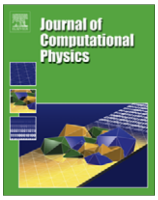
Journal of Computational Physics 228 (2009) 5280–5297

Contents lists available at ScienceDirect

Journal of Computational Physics

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



Nondissipative and energy-stable high-order finite-difference interface schemes for 2-D patch-refined grids

R.M.J. Kramer [a,*], C. Pantano [b], D.I. Pullin [a]

[a] *Graduate Aeronautical Laboratories, California Institute of Technology, MC 205-45, Pasadena, CA 91125, United States*

[b] *Department of Mechanical Science and Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801, United States*

ARTICLE INFO

Article history:
Received 27 May 2008
Received in revised form 3 April 2009
Accepted 8 April 2009
Available online 22 April 2009

Keywords:
High-order finite difference
Mesh interface
Stable stencil
Adaptive mesh refinement
Summation by parts

ABSTRACT

A class of finite-difference interface schemes suitable for two-dimensional cell-centered grids with patch-refinement and step-changes in resolution is presented. Grids of this type are generated by adaptive mesh refinement methods according to resolution needs dictated by the physics of the problem being modeled. For these grids, coarse and fine nodes are not aligned at the mesh interfaces, resulting in hanging nodes. Three distinct geometries are identified at the interfaces of a domain with interior patch-refinement: edges, concave corners and convex corners. Asymptotic stability in time of the numerical scheme is achieved by imposing a summation-by-parts condition on the interface closure, which is thus also nondissipative. Interface stencils corresponding to an explicit fourth-order finite-difference scheme are presented for each geometry. To preserve stability, a reduction in local accuracy is required at the corner geometries. It is also found that no second-order accurate solution exists that satisfies the summation-by-parts condition. Tests using the 2-D scalar advection equation and an inviscid compressible vortex support the stability and accuracy of these stencils for both linear and nonlinear problems.

1. Introduction

Adaptive mesh refinement (AMR) methods are increasingly applied to fluid dynamics and other physical problems, where the ability to locally refine the computational grid allows efficient resolution of features involving a wide range of length scales. However, for a given truncation error, high-order uniform-grid methods can be more efficient computationally than the usual second-order AMR [1,2], owing to gains realized by the simplicity of the data structures. High-order AMR offers potentially the best of both worlds if stable closures at refinement interfaces are available. In this paper, the focus is primarily on problems of hyperbolic type, i.e., with a strong wave propagation character.

A mathematical challenge for AMR methods is proper treatment of interfaces where there is a step-change in grid resolution. The method of Berger and Colella [3] uses a computationally efficient interpolation/restriction operation to transfer information between levels of a refinement hierarchy. Examples of utilization of this idea in second-order accurate approximations are abundant [3–5]. Unfortunately, this technique cannot be extended easily to higher-order accuracy, while still preserving stability.

High-order closures for one-dimensional step-changes in resolution are discussed in more detail in [6]. For two-dimensional meshes, interface closures have been constructed by enforcing Lax stability in a linear scalar advection equation model [7,8]. This approach suffers from dissipation introduced at the interface to stabilize the closure and by the fact that stability is ensured in general only for waves traveling along specific directions. This is a serious limitation when dealing with systems

[*] Corresponding author. Tel.: +1 626 395 4747.
E-mail address: rmjk@caltech.edu (R.M.J. Kramer).

doi:10.1016/j.jcp.2009.04.010

of partial differential equations like the compressible Euler equations of fluid mechanics, which support waves traveling in multiple directions simultaneously. Moreover, many hyperbolic problems have features sensitive to numerical dissipation, including turbulence and wave-propagation, and there is a need for a more advanced interface treatment that does not involve numerical dissipation.

