High-order dimensionally-split Cartesian embedded boundary method for non-dissipative schemes

Brady, Peter T.
Livescu, Daniel
Sharan, Nek

To be published in: Journal of Computational Physics

DOI to publisher's version: 10.1016/j.jcp.2022.111341

High-order dimensionally-split Cartesian embedded boundary method for non-dissipative schemes

Nek Sharan[a], Peter T. Brady[b], and Daniel Livescu[b]

[a] *Department of Aerospace Engineering, Auburn University, Auburn, AL 36849, USA*
[b] *CCS-2, Los Alamos National Laboratory, Los Alamos, NM 87545, USA*

Abstract

Centered finite-difference schemes are commonly used for high-fidelity turbulent flow simulations in canonical configurations because of their non-dissipative property and computational efficiency. However, their use in flow simulations over complex geometries is limited by the requirements of a structured grid and a stable boundary treatment in the absence of artificial (numerical) dissipation. Cartesian embedded boundary (EB) approaches provide an efficient structured-grid framework to apply difference schemes over complex domains. However, they are often restricted to low orders of accuracy because of numerical instabilities at the embedded boundaries and the issues of small-cell problem that are difficult to address with high-order accuracy. The present work discusses a systematic approach to obtain high-order EB methods with non-dissipative centered schemes in the interior. This approach, based on satisfying the primary and secondary conservation conditions, is employed to derive EB schemes that are up to sixth-order accurate in the interior and fourth-order accurate globally for hyperbolic, parabolic as well as incompletely parabolic problems. The proposed finite-difference discretization is, by construction, dimensionally split and addresses the small-cell problem without any cell/geometry transformations, thus, highly simplifying implementation in a flow solver. Various linear and non-linear numerical tests are performed to evaluate the stability and the accuracy of the proposed EB schemes.

Keywords:
embedded boundary method, conservative schemes, high order, finite-difference schemes

1. Introduction

Small-scale fluctuations play a crucial role in applications of flow-generated noise [1, 2, 3] and turbulent mixing [4, 5, 6]. Retaining energy at the small scales is thus important in numerical simulations of such applications, and hence energy-conservative (non-dissipative) centered schemes are often preferred for discretization of convective terms in unsteady turbulence simulations [7, 8]. Use of centered schemes in a finite-difference framework simplifies code implementation and improves computational efficiency by providing a dimensionally-split discretization over structured grids. Structured grids are well suited for simple computational domains, however, flow simulations over complex geometries require unstructured grids [9] or the use of immersed boundary methods (IBMs) [10] over structured grids.

Cartesian embedded boundary (EB) methods are a class of IBMs that allow sharp representation of solid boundaries by simply cutting out the solid body from a Cartesian fluid grid [11, 12], as illustrated in figure 1. The fluid and the solid domain, denoted by Ω_f and Ω_s, respectively, intersect at the embedded boundary, labeled Γ in figure 1(a). Γ creates non-uniform cells at the EB and the challenge with the EB methods is to discretize the governing equations in those cells and to apply the boundary conditions at the EB in an

Email address: nsharan@auburn.edu (Nek Sharan)

accurate, stable, and conservative manner. Issues that make those steps challenging, especially with non-dissipative convection schemes, include: a) numerical instabilities arising at the EB in long-time fluid-flow simulations, b) small-cell problems that affect the time integration step sizes and the overall accuracy of the scheme, and c) conservation losses in addressing the previous two issues.

The sharp interface IBMs, referred to here as the EB methods, have also been called in the literature as immersed interface methods (IIMs) or cut-cell methods based on the interface treatment and the discretization. IIMs, as discussed in [13, 14, 15], use a finite-difference (FD) discretization with interface treatments involving the jump conditions for the solution (and, in some cases, the solution derivatives) at the interface. In the IIMs, the jump conditions are incorporated into the stencil coefficients allowing computations on both sides of the interface. Cut-cell methods commonly use a finite-volume (FV) framework (e.g. [16, 17, 18, 19, 20, 21, 22, 23, 24]) where no computations are performed on the solid side. A FV discretization requires geometry and solution reconstructions to approximate the embedded boundary surface area and the fluxes on cell faces. For example, in figure 1(c), time advancement of the solution in cut cell $\mathcal{ABCDE}$ (shaded red) using a FV scheme requires an estimate of the embedded boundary (denoted $\mathcal{CD}$) face area as well as approximations of solution integrals on cell faces, where the integrals on irregular faces ($\mathcal{BC}$ and $\mathcal{DE}$) require high-order solution reconstructions for accuracies of second order or higher. In addition, procedures such as cell mixing [18], cell merging [17], cell linking [19] or flux redistribution [25] are required to address the small-cell problem that arises when a cut cell becomes small relative to the regular cells, e.g. the cell $\mathcal{EDF}$ in figure 1(c). The complexity of these procedures often restricts the global order-of-accuracy of the cut-cell methods to second order.

A finite-difference (FD) method (e.g. [26, 27]) alleviates some of the geometric complexities because of a dimensionally-split discretization that only requires information along one-dimensional grid lines. For example, time advancement of the solution at node $\mathcal{K}$ in figure 1(c) requires derivative approximations along the x- and the y-grid lines which simply needs derivative stencils for regularly spaced nodes (as illustrated by the green dashed box) as well as stencils that account for non-uniform node spacing at the EB (as illustrated by the red dashed box). The one-dimensional configuration of interest at the EB is shown in figure 2, and the derivative stencils for such configurations are referred to hereafter as the EB stencils. The small-cell treatment in a finite-difference EB discretization boils down to handling the cases of $\alpha \to 0$ in the EB stencils.

The dimensionally-split FD discretization avoids the cost and the complications of surface/volume reconstructions, however, derivation of stable and conservative EB stencils for the simple EB configuration of figure 2 has proven notoriously difficult for inviscid problems. In [26], for example, the authors combined local Taylor-series expansion with stability constraints to derive EB stencils for centered schemes that were up to fourth-order accurate in the interior. But, as the authors noted, the derived high-order stencils were not stable in the inviscid limit. Due to the challenge of theoretically proving stability, a non-linear optimization approach was employed in [27] to numerically examine the stability of high-order EB stencils. To the best of our knowledge, EB stencils that are theoretically stable in the inviscid limit without introducing artificial dissipation, explicitly or implicitly (by using dissipative schemes in the interior), have not been reported.

Non-dissipative FD convection schemes that ensure primary and secondary conservation are favored for long time integrations in unsteady turbulent flow simulations [28, 29, 30]. Primary conservation ensures conservation of a transport variable, whereas secondary (or energy) conservation ensures conservation of the quadratic quantity of the transport variable in the inviscid limit. For example, in the scalar transport over a bounded domain,

$$\phi_t + c\phi_x = \mu\phi_{xx}, \qquad \text{in} \qquad x_0 \le x \le x_n, \tag{1}$$

where c and μ are constants, primary conservation implies

$$\frac{d}{dt}\int_{x_0}^{x_n} \phi \, dx = \int_{x_0}^{x_n} \left(-c\frac{\partial \phi}{\partial x} + \mu\frac{\partial^2 \phi}{\partial x^2}\right) dx = c\left[\phi\left(x_0,t\right) - \phi\left(x_n,t\right)\right] + \mu\left[\phi_x\left(x_n,t\right) - \phi_x\left(x_0,t\right)\right], \tag{2}$$

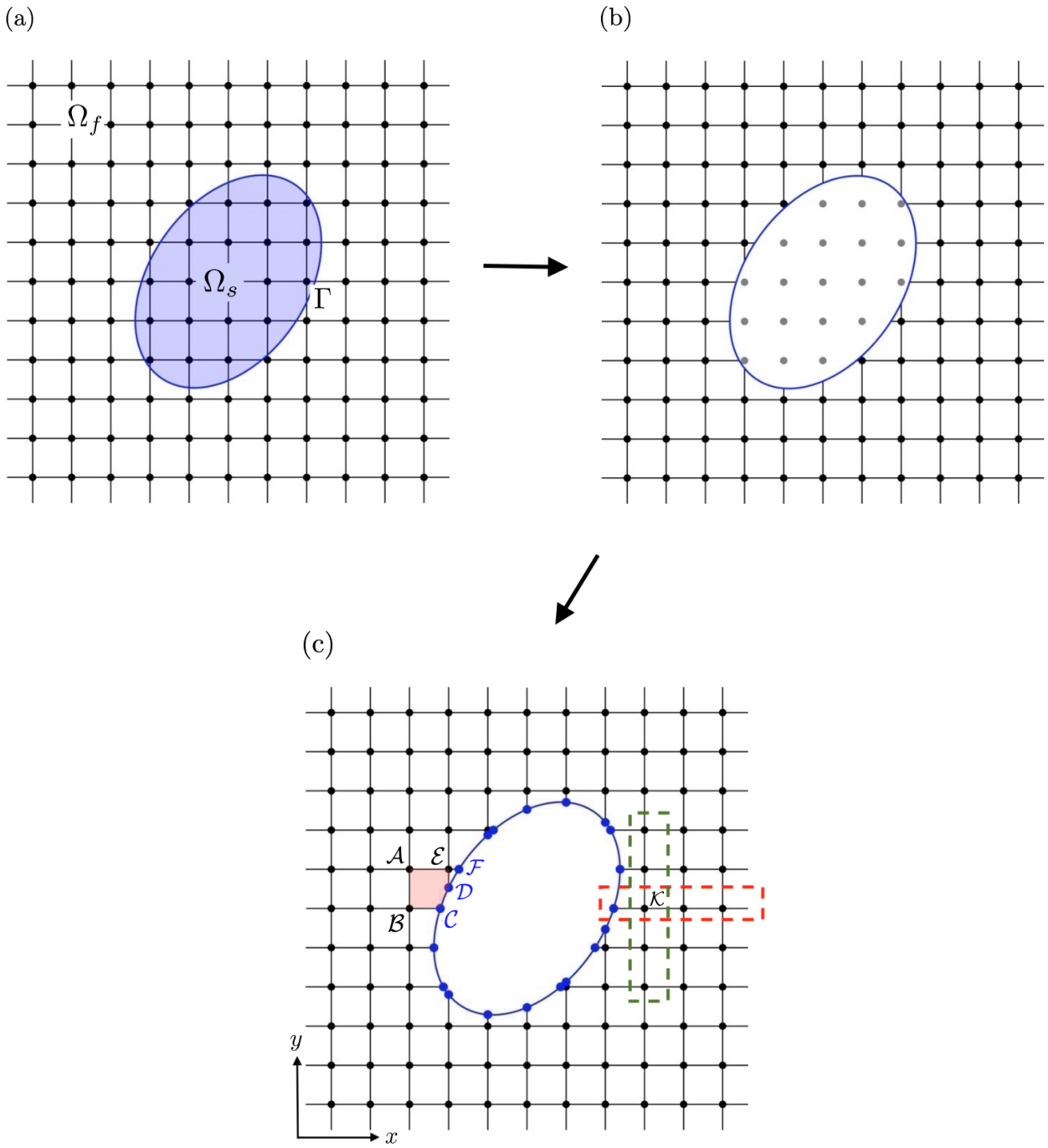


Figure 1: Schematic of the steps in an embedded boundary (or cut-cell) method: (a) embedded boundary, Γ, at the intersection of the solid body Ω_s with the fluid domain Ω_f, (b) the Cartesian grid points (or nodes) inside the solid body, shown as gray nodes, are excluded from fluid calculations, (c) EB points, shown as blue nodes, are introduced at the intersection of Γ with the Cartesian grid lines. A finite-difference EB discretization requires derivative approximations along grid lines, as highlighted at node $\mathcal{K}$ by the x- and the y-grid lines in the red and the green dashed boxes, respectively, of subfigure (c).

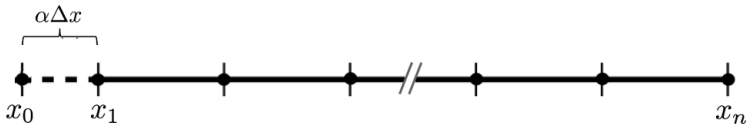


Figure 2: One-dimensional grid with non-uniform spacing at the first grid point, analogous to a embedded (or cut-cell) boundary point ($0 \leq \alpha \leq 1$).

whereas secondary conservation implies

$$\frac{d}{dt}\int_{x_0}^{x_n} \phi^2\, dx = 2\int_{x_0}^{x_n} \left(-c\phi\frac{\partial \phi}{\partial x} + \mu\phi\frac{\partial^2 \phi}{\partial x^2}\right) dx = c\left[\phi^2\left(x_0,t\right) - \phi^2\left(x_n,t\right)\right] + 2\mu\left[\phi\phi_x|_{x_0}^{x_n} - \|\phi_x\|^2\right], \quad (3)$$

where $\phi\phi_x|_{x_0}^{x_n} = \phi\left(x_n,t\right)\phi_x\left(x_n,t\right) - \phi\left(x_0,t\right)\phi_x\left(x_0,t\right)$ and $\|\phi_x\|^2 = \int_{x_0}^{x_n} \phi_x^2\, dx$. In the inviscid limit ($\mu = 0$), (3) states that ϕ^2 is conserved on accounting for the boundary fluxes. Long-time stability benefits of discretely satisfying the primary and secondary conservation property in the inviscid limit for periodic domains (where the RHS of (2) and (3) vanish, except for the $\|\phi_x\|^2$ term in (3)) have been reported for incompressible [31] as well as compressible [30] turbulent flow simulations. The present study shows that EB stencils for spatial first- and second-derivative approximation that satisfy the secondary conservation property (3) yield stable EB methods for inviscid as well as viscous problems, and derives those stencils for schemes that are up to sixth-order accurate in the interior and fourth-order accurate globally. While similar accuracy was reported in the authors' previous work [27], to the best of our knowledge, a systematic way to obtain high-order EB schemes that are theoretically stable (without artificial/numerical dissipation) and that address the small-cell problem (without geometry/solution reconstructions) have not been reported.

The paper is organized as follows. Section 2 describes the key steps in the proposed finite-difference EB scheme. Section 3 discusses the derivation of the EB schemes to address the small-cell problem and to ensure primary and secondary conservation. The construction of the first-derivative approximations for embedded boundaries is described in section 3.1 and second-derivative approximations in section 3.2. Time stability of the derived schemes is discussed in section 4 and the numerical results for various linear and non-linear problems are evaluated in section 5. Finally, the conclusions are provided in section 6.

2. Finite-difference Cartesian embedded boundary method

The steps in the proposed finite-difference EB discretization are shown schematically in figure 1. The pre-processing step involves identifying the fluid nodes where the fluid-flow equations need to be solved. The grid points shown as gray nodes in figure 1(b) lie in Ω_s and they are, therefore, excluded from fluid calculations. To account for the fluid-solid boundary, the EB nodes, colored blue in figure 1(c), are introduced at the intersection of the solid body with the Cartesian grid lines. The presence of a solid body and the EB nodes modify the overall fluid discretization as follows.

Assuming $n_x + 1$ and $n_y + 1$ grid points (black nodes) in the x- and y-direction, respectively, of the Cartesian grid shown in figure 1(a), the derivative approximations can be written as

$$\frac{\partial^\nu}{\partial x^\nu} \approx \mathbf{D}_\nu^x = I^y \otimes D_\nu^x, \qquad \frac{\partial^\nu}{\partial y^\nu} \approx \mathbf{D}_\nu^y = D_\nu^y \otimes I^x, \quad (4)$$

where D_ν^x is a $(n_x+1)\times(n_x+1)$ operator that approximates $\partial^\nu/\partial x^\nu$ along a x-grid line, I^y is a $(n_y+1)\times(n_y+1)$ identity matrix and $\otimes$ denotes a Kronecker product. The operators in the approximation of $\partial^\nu/\partial y^\nu$ are similarly defined. $\mathbf{D}_\nu^x$ and $\mathbf{D}_\nu^y$ are matrices of size $N\times N$, where $N = (n_x+1)(n_y+1)$, and they assume a grid function with elements Φ_{ij} $(0 \leq i \leq n_x,\, 0 \leq j \leq n_y)$ ordered as

$$\mathbf{\Phi} = \begin{bmatrix} \Phi_{00} & \Phi_{10} & \cdots & \Phi_{n_x 0} & \Phi_{01} & \cdots & \Phi_{n_x n_y} \end{bmatrix}^T. \quad (5)$$

When the solid (embedded) body and the EB nodes are introduced in figure 1(c), (4) is no longer applicable because all the x- and y-grid lines are not equivalent. The grid lines that cut through the solid body now include the EB nodes and the nodes in Ω_s (where fluid calculations are not required). The derivative approximations $\mathbf{D}_\nu^x$ and $\mathbf{D}_\nu^y$ are then given by

$$\mathbf{D}_\nu^x = \begin{bmatrix} D_\nu^{x,0} & & & & \\ & D_\nu^{x,1} & & & \\ & & \ddots & & \\ & & & D_\nu^{x,n_y-1} & \\ & & & & D_\nu^{x,n_y} \end{bmatrix}, \qquad \mathbf{D}_\nu^y = \begin{bmatrix} D_\nu^{0,y} & & & & \\ & D_\nu^{1,y} & & & \\ & & \ddots & & \\ & & & D_\nu^{n_x-1,y} & \\ & & & & D_\nu^{n_x,y} \end{bmatrix} \mathcal{P}, \tag{6}$$

where $D_\nu^{x,j}$, for $0 \le j \le n_y$, is the $(n_x+1)\times(n_x+1)$ operator approximating $\partial^\nu/\partial x^\nu$ along the x-grid line at the j^{th} y-coordinate and $D_\nu^{i,y}$, for $0 \le i \le n_x$, is the $(n_y+1)\times(n_y+1)$ operator approximating $\partial^\nu/\partial y^\nu$ along the y-grid line at the i^{th} x-coordinate. $\mathcal{P}$ is a permutation matrix that transforms the order of the elements in the grid function (5) from x-first traversal to y-first, given by

$$\mathcal{P}\mathbf{\Phi} = \begin{bmatrix} \Phi_{00} & \Phi_{01} & \cdots & \Phi_{0n_y} & \Phi_{10} & \cdots & \Phi_{n_x n_y} \end{bmatrix}^T. \tag{7}$$

In the absence of a solid body, all the $D_\nu^{x,j}$ operators are the same and, similarly, all the $D_\nu^{i,y}$ operators are the same allowing $\mathbf{D}_\nu^x$ and $\mathbf{D}_\nu^y$ to be written as (4). Introduction of the solid body and the EB nodes require modifications to the operators for the grid lines that intersect the solid body. EB stencils, which include only the fluid grid points and the EB nodes, constitute the modifications to account for the EB. These stencils, by excluding the grid points in Ω_s, ensure that the solution values at grid points inside the solid body are ignored in advancing the fluid solution. The EB stencils differ from a regular boundary stencil because of the non-uniform node spacing near the EB, as shown, for example, by the node spacings in the red dashed box of figure 1(c). A one-dimensional configuration that represents the EB setup is shown in figure 2, where the first node location is given by

$$x_0 = x_1 - \alpha\Delta x, \qquad \text{for} \qquad 0 \le \alpha \le 1. \tag{8}$$

Δx denotes the uniform grid spacing in the interior, *e.g.* $\Delta x = x_2 - x_1$. $\alpha = 1$ yields a uniform grid, whereas $\alpha = 0$ leads to coincident first and second grid points. Derivative approximations for $0 \le \alpha \le 1$ in the simple configuration of figure 2 provide all the EB stencils needed to construct the operators in (6). The goal here is to derive those approximations such that the discrete operators satisfy the primary and the secondary conservation properties outlined in the previous section.

2.1. Discrete conservation

A semi-discretization of (1) over the grid shown in figure 2 can be written as

$$\frac{d\mathbf{\Phi}}{dt} = -cD_1\mathbf{\Phi} + \mu D_2\mathbf{\Phi}, \tag{9}$$

where the grid function $\mathbf{\Phi}(t) = \begin{bmatrix} \Phi_0(t) & \cdots & \Phi_n(t) \end{bmatrix}^T$, and D_1 and D_2 denote a first- and second-derivative approximation, respectively.

The discrete primary conservation condition for (9) that mimics (2) can be written as

$$\frac{d}{dt}\sum_{i=0}^{n}(W\mathbf{\Phi})_i = \sum_{i=0}^{n}\left(W\frac{d\mathbf{\Phi}}{dt}\right)_i = \sum_{i=0}^{n}\left(-cWD_1\mathbf{\Phi} + \mu WD_2\mathbf{\Phi}\right)_i = c\left(\Phi_0 - \Phi_n\right) + \mu\left[(S_1\mathbf{\Phi})_n - (S_1\mathbf{\Phi})_0\right], \tag{10}$$

where $(\bullet)_i$ denotes the i-th component of a vector and the entries of the $(n+1)\times(n+1)$ matrix W constitute a quadrature for the domain $x_0 \le x \le x_n$. S_1 denotes a first-derivative approximation that may be same as or different from D_1 depending on the operator that satisfies all the conservation constraints.

