshaw, *A fourth-order accurate method for the incompressible Navier-Stokes equations on overlapping grids*, J. Comput. Phys., 113 (1994), pp. 13–25.

[18] W. D. Henshaw, *A high-order accurate parallel solver for Maxwell's equations on overlapping grids*, SIAM J. Sci. Comput., 28 (2006), pp. 1730–1765.

[19] W. D. Henshaw and K. K. Chand, *A composite grid solver for conjugate heat transfer in fluid-structure systems*, J. Comput. Phys., 228 (2009), pp. 3708–3741.

[20] W. D. Henshaw, H.-O. Kreiss, and L. G. M. Reyna, *A fourth-order accurate difference approximation for the incompressible Navier-Stokes equations*, Comput. Fluids, 23 (1994), pp. 575–593.

[21] W. D. Henshaw and D. W. Schwendeman, *Moving overlapping grids with adaptive mesh refinement for high-speed reactive and non-reactive flow*, J. Comput. Phys., 216 (2006), pp. 744–779.

[22] J. S. Hesthaven and T. Warburton, *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications.*, Springer Verlag, New York, 2008.

[23] T. J. Hughes, J. A. Evans, and A. Reali, *Finite element and NURBS approximations of eigenvalue, boundary-value, and initial-value problems*, Comput. Methods Appl. Mech. Engrg., 272 (2014), pp. 290–320.

[24] T. R. Hughes, *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*, Dover Publications, New York, 2000.

[25] J. J. Jacangelo, J. W. Banks, and T. Hagstrom, *Galerkin differences for high-order partial differential equations*, SIAM J. Sci. Comput., 412 (2020), pp. B447–B471.

[26] H.-O. Kreiss and N. A. Petersson, *A second order accurate embedded boundary method for the wave equation with Dirichlet data*, SIAM Journal on Scientific Computing, 27 (2006), pp. 1141–1167.

[27] R. J. LeVeque and D. Calhoun, *Cartesian grid methods for fluid flow in complex geometries*, in Computational Modeling in Biological Fluid Dynamics, L. J. Fauci and S. Gueron, eds., vol. 124 of IMA Volumes in Mathematics and its Applications, Springer-Verlag, 2001, pp. 117–143.
[28] R. J. LeVeque and Z. Li, *The immersed interface method for elliptic equations with discontinuous coefficients and singular sources*, SIAM Journal on Numerical Analysis, 31 (1994), pp. 1019–1044.
[29] R. J. LeVeque and Z. Li, *Immersed interface methods for Stokes flow with elastic boundaries or surface tension*, SIAM Journal on Scientific Computing, 18 (1997), pp. 709–735.
[30] T. Li, C.-W. Shu, and 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 (2017), pp. 576–607.
[31] T. Li, C.-W. Shu, and 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 (2017), pp. 576–607.
[32] J. Lu, J. Fang, S. Tan, C.-W. Shu, and M. Zhang, *Inverse Lax–Wendroff procedure for numerical boundary conditions of convection–diffusion equations*, Journal of Computational Physics, 317 (2016), pp. 276–300.
[33] J. Lu, C.-W. Shu, S. Tan, and M. Zhang, *An inverse Lax-Wendroff procedure for hyperbolic conservation laws with changing wind direction on the boundary*, Journal of Computational Physics, 426 (2021), p. 109940.
[34] F. Meng, J. W. Banks, W. D. Henshaw, and D. W. Schwendeman, *A stable and accurate partitioned algorithm for conjugate heat transfer*, J. Comput. Phys., 344 (2017), pp. 51–85.
[35] F. Meng, J. W. Banks, W. D. Henshaw, and 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), p. 113040.
[36] S. Nilsson, N. A. Petersson, B. Sjögreen, and H.-O. Kreiss, *Stable difference approximations for the elastic wave equation in second order formulation*, SIAM Journal on Numerical Analysis, 45 (2007), pp. 1902–1936.
[37] D. A. Serino, J. W. Banks, W. D. Henshaw, and D. W. Schwendeman, *A stable added-mass partitioned (AMP) algorithm for elastic solids and incompressible flow*, J. Comput. Phys., 399 (2019), pp. 1–30.
[38] D. A. Serino, J. W. Banks, W. D. Henshaw, and D. W. Schwendeman, *A stable added-mass partitioned (AMP) algorithm for elastic solids and incompressible flow: Model problem analysis*, SIAM J. Sci. Comput., 41 (2019), pp. A2464–A2484.
[39] C.-W. Shu and S. Tan, *Chapter 2 - inverse Lax–Wendroff procedure for numerical boundary treatment of hyperbolic equations*, in Handbook of Numerical Methods for Hyperbolic Problems, R. Abgrall and C.-W. Shu, eds., vol. 18 of Handbook of Numerical Analysis, Elsevier, 2017, pp. 23–52.
[40] B. Sjögreen and N. A. Petersson, *A fourth order finite difference scheme for the elastic wave equation in second order formulation*, Scient. Comput., 52 (2012), pp. 17–48.
[41] G. Strang and G. Fix, *An Analysis of the Finite Element Method 2nd Edition*, Wellesley-Cambridge, 2nd ed., 2008.
[42] S. Tan and C.-W. Shu, *Inverse Lax-Wendroff procedure for numerical boundary conditions of conservation laws*, Journal of Computational Physics, 229 (2010), pp. 8144–8166.