As pointed out by Trefethen [9], global stability criteria are necessary to ensure that local stability is preserved in the presence of multiple interfaces, since local criteria do not capture interactions between interfaces. Energy-stable methods are preferred for this reason, and for a number of other advantages mentioned in [6]. The global stability property may be imbedded in finite-difference methods by preserving the summation-by-parts (SBP) property of the differential equation. First proposed by Kreiss and Scherer [10], this has been shown [11,12] to impart global stability by the formal GKS definition of Gustafsson et al. [13] and to be asymptotically stable in time when the boundary condition is implemented by the simultaneous approximation term (SAT) technique. Both definitions of stability must be satisfied to prevent nonphysical growth of the numerical solution in hyperbolic problems [14]. This approach also ensures that stability extends to nonlinear problems and systems of equations [11].

The focus of development of SBP operators has been primarily on boundary schemes. For grid interfaces, Nordström and Carpenter [15] propose a high-order method that uses a penalty-type technique to match the function value and first derivative at a common interface point. This assumes a vertex-type mesh, compared to the cell-centered finite volume-type mesh topology considered in this work that avoids overlapping nodes under refinement. The penalty method introduces some numerical dissipation, but has been applied successfully to fluid dynamics and electromagnetic problems where this effect was found to be small [16].

An alternative interface formulation is described in [6]. In this case, a customized stencil that satisfies the SBP property is developed for the nodes near an interface in a manner similar to the boundary stencils of Strand [17] and Carpenter et al. [11]. These nondissipative high-order schemes are shown to be energy-stable, but apply only to interfaces that are essentially one-dimensional in character. The same methodology is used in this work to extend the previous result to fully two-dimensional patch-refined grids. In the present case, the focus is on satisfying the stability criteria such that a truly two-dimensional SBP property is enforced throughout the domain. It is found that this has a cost in local accuracy at refinement corners. To our knowledge, no scheme enforcing the SBP property across grid interfaces in this way has been published for any order of accuracy.

In Section 2 the geometry of the patch-refined grid is defined, followed in Section 3 by the SBP formulation of the first-derivative operators. The interface stability criteria results from a more general condition for stability of the semidiscrete linear advection equation, and is used to generate the interface schemes presented in Section 4. Numerical tests of these schemes follow in Section 5.

2. Two-dimensional patch refinement

2.1. Definition of the grid

The starting point is the two-dimensional Cauchy problem, the scalar advection equation

$$\frac{\partial u}{\partial t}+a\frac{\partial u}{\partial x}+b\frac{\partial u}{\partial y}=0, \quad (x,y)\in\Omega, \quad t\geqslant 0, \tag{1}$$

where a square domain $\Omega=\{(x,y)\in[0,1]\times[0,1]\}$ is chosen for simplicity, with initial condition $u(x,y,0)=u_0(x,y)$ and boundary conditions (for $a>0$ and $b>0$),

$$u(0,y,t)=g^x(y,t), \quad t\geqslant 0, \tag{2}$$

$$u(x,0,t)=g^y(x,t), \quad t\geqslant 0. \tag{3}$$

The numerical solution to this problem is sought on a node-based finite-volume partition of the domain, chosen over a vertex-based discretization because of its wide use in adaptive mesh refinement methods [3,5]. Certain regions of this domain are locally refined relative to the original discretization; first, we define the nature of this refinement.

Consider a uniform discretization of the domain Ω into computational cells of size $\Delta x\times\Delta y$, with $\Delta x=1/N$ and $\Delta y=1/M$. Under local refinement, Ω is partitioned into two subdomains by $\Omega=\Omega^f\cup\Omega^c$, where the fine region is a block of refined cells defined by

$$\Omega^f\equiv\{(x,y)\in[n_L\Delta x,n_H\Delta x]\times[m_L\Delta y,m_H\Delta y]\},$$