Similarly, the discrete secondary conservation condition that mimics (3) can be written as

$$\frac{d}{dt}\|\mathbf{\Phi}\|_W^2 = \mathbf{\Phi}^T W\frac{d\mathbf{\Phi}}{dt} + \left(\frac{d\mathbf{\Phi}}{dt}\right)^T W\mathbf{\Phi} = -c\mathbf{\Phi}^T\left[WD_1 + (WD_1)^T\right]\mathbf{\Phi} + \mu\mathbf{\Phi}^T\left[WD_2 + (WD_2)^T\right]\mathbf{\Phi}$$

$$= c\left(\Phi_0^2 - \Phi_n^2\right) + 2\mu\left[\Phi_n\left(S_1\Phi\right)_n - \Phi_0\left(S_1\Phi\right)_0 - \|D_1\mathbf{\Phi}\|_W^2\right], \tag{11}$$

where $\|\mathbf{v}\|_W^2 = \mathbf{v}^T W \mathbf{v}$ is the discrete norm of a vector $\mathbf{v}$. A valid definition of the norm requires that the matrix W be symmetric and positive definite (s.p.d.).

Our aim here is to derive high-order approximations D_1 and D_2 for the EB configuration of figure 2 such that the conservation conditions (10) and (11) are satisfied.

2.1.1. First-derivative approximation

The first-derivative approximation, D_1, is determined to satisfy the inviscid components of (10) and (11) given by, respectively,

$$\sum_{i=0}^{n} \left(WD_1\mathbf{\Phi}\right)_i = \Phi_n - \Phi_0 \tag{12}$$

and

$$\mathbf{\Phi}^T\left[WD_1 + \left(WD_1\right)^T\right]\mathbf{\Phi} = \Phi_n^2 - \Phi_0^2. \tag{13}$$

(12) implies, for the entries a_{ij} $(0 \le i, j \le n)$ of the matrix $A = WD_1$,

$$\sum_{i=0}^{n} a_{ij} = \begin{cases} -1 & j = 0 \\ 0 & 0 < j < n \\ 1 & j = n \end{cases}, \tag{14}$$

and (13) implies (e.g. [32])

$$WD_1 + \left(WD_1\right)^T = E = \mathrm{diag}\left(-1, 0, \cdots 0, 1\right). \tag{15}$$

It can be shown that a first derivative approximation D_1 that satisfies (15) also ensures (14); see Appendix A for the proof.

2.1.2. Second-derivative approximation

The second-derivative approximation, D_2, is determined to satisfy the viscous components of (10) and (11) given by, respectively,

$$\sum_{i=0}^{n} \left(WD_2\mathbf{\Phi}\right)_i = \left(S_1\mathbf{\Phi}\right)_n - \left(S_1\mathbf{\Phi}\right)_0 \tag{16}$$

and

$$\mathbf{\Phi}^T\left[WD_2 + \left(WD_2\right)^T\right]\mathbf{\Phi} = 2\left(\Phi_n\left(S_1\Phi\right)_n - \Phi_0\left(S_1\Phi\right)_0 - \|D_1\mathbf{\Phi}\|_W^2\right). \tag{17}$$

A s.p.d. W and a D_2 of the form (*e.g.* [33, Section 7.2])

$$WD_2 = -M + ES_1 \tag{18}$$

is considered, where E is given by (15) and S_1 is a first-derivative approximation. (18) ensures (16) if the entries of M, denoted by m_{ij}, satisfy

$$\sum_{i=0}^{n} m_{ij} = 0 \qquad \text{for} \qquad 0 \le j \le n. \tag{19}$$

Substituting (18) in LHS of (17) yields

$$\mathbf{\Phi}^T\left[WD_2 + \left(WD_2\right)^T\right]\mathbf{\Phi} = 2\left[\Phi_n\left(S_1\Phi\right)_n - \Phi_0\left(S_1\Phi\right)_0\right] - \mathbf{\Phi}^T\left[M + M^T\right]\mathbf{\Phi}. \tag{20}$$

If M is an approximation of $-\partial^2/\partial x^2$ at the interior grid points then $-\mathbf{\Phi}^T M \mathbf{\Phi} \approx \int \phi\phi_{xx}\ dx$, ignoring the boundary terms. For the continuous case, $\int \phi\phi_{xx}\ dx = -\int \phi_x^2\ dx = -\|\phi_x\|^2$, ignoring the boundary terms. The last term of (20), therefore, provides a reasonable approximation to the last term of (17), where $\|D_1\mathbf{\Phi}\|_W^2 \approx \|\phi_x\|^2$, if $M + M^T$ is positive semidefinite and M is an approximation of $-\partial^2/\partial x^2$ at the interior grid points [33]. Hence, W and D_2 of the form (18) ensure that (16) and (17) are satisfied if M has the following properties:

p	1	2	3
$c_{2,0}$	-2	$-5/2$	$-49/18$

Table 1: Values of $c_{2,0}$ for the centered schemes of order-of-accuracy $2p$ in the interior.

- its entries satisfy (19),
- it is an approximation of $-\partial^2/\partial x^2$ at the interior grid points, and
- $M + M^T$ is positive semidefinite.

To summarize, W, D_1 and D_2 that ensure (15) and (18) with a first-derivative approximation S_1 and a M that satisfies the above properties ensure that the discrete primary and secondary conservation conditions, (10) and (11), are satisfied. The next section describes the steps in deriving those operators for the EB configuration of figure 2 for various order of accuracies.

3. EB stencil derivation and addressing the small-cell problem

FD derivative approximations, D_ν, for the bounded domain of figure 2 can be written as

$$D_\nu = \frac{1}{\Delta x^\nu}\begin{bmatrix} d_{\nu,00} & \cdots & \cdots & \cdots & d_{\nu,0\beta} & & & & & \\ \vdots & \ddots & \ddots & \ddots & \vdots & & & & & \\ \vdots & \ddots & \ddots & \ddots & \vdots & & & & & \\ d_{\nu,\kappa 0} & \cdots & \cdots & \cdots & d_{\nu,\kappa\beta} & & & & & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & & -c_{\nu,\omega} & \cdots & -c_{\nu,1} & c_{\nu,0} & c_{\nu,1} & \cdots & c_{\nu,\omega} \\ & & & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \end{bmatrix}, \tag{21}$$

where $\kappa+1$ and $\beta+1$ are the depth and the width of the EB stencil, respectively, and ω $(= p)$ denotes the half-width of the centered scheme of order $2p$ in the interior. The coefficients of the centered schemes for $\nu = 1, 2$ and order-of-accuracy $2p$ are given by

$$c_{1,i} = \begin{cases} 0 & i = 0 \\ -\frac{(-1)^i\,(\omega!)^2}{i\,(\omega+i)!\,(\omega-i)!} & 1 \le i \le \omega = p \end{cases}, \tag{22}$$

$$c_{2,i} = \begin{cases} c_{2,0} & i = 0 \\ -\frac{2(-1)^i\,(\omega!)^2}{i^2\,(\omega+i)!\,(\omega-i)!} & 1 \le i \le \omega = p \end{cases}, \tag{23}$$

where $c_{2,0}$ for various values of p considered in this study are provided in table 1. The boundary stencil coefficients $d_{\nu,ij}$ $(0 \le i \le \kappa,\, 0 \le j \le \beta)$ for the EB configuration of figure 2 are the unknowns of interest. Only the coefficients for the left boundary are shown in (21) and considered in the discussion hereafter. The stencil coefficients for the right boundary can be obtained from the stencils derived for the left boundary (in terms of α), for example, by using $\alpha = 1$ for the uniform grid spacing at the right boundary in figure 2.

Quadrature matrices W of the form

$$W = \Delta x\,\mathrm{diag}\,(w_0, \cdots, w_\kappa, 1, \cdots) \tag{24}$$

are considered, where the coefficients at the right boundary can again be obtained from the expressions of the left boundary coefficients. We show below that a conservative discretization can be transformed from

a node-based representation to a flux form using a W that ensures primary conservation, *i.e.* that ensures (12) and (16) for the semi-discretization (9). The diagonal structure of W simplifies the transformation by directly providing the cell sizes. Using those cell sizes, the semi-discretization (9) can be written in the flux form as

$$\left.\frac{d\mathbf{\Phi}}{dt}\right|_i = -c\frac{\Phi_{i+\frac{1}{2}} - \Phi_{i-\frac{1}{2}}}{w_i\Delta x} + \mu\frac{(\Phi_x)_{i+\frac{1}{2}} - (\Phi_x)_{i-\frac{1}{2}}}{w_i\Delta x}. \tag{25}$$

The semi-discretizations (9) and (25) differ only in form and are otherwise identical in that the numerical solutions from (9) and (25) will be the same. The cell interface fluxes of (25) can be assembled in vectors

$$\mathbf{f}_1 = \begin{bmatrix}\Phi_{-\frac{1}{2}} & \Phi_{\frac{1}{2}} & \cdots & \Phi_{n+\frac{1}{2}}\end{bmatrix}^T \qquad \text{and} \qquad \mathbf{f}_2 = \begin{bmatrix}(\Phi_x)_{-\frac{1}{2}} & (\Phi_x)_{\frac{1}{2}} & \cdots & (\Phi_x)_{n+\frac{1}{2}}\end{bmatrix}^T$$

of size $n+2$, and the fluxes can be obtained iteratively using

$$(\mathbf{f}_1)_j = (\mathbf{f}_1)_{j-1} + (WD_1\mathbf{\Phi})_{j-1}, \qquad \text{for} \qquad 1 \le j \le n+1, \tag{26}$$

$$(\mathbf{f}_2)_j = (\mathbf{f}_2)_{j-1} + (WD_2\mathbf{\Phi})_{j-1}, \qquad \text{for} \qquad 1 \le j \le n+1, \tag{27}$$

with $(\mathbf{f}_1)_0 = \Phi_{-\frac{1}{2}} = \Phi_0$ and $(\mathbf{f}_2)_0 = (\Phi_x)_{-\frac{1}{2}} = (S_1\mathbf{\Phi})_0$. The notation $(\bullet)_j$ denotes the j-th component of a vector. Δx in (25) is the uniform grid spacing of the interior points, as defined in (8), and the cell sizes are determined by w_i. The small-cell problem can thus be avoided by choosing w_i that remain sufficiently large as $\alpha \to 0$. Hence the constraint $w_i \ge \varepsilon$ for $0 \le \alpha \le 1$ will be imposed in the following derivations with a suitable positive value ε.

The goal here is to derive the approximations D_1 and D_2 and the matrix W for various order of accuracies such that:

1. D_1 and D_2 use non-dissipative centered schemes of order-of-accuracy $\mathcal{O}\left(\Delta x^{2p}\right)$ in the interior with EB stencils that are at least $\mathcal{O}\left(\Delta x^p\right)$ accurate
2. WD_1 satisfies (15) and WD_2 of the form (18) satisfies (19) with a positive semidefinite M that is an approximation of $-\partial^2/\partial x^2$ at the interior grid points
3. The EB stencils must not cause small-cell issues when $\alpha \to 0$ in figure 2
4. The overall scheme must be time stable with appropriate boundary conditions for hyperbolic, parabolic as well as incompletely parabolic problems

Condition 1, 2 and 4 ensure high resolution, discrete primary and secondary conservation, and stability, respectively. Condition 3 addresses the small-cell problem. The methodology employed to satisfy Conditions 1, 2 and 3 for D_1 and D_2 is described in section 3.1 and 3.2, respectively, and the stability of the derived schemes is discussed in section 4.

3.1. First-derivative approximations

Algorithm 1 describes the steps in the derivation of D_1 and W to satisfy Conditions 1, 2 and 3 for various order of accuracies. The algorithm takes as input the boundary/interior order of accuracy (defined by p) and the maximum acceptable depth of the EB stencil (defined by N_κ), and provides as output the operators D_1 and W if EB stencils that satisfy Conditions 1, 2 and 3 are found. The EB stencil depth $\kappa+1$, see (21), determines the number of free parameters, and if they are insufficient to satisfy all the conditions, κ is incremented. The initial value of $\kappa+1$ is assumed equal to p, which is the half-width of the interior central scheme. A $2p^{\text{th}}$-order accurate central scheme requires boundary closures for at least p boundary points. The width of the EB stencil is specified by $\beta = \kappa+p$, consistent with the structure of the stable FD boundary stencils on a uniform grid [34, 35]. The flags for Conditions 1, 2, and 3 are labeled `AccuracyConstraint`, `Consv`, and `AddressSCellIssue`, respectively. The EB stencil free parameters are first chosen to satisfy the order-of-accuracy constraints (Condition 1), followed by the primary and secondary conservation constraints

(Condition 2). The remaining free parameters are then chosen to ensure $w_j \geq \varepsilon$ for $0 \leq j \leq \kappa$ and $0 \leq \alpha \leq 1$ to address the small-cell problem (Condition 3). $\varepsilon = 0.1$ is used for the derivations here and it suffices to avoid the small-cell issue.

The EB stencils for $p = 1$ derived from Algorithm 1 are given by

$$D_1 = \frac{1}{\Delta x}\begin{bmatrix} -\frac{2}{\alpha+1} & 1 & \frac{1-\alpha}{\alpha+1} & 0 & 0 & \\ -\frac{1}{\alpha+1} & 0 & \frac{1}{\alpha+1} & 0 & 0 & \\ \frac{1-\alpha}{\alpha^2-\alpha-4} & \frac{\alpha+1}{\alpha^2-\alpha-4} & 0 & \frac{2}{-\alpha^2+\alpha+4} & 0 & \\ 0 & 0 & -\frac{1}{2} & 0 & \frac{1}{2} & \\ & & & \ddots & \ddots & \ddots \end{bmatrix}, \tag{28}$$

$$W = \Delta x \,\mathrm{diag}\left(\frac{\alpha+1}{4}, \frac{(\alpha+1)^2}{4}, \frac{-\alpha^2+\alpha+4}{4}, 1, \cdots\right). \tag{29}$$

$\kappa = 2$ in (28) and (29). The small-cell problem does not arise with the above operators because the choice of $w_j \geq \varepsilon$ ensures that, in the flux-based discretization (25), the cell sizes near the irregular nodes, given by (29), do not become small when $\alpha \to 0$ and, as a result, the denominator of the stencil coefficients, given in (28), are significantly large for $0 \leq \alpha \leq 1$, thus, avoiding the issue of division by zero or a small number.

Since Algorithm 1 ensures that the scheme is conservative, (28) and (29) can be used to write the inviscid terms of the flux-based discretization (25) with fluxes, calculated using (26), given by

$$(\mathbf{f}_1)_i = \Phi_{i-\frac{1}{2}} = \begin{cases} \Phi_0 & \text{if} \quad i = 0, \\ \frac{1}{2}\Phi_0 + \frac{1+\alpha}{4}\Phi_1 + \frac{1-\alpha}{4}\Phi_2 & \text{if} \quad i = 1, \\ \frac{1-\alpha}{4}\Phi_0 + \frac{1+\alpha}{4}\Phi_1 + \frac{1}{2}\Phi_2 & \text{if} \quad i = 2, \\ \frac{1}{2}\left(\Phi_{i-1} + \Phi_i\right) & \text{if} \quad i \geq 3, \end{cases} \tag{30}$$

and the cell sizes given by (29). For $\alpha = 0$, *i.e.* coincident first and second grid points, the cell sizes, using (29), are

$$W = \Delta x \,\mathrm{diag}\left(\frac{1}{4}, \frac{1}{4}, 1, 1, \cdots\right), \tag{31}$$

showing that while the grid points coincide at the EB, the effective cell sizes are not zero. This, by construction, avoids the small-cell problem without any geometry or solution reconstructions.

The EB stencils for $p = 2$ and 3 that use the fourth- and the sixth-order central scheme, respectively, in the interior are also derived using Algorithm 1. The stencil coefficients for $p = 2$ ($\kappa = 4$) are provided in Appendix C and for $p = 3$ ($\kappa = 6$) in the supplementary document. Matlab scripts containing these coefficients are included in the supplementary material and the instructions to access them are described in Appendix D.

3.2. Second-derivative approximations

Second-derivative approximations, D_2, that satisfy Conditions 1, 2 and 3 for various order of accuracies are derived using Algorithm 2. The depth and the width of the EB stencil for D_2 are chosen to be the same as that of D_1 to allow conservation analyses using the W matrix derived for D_1. A common W for D_1 and D_2 helps ensure numerical stability for convection-diffusion problems, as discussed in the next section. Similar to the algorithm for derivation of D_1, the free parameters in the EB stencils for D_2 are first chosen to satisfy the order-of-accuracy constraints. Then the ES_1 matrix in (18) of the form

$$ES_1 = \begin{bmatrix} -s_0 & \cdots & -s_\beta & 0 & \cdots \\ 0 & \cdots & 0 & 0 & \cdots \\ & \ddots & \ddots & \ddots & \ddots \\ & \ddots & \ddots & \ddots & \ddots \end{bmatrix} \tag{32}$$

Algorithm 1 Determine D_1 and W to satisfy Conditions 1, 2 and 3.

input : Boundary and interior order of accuracy (p, $2p$)
input : Maximum value of κ (N_κ)

AccuracyConstraint, Consv, AddressSCellIssue = `false`
$\kappa \leftarrow p - 1$
while $\kappa \leq N_\kappa$ **do**
  $\beta \leftarrow \kappa + p$
  Use κ, β and p to construct D_1 and W as given by (21), (22) and (24)
  AccuracyConstraint $\leftarrow$ Can the free parameters in D_1 satisfy the order-of-accuracy constraints?
  if (AccuracyConstraint) **then**
    Update the D_1 boundary coefficients to satisfy the order-of-accuracy constraints
    Consv $\leftarrow$ Can the remaining free parameters in D_1 and W satisfy (15)?
    if (Consv) **then**
      Update the D_1 and W boundary coefficients to satisfy (15)
      AddressSCellIssue $\leftarrow$ Can the remaining free parameters in W ensure $w_j \geq \varepsilon$ $(0 \leq j \leq \kappa)$ for $0 \leq \alpha \leq 1$?
      if (AddressSCellIssue) **then**
        Update the W boundary coefficients to ensure $w_j \geq \varepsilon$ for $0 \leq \alpha \leq 1$
        return D_1 and W
      else
        κ++
      end if
    else
      κ++
    end if
  else
    κ++
  end if
end while

return `no solution found`

is constructed. The operator E, given by (15), zeros out all the rows of the first-derivative approximation S_1 except the first and the last. Only the left boundary coefficients are shown in (32); the coefficients for the right boundary can be obtained from the left boundary coefficients (derived in terms of α). To simplify conservation and stability analyses, the width of the S_1 boundary stencil is assumed to not exceed the width of the corresponding D_1 and D_2 operator.

D_2 for $p = 1$ derived from Algorithm 2 is

$$D_2 = \frac{1}{\Delta x^2}\begin{bmatrix} \frac{6}{\alpha^2+3\alpha+2} & -2 & \frac{4\alpha-2}{\alpha+1} & \frac{2-2\alpha}{\alpha+2} & 0 & \\ \frac{6}{\alpha^2+3\alpha+2} & -2 & \frac{4\alpha-2}{\alpha+1} & \frac{2-2\alpha}{\alpha+2} & 0 & \\ \frac{2(\alpha-1)}{\alpha^3-5\alpha-4} & \frac{2(\alpha+1)}{-\alpha^2+\alpha+4} & \frac{2\left(\alpha^2+3\alpha+4\right)}{\alpha^3-5\alpha-4} & \frac{4}{-\alpha^2+\alpha+4} & 0 & \\ 0 & 0 & 1 & -2 & 1 & \\ & & & \ddots & \ddots & \ddots \end{bmatrix}, \tag{33}$$

and the entries of ES_1 for $p = 1$ are

$$s_0 = -\frac{2+\alpha}{1+\alpha}, \quad s_1 = \frac{1}{2}(1+\alpha)^2, \quad s_2 = -\frac{-2+2\alpha^2+\alpha^3}{1+\alpha}, \quad s_3 = \frac{1}{2}\left(-1+\alpha^2\right). \tag{34}$$

As in the case of stencils (28)–(29), the small-cell problem does not arise with the above operators because the choice of $w_j \geq \varepsilon$ ensures that the denominators of the stencil coefficients in (33) and (34) do not become small for $0 \leq \alpha \leq 1$, thus, avoiding division by zero or a small number.

(29), (33) and (34) can be used to write the viscous terms of the flux-based discretization (25) with fluxes calculated using (27), given by

$$(\mathbf{f}_2)_i = (\Phi_x)_{i-\frac{1}{2}} = \frac{1}{\Delta x}\begin{cases} -\frac{2+\alpha}{1+\alpha}\Phi_0 + \frac{1}{2}(1+\alpha)^2\Phi_1 - \frac{-2+2\alpha^2+\alpha^3}{1+\alpha}\Phi_2 + \frac{1}{2}\left(-1+\alpha^2\right)\Phi_3 & \text{if} \quad i = 0, \\ -\frac{5+5\alpha+2\alpha^2}{4+6\alpha+2\alpha^2}\Phi_0 + \frac{1}{2}\alpha(1+\alpha)\Phi_1 + \frac{3+\alpha-2\alpha^2-2\alpha^3}{2(1+\alpha)}\Phi_2 + \frac{(-1+\alpha)(1+\alpha)^2}{2(2+\alpha)}\Phi_3 & \text{if} \quad i = 1, \\ \frac{\alpha-1}{2(1+\alpha)}\Phi_0 - \frac{1}{2}(1+\alpha)\Phi_1 + \frac{2+\alpha+\alpha^2}{2(1+\alpha)}\Phi_2 & \text{if} \quad i = 2, \\ \Phi_i - \Phi_{i-1} & \text{if} \quad i \geq 3, \end{cases} \tag{35}$$

and the cell sizes given by (29). As discussed in the previous section, the cell sizes remain finite with $\alpha \to 0$, thus avoiding the small-cell problem. The fluxes (35) represent first-derivative approximations at the cell interface locations determined by (29).

Algorithm 2 has also been applied to derive EB stencils for D_2 with $p = 2$ and 3 that use the fourth- and the sixth-order central scheme, respectively, in the interior. The stencil coefficients are provided in the supplementary document and Matlab scripts containing those coefficients are included in the supplementary material. Algorithms 1 and 2 were executed in Mathematica [36].

Algorithm 2 Determine D_2 to satisfy Conditions 1, 2 and 3.

input : Boundary and interior order of accuracy (p, $2p$)

input : κ and β from D_1 for (p, $2p$) order of accuracy

Use κ, β and p to construct D_2 as given by (21), (23) and table 1

Update the boundary coefficients in D_2 to satisfy the order-of-accuracy constraints

Construct the matrix ES_1 of the form (32)

Update the coefficients of S_1 to approximate first derivative with p^{th} order of accuracy

Choose the free parameters in D_2 and S_1 such that the entries of $M\,(= ES_1 - WD_2)$ satisfy (19)

Optimize the remaining free parameters to ensure $M + M^T$ is positive semidefinite

4. Stability analysis

In this section, we show that the conservation conditions satisfied by the derivative approximations derived in the previous section ensure that the proposed EB method is time stable for convection-diffusion problems, including pure convection and pure diffusion problems. Pure convection problems lack physical diffusion that may damp the numerical instabilities arising at the EB, and hence they provide a strict test of numerical stability, especially with the non-dissipative interior schemes considered in this study.

To begin, consider the linear convection-diffusion equation (1) with an initial condition

$$\phi\left(x,0\right)=f\left(x\right), \tag{36}$$

and Dirichlet boundary conditions (BCs),

$$\phi\left(x_0,t\right)=g_1\left(t\right), \qquad \phi\left(x_n,t\right)=g_2\left(t\right). \tag{37}$$

The case with a Neumann BC will be analysed later in this section. The semi-discretization (9) for solving (1) with the BCs (37) is *time stable* if for homogeneous BCs, *i.e.* for $g_1=g_2=0$, there is a unique solution $\mathbf{\Phi}(t)$ satisfying [33, 34]

$$\left\|\mathbf{\Phi}\right\|_W \leq K\left\|\mathbf{f}\right\|_W, \qquad \text{or} \qquad \frac{d}{dt}\left\|\mathbf{\Phi}\right\|_W^2 \leq 0, \tag{38}$$

where K is independent of Δx, $\mathbf{f}$ and t. $\mathbf{f}=\begin{bmatrix} f(x_0) & \cdots & f(x_n)\end{bmatrix}^T$ denotes the discrete initial data and $\left\|\mathbf{\Phi}\right\|_W^2$ denotes the discrete norm defined as in (11). The stability condition (38) guarantees that in the absence of energy influx from the boundaries, the discrete solution energy, $\left\|\mathbf{\Phi}\right\|_W^2$, in the domain remains bounded.

Pre-multiplying (9) by $\mathbf{\Phi}^T W$ and adding it to its transpose yields, as in (11),

$$\frac{d}{dt}\left\|\mathbf{\Phi}\right\|_W^2=-c\mathbf{\Phi}^T\left[WD_1+\left(WD_1\right)^T\right]\mathbf{\Phi}+\mu\mathbf{\Phi}^T\left[WD_2+\left(WD_2\right)^T\right]\mathbf{\Phi}. \tag{39}$$

Using the secondary conservation property (15) and (20) of D_1 and D_2, respectively, (39) can be rewritten as

$$\frac{d}{dt}\left\|\mathbf{\Phi}\right\|_W^2=c\left(\Phi_0^2-\Phi_n^2\right)+2\mu\left[\Phi_n\left(S_1\Phi\right)_n-\Phi_0\left(S_1\Phi\right)_0\right]-\mu\mathbf{\Phi}^T\left[M+M^T\right]\mathbf{\Phi}, \tag{40}$$

where it is assumed that the norm matrix W is the same for D_1 and D_2, *i.e.* D_1 and D_2 are conservative with the same quadrature weights. This was ensured by considering the same W matrices in sections 3.1 and 3.2.

In this study, the BCs are applied directly (or exactly) by injecting the boundary data (e.g. [37]). For the time-stability analysis, it suffices to consider homogeneous boundary data, as mentioned above. Injecting homogeneous BCs, *i.e.* $\Phi_0=g_1=0$ and $\Phi_n=g_2=0$, in the semi-discretization (9) yields, using (40),

$$\frac{d}{dt}\left\|\mathbf{\Phi}\right\|_W^2=-\mu\widetilde{\mathbf{\Phi}}^T\left[\widetilde{M}+\widetilde{M}^T\right]\widetilde{\mathbf{\Phi}}, \tag{41}$$

where $\widetilde{\mathbf{\Phi}}(t)=\begin{bmatrix}\Phi_1(t) & \cdots & \Phi_{n-1}(t)\end{bmatrix}^T$ is the solution vector $\mathbf{\Phi}$ without the first and the last element (that correspond to the grid points where the homogeneous boundary data is injected) and $\widetilde{M}$ denotes the $(n-1)\times(n-1)$ matrix obtained by removing the first and the last rows and columns of M. The principal submatrices of a positive (semi)definite matrix are positive (semi)definite [38]. Hence $\widetilde{M}+\widetilde{M}^T$, a principal submatrix of $M+M^T$, is positive semidefinite because M derived from Algorithm 2 ensures that $M+M^T$ is positive semidefinite. Thus, from (41), $\frac{d}{dt}\left\|\mathbf{\Phi}\right\|_W^2\leq 0$, which proves that the semi-discretization (9) for solving the convection-diffusion equation (1) with Dirichlet BCs is time stable with the EB derivative approximations derived in the previous section.

Now, we consider another initial-boundary value problem (IBVP) for the convection-diffusion equation (1). For $c > 0$, the IBVP is well-posed with Dirichlet BC at the left boundary and Neumann BC at the right boundary,

$$\phi(x_0, t) = g_1(t) \quad \text{and} \quad \left.\frac{\partial \phi}{\partial x}\right|_{x_n} = g_3(t). \tag{42}$$

The discrete BCs are then given by

$$\Phi_0 = g_1 \quad \text{and} \quad (S_1 \mathbf{\Phi})_n = g_3. \tag{43}$$

The semi-discretization (9) on incorporating the BCs (43) can be written as

$$\frac{d\widehat{\mathbf{\Phi}}}{dt} = -c\widehat{D}_1 \widehat{\mathbf{\Phi}} + \mu \widehat{D}_2 \widehat{\mathbf{\Phi}} - \left(c\widehat{\mathbf{d}}_1^0 - \mu \widehat{\mathbf{d}}_2^0\right) g_1 - \mu \widehat{\mathbf{e}}_n \left[(S_1 \mathbf{\Phi})_n - g_3\right], \tag{44}$$

where $\widehat{\mathbf{\Phi}}(t) = \begin{bmatrix} \Phi_1(t) & \cdots & \Phi_n(t) \end{bmatrix}^T$ is the solution vector $\mathbf{\Phi}$ without its first element; the Dirichlet BC from (43) is injected at the first grid point. The notation $\widehat{\bullet}$ over a matrix denotes the matrix without its first row and column. Removing the first row from D_1 and D_2 avoids calculations at the first grid point, where g_1 is injected. The third term in the RHS of (44) includes stencil coefficients that use the Dirichlet boundary data. $\widehat{\mathbf{d}}_1^0$ and $\widehat{\mathbf{d}}_2^0$ denote the first column of D_1 and D_2, respectively, after removing their first row. The last term in the RHS of (44) imposes the Neumann BC at the right boundary. D_2 operators of the form (18) derived in section 3.2 contain the $W^{-1}ES_1$ operator, and the last term in the RHS of (44) substitutes the ES_1 operator at the right boundary with the boundary data g_3. The vector $\widehat{\mathbf{e}}_n = \begin{bmatrix} 0 & \cdots & 0 & 1/w_n \end{bmatrix}^T$ of size n, where w_n is the last diagonal element of W, ensures that the Neumann BC is applied only at the last grid point.

To prove time stability, condition (38) must be satisfied with homogeneous boundary data: $g_1 = g_3 = 0$. Injecting $g_1 = 0$ at x_0 provides

$$\frac{d\Phi_0}{dt} = 0, \tag{45}$$

and $g_1 = g_3 = 0$ in (44) provides

$$\frac{d\widehat{\mathbf{\Phi}}}{dt} = \left[-c\widehat{D}_1 + \mu\left(\widehat{D}_2 - E_n \widehat{S}_1\right)\right] \widehat{\mathbf{\Phi}}, \tag{46}$$

where $E_n = w_n \widehat{\mathbf{e}}_n \widehat{\mathbf{e}}_n^T$. Equations (45) and (46) can be combined to write

$$\frac{d\mathbf{\Phi}}{dt} = \left[-c\overline{D}_1 + \mu\left(\overline{D}_2 - W^{-1}E\overline{S}_1\right)\right] \mathbf{\Phi}, \tag{47}$$

where the notation $\overline{\bullet}$ over a matrix denotes the matrix with its first row and column zeroed out. Pre-multiplying (47) by $\mathbf{\Phi}^T W$ and adding it to its transpose yields

$$\frac{d}{dt}\|\mathbf{\Phi}\|_W^2 = -c\mathbf{\Phi}^T\left[W\overline{D}_1 + \left(W\overline{D}_1\right)^T\right]\mathbf{\Phi} - \mu\mathbf{\Phi}^T\left[\overline{M} + \overline{M}^T\right]\mathbf{\Phi} = -c\Phi_n^2 - \mu\widehat{\mathbf{\Phi}}^T\left[\widehat{M} + \widehat{M}^T\right]\widehat{\mathbf{\Phi}}, \tag{48}$$

where $W\overline{D}_2 = -\overline{M} + E\overline{S}_1$ is used to obtain the first equality and $W\overline{D}_1 + \left(W\overline{D}_1\right)^T = \text{diag}\,(0, \cdots, 0, 1)$ as well as $\mathbf{\Phi}^T\left[\overline{M} + \overline{M}^T\right]\mathbf{\Phi} = \widehat{\mathbf{\Phi}}^T\left[\widehat{M} + \widehat{M}^T\right]\widehat{\mathbf{\Phi}}$ are used to obtain the second equality. The latter follows from the fact that all entries in the first row and column of $\overline{M}$ are zero. The proof of $W\overline{D}_1 + \left(W\overline{D}_1\right)^T = \text{diag}\,(0, \cdots, 0, 1)$ using (15) for a diagonal W is provided in Appendix B. $\widehat{M} + \widehat{M}^T$ is a principal submatrix of $M + M^T$, which is derived to be positive semidefinite from Algorithm 2, and hence $\widehat{M} + \widehat{M}^T$ must also be positive semidefinite. Thus $\frac{d}{dt}\|\mathbf{\Phi}\|_W^2 \leq 0$, from (48), and the semi-discretization (44) is time stable with the EB derivative approximations derived in this study. The above analysis assumed a Neumann BC at the right boundary; a similar analysis shows time-stability for an IBVP with $c < 0$ and Neumann BC at the left boundary.

The above results prove time stability of the proposed EB discretization for convection-diffusion problems. Time stability for purely convective and diffusive problems follows from the above results and is discussed below.

Pure convection problem

Consider an IBVP for (1) in the inviscid limit ($\mu = 0$) with $c > 0$ and the initial and the boundary condition given by

$$\phi\,(x, 0) = f\,(x) \qquad \text{and} \qquad \phi\,(x_0, t) = g_1\,(t)\,, \tag{49}$$

respectively. A semi-discretization of this problem can be obtained from (44) and the time rate of change of the discrete solution energy with homogeneous boundary data can be obtained from (48) by substituting $\mu = 0$. The substitution in (48) provides $\frac{d}{dt}\left\|\mathbf{\Phi}\right\|_W^2 = -c\Phi_n^2 \leq 0$, thus ensuring time stability for the pure convection problem.

Pure diffusion problem

Consider an IBVP for (1) in the viscous limit ($c = 0$) with the initial condition given by (36) and the boundary conditions given by (37) or (42). In the former case (with both Dirichlet BC), the time rate of change of the discrete solution energy with homogeneous boundary data is given by (41) and in the latter case (with a Dirichlet and a Neumann BC), the time rate of change of the discrete solution energy is obtained by substituting $c = 0$ in (48). $\widehat{M} + \widehat{M}^T$ is positive semidefinite, as discussed before, and hence the proposed discretization is time stable in both cases for the pure diffusion problem.

The theoretical stability results discussed in this section are verified numerically in the next section for the schemes derived in section 3.

5. Numerical results

Numerical stability and accuracy of the EB schemes derived in section 3 are evaluated for various linear and non-linear problems in this section. The methods based on various schemes will be denoted here by D_ν^p, where $\nu = 1$ and 2 denote the first- and the second-derivative approximation, respectively, and $p = 1, 2, 3$ denote the order-of-accuracy of the boundary stencil for a $2p^{\text{th}}$-order accurate interior scheme. In all cases, the time integration is performed using the classical fourth-order Runge-Kutta (RK4) method. For convergence studies, a small enough time step is taken to ensure that the temporal errors are insignificant compared to the spatial truncation errors.

5.1. One-dimensional scalar convection problem

Consider equation (1) with $c = 1$ and $\mu = 0$ on a spatial domain $x_0 \leq x \leq x_n = 1$, shown in figure 2, where $x_0 = x_1 - \alpha\Delta x$ for $0 \leq \alpha \leq 1$. Let the initial and the boundary condition be given by (49) with

$$f\,(x) = \sin 2\pi x, \qquad g_1\,(t) = \sin 2\pi\,(x_0 - t)\,.$$

The exact solution to the problem is $\phi(x, t) = \sin 2\pi\,(x - t)$. A semi-discretization of the problem is given by (44) on substituting $\mu = 0$ and $c = 1$. The discretization (or system) matrix is then $\mathcal{M} = -\widehat{D}_1$, and the stability of the method is determined by the eigenvalues of $\mathcal{M}$ [39, 40].

Figure 3 illustrates the eigenvalues of the system matrix, $\mathcal{M}$, using the D_1^p operators for $p = 1$, 2 and 3 over a domain as shown in figure 2 with various number of grid points. The left column of the figure plots the eigenvalues for $\alpha = 0$ (coincident first and second grid points) and the right column plots the eigenvalues for an arbitrarily chosen value of $\alpha = 2/3$. All eigenvalues in each subfigure lie in the left half of the complex plane confirming time stability of the schemes using the D_1^p operators derived in section 3.1. The spectra for $\alpha = 0$ and $2/3$ in figure 3 for each D_1^p operator are similar, and the same is true for other values of α (not shown here for brevity), indicating that as $\alpha \to 0$ the system does not become stiffer. The conservative schemes derived in this study ensure that the cell sizes, defined by the entries of the W matrix, remain finite when $\alpha \to 0$, as discussed in section 3. As a result, the small-cell problem does not arise and a CFL $= c\Delta t/\Delta x$ value up to one is admissible with all D_1^p operators for this problem at all values of α.

Figures 4(a) and (b) depict the L_∞-error and the convergence rates of various schemes for the two values of α considered above. The profiles for other values of α are similar, indicating that the numerical truncation errors are not sensitive to the location of the EB. A global $p+1$ order-of-accuracy is observed for various schemes, consistent with the theoretical result of [41, 33]. The results in figures 3 and 4 confirm stable EB schemes that are up to fourth-order accurate globally (with a sixth-order non-dissipative scheme in the interior) for a linear inviscid (hyperbolic) problem. This will be confirmed for quasi-linear and non-linear inviscid problems in the subsequent sections.

5.2. One-dimensional diffusion problem

Consider equation (1) with $c = 0$ and $\mu = 0.1$ on a spatial domain $x_0 \leq x \leq x_n = 1$, as shown in figure 2. Let the initial and the boundary condition be given by (36) and (37), respectively, with

$$f(x) = \sin \pi x, \qquad g_1(t) = e^{-\mu \pi^2 t} \sin \pi x_0, \qquad g_2(t) = 0. \tag{50}$$

The exact solution to the problem is $\phi(x,t) = e^{-\mu\pi^2 t} \sin \pi x$. Figure 5 shows the the exact solution and the numerical solution computed using $D_2^{p=3}$ at various times.

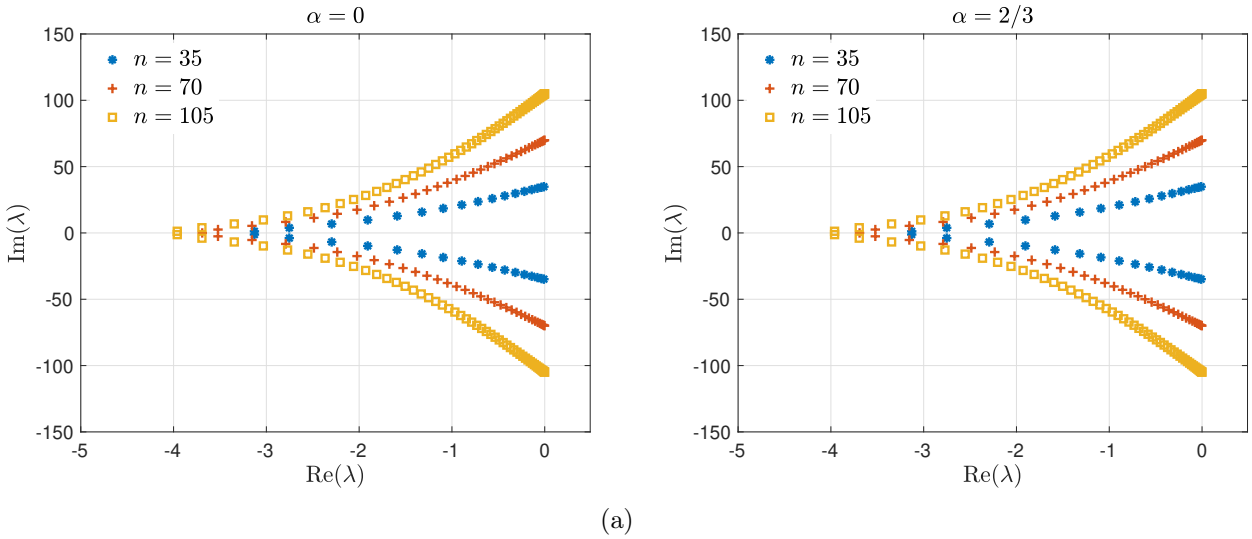
A semi-discretization to the above problem that imposes the BCs exactly is given by

$$\frac{d\widetilde{\mathbf{\Phi}}}{dt} = \mu \left(\widetilde{D}_2 \widetilde{\mathbf{\Phi}} + \widetilde{\mathbf{d}}_2^0 g_1 + \widetilde{\mathbf{d}}_2^n g_2 \right), \tag{51}$$

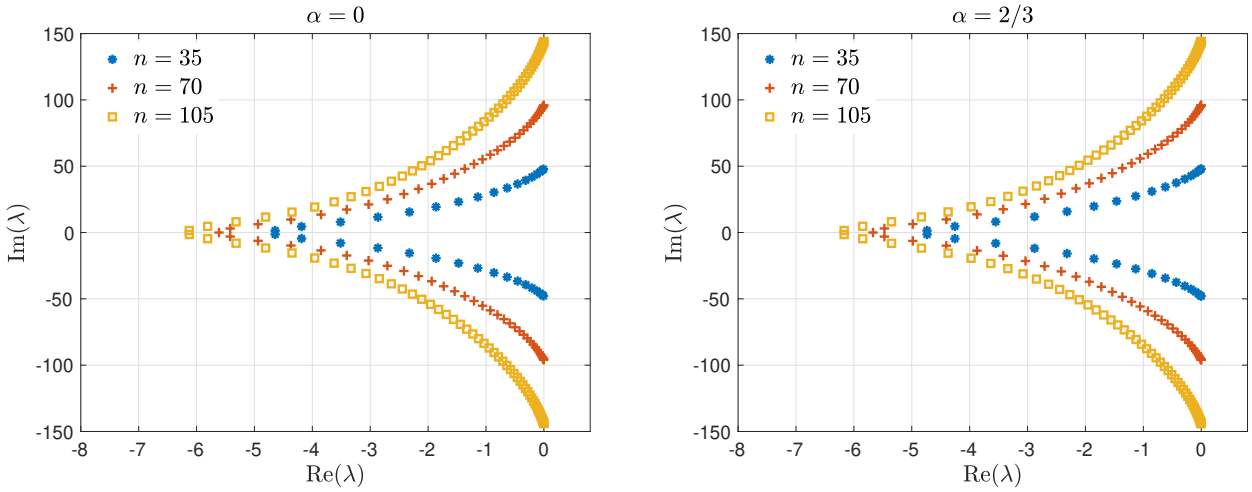
where, as in (41), $\widetilde{\mathbf{\Phi}}(t) = \begin{bmatrix} \Phi_1(t) & \cdots & \Phi_{n-1}(t) \end{bmatrix}^T$ and $\widetilde{D}_2$ denotes the $(n-1) \times (n-1)$ matrix obtained by removing the first and the last rows and columns of D_2. $\widetilde{\mathbf{d}}_2^0$ and $\widetilde{\mathbf{d}}_2^n$ denote the first and the last columns of D_2 after removing its first and last rows. The stability of the method (51) can be determined from the eigenvalues of the discretization (or system) matrix $\mathcal{M} = \mu \widetilde{D}_2$.