with $0<n_L<n_H<N, 0<m_L<m_H<M$ and $(n_L,n_H,m_L,m_H)\in\mathbb{N}$. The fine region thus occupies an interior part of Ω, with each edge extending over an integer number of coarse cells. Note that in general, Ω^f may include a number of such refined blocks of cells. Within Ω^f, the discretization is refined by ratios r_x and r_y in the x- and y-directions, respectively, for computational cells of size $(\Delta x/r_x)\times(\Delta y/r_y)$. It is generally required that $1<r_x,r_y\in\mathbb{N}$. Additional resolution is thus provided for the problem in Ω^f. The remaining coarse region $\Omega^c=\Omega\setminus\Omega^f$, has the original cell dimensions. A sketch of such a partition is shown in Fig. 1. Dirichlet boundary conditions (2) and (3) are imposed on $\Gamma^D\equiv\{(x,y)\in([0,1]\times 0)\cup(0\times[0,1])\}$, where $\partial\Omega=\Gamma^D+\Gamma^N$. Note that this partitioning of the domain does not affect the equations or boundary conditions.

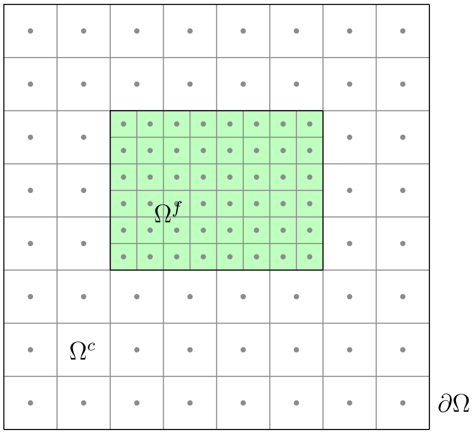


Fig. 1. Sketch of the domain Ω, showing a partition into a fine region Ω^f (shaded) and coarse region Ω^c.

A numerical method to solve (1) is constructed by approximation of $u(x,y,t) = u^c(x,y,t) + u^f(x,y,t)$, where u^c and u^f are the partitions of u to Ω^c and Ω^f, respectively. Locating nodes at the cell centers in each subdomain, u is discretized such that $u^c(x_I, y_J, t) \approx u^c_{I,J}(t)$, where $x_I = \Delta x(I - 1/2)$ and $y_J = \Delta y(J - 1/2)$, over the index set

$$\mathcal{I}^c = \{(I,J) \in [1,N] \times [1,M] \setminus [n_L + 1, n_H] \times [m_L + 1, m_H]\},$$

and $u^f(x_i, y_j, t) \approx u^f_{i,j}(t)$, where $x_i = \Delta x(n_L + (i - 1/2)/r_x)$ and $y_j = \Delta y(m_L + (j - 1/2)/r_y)$, over the index set

$$\mathcal{I}^f = \{(i,j) \in [1,(n_H - n_L)r_x] \times [1,(m_H - m_L)r_y]\},$$

where the complete index set $\mathcal{I} = \mathcal{I}^c \cup \mathcal{I}^f$. Let $\mathbf{u} = \{u^c_{I,J}, u^f_{i,j}\} = \{u_{\mathcal{I}}\}$ denote the discrete approximation to u at the nodes $\mathcal{I}$; no particular ordering of the index set is implied. The resulting approximation of (1) yields a system of ordinary differential equations, which are integrated numerically in time using an appropriate time-integration method, e.g., Runge–Kutta.

2.2. Interface types

The present objective is to develop an explicit finite-difference scheme of the two-dimensional domain with a cell-centered mesh. Approximation of the first derivative at interior cells, away from the interface at $\partial\Omega^f$, is accomplished using standard centered finite-difference techniques. Near $\partial\Omega^f$, there are two prevailing approaches to dealing with the irregular node locations at the interface. The first solves (1) as separate problems on each subdomain Ω^c and Ω^f, using a compatibility condition to transfer information between subdomains. This compatibility condition can be viewed as an addition to the original problem. Examples of this approach include [15,18,19]. The second approach, similar to [6], solves (1) on the full domain, assuming that a continuous and differentiable function $u(x,y,t)$ exists and is well defined across the grid interfaces. This leaves the problem of the change in resolution to be resolved by the numerical method, where special stencils are required to approximate the spatial derivative in the vicinity of $\partial\Omega^f$. This is the approach considered for this work.