Figure 6 plots the eigenvalues of the system matrix, $\mathcal{M}$, using the D_2^p operators for $p = 1$, 2 and 3 over a domain as shown in figure 2 with various number of grid points. The left column of the figure shows the eigenvalues for $\alpha = 0$ and the right column shows the eigenvalues for $\alpha = 2/3$. The eigenvalue structure for other values of α (not shown here for brevity) is similar for each D_2^p operator, indicating that the spectral property of the derivative approximations is independent of α. In all cases, the eigenvalues lie strictly in the left half of the complex plane confirming time stability with the D_2^p operators derived in section 3.2. A comparison of figures 6(a), 6(b) and 6(c) corresponding to the D_2^p operators for $p = 1$, 2 and 3, respectively, suggests that the system matrix becomes progressively stiffer as p increases. Consequently, while a CFL $= \mu \Delta t / \left(\Delta x^2 / 2 \right)$ value up to one is admissible with $p = 1$ and 2, the $p = 3$ operator requires a CFL of 0.1 or smaller.

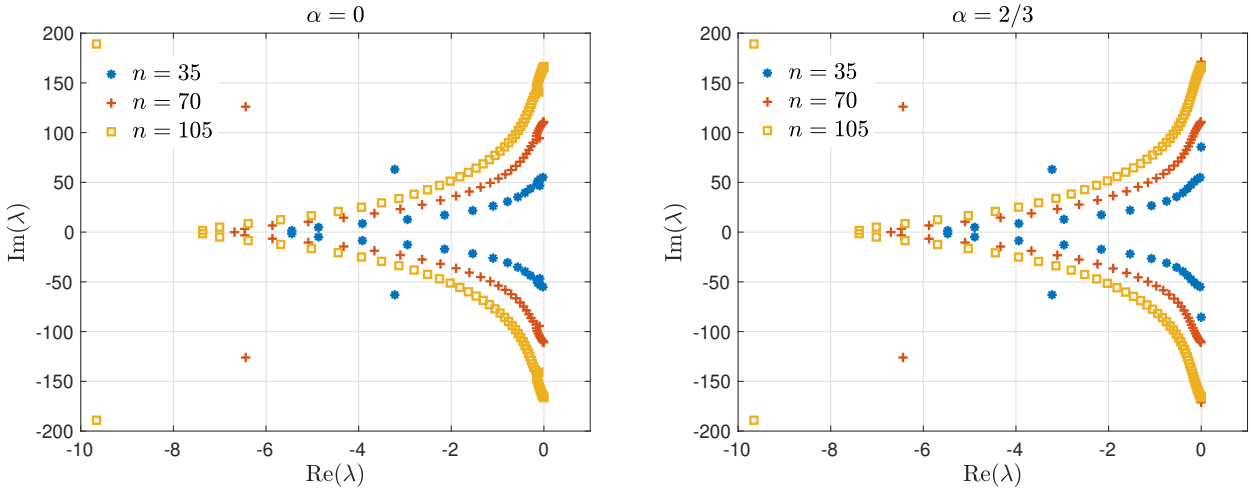
Figures 7(a) and (b) show the L_∞-error and the convergence rates of various schemes for the two values of α considered above. The profiles for other values of α are similar. A global $p+2$ order-of-accuracy is observed for $p \geq 2$ schemes, consistent with the theoretical result that for a differential operator of degree r in space, the boundary order-of-accuracy can be lowered by r locally if the scheme is energy stable [33, Chapter 3]. A global second-order accuracy is observed for the $p = 1$ scheme, where the interior scheme is second-order accurate. The results in figures 6 and 7 confirm stable EB schemes that are up to fifth-order accurate globally (with a sixth-order scheme in the interior) for a linear parabolic problem. This will be confirmed for quasi-linear and non-linear viscous problems in the subsequent sections.



(a)



(b)



(c)

Figure 3: Eigenvalue spectra of the system matrix to solve the one-dimensional scalar convection problem of section 5.1 using (a) $D_1^{p=1}$, (b) $D_1^{p=2}$, and (c) $D_1^{p=3}$ for $\alpha = 0$ (left column) and $\alpha = 2/3$ (right column) over a domain as shown in figure 2 with various number of grid points. Note that the axis limits are slightly different for the plots of different D_1^p operators.

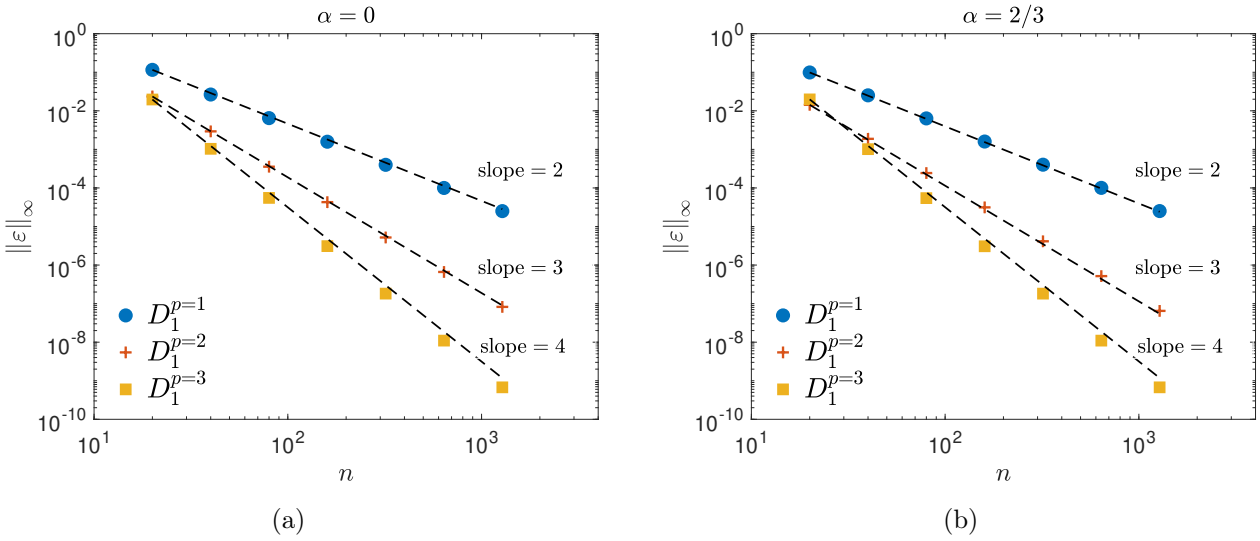


Figure 4: $L_\infty-$norm of the solution error and the convergence rate in solving the one-dimensional scalar convection problem of section 5.1 using various schemes for (a) $\alpha = 0$ and (b) $\alpha = 2/3$ on a grid as shown in figure 2. Error calculations performed at $t = 1.0$.

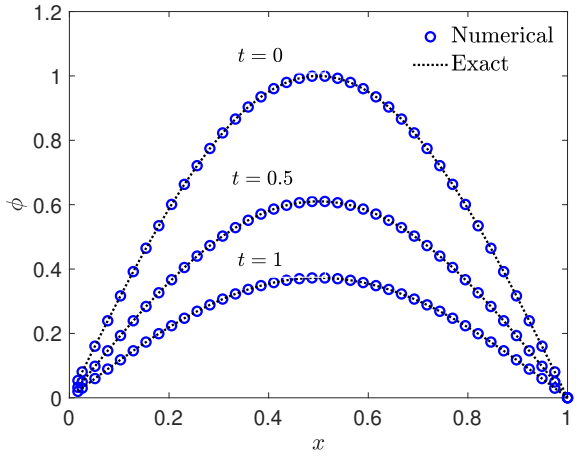


Figure 5: Solution at various times of the diffusion problem of section 5.2 using $D_2^{p=3}$ over the grid shown in figure 2 with $\alpha = 1/3$ and 40 grid points.

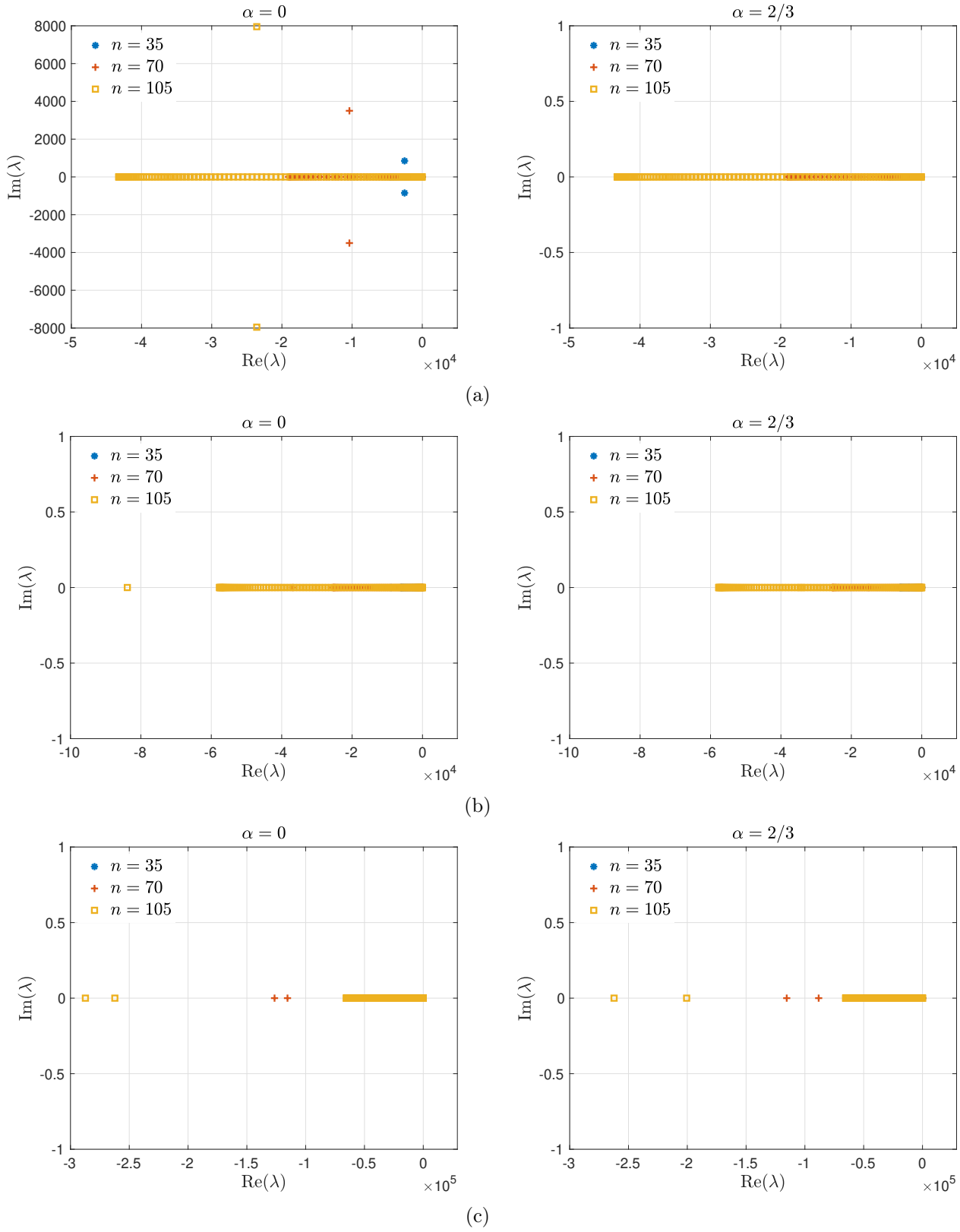


Figure 6: Eigenvalue spectra of the system matrix to solve the one-dimensional diffusion problem of section 5.2 using (a) $D_2^{p=1}$, (b) $D_2^{p=2}$, and (c) $D_2^{p=3}$ for $\alpha = 0$ (left column) and $\alpha = 2/3$ (right column) over a domain as shown in figure 2 with various number of grid points. Note the difference in axis limits for various D_2^p operators.

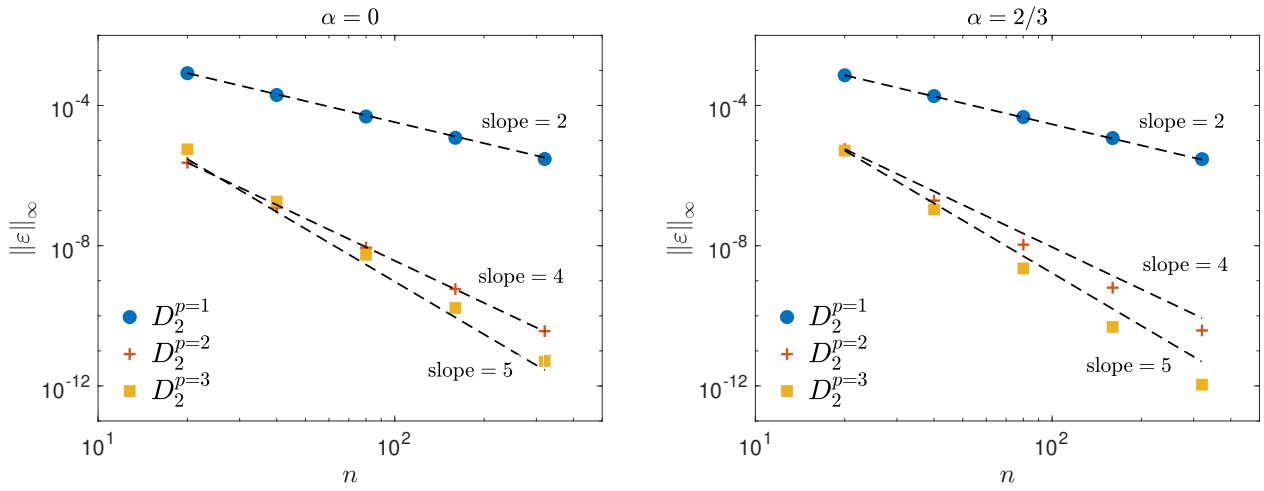


Figure 7: $L_\infty-$norm of the solution error and the convergence rate in solving the one-dimensional diffusion problem of section 5.2 using various schemes for (a) $\alpha = 0$ and (b) $\alpha = 2/3$ over a grid as shown in figure 2. Error calculations performed at $t = 1.0$.

5.3. One-dimensional viscous Burgers' problem

In this section, the stability and the accuracy of the derived EB stencils is examined for an incompletely parabolic non-linear problem. Consider the viscous Burgers' equation with a source term,

$$\frac{\partial U}{\partial t} + \frac{\partial}{\partial x}\left(\frac{U^2}{2}\right) = \mu\frac{\partial^2 U}{\partial x^2} + f_U. \qquad x_0 \le x \le x_n,\ t \ge 0, \tag{52}$$

The method of manufactured solutions [42] is employed to perform long-time simulations to assess the time stability of the derived schemes. The source term allows error evaluation from the exact solution, which is assumed to be

$$U(x,t) = \sin 2\pi (x - t) + C, \tag{53}$$

where $C = 1.0$ is a constant. (53) prescribes the initial and the boundary data. The source term, assuming $\mu = (2\pi)^{-1}$, is given by

$$f_U(x,t) = 8\pi \cos^3 \pi (x - t) \sin \pi (x - t). \tag{54}$$

Equation (52) is solved on a grid shown in figure 8, where the first and the last grid point have non-uniform spacing specified by α_l and α_r, respectively. Figure 9 shows the long-time solution errors from solving (52) with the initial condition

$$U(x,0) = \sin 2\pi x + 1, \tag{55}$$

and Dirichlet BCs

$$U(x_0,t) = g_1(t) = \sin 2\pi (x_0 - t) + 1, \qquad U(x_n,t) = g_2(t) = \sin 2\pi (x_n - t) + 1. \tag{56}$$

A semi-discretization of (52) that imposes the BCs (56) exactly is given by

$$\frac{d\widetilde{\mathbf{u}}}{dt} = -\widetilde{D}_1\left(\widetilde{\mathbf{u}}^2/2\right) + \mu\widetilde{D}_2\widetilde{\mathbf{u}} - \left[\widetilde{\mathbf{d}}_1^0\left(g_1/2\right) - \mu\widetilde{\mathbf{d}}_2^0\right] g_1 - \left[\widetilde{\mathbf{d}}_1^n\left(g_2/2\right) - \mu\widetilde{\mathbf{d}}_2^n\right] g_2, \tag{57}$$

where $\widetilde{\mathbf{u}}(t) = \begin{bmatrix} u_1(t) & \cdots & u_{n-1}(t) \end{bmatrix}^T$ is the solution vector $\mathbf{u}$ without the first and the last element (that correspond to the grid points where the boundary data is injected). The operator $\widetilde{D}_{1,2}$ and the vectors $\widetilde{\mathbf{d}}_{1,2}^0$ and $\widetilde{\mathbf{d}}_{1,2}^n$ are defined as in section 5.2.

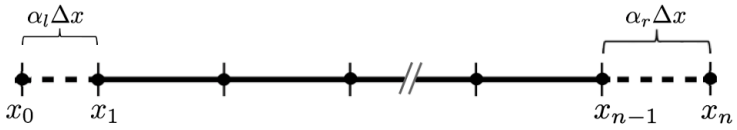


Figure 8: One-dimensional grid with non-uniform spacing at the first and the last grid point $\left(0 \leq \alpha_{l,r} \leq 1\right)$.

The long-time solution error is evaluated for various α_l and α_r values. The results from the limit cases of $\alpha_l = \alpha_r = 0$ and $\alpha_l = \alpha_r = 1$ are shown in figure 9(a) and 9(d), respectively. Two sets of intermediate α_l and α_r values are considered in figures 9(b) and 9(c). In all cases, the error remains nearly constant with time indicating a time-stable behavior for all D_1 and D_2 operators. The convergence results from various schemes for $\alpha_l = \alpha_r = 0$ and $\alpha_l = 1/3$, $\alpha_r = 2/3$ are illustrated in figure 10(a) and 10(b), respectively. Results from other α_l and α_r values are similar. A global $p+2$ order of accuracy is observed with the $p \geq 2$ schemes and a second-order accuracy with the $p = 1$ scheme.

To test the robustness of the developed schemes with Neumann BCs, (52) is solved with the BCs:

$$U\left(x_0, t\right) = g_1\left(t\right) = \sin 2\pi\left(x_0 - t\right) + 1, \qquad \left.\frac{\partial U}{\partial x}\right|_{(x_n, t)} = g_3\left(t\right) = 2\pi \cos 2\pi\left(x_n - t\right). \tag{58}$$

A semi-discretization of (52) that imposes the BCs (58) is given by

$$\frac{d\widehat{\mathbf{u}}}{dt} = -\widehat{D}_1\left(\widehat{\mathbf{u}}^2/2\right) + \mu \widehat{D}_2 \widehat{\mathbf{u}} - \left[\widehat{\mathbf{d}}_1^0\left(g_1/2\right) - \mu \widehat{\mathbf{d}}_2^0\right] g_1 - \mu \widehat{\mathbf{e}}_n\left[\left(S_1 \mathbf{u}\right)_n - g_3\right], \tag{59}$$

where the notation $\widehat{\bullet}$ over vectors and matrices are defined as in (44).

Figure 11 shows the solution errors from long-time integrations of (59) with the operators for $p = 1$, 2 and 3. The errors from the case of $\alpha_l = \alpha_r = 0$, where the first two and the last two grid points coincide, is depicted in figure 11(a), and the errors from the grid with $\alpha_l = 1/3$ and $\alpha_r = 2/3$ is depicted in figure 11(b). In both cases, the error remains bounded at all times indicating time stability. The errors from other α_l and α_r values and with different number of grid points (not shown here) yield similar time-stable behavior. The convergence results from various schemes are shown in figure 12 for the two sets of α_l and α_r values used in figure 11. As in the case of Dirichlet BCs (figure 10), a global order-of-accuracy of approximately $p+2$ is observed with $p \geq 2$ schemes and a second-order accuracy with $p = 1$ scheme.

The results in this and the previous sections illustrate time stability and high-order accuracy of the EB stencils derived from the procedures outlined in section 3 for one-dimensional inviscid as well as viscous calculations. They also show that the proposed schemes do not have a small-cell problem. In the following sections, the derived EB stencils' performance is evaluated for two-dimensional problems.

5.4. Two-dimensional variable-coefficient convection problem

Consider the two-dimensional scalar inviscid problem

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = 0, \qquad -L \leq x, y \leq L \qquad t \geq 0, \tag{60a}$$

with the velocity vector $\mathbf{u} = (u, v)^T$, where

$$u(x, y) = \frac{\partial r}{\partial x}, \;\; v(x, y) = \frac{\partial r}{\partial y}, \tag{60b}$$

$$r\left(x, y\right) = \sqrt{\left(x - x_0\right)^2 + \left(y - y_0\right)^2}. \tag{60c}$$

where $L = 1$, $x_0 = 0$ and $y_0 = 0$. Let the initial and the boundary conditions be

$$\phi(x, y, t = 0) = \sin 2\pi r, \tag{61}$$

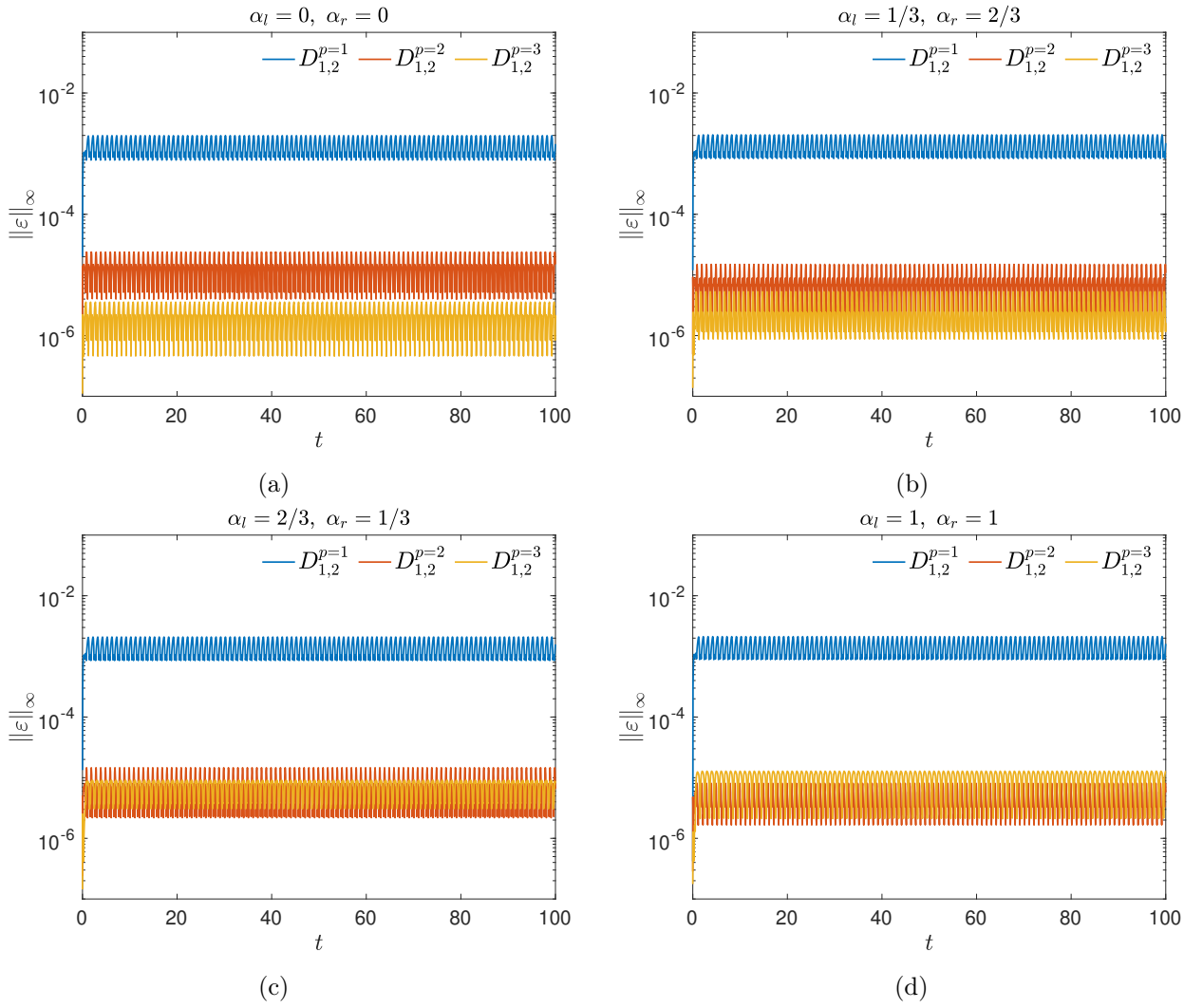


Figure 9: $L_\infty-$error from the long-time simulations of (52) with the initial condition (55) and the boundary condition (56) over the grid shown in figure 8 with 80 grid points and (a) $\alpha_l = \alpha_r = 0$, (b) $\alpha_l = 1/3$, $\alpha_r = 2/3$, (c) $\alpha_l = 2/3$, $\alpha_r = 1/3$, and (d) $\alpha_l = \alpha_r = 1$.

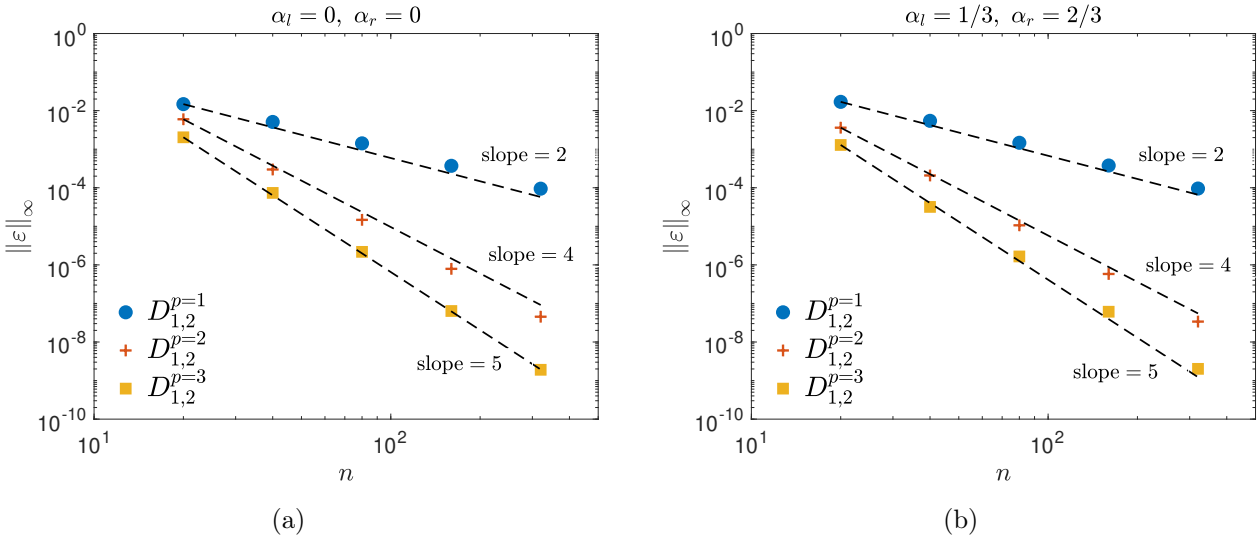


Figure 10: $L_\infty-$norm of the solution error and the convergence rate in solving (52) with the initial condition (55) and the boundary condition (56). Semi-discretization (57) is used with various schemes for (a) $\alpha_l = \alpha_r = 0$ and (b) $\alpha_l = 1/3$, $\alpha_r = 2/3$ over a grid as shown in figure 8. Error calculations performed at $t = 1.0$.

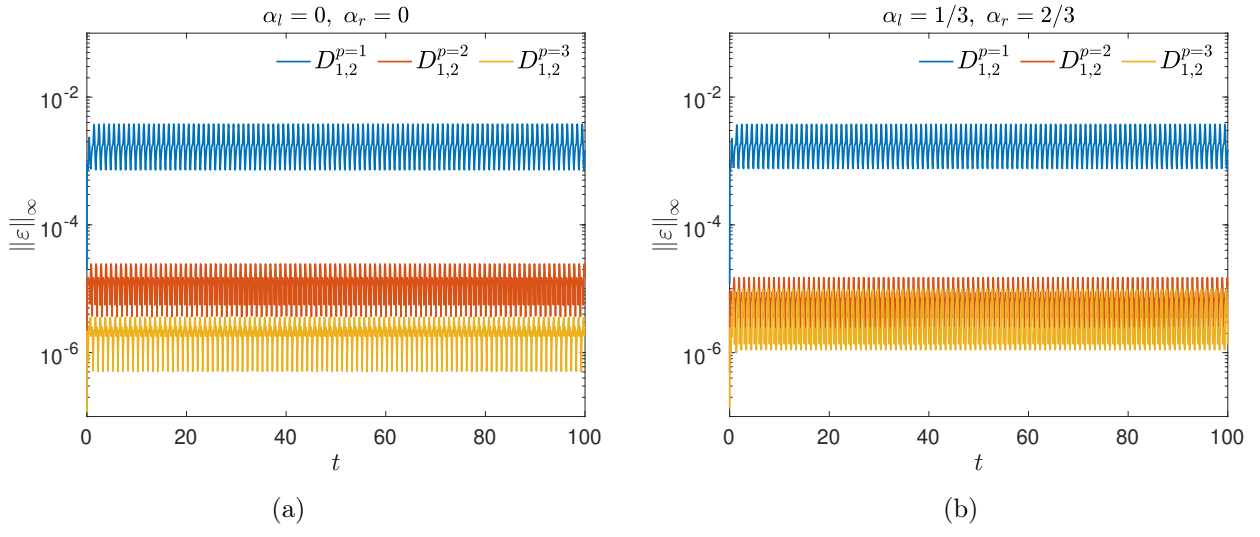
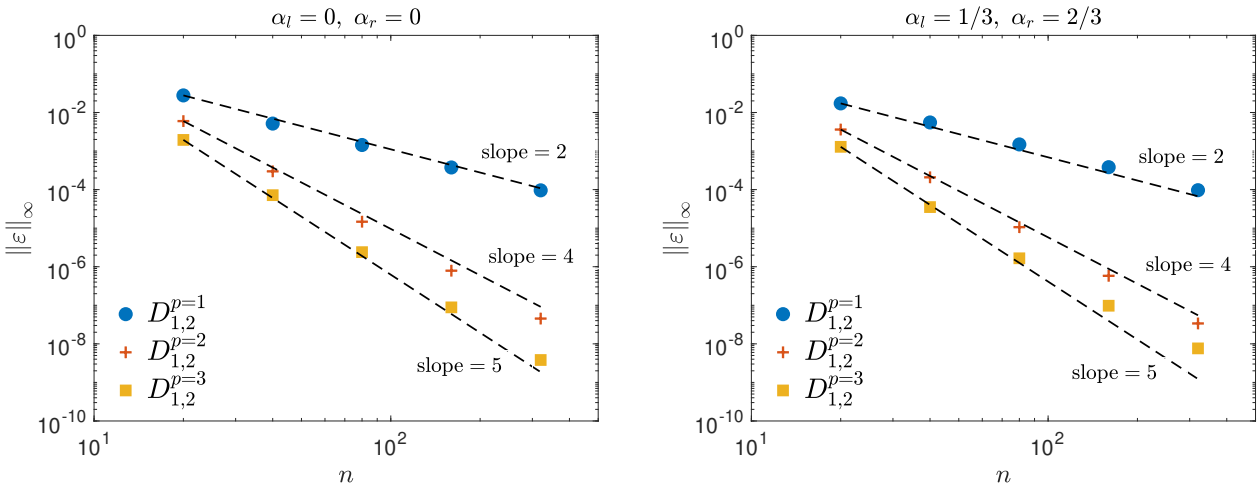


Figure 11: $L_\infty-$error from the long-time simulations of (52) with the initial condition (55) and the boundary condition (58). Semi-discretization (59) is used over the grid shown in figure 8 with 80 grid points and (a) $\alpha_l = \alpha_r = 0$, and (b) $\alpha_l = 1/3$, $\alpha_r = 2/3$.



(a) (b)

Figure 12: $L_\infty-$norm of the solution error and the convergence rate in solving (52) with the initial condition (55) and the boundary conditions (58). Semi-discretization (59) is used with operators of various order of accuracy for (a) $\alpha_l = \alpha_r = 0$ and (b) $\alpha_l = \alpha_r = 1$ over a grid as shown in figure 8. Error calculations performed at $t = 1.0$.

and

$$\phi(x, y, t) = \sin 2\pi\,(r_0 - t) \qquad \text{at} \qquad \sqrt{(x - x_0)^2 + (y - y_0)^2} = r_0, \tag{62}$$

respectively. The exact solution to the problem is

$$\phi(x, y, t) = \sin 2\pi\,(r - t)\,. \tag{63}$$

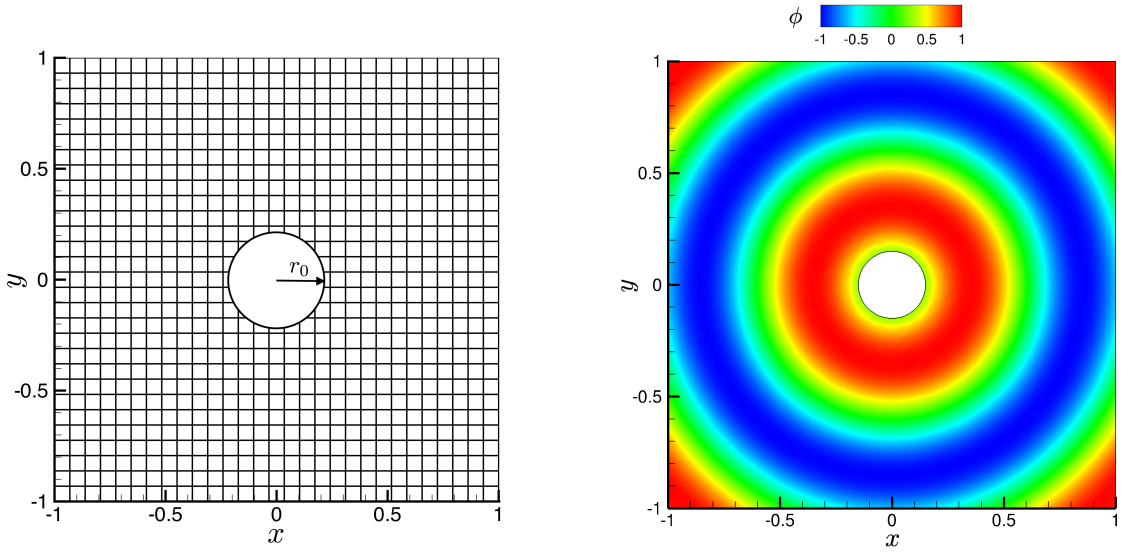
The initial-boundary value problem (60)–(62) is solved on an EB grid as shown in figure 13(a). The problem is well posed with the boundary conditions (62) applied at $r = r_0$. Figure 13(b) depicts the solution at $t \approx 5$ using $r_0 = 0.15$.

The stability of the derived EB schemes is examined from long-time simulations with various number of grid points. The $L_\infty-$error from these calculations with $N \times N$ grid points is shown in figures 14(a)–(c). As evident, all derived first-derivative approximations, D_1^p, produce stable results and the error reduces with increase in the number of grid points and increase in the order of accuracy. The time steps in the long-time simulations are calculated using

$$\Delta t = \text{CFL} \min_{i,j \in \Omega_f} \left(\frac{|u_{ij}|}{\Delta x} + \frac{|v_{ij}|}{\Delta y} \right)^{-1}, \tag{64}$$

with a CFL of 0.5, illustrating that the derived schemes do not have a small-cell problem. The minimum in (64) is calculated over all the fluid grid points and Δx and Δy denote the uniform grid spacing in the Cartesian directions in the interior of the domain.

Figure 14(d) shows the errors and the convergence rates from various D_1^p operators. As noted in section 5.1 and expected from the theoretical result of [41], an approximately $p + 1$ global order of accuracy is observed for various schemes. Additional simulations with different domain sizes and circle locations and radii (not shown here for brevity) displayed similar stable results confirming the robustness of the derived EB stencils for this problem.



(a) (b)

Figure 13: (a) EB grid over which the two-dimensional variable-coefficient convection problem of section 5.4 is solved, and (b) the solution to the problem at $t \approx 5$ using $D_1^{p=2}$.

5.5. Two-dimensional isentropic convecting vortex

In this section, we examine the performance of the derived EB stencils for solving the Euler equations. The conservative form of the two-dimensional Euler equations is given by

$$\frac{\partial Q}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = 0,$$

$$Q = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ E \end{bmatrix}, \qquad F = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u\,(E+p) \end{bmatrix}, \qquad G = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v\,(E+p) \end{bmatrix}, \tag{65}$$

$$E = \frac{p}{\gamma - 1} + \rho\left(\frac{u^2 + v^2}{2}\right),$$

where u, v are the Cartesian velocity components, ρ denotes the density, p is the pressure and E is the total energy. γ denotes the ratio of specific heats.

The Euler equations is solved for the propagation of a compressible isentropic vortex on domains with different EB shapes, as shown in figure 15. The circle in figure 15(a) has a diameter of 0.6, the square in figure 15(b) has sides of length 0.4 and the equilateral triangle in figure 15(c) has sides of length 0.6. These EB configurations for the Euler equations help determine the robustness of the developed schemes for non-linear inviscid problems over a variety of EBs. The domain boundaries in the x- and the y-direction are assumed to be periodic allowing the vortex to loop through the domain multiple times, providing an assessment of the long-time behavior of the EB stencils. The exact solution to the problem is given by

$$\begin{aligned}
&\rho = \left(1 - \frac{\varpi^2(\gamma-1)}{8\pi^2 c_0^2} e^{1-\varphi^2 r^2}\right)^{\frac{1}{\gamma-1}}, \qquad u = u_0 - \frac{\varpi}{2\pi}\varphi(y - y_0 - v_0 t) e^{\frac{1-\varphi^2 r^2}{2}}, \\
&v = v_0 + \frac{\varpi}{2\pi}\varphi(x - x_0 - u_0 t) e^{\frac{1-\varphi^2 r^2}{2}}, \qquad E = \frac{p}{\gamma-1} + \frac{1}{2}\rho(u^2 + v^2), \\
&p = \rho^\gamma, \qquad\qquad r^2 = (x - x_0 - u_0 t)^2 + (y - y_0 - v_0 t)^2,
\end{aligned} \tag{66}$$

where $(x_0,\, y_0)$ denotes the initial position of the vortex, (u_0, v_0) denotes the vortex convective velocity, φ is a scaling factor and ϖ denotes the non-dimensional circulation. We use $(x_0,\, y_0) = (-0.6,\, 0)$, $v_0 = 0$,

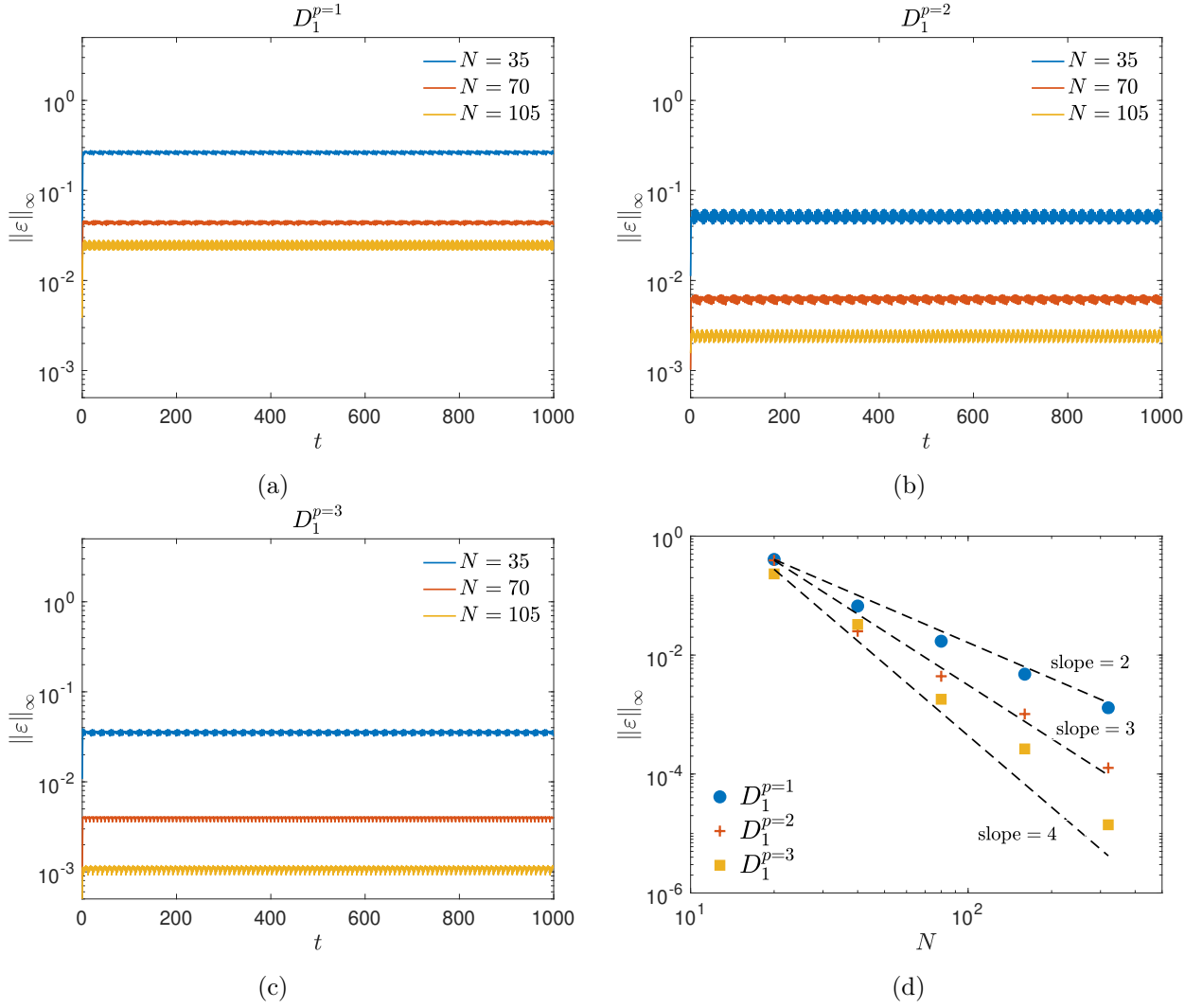


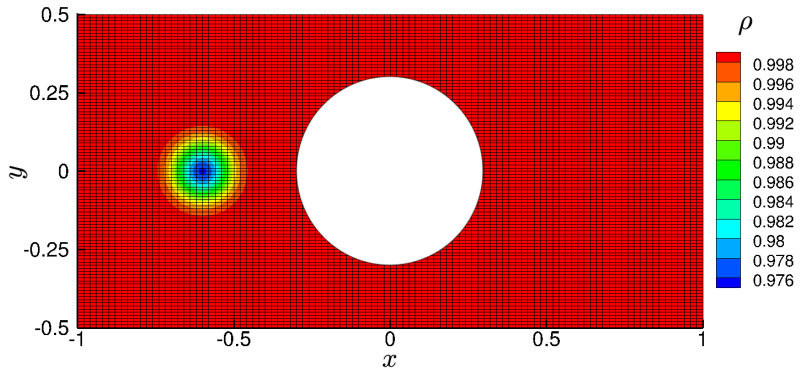
Figure 14: $L_\infty-$error from long-time simulations of (60)–(62) using $N \times N$ grid points and derivative approximations D_1 for (a) $p = 1$, (b) $p = 2$, and (c) $p = 3$. Subfigure (d) shows the $L_\infty-$error and the convergence rate in solving the problem using various schemes. The errors are calculated at $t = 1.0$.