The historical difficulty encountered with this approach is the appearance of "hanging nodes" at the grid interface, as nodes do not align with Cartesian lines passing through the cell centers on each side. This problem has made achieving simultaneous high-order accuracy and long-time stability of the interface closure difficult, and is addressed in this work. Our previous work [6] addressed the interface problem only for one-dimensional geometries.

A consequence of the hanging-node geometry is a strong dependence of the interface closure on the refinement ratio. Fig. 2(a) shows an interface with a refinement factor of 2 in each direction, where an interface stencil for the coarse node would include dependence on four fine nodes, whereas in Fig. 2(b), with a factor 4 refinement, the same node is now dependent on 16 fine nodes in an equivalent area. In general, this means that an interface stencil cannot have a simple functional

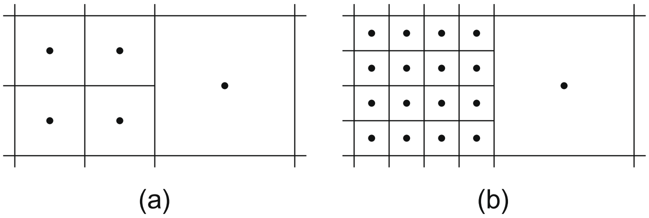


Fig. 2. The topology of an interface is dependent on the refinement factor: (a) refinement factor 2 in each direction, (b) refinement factor 4 in each direction.

dependence on r_x or r_y in the way that the one-dimensional scheme could, cf. [6]. We therefore fix the refinement ratios at $r_x = r_y = 2$ from this point forward, but this presentation may be followed for any integer refinement ratio (without guarantee that a solution exists).

For a general refined region Ω^f, three basic interfaces may be identified, each with a unique topology. These are sketched in Fig. 3: (a) shows the edge interface, corresponding to the Cartesian edges of $\partial\Omega^f$ where the change in discretization is essentially in one direction only; (b) shows the convex corner interface, where two edges meet at a corner of $\partial\Omega^f$ that is convex relative to Ω^f; and (c) shows the concave corner interface, where the corner is concave relative to the fine region. Other more complex interfaces may be devised, but these three elements are sufficient to build in a domain any arbitrarily refined interior region satisfying simple rules regarding its size such that each interface geometry remains distinct. Multiple levels of refinement, obtained by embedding additionally refined subdomains within Ω^f, present no additional closure problems, as the interface geometries are logically identical to those shown.

3. Formulation and stability

3.1. One-dimensional SBP formulation

The 1-D summation-by-parts operator is defined for the 1-D version of the advection equation

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0, \quad t \geqslant 0, \tag{4}$$

following [11,6], for a general approximation to the spatial derivative given by

$$P\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}x} = \frac{1}{\Delta x}Q\mathbf{u}, \tag{5}$$

where $P = \{p_{ij}\}$ and $Q = \{q_{ij}\}$ are matrices containing the coefficients of the stencil. An explicit finite-difference scheme is used in the interior of the domain, where centered second- and fourth-order schemes are given by

$$\frac{\mathrm{d}u(x)}{\mathrm{d}x} \approx \frac{1}{\Delta x}\left[\frac{1}{2}u(x+\Delta x) - \frac{1}{2}u(x-\Delta x)\right], \tag{6}$$

and

$$\frac{\mathrm{d}u(x)}{\mathrm{d}x} \approx \frac{1}{\Delta x}\left[\frac{1}{12}u(x-2\Delta x) - \frac{2}{3}u(x-\Delta x) + \frac{2}{3}u(x+\Delta x) - \frac{1}{12}u(x+2\Delta x)\right], \tag{7}$$