$\gamma = 1.4$, $\varphi = 11$ and $\varpi = 1$. All quantities in (66) are non-dimensional, obtained from the density scale $= \rho_0^*$, velocity scale $u_0^* = \frac{c_0^*}{\sqrt{\gamma}}$, unit length scale and pressure scale $= \rho_0^* u_0^{*2}$, where $*$ denotes the dimensional quantities. The non-dimensional ambient speed of sound is $c_0 = \sqrt{\gamma}$.

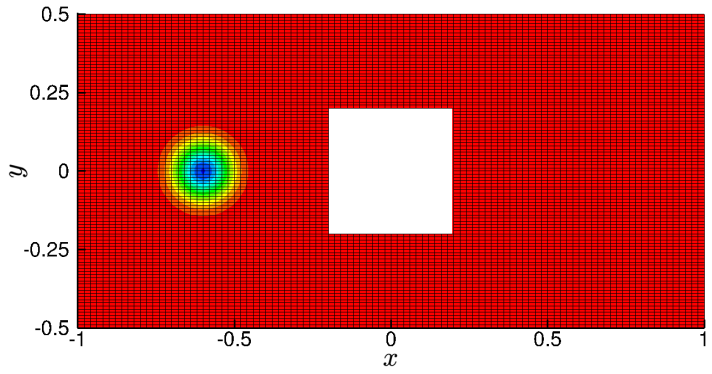
A characteristic decomposition, where only the incoming characteristic variables are specified at the boundary [43], is employed at the EB to apply the BCs using (66). Figure 16 shows the L_∞- and the L_2-norm of the density and the velocity magnitude error for various EB configurations (depicted in figure 15). Both the L_∞ and the L_2 errors are calculated over the grid points in the fluid domain, Ω_f (see figure 1). While the L_∞-norm provides the maximum error, the L_2-norm offers an error estimate for the entire fluid domain. A large difference between the two would imply that the errors are concentrated at only a few grid points. Figure 16 shows the errors from a subsonic convective velocity where each (embedded) boundary grid point has incoming as well as outgoing characteristic waves. The time-periodic peaks in the error profiles, noticeable especially in the $p = 1$ results, correspond to the times when the vortex crosses the EB. The vortex traversal through the EB generates some numerical reflections, thus increasing the errors, however, the errors do not grow to destabilize the simulation. The EB stencils derived to satisfy the secondary conservation constraints ensure stability without the use of global numerical filters/artificial dissipation. The results from the D_1 operators for $p = 1$ and 3 are plotted in figure 16; the results from the $p = 2$ operators (not shown here to avoid clutter) are similar. In the duration of the simulation, the vortex crosses the EB ten times allowing a long-time stability assessment. The error profiles and magnitudes for various EB configurations are similar for each operator indicating that the derived EB schemes are not very sensitive to the EB geometry.

To further examine the behavior of the EB stencils, figures 17(a)–(c) depict the L_∞-norm of the velocity magnitude error at $u_0 = 1$ (subsonic) and $u_0 = 2$ (supersonic) and at various grid resolutions using the D_1 operators for $p = 1$, 2 and 3. The errors remain bounded with time in all cases indicating a time-stable behavior and, as expected, the errors reduce with increase in the number of grid points for each operator. All calculations were performed at a CFL number of 0.5, showing that for this non-linear problem with various EB configurations the developed method does not have a small-cell problem. The convergence rates from various D_1 operators are depicted in figure 17(d). An approximately $p + 1$ global order of accuracy is observed for various schemes.

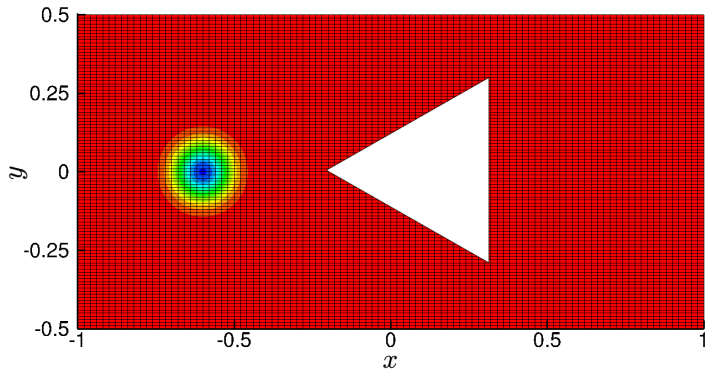
The results in this section demonstrate the robustness of the derived EB stencils for a non-linear inviscid problem. It is thus expected that these stencils will provide stable and accurate solutions with physical viscosity in viscous calculations. The next section discusses one such problem.



(a)



(b)



(c)

Figure 15: Computational domain/grid and initial density contour for isentropic convecting vortex on various EB domains labeled: (a) Circle, (b) Square, and (c) Triangle. The legend for the density contours is the same for all plots.

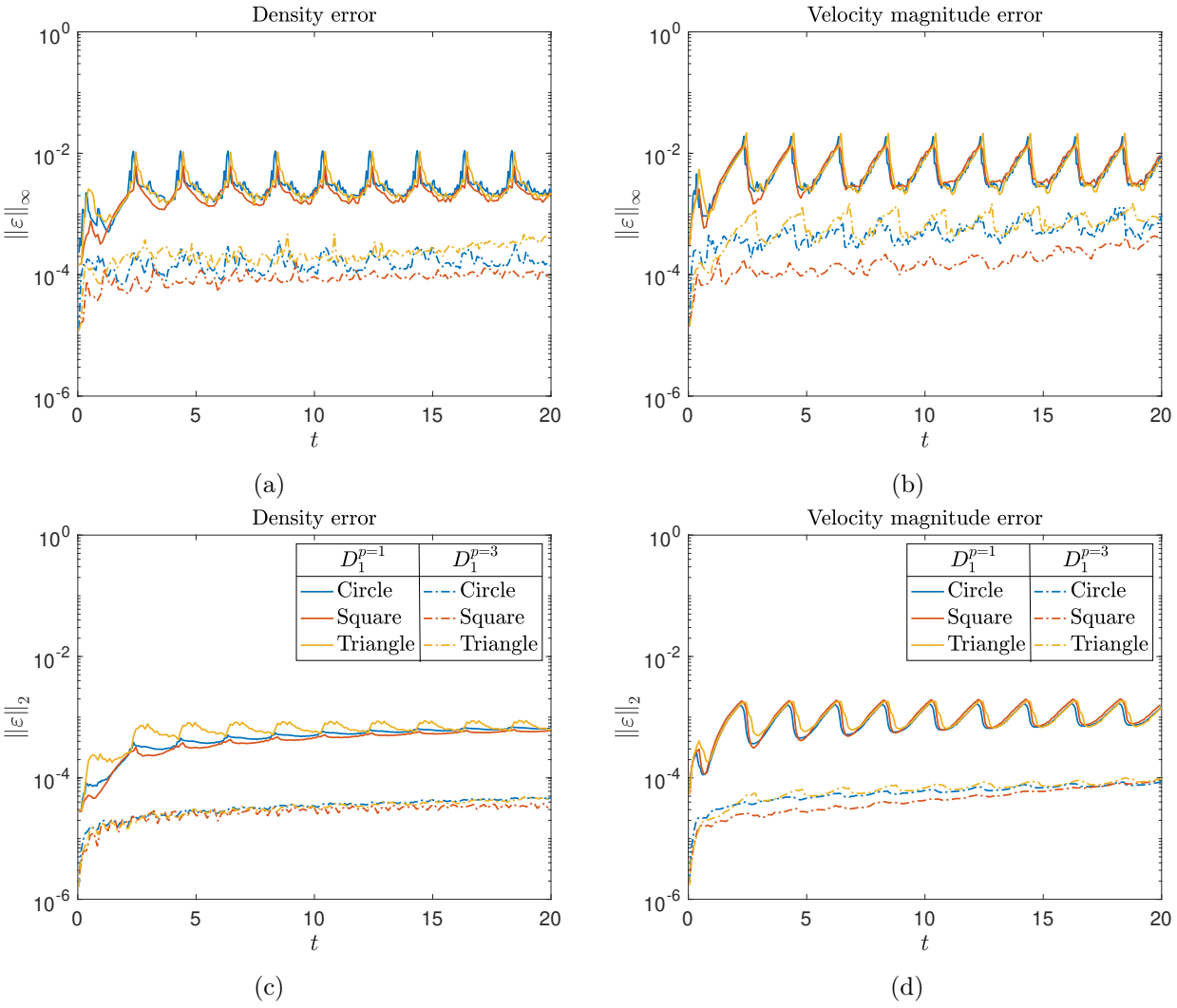


Figure 16: $L_\infty-$error of (a) density and (b) velocity magnitude & L_2-error of (a) density and (b) velocity magnitude from solving the two-dimensional Euler equations (65) using the D_1 operators for $p = 1$ and 3 on various EB grids shown in figure 15. The Euler equations are solved for the isentropic convecting vortex problem at subsonic convective velocity $u_0 = 1$ using 200×200 grid points. The legend is the same for all plots.

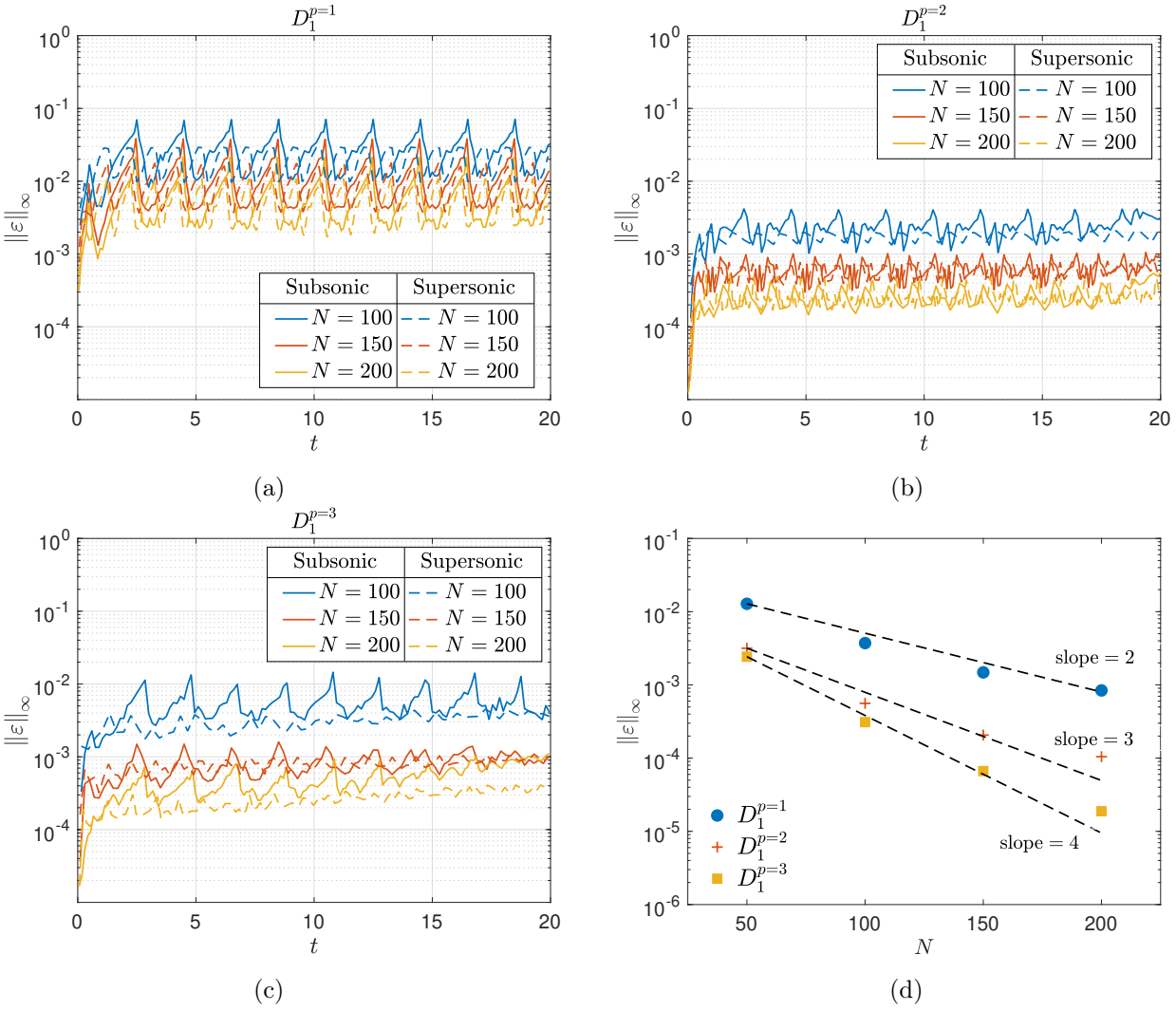


Figure 17: L_∞-norm of the velocity magnitude error from long-time simulations of the Euler equations for the convecting vortex problem on the EB grid shown in figure 15(c) using subsonic ($u_0 = 1$) and supersonic ($u_0 = 2$) convective velocity with $N \times N$ grid points and D_1 operators for (a) $p = 1$, (b) $p = 2$ and (c) $p = 3$. Subfigure (d) shows the $L_\infty-$error and the convergence rates from solving the problem using various schemes. The errors are calculated at $t = 2.0$ with $u_0 = 1$.

5.6. Two-dimensional convection-diffusion problem

To examine the performance of the derived EB stencils for a two-dimensional viscous problem, we consider equation (60) supplemented by viscous and source terms,

$$\frac{\partial \phi}{\partial t} + u\frac{\partial \phi}{\partial x} + v\frac{\partial \phi}{\partial y} = \mu\left(\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2}\right) + f, \qquad -L \leq x, y \leq L \qquad t \geq 0, \tag{67a}$$

$$u(x,y) = \frac{\partial r}{\partial x}, \;\; v(x,y) = \frac{\partial r}{\partial y}, \tag{67b}$$

$$r\,(x,y) = \sqrt{(x-x_0)^2 + (y-y_0)^2}, \tag{67c}$$

where $L = 3/2$, $x_0 = 0$ and $y_0 = 0$ is assumed. For $f = 2\pi\mu e^{-2\pi\mu t}\left[(2\pi - 1)\sin 2\pi\,(r-t) - \frac{\cos 2\pi(r-t)}{r}\right]$ and the initial condition

$$\phi(x,y,0) = \sin 2\pi r, \tag{68}$$

the exact solution of (67) is given by

$$\phi(x,y,t) = e^{-2\pi\mu t}\sin 2\pi\,(r-t)\,. \tag{69}$$

The above problem is solved on an EB grid similar to figure 13(a) with $\mu = 10^{-3}$ and boundary conditions specified using (69). The performance of the developed EB approach is evaluated with Dirichlet as well as Neumann BCs at the outer boundaries ($x = -L, L$ and $y = -L, L$), while Dirichlet BC is imposed at $r = r_0$. Figure 18(a) and 18(b) plot the L_∞-norm of the solution error from long-time simulations with Dirichlet and Neumann BC at the outer boundary, respectively. The figure shows the results from the $p = 1$, 2 and 3 derivative approximations with 80×80 grid points over the domain. Results from other tested grid resolutions were similar. The solution magnitude decays with time, as evident from (69), and accordingly the error magnitude decays as well. Increase in the order of accuracy reduces the errors, as expected. and the errors in the run with the Neumann BC at the outer boundary are slightly larger than those with the Dirichlet BC. All simulations were performed with time step evaluated from

$$\Delta t = \mathrm{CFL}\min_{i,j\in\Omega_f}\left[\frac{|u_{ij}|}{\Delta x} + \frac{|v_{ij}|}{\Delta y} + 2\mu\left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2}\right)\right]^{-1}, \tag{70}$$

using a CFL of 0.5. The minimum in (70) is calculated over all the fluid grid points and Δx and Δy denote the uniform grid spacing in the Cartesian directions in the interior of the domain. In general, ensuring time stability for inviscid problems is more challenging, therefore, in light of the stable results of section 5.4, time stability of the derived EB stencils for the above problem is not surprising.

5.7. Three-dimensional variable-coefficient convection problem

The dimensionally-split discretization proposed in the present EB method simplifies the extension from two to three dimensions. The one-dimensional operations of the first two dimensions are simply applied in the third dimension and, since the proposed approach avoids surface/volume reconstructions, no additional complexity arises in the code implementation. To evaluate the performance of the derived EB schemes for a three-dimensional problem, the inviscid (variable-coefficient) convection equation (60a) is solved with the velocity vector $\mathbf{u} = (u, v, w)^T$, where

$$u(x,y,z) = \frac{\partial r}{\partial x}, \;\; v(x,y,z) = \frac{\partial r}{\partial y},, \;\; w(x,y,z) = \frac{\partial r}{\partial z}, \tag{71}$$

$$r\,(x,y,z) = \sqrt{(x-x_0)^2 + (y-y_0)^2 + (z-z_0)^2}, \tag{72}$$

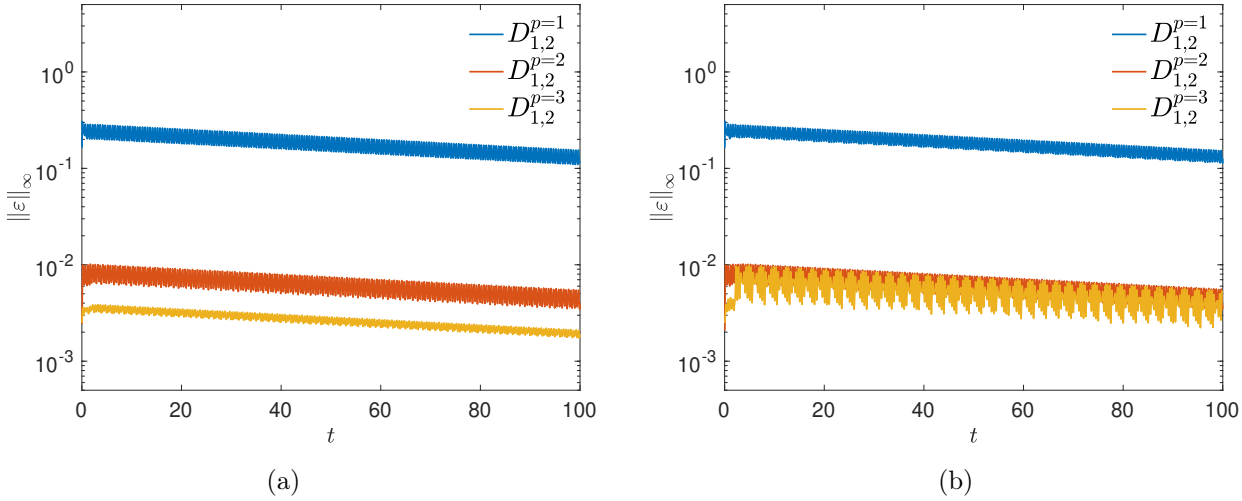


Figure 18: $L_\infty-$error from long-time simulations of (67) using 80×80 grid points and various EB derivative approximations with Dirichlet BC at $r = r_0$ and (a) Dirchlet BC and (b) Neumann BC at $x = -L, L$ and $y = -L, L$.

and $L = 1$, $x_0 = 0$ and $y_0 = 0$ is assumed. Let the initial and the boundary conditions be

$$\phi(x, y, z, t = 0) = \sin 2\pi r, \tag{73}$$

and

$$\phi(x, y, z, t) = \sin 2\pi \left(r_0 - t\right) \qquad \text{at} \qquad \sqrt{\left(x - x_0\right)^2 + \left(y - y_0\right)^2 + \left(z - z_0\right)^2} = r_0, \tag{74}$$

respectively. The exact solution to the problem is $\phi = \sin 2\pi \left(r - t\right)$.

The above problem is solved on a Cartesian EB grid, depicted in figure 19(a), with $r_0 = 0.15$. The stability of the derived EB stencils is evaluated from long-time simulations with various number of grid points. The L_∞-error from the calculations using 60 grid points in each direction with a CFL of 0.5 is shown in figure 19(b). A non-growing error in time with each first-derivative EB approximation indicates time-stable behavior for a three-dimensional inviscid problem without the use of artificial dissipation or filters, and a reasonable CFL value illustrates that the derived schemes do not have a small-cell problem. The convergence rates from various D_1^p operators is examined in figure 19(c). Approximately $p + 1$ global order of accuracy is observed for various schemes. Additional simulations with different sphere sizes/locations, grid resolutions, and domain sizes (not included here for brevity) showed similar stable results confirming the robustness of the derived EB stencils in extension to three-dimensional problems.

5.8. Two-dimensional flow past a circular cylinder

This section discusses the application of the proposed EB method to compute the flow over a circular cylinder. The two-dimensional compressible Navier-Stokes equations are solved, given in terms of non-dimensional quantities by

$$\frac{\partial Q}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = \frac{1}{Re}\left(\frac{\partial F_v}{\partial x} + \frac{\partial G_v}{\partial y}\right), \tag{75}$$

where Q, F, and G are defined as in (65), Re is the Reynolds number, and F_v and G_v denote the viscous fluxes (e.g. [44])

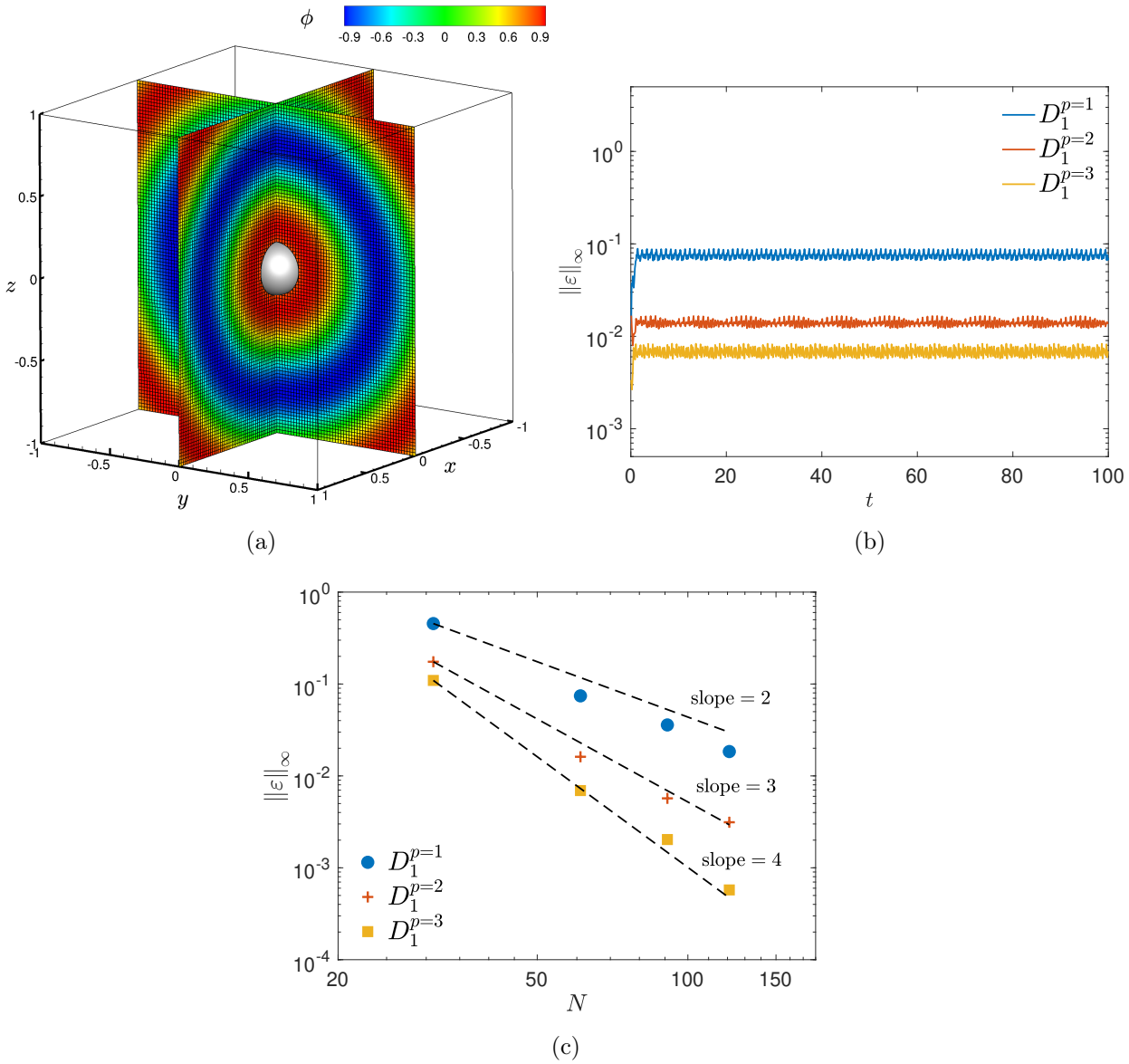


Figure 19: Results from the application of various EB schemes to solving the three-dimensional variable-coefficient convection problem. (a) The computational domain with two slices showing the spherical embedded boundary, (b) $L_\infty-$error from long-time simulations of the problem using $60 \times 60 \times 60$ grid points and various EB derivative approximations with Dirichlet BC at $r = r_0$, and (c) $L_\infty-$error and the convergence rates from solving the problem using various schemes, where the errors are calculated at $t = 10$.

$$F_v = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ u\tau_{xx} + v\tau_{xy} + \frac{\mu}{Pr(\gamma-1)}\frac{\partial(c^2)}{\partial x} \end{bmatrix}, \qquad G_v = \begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ u\tau_{xy} + v\tau_{yy} + \frac{\mu}{Pr(\gamma-1)}\frac{\partial(c^2)}{\partial y} \end{bmatrix}. \tag{76}$$

The stress tensor is given by

$$\tau_{ij} = \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right),$$

where $(x_1, x_2) \equiv (x, y)$ and $(u_1, u_2) \equiv (u, v)$ is adopted. $c^2 = \gamma p/\rho$ is the speed of sound for ideal gases, μ denotes the dynamic viscosity, and Pr is the Prandtl number. The non-dimensional quantities are obtained using the freestream values of density ρ^*_∞, speed of sound c^*_∞, viscosity μ^*_∞, and the cylinder diameter D^*, where the superscript $*$ denotes dimensional quantities and the subscript ∞ denotes freestream values. The pressure is non-dimensionalized using $\rho^*_\infty c^{*2}_\infty$ and the temperature using $(\gamma-1)\,T^*_\infty$, providing the non-dimensional perfect gas equation of state $p = \left(\frac{\gamma-1}{\gamma}\right)\rho T$.

Equation (75) is solved for the cylinder flow at a Reynolds number Re_D (based on the cylinder diameter) of 100 and a freestream Mach number of 0.2. The computational domain extends from $-5D$ to $20D$ in the streamwise direction and from $-5D$ to $5D$ in the crossstream direction. The instantaneous vorticity magnitude contours from a section of the domain is shown in figure 20. Subsonic inflow boundary conditions are applied at $x = -5D$ and non-reflecting outflow boundary conditions [45, 46] at other (outer) boundaries. Isothermal ($T_{\text{wall}} = T_\infty$) no-slip boundary conditions are applied at the cylinder walls, i.e., at the EB points.

The $D^{p=2}_{1,2}$ operators are used for the results discussed here. The $p = 1$ and 3 operators yielded similar results with larger grid-point requirement for the low-order $p = 1$ scheme. To test the performance of the D_2 operators, the viscous terms in (75) were discretized in a non-conservative form. For a conservative treatment of the viscous terms, repeated first derivative approximations may be used or boundary stencils for second derivatives involving variable coefficients (e.g. [47]) need to be derived. A detailed discussion of the latter is beyond the scope of this article.

Time advancement of the solution at the EB points requires derivative calculations along the direction in which a grid line cuts the EB as well as the direction perpendicular to it. For example, at the EB points $\mathcal{A}$ and $\mathcal{B}$ in figure 21, the solution advancement requires derivative calculations in both the x- and the y-directions. While the y-derivative at $\mathcal{A}$ and the x-derivative at $\mathcal{B}$ may be computed using the stencils derived in section 3, additional grid points shown in green and red (in figure 21) are required for x-derivative calculations at $\mathcal{A}$ and y-derivative at $\mathcal{B}$, respectively. In this study, values at those grid points are obtained using linear interpolation from the neighboring points.

The lift and drag forces acting on the cylinder is calculated from a numerical approximation of the integral (e.g. [48])

$$F_i = \oint (\sigma_{ij} n_j)\, ds,$$

where n_j denotes the components of the unit vector normal to the EB and $\sigma_{ij} = \tau_{ij} + p\delta_{ij}$. The drag coefficient from the $Re_D = 100$ simulation (shown in figure 20) is 1.37 ± 0.01 and the lift coefficient is ± 0.34, which compares favorably with the values reported in previous studies (e.g. [48, 49]).

6. Conclusions

High-order finite-difference embedded boundary (EB) schemes are presented for high-fidelity fluid-flow simulations over practical geometries with structured grids. EB stencils are derived for centered (non-dissipative) interior schemes to satisfy the primary and the secondary conservation constraints. These

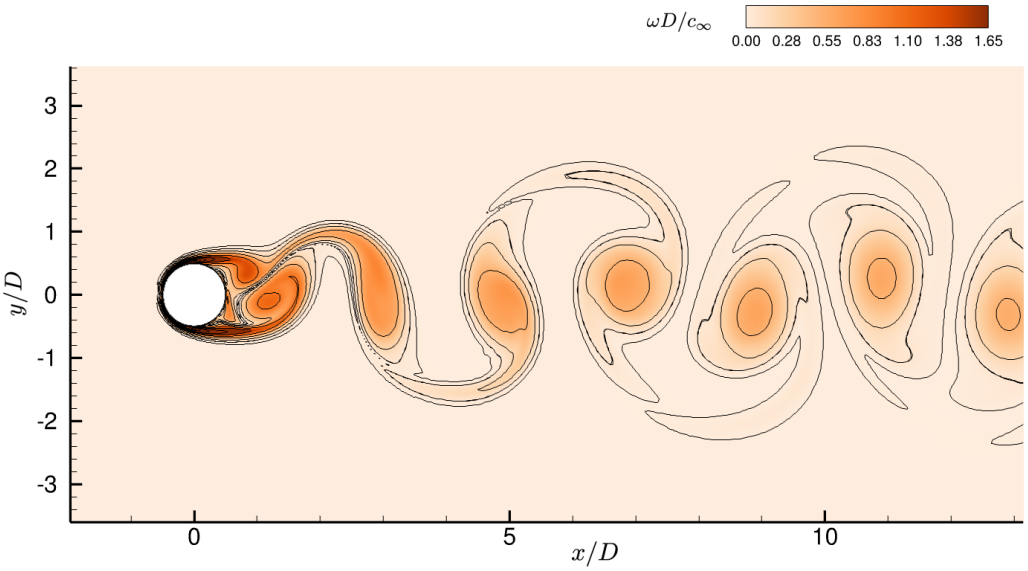


Figure 20: Contours of vorticity magnitude at $tc_\infty/D \approx 800$. The contour lines depict 7 levels in the range $[0, 1.65]$.

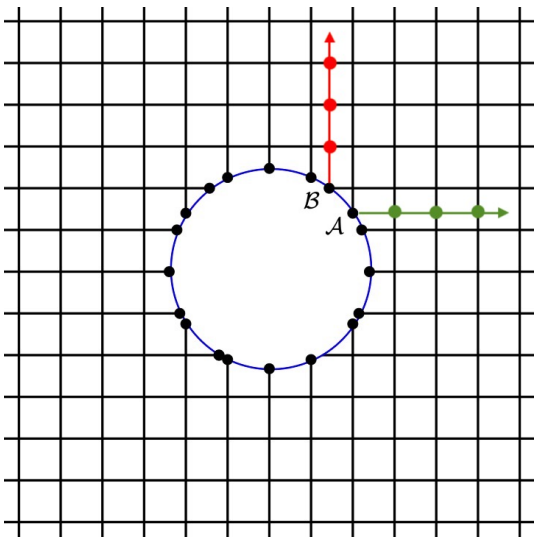


Figure 21: A schematic showing the additional grid points required for time advancement of the solutions at the EB points $\mathcal{A}$ and $\mathcal{B}$.

stencils ensure time stability without ad hoc stabilization measures, *e.g.* artificial dissipation or numerical filter. Moreover, a simple solution to the small-cell problem is proposed by recognizing that the cell size in a conservative finite-difference scheme may not be equal to the grid spacing. The proposed solution addresses the small-cell problem by construction and does not require tedious solution or geometry reconstructions over the cut cells. A systematic framework to derive conservative first- and second-derivative embedded boundary closures is discussed. The framework is applied to obtain EB schemes that are up to sixth-order accurate in the interior and at least fourth-order accurate globally for hyperbolic, parabolic as well as incompletely parabolic problems. The performance of the derived stencils is evaluated for various linear and non-linear inviscid as well as viscous problems: (a) 1-D scalar convection, (b) 1-D diffusion, (c) 1-D viscous Burgers', (d) 2-D variable-coefficient convection, (e) 2-D Euler equation, (f) 2-D convection-diffusion, (g) 3-D variable-coefficient convection, and (h) 2-D flow past a circular cylinder.

Acknowledgements

This work was supported by the US Department of Energy through the Los Alamos National Laboratory. Los Alamos National Laboratory is operated by Triad National Security, LLC, for the National Nuclear Security Administration of U.S. Department of Energy (Contract No. 89233218CNA000001). Research presented in this article was supported by the Laboratory Directed Research and Development program of Los Alamos National Laboratory under project number 20190227ER. Computational resources were provided by the LANL Institutional Computing (IC) Program and the National Science Foundation XSEDE resources under grant TG-PHY210037.

Appendix A. Proof showing (15) implies (14)

The rows of a derivative approximation must sum to zero to satisfy the zeroth order-of-accuracy constraint, *i.e.*,

$$\sum_{j=0}^{n} d_{1,ij} = 0, \qquad \text{for} \qquad 0 \le i \le n, \tag{A.1}$$

where $d_{1,ij}/\Delta x$ denotes the i,j entry of D_1. The entries of the matrix $A = WD_1$ are given by

$$a_{ij} = \sum_{k=0}^{n} w_{ik} d_{1,kj}, \qquad \text{for} \qquad 0 \le i,j \le n,$$

where a_{ij} and $w_{ij}\Delta x$ denote the i,j entry of A and W, respectively. The sum of the entries of the i-th row of $A = WD_1$ is then given by

$$\sum_{j=0}^{n} a_{ij} = \sum_{j=0}^{n}\sum_{k=0}^{n} w_{ik} d_{1,kj} = \sum_{k=0}^{n} w_{ik} \left(\sum_{j=0}^{n} d_{1,kj} \right) = 0, \qquad \text{for} \qquad 0 \le i \le n, \tag{A.2}$$

where the last equality follows from (A.1).

In addition, $WD_1 + (WD_1)^T = A + A^T = \text{diag}\,(-1, 0, \cdots 0, 1)$ from (15) implies

$$a_{ij} = \begin{cases} -\frac{1}{2} & \text{if} \quad i = j = 0, \\ \frac{1}{2} & \text{if} \quad i = j = n, \\ 0 & \text{if} \quad 1 \le i = j \le n-1, \\ -a_{ji} & \text{otherwise.} \end{cases} \tag{A.3}$$

$A = WD_1$ provides

$$\sum_{i=0}^{n} (WD_1\mathbf{u})_i = \sum_{i=0}^{n} (A\mathbf{u})_i = \sum_{i=0}^{n}\sum_{j=0}^{n} a_{ij}u_j = \sum_{i=0}^{n} a_{i0}u_0 + \sum_{i=0}^{n}\sum_{j=1}^{n-1} a_{ij}u_j + \sum_{i=0}^{n} a_{in}u_n. \tag{A.4}$$

Application of (A.3) to the middle term in the RHS of (A.4) yields

$$\sum_{i=0}^{n}\sum_{j=1}^{n-1} a_{ij}u_j = \sum_{i=0}^{n}\sum_{j=1}^{n-1} -a_{ji}u_j = -\sum_{j=1}^{n-1}\left(\sum_{i=0}^{n} a_{ji}\right)u_j = 0, \tag{A.5}$$

where the last equality in (A.5) follows from (A.2). Applying (A.2) for $i = 0$ and n yields, using (A.3),

$$\sum_{j=0}^{n} a_{0j} = a_{00} + \sum_{j=1}^{n} a_{0j} = 0 \qquad \Rightarrow \sum_{j=1}^{n} a_{j0} = a_{00} = -\frac{1}{2}, \tag{A.6}$$

$$\sum_{j=0}^{n} a_{nj} = a_{nn} + \sum_{j=0}^{n-1} a_{nj} = 0 \qquad \Rightarrow \sum_{j=0}^{n-1} a_{jn} = a_{nn} = \frac{1}{2}. \tag{A.7}$$

Substituting (A.5), (A.6) and (A.7) in (A.4) provides

$$\sum_{i=0}^{n} \left(WD_1\mathbf{u}\right)_i = \left(\sum_{j=1}^{n} a_{j0} + a_{00}\right)u_0 + \left(\sum_{j=0}^{n-1} a_{jn} + a_{nn}\right)u_n = u_n - u_0.$$

This completes the proof. □

Appendix B. Proof showing (15) implies $W\overline{D}_1 + \left(W\overline{D}_1\right)^T = \text{diag}\,(0, \cdots, 0, 1)$

For a diagonal $W = \Delta x\,\text{diag}\,(w_0, w_1, \cdots, w_{n-1}, w_n)$ and a D_1 with the i, j coefficient given by $d_{1,ij}/\Delta x$, the coefficients of $A = WD_1$ are

$$a_{ij} = w_i d_{1,ij}, \qquad \text{for} \qquad 0 \le i, j \le n. \tag{B.1}$$

Similarly, the coefficients of $\overline{A} = W\overline{D}_1$, denoted by $\overline{a}_{ij}$, are

$$\overline{a}_{ij} = \begin{cases} 0 & \text{if} \quad i \text{ or } j = 0, \\ a_{ij} & \text{otherwise.} \end{cases} \tag{B.2}$$

Condition (15) implies that a_{ij} satisfies (A.3). The relations (B.2) with (A.3) provide

$$\overline{a}_{ij} = \begin{cases} 0 & \text{if} \quad i \text{ or } j = 0, \\ a_{nn} = \frac{1}{2} & \text{if} \quad i = j = n, \\ a_{ij} = -a_{ji} & \text{otherwise.} \end{cases} \tag{B.3}$$

(B.3) can be used to show that $\overline{A} + \overline{A}^T = \text{diag}\,(0, \cdots, 0, 1)$. This completes the proof. □

Appendix C. Operators for $p = 2$

The first-derivative approximation D_1 and the quadrature matrix W for $p = 2$ are given by (21)-(22) and (24) with $\kappa = 4$ and $\beta = 6$, *i.e.*

$$D_1 = \frac{1}{\Delta x}\begin{bmatrix} d_{1,00} & d_{1,01} & d_{1,02} & d_{1,03} & d_{1,04} & d_{1,05} & d_{1,06} & & \\ d_{1,10} & d_{1,11} & d_{1,12} & d_{1,13} & d_{1,14} & d_{1,15} & d_{1,16} & & \\ d_{1,20} & d_{1,21} & d_{1,22} & d_{1,23} & d_{1,24} & d_{1,25} & d_{1,26} & & \\ d_{1,30} & d_{1,31} & d_{1,32} & d_{1,33} & d_{1,34} & d_{1,35} & d_{1,36} & & \\ d_{1,40} & d_{1,41} & d_{1,42} & d_{1,43} & d_{1,44} & d_{1,45} & d_{1,46} & & \\ & & & \frac{1}{12} & -\frac{2}{3} & 0 & \frac{2}{3} & -\frac{1}{12} & \\ & & & & \ddots & \ddots & \ddots & \ddots & \ddots \end{bmatrix}, \tag{C.1}$$

$$W = \Delta x \,\mathrm{diag}\,(w_0,\, w_1,\, w_2,\, w_3,\, w_4,\, 1,\, \cdots)\,, \tag{C.2}$$

where

$w_0 = \gamma,$

$w_1 = \frac{1}{48}\left(2\alpha^4 - 8\alpha^3(\gamma-2) + \alpha^2(44-48\gamma) + \alpha(48-88\gamma) - 48\gamma + 17\right),$

$w_2 = \frac{1}{48}\left(-6\alpha^4 + 8\alpha^3(3\gamma-5) + 24\alpha^2(5\gamma-3) + 144\alpha\gamma + 59\right),$

$w_3 = \frac{1}{48}\left(6\alpha^4 - 8\alpha^3(3\gamma-4) + \alpha^2(36-96\gamma) - 72\alpha\gamma + 43\right),$

$w_4 = \frac{1}{48}\left(-2\alpha^4 + 8\alpha^3(\gamma-1) + 8\alpha^2(3\gamma-1) + 16\alpha\gamma + 49\right),$

$d_{1,00} = -\frac{1}{2\gamma},$

$d_{1,01} = \frac{-2\alpha^4 k + 8\alpha^3(\gamma-1)k + 4\alpha^2((6\gamma-2)k+3) + 4\alpha(4\gamma(k-3)+9) + 49k - 72\gamma + 24}{48\gamma},$

$d_{1,02} = -\frac{-2\alpha^4 k + 8\alpha^3(\gamma-1)k + 8\alpha^2((3\gamma-1)k+1) + 16\alpha(\gamma(k-2)+1) + 49k - 32\gamma}{16\gamma},$