respectively. P is a dense matrix at the boundaries and diagonal elsewhere. Boundary conditions are applied using the SAT method [11] to give a semidiscrete approximation for (4)

$$P\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = -\frac{a}{\Delta x}Q\mathbf{u} + \frac{a}{\Delta x}\tau\mathbf{s}(u_1 - g(t)), \tag{8}$$

where τ is the penalty coefficient with $\tau \geqslant 1$ for stability, and

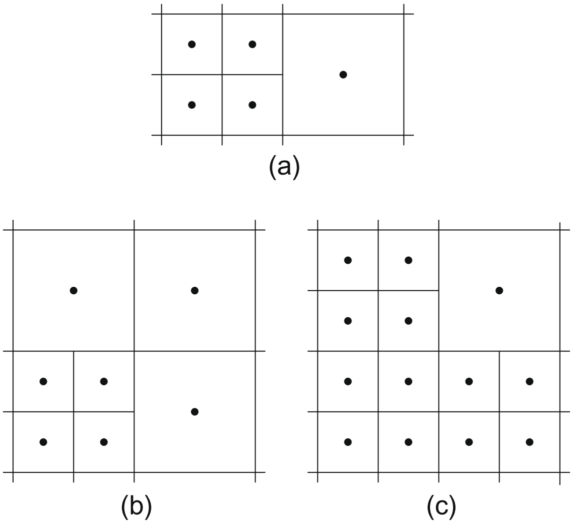


Fig. 3. Sketches of the three principal interior interface geometries encountered in two-dimensions: (a) edge, (b) convex corner, (c) concave corner.

$$\mathbf{s} = [q_{11}, 0, \dots, 0]^T. \tag{9}$$

Eq. (8) may be recast in the more convenient form

$$P\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = -\frac{a}{\Delta x}\widetilde{Q}\mathbf{u} - \frac{a}{\Delta x}\tau\mathbf{s}g(t), \tag{10}$$

where

$$\widetilde{Q} = Q - \tau[\mathbf{s}\,\mathbf{0}\cdots\mathbf{0}]. \tag{11}$$

The summation-by-parts property requires that:

1. P is symmetric positive definite, and
2. Q is anti-symmetric except at the corners, such that the matrix $Q + Q^T$ is the diagonal matrix $\mathrm{diag}(2q_{11}, 0, \dots, 0, 2q_{NN})$, with $q_{11} < 0 < q_{NN}$.

It is straightforward to show that these properties lead to an energy-stable discretization of the linear advection equation [11,12].

3.2. Two-dimensional SBP formulation

For (1), the 1-D theory is extended in a manner analogous to that described by [12]. Define general 2-D finite-difference approximations to the spatial derivatives on the entire domain Ω,

$$\frac{\partial\mathbf{u}}{\partial x} = \frac{1}{\Delta x}D_x\mathbf{u}, \tag{12}$$

$$\frac{\partial\mathbf{u}}{\partial y} = \frac{1}{\Delta y}D_y\mathbf{u}, \tag{13}$$

such that a projection $\mathbf{v}$ of the exact solution $u(x,y,t)$ onto the 2-D grid satisfies

$$\frac{\partial\mathbf{v}}{\partial x} = \frac{1}{\Delta x}D_x\mathbf{v} + \mathbf{t}_x, \tag{14}$$

$$\frac{\partial\mathbf{v}}{\partial y} = \frac{1}{\Delta y}D_y\mathbf{v} + \mathbf{t}_y, \tag{15}$$

where $\mathbf{t}_x$ and $\mathbf{t}_y$ are the truncation errors in each direction. The discretization spacings Δx and Δy now represent the finest length scale in each direction, following the convention of [6]. In this way, the scaling factor of the discretization for coarse regions is incorporated into the coefficients of D_x and D_y. These finite-difference approximations, with the SAT boundary implementation from [12], give a semidiscrete form of (1),