$d_{1,03} = \frac{-2\alpha^4 k + 8\alpha^3(\gamma-1)k + 4\alpha^2((6\gamma-2)k+1) + 4\alpha(4\gamma(k-1)+1) + 49k - 8\gamma}{16\gamma},$

$d_{1,04} = \frac{k\left(2\alpha^4 - 8\alpha^3(\gamma-1) + \alpha^2(8-24\gamma) - 16\alpha\gamma - 49\right)}{48\gamma},$

$d_{1,05} = 0,$

$d_{1,06} = 0,$

$$d_{1,10} = -\frac{-2\alpha^4 k + 8\alpha^3(\gamma-1)k + 4\alpha^2((6\gamma-2)k+3) + 4\alpha(4\gamma(k-3)+9) + 49k - 72\gamma + 24}{2\alpha^4 - 8\alpha^3(\gamma-2) + \alpha^2(44-48\gamma) + \alpha(48-88\gamma) - 48\gamma + 17},$$

$$d_{1,11} = 0,$$

$$d_{1,12} = \frac{-2\alpha^6 k + 2\alpha^5(4\gamma-9)k + \alpha^4((64\gamma-60)k+2) + 8\alpha^3((23\gamma-11)k-\gamma+2) + \alpha^2((224\gamma+1)k-72\gamma+56) + \alpha((96\gamma+245)k-208\gamma+96) + 294k - 192\gamma + 59}{2\left(2\alpha^4 - 8\alpha^3(\gamma-2) + \alpha^2(44-48\gamma) + \alpha(48-88\gamma) - 48\gamma + 17\right)},$$

$$d_{1,13} = -\frac{-2\alpha^6 k + 8\alpha^5(\gamma-2)k + \alpha^4((56\gamma-46)k+4) + 4\alpha^3(2(17\gamma-7)k-4\gamma+5) + \alpha^2((136\gamma+25)k-72\gamma+28) + 4\alpha((12\gamma+49)k-20\gamma+3) + 147k - 24\gamma + 4}{2\alpha^4 - 8\alpha^3(\gamma-2) + \alpha^2(44-48\gamma) + \alpha(48-88\gamma) - 48\gamma + 17},$$

$$d_{1,14} = \frac{-2\alpha^6 k + 2\alpha^5(4\gamma-7)k + 6\alpha^4((8\gamma-6)k+1) + 8\alpha^3((13\gamma-5)k-3\gamma+3) + 3\alpha^2((32\gamma+11)k-24\gamma+8) + \alpha((32\gamma+147)k-48\gamma) + 98k - 3}{2\left(2\alpha^4 - 8\alpha^3(\gamma-2) + \alpha^2(44-48\gamma) + \alpha(48-88\gamma) - 48\gamma + 17\right)},$$

$$d_{1,15} = 0,$$

$$d_{1,16} = 0,$$

$$d_{1,20} = -\frac{3\left(2\alpha^4 k - 8\alpha^3(\gamma-1)k - 8\alpha^2((3\gamma-1)k+1) - 16\alpha(\gamma(k-2)+1) - 49k + 32\gamma\right)}{-6\alpha^4 + 8\alpha^3(3\gamma-5) + 24\alpha^2(5\gamma-3) + 144\alpha\gamma + 59},$$

$$d_{1,21} = -\frac{-2\alpha^6 k + 2\alpha^5(4\gamma-9)k + \alpha^4((64\gamma-60)k+2) + 8\alpha^3((23\gamma-11)k-\gamma+2) + \alpha^2((224\gamma+1)k-72\gamma+56) + \alpha((96\gamma+245)k-208\gamma+96) + 294k - 192\gamma + 59}{2\left(-6\alpha^4 + 8\alpha^3(3\gamma-5) + 24\alpha^2(5\gamma-3) + 144\alpha\gamma + 59\right)},$$

$$d_{1,22} = 0,$$

$$d_{1,23} = -\frac{6\alpha^6 k - 6\alpha^5(4\gamma-7)k - 6\alpha^4(8(3\gamma-2)k+3) - 8\alpha^3((33\gamma-9)k-9\gamma+10) - 3\alpha^2((48\gamma+49)k-88\gamma+24) - 9\alpha(49k-16\gamma) - 59}{2\left(-6\alpha^4 + 8\alpha^3(3\gamma-5) + 24\alpha^2(5\gamma-3) + 144\alpha\gamma + 59\right)},$$

$$d_{1,24} = \frac{\alpha(\alpha+2)\left(2\alpha^4 k - 8\alpha^3(\gamma-1)k - 8\alpha^2((3\gamma-1)k+1) - 16\alpha(\gamma(k-2)+1) - 49k + 32\gamma\right)}{-6\alpha^4 + 8\alpha^3(3\gamma-5) + 24\alpha^2(5\gamma-3) + 144\alpha\gamma + 59},$$

$$d_{1,25} = 0,$$

$$d_{1,26} = 0$$

$$d_{1,30} = \frac{3\left(2\alpha^4 k - 8\alpha^3(\gamma-1)k - 4\alpha^2((6\gamma-2)k+1) - 4\alpha(4\gamma(k-1)+1) - 49k + 8\gamma\right)}{6\alpha^4 - 8\alpha^3(3\gamma-4) + \alpha^2(36-96\gamma) - 72\alpha\gamma + 43},$$

$$d_{1,31} = \frac{-2\alpha^6 k + 8\alpha^5(\gamma-2)k + \alpha^4((56\gamma-46)k+4) + 4\alpha^3(2(17\gamma-7)k-4\gamma+5) + \alpha^2((136\gamma+25)k-72\gamma+28) + 4\alpha((12\gamma+49)k-20\gamma+3) + 147k - 24\gamma + 4}{6\alpha^4 - 8\alpha^3(3\gamma-4) + \alpha^2(36-96\gamma) - 72\alpha\gamma + 43},$$

$$d_{1,32} = -\frac{-6\alpha^6 k + 6\alpha^5(4\gamma-7)k + 6\alpha^4(8(3\gamma-2)k+3) + 8\alpha^3((33\gamma-9)k-9\gamma+10) + 3\alpha^2((48\gamma+49)k-88\gamma+24) + 9\alpha(49k-16\gamma)+59}{2\left(6\alpha^4 - 8\alpha^3(3\gamma-4) + \alpha^2(36-96\gamma) - 72\alpha\gamma + 43\right)},$$

$$d_{1,33} = 0,$$

$$d_{1,34} = \frac{-2\alpha^6 k + 2\alpha^5(4\gamma-5)k + 2\alpha^4(8(2\gamma-1)k+5) + 8\alpha^3((5\gamma-1)k-5\gamma+5) + \alpha^2((16\gamma+49)k-120\gamma+40) + \alpha(49k-80\gamma)+59}{2\left(6\alpha^4 - 8\alpha^3(3\gamma-4) + \alpha^2(36-96\gamma) - 72\alpha\gamma + 43\right)},$$

$$d_{1,35} = -\frac{4}{6\alpha^4 - 8\alpha^3(3\gamma-4) + \alpha^2(36-96\gamma) - 72\alpha\gamma + 43},$$

$$d_{1,36} = 0,$$

$$d_{1,40} = k,$$

$$d_{1,41} = \frac{-2\alpha^6 k + 2\alpha^5(4\gamma-7)k + 6\alpha^4((8\gamma-6)k+1) + 8\alpha^3((13\gamma-5)k-3\gamma+3) + 3\alpha^2((32\gamma+11)k-24\gamma+8) + \alpha((32\gamma+147)k-48\gamma)+98k-3}{2\left(2\alpha^4 - 8\alpha^3(\gamma-1) + \alpha^2(8-24\gamma) - 16\alpha\gamma - 49\right)},$$

$$d_{1,42} = \frac{\alpha(\alpha+2)\left(2\alpha^4 k - 8\alpha^3(\gamma-1)k - 8\alpha^2((3\gamma-1)k+1) - 16\alpha(\gamma(k-2)+1) - 49k + 32\gamma\right)}{2\alpha^4 - 8\alpha^3(\gamma-1) + \alpha^2(8-24\gamma) - 16\alpha\gamma - 49},$$

$$d_{1,43} = \frac{-2\alpha^6 k + 2\alpha^5(4\gamma-5)k + 2\alpha^4(8(2\gamma-1)k+5) + 8\alpha^3((5\gamma-1)k-5\gamma+5) + \alpha^2((16\gamma+49)k-120\gamma+40) + \alpha(49k-80\gamma)+59}{2\left(2\alpha^4 - 8\alpha^3(\gamma-1) + \alpha^2(8-24\gamma) - 16\alpha\gamma - 49\right)},$$

$$d_{1,44} = 0,$$

$$d_{1,45} = -\frac{32}{2\alpha^4 - 8\alpha^3(\gamma-1) + \alpha^2(8-24\gamma) - 16\alpha\gamma - 49},$$

$$d_{1,46} = \frac{4}{2\alpha^4 - 8\alpha^3(\gamma-1) + \alpha^2(8-24\gamma) - 16\alpha\gamma - 49},$$

with parameter values chosen to minimize the spectral radius. We choose $k = 0$ and

$$\gamma = s \max(0, \beta_1) + (1 - s) \min(\beta_2, \beta_3), \tag{C.3}$$

$$s = \frac{147}{233},$$

$$\beta_1 = \frac{-59 + 72\alpha^2 + 40\alpha^3 + 6\alpha^4}{144\alpha + 120\alpha^2 + 24\alpha^3},$$

$$\beta_2 = \frac{43 + 36\alpha^2 + 32\alpha^3 + 6\alpha^4}{72\alpha + 96\alpha^2 + 24\alpha^3},$$

$$\beta_3 = \frac{17 + 48\alpha + 44\alpha^2 + 16\alpha^3 + 2\alpha^4}{48 + 88\alpha + 48\alpha^2 + 8\alpha^3}.$$

The second-derivative approximation D_2 for $p = 2$ is given by (21) and (23) with $\kappa = 4$ and $\beta = 6$, *i.e.*

$$D_2 = \frac{1}{\Delta x^2} \begin{bmatrix} d_{2,00} & d_{2,01} & d_{2,02} & d_{2,03} & d_{2,04} & d_{2,05} & d_{2,06} & & \\ d_{2,10} & d_{2,11} & d_{2,12} & d_{2,13} & d_{2,14} & d_{2,15} & d_{2,16} & & \\ d_{2,20} & d_{2,21} & d_{2,22} & d_{2,23} & d_{2,24} & d_{2,25} & d_{2,26} & & \\ d_{2,30} & d_{2,31} & d_{2,32} & d_{2,33} & d_{2,34} & d_{2,35} & d_{2,36} & & \\ d_{2,40} & d_{2,41} & d_{2,42} & d_{2,43} & d_{2,44} & d_{2,45} & d_{2,46} & & \\ & & & -\frac{1}{12} & \frac{4}{3} & -\frac{5}{2} & \frac{4}{3} & -\frac{1}{12} & \\ & & & & \ddots & \ddots & \ddots & \ddots & \ddots \end{bmatrix}. \tag{C.4}$$

The coefficients $d_{2,ij}$ for $0 \le i \le 4$ and $0 \le j \le 6$ are provided in the supplementary document and Matlab scripts containing those coefficients are included in the supplementary material.
The ES_1 operator corresponding to (C.4) is given by (32) with $\beta = 6$, where

$$s_0 = -\frac{3\alpha^2 + 12\alpha + 29}{\alpha^3 + 6\alpha^2 + 11\alpha + 6},$$

$$s_1 = 3,$$

$$s_2 = \frac{\alpha^2 - 13\alpha + 6}{2\alpha + 2},$$

$$s_3 = -\frac{\alpha^2 - 5\alpha + 3}{\alpha + 2},$$

$$s_4 = \frac{\alpha^2 - 3\alpha + 2}{2\alpha + 6},$$

$$s_5 = 0,$$

$$s_6 = 0,$$

Appendix D. Supplementary material

The supplementary material includes a document that contains the D_2 stencil coefficients for $p = 2$ and all the operators for $p = 3$. The operators for various order of accuracies can also be accessed through Matlab scripts provided in the supplementary material. The main script that generates operators for a one-dimensional embedded boundary grid configuration as shown in figure 8 is copied below:

```
n = 15;                 % number of grid points
```

```
L = 1;                % domain length to calculate grid spacing
dx = L/(n-1);         % grid spacing

alphaL = 1/2;         % 0 <= alphaL <= 1
alphaR = 2/3;         % 0 <= alphaR <= 1

x = 0:dx:L;
x(1) = x(2) - alphaL*dx;
x(n) = x(n-1) + alphaR*dx;

derv_approx = 1;      % first (=1) or second (=2) derivative approximation
p = 2;                % boundary order of accuracy (1, 2 or 3)

%%%%%%%%%% OPERATORS %%%%%%%%%%%%%
[D, W, ES] = operators(n,derv_approx,p,alphaL,alphaR);
D = (1/dx)^derv_approx * D;
W = dx * W;
ES = 1/dx * ES;
```

The variables `alphaL` and `alphaR` define α_l and α_r in figure 8, respectively. Choose `derv_approx = 1` for a first-derivative approximation and `derv_approx = 2` for a second-derivative approximation.

References

[1] I. Proudman, The generation of noise by isotropic turbulence, Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences 214 (1116) (1952) 119–132.

[2] C. K. Tam, L. Auriault, Jet mixing noise from fine-scale turbulence, AIAA journal 37 (2) (1999) 145–153.

[3] J. B. Freund, Noise sources in a low-reynolds-number turbulent jet at mach 0.9, Journal of Fluid Mechanics 438 (2001) 277.

[4] K. R. Sreenivasan, R. Antonia, The phenomenology of small-scale turbulence, Annual review of fluid mechanics 29 (1) (1997) 435–472.

[5] D. Livescu, J. Ristorcelli, Variable-density mixing in buoyancy-driven turbulence, Journal of Fluid Mechanics 605 (2008) 145.

[6] N. Sharan, G. Matheou, P. E. Dimotakis, Turbulent shear-layer mixing: initial conditions, and direct-numerical and large-eddy simulations, Journal of Fluid Mechanics 877 (2019) 35–81.

[7] R. Mittal, P. Moin, Suitability of upwind-biased finite difference schemes for large-eddy simulation of turbulent flows, AIAA journal 35 (8) (1997) 1415–1417.

[8] N. Park, J. Y. Yoo, H. Choi, Discretization errors in large eddy simulation: on the suitability of centered and upwind-biased compact difference schemes, Journal of Computational Physics 198 (2) (2004) 580–616.

[9] D. Mavriplis, Unstructured grid techniques, Annual Review of Fluid Mechanics 29 (1) (1997) 473–514.

[10] R. Mittal, G. Iaccarino, Immersed boundary methods, Annu. Rev. Fluid Mech. 37 (2005) 239–261.

[11] D. M. Ingram, D. M. Causon, C. G. Mingham, Developments in cartesian cut cell methods, Mathematics and Computers in Simulation 61 (3-6) (2003) 561–572.

[12] M. Berger, M. Aftosmis, Progress towards a cartesian cut-cell method for viscous compressible flow, in: 50th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, 2012, p. 1301.

[13] 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.

[14] C. Zhang, R. J. LeVeque, The immersed interface method for acoustic wave equations with discontinuous coefficients, Wave motion 25 (3) (1997) 237–263.

[15] L. Lee, R. J. LeVeque, An immersed interface method for incompressible navier–stokes equations, SIAM Journal on Scientific Computing 25 (3) (2003) 832–856.

[16] T. Ye, R. Mittal, H. Udaykumar, W. Shyy, An accurate cartesian grid method for viscous incompressible flows with complex immersed boundaries, Journal of computational physics 156 (2) (1999) 209–240.

[17] M.-H. Chung, Cartesian cut cell approach for simulating incompressible flows with rigid bodies of arbitrary shape, Computers & Fluids 35 (6) (2006) 607–623.

[18] X. Y. Hu, B. Khoo, N. A. Adams, F. Huang, A conservative interface method for compressible flows, Journal of Computational Physics 219 (2) (2006) 553–578.

[19] D. Hartmann, M. Meinke, W. Schröder, A strictly conservative cartesian cut-cell method for compressible viscous flows on adaptive grids, Computer Methods in Applied Mechanics and Engineering 200 (9-12) (2011) 1038–1052.
[20] P. T. Barton, B. Obadia, D. Drikakis, A conservative level-set based method for compressible solid/fluid problems on fixed grids, Journal of Computational Physics 230 (21) (2011) 7867–7890.
[21] M. Meinke, L. Schneiders, C. Günther, W. Schröder, A cut-cell method for sharp moving boundaries in cartesian grids, Computers & Fluids 85 (2013) 135–142.
[22] L. Schneiders, C. Günther, M. Meinke, W. Schröder, An efficient conservative cut-cell method for rigid bodies interacting with viscous compressible flows, Journal of Computational Physics 311 (2016) 62–86.
[23] B. Muralidharan, S. Menon, A high-order adaptive cartesian cut-cell method for simulation of compressible viscous flow over immersed bodies, Journal of Computational Physics 321 (2016) 342–368.
[24] M. Natarajan, R. Grout, W. Zhang, M. Day, A moving embedded boundary approach for the compressible navier-stokes equations in a block-structured adaptive refinement framework, Journal of Computational Physics (2022) 111315.
[25] P. Colella, D. T. Graves, B. J. Keen, D. Modiano, A cartesian grid embedded boundary method for hyperbolic conservation laws, Journal of Computational Physics 211 (1) (2006) 347–366.
[26] C. Brehm, H. F. Fasel, A novel concept for the design of immersed interface methods, Journal of Computational Physics 242 (2013) 234–267.
[27] P. T. Brady, D. Livescu, Foundations for high-order, conservative cut-cell methods: Stable discretizations on degenerate meshes, Journal of Computational Physics 426 (2021) 109794.
[28] F. Nicoud, Conservative high-order finite-difference schemes for low-mach number flows, Journal of Computational Physics 158 (1) (2000) 71–97.
[29] O. Desjardins, G. Blanquart, G. Balarac, H. Pitsch, High order conservative finite difference scheme for variable density low mach number turbulent flows, Journal of Computational Physics 227 (15) (2008) 7125–7159.
[30] Y. Morinishi, Skew-symmetric form of convective terms and fully conservative finite difference schemes for variable density low-mach number flows, Journal of Computational Physics 229 (2) (2010) 276–300.
[31] Y. Morinishi, T. S. Lund, O. V. Vasilyev, P. Moin, Fully conservative higher order finite difference schemes for incompressible flow, Journal of computational physics 143 (1) (1998) 90–124.
[32] H.-O. Kreiss, G. Scherer, On the existence of energy estimates for difference approximations for hyperbolic systems, Tech. rep., Technical report, Dept. of Scientific Computing, Uppsala University (1977).
[33] B. Gustafsson, High order difference methods for time dependent PDE, Vol. 38, Springer Science & Business Media, 2007.
[34] N. Sharan, Time-stable high-order finite difference methods for overset grids, Ph.D. thesis, University of Illinois at Urbana-Champaign (2016).
[35] N. Sharan, C. Pantano, D. J. Bodony, Time-stable overset grid method for hyperbolic problems using summation-by-parts operators, Journal of Computational Physics 361 (2018) 199–230.
[36] W. R. Inc., Mathematica, Version 12.2, Champaign, IL, 2020.
URL `https://www.wolfram.com/mathematica`
[37] N. Sharan, P. T. Brady, D. Livescu, Stable and conservative boundary treatment for difference methods, with application to cut-cell discretizations, in: AIAA Scitech 2020 Forum, 2020, p. 0807.
[38] F. Zhang, Matrix theory: basic results and techniques, Springer Science & Business Media, 2011.
[39] M. H. Carpenter, D. Gottlieb, S. Abarbanel, The stability of numerical boundary treatments for compact high-order finite-difference schemes, Journal of Computational Physics 108 (2) (1993) 272–295.
[40] N. Sharan, C. Pantano, D. J. Bodony, Energy stable overset grid methods for hyperbolic problems, in: 7th AIAA Theoretical Fluid Mechanics Conference, 2014, p. 2924.
[41] B. Gustafsson, The convergence rate for difference approximations to mixed initial boundary value problems, Mathematics of Computation 29 (130) (1975) 396–406.
[42] K. Salari, P. Knupp, Code verification by the method of manufactured solutions, Tech. rep., Sandia National Labs., Albuquerque, NM (US) (2000).
[43] K. W. Thompson, Time-dependent boundary conditions for hyperbolic systems, ii, Journal of computational physics 89 (2) (1990) 439–461.
[44] T. H. Pulliam, Implicit solution methods in computational fluid dynamics, Applied numerical mathematics 2 (6) (1986) 441–474.
[45] T. J. Poinsot, S. K. Lele, Boundary conditions for direct simulations of compressible viscous flows, Journal of computational physics 101 (1) (1992) 104–129.
[46] G. Lodato, P. Domingo, L. Vervisch, Three-dimensional boundary conditions for direct and large-eddy simulation of compressible viscous flows, Journal of computational physics 227 (10) (2008) 5105–5143.
[47] R. Kamakoti, C. Pantano, High-order narrow stencil finite-difference approximations of second-order derivatives involving variable coefficients, SIAM Journal on Scientific Computing 31 (6) (2010) 4222–4243.
[48] M. Al-Marouf, R. Samtaney, A versatile embedded boundary adaptive mesh method for compressible flow in complex geometry, Journal of Computational Physics 337 (2017) 339–378.
[49] C. Brehm, C. Hader, H. F. Fasel, A locally stabilized immersed boundary method for the compressible navier–stokes equations, Journal of Computational Physics 295 (2015) 475–504.