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = -\left(\frac{a}{\Delta x}D_x + \frac{b}{\Delta y}D_y\right)\mathbf{u} - \frac{a}{\Delta x}\mathbf{g}_x - \frac{b}{\Delta y}\mathbf{g}_y, \tag{16}$$

where the vectors $\mathbf{g}_x$ and $\mathbf{g}_y$ contain the contribution of the boundary conditions. To better understand the structure of the finite-difference matrices in 2-D, see that on a uniform grid ($\Omega^f = \{\}$), the matrices D_x and D_y are block diagonal, and may be written explicitly as

$$D_x = \begin{bmatrix} P^{-1}\widetilde{Q} & & & \\ & P^{-1}\widetilde{Q} & & \\ & & \ddots & \\ & & & P^{-1}\widetilde{Q} \end{bmatrix}, \quad \mathbf{g}_x = \begin{bmatrix} P^{-1}\tau\mathbf{s}g_1^x(t) \\ P^{-1}\tau\mathbf{s}g_2^x(t) \\ \vdots \\ P^{-1}\tau\mathbf{s}g_N^x(t) \end{bmatrix},$$

where $P, \widetilde{Q}, \tau$ and $\mathbf{s}$ are all as defined in 1-D, and similarly in the y-direction by the appropriate transformation.

On a patch-refined grid, D_x and D_y will have this block-diagonal structure in uniform regions, but near interfaces a more general form is required. For a grid with a total of $\mathcal{N} = |\mathcal{I}|$ nodes, there will be a subset of nodes, $\mathcal{I}^i \subset \mathcal{I}$, of length $n = |\mathcal{I}^i|$ in the vicinity of $\partial\Omega^f$ that require special interface stencils. These stencils depend on a larger subset of nodes, $\mathcal{I}^d \subset \mathcal{I}$, of length $m = |\mathcal{I}^d| > n$ (which will be different for each derivative). Thus D_x and D_y have potentially dense $n \times m$ submatrices over the interface nodes $\mathcal{I}^i \times \mathcal{I}^d$, which we label $\widehat{D}_x$ and $\widehat{D}_y$, respectively.

These matrices are defined as general explicit finite difference approximations

$$\frac{\partial u_k}{\partial x} = \frac{1}{\Delta x}\sum_{j\in\mathcal{I}^{dx}} \widehat{D}_{x,kj}u_j, \tag{17}$$

$$\frac{\partial u_k}{\partial y} = \frac{1}{\Delta y}\sum_{j\in\mathcal{I}^{dy}} \widehat{D}_{y,kj}u_j, \tag{18}$$

for all $k \in \mathcal{I}^i$, where Δx and Δy are the discretization scales in the fine region. The formal order of each approximation is determined by Taylor series expansion of the polynomial test function

$$f_{(z_1,z_2)}(x,y) = x^{z_1}y^{z_2}, \tag{19}$$

to degree specified by the index pair (z_1, z_2), in the combinations shown in Table 1. For either (17) or (18) to be accurate to a given order, all index pairs up to that order must be satisfied exactly: for a third order-accurate x-derivative, (17) must satisfy all ten index pairs up to third-order. Both directions must be considered for each derivative because of the off-direction perturbations introduced by the hanging-node geometry of the grid. Simple algebra will show that this does not result in a cross-dependence on the discretizations; therefore, the x-derivative is independent of Δy and vice versa. Note that in order for the discretization to preserve the global convergence rate of an interior scheme of order σ, it is expected that the interface schemes must be accurate to at least order $(\sigma - 1)$ [20].

3.3. Error bound and stability criteria

The stability of (16) is examined in the context of a positive definite norm matrix H, such that for $\mathbf{u} \in \mathbb{R}$,

$$\|\mathbf{u}\|_H^2 = (\mathbf{u}, H\mathbf{u}) = \mathbf{u}^T H\mathbf{u} > 0, \tag{20}$$

with the equivalence of the norms,

$$h_{\text{lower}}\|\mathbf{e}\|^2 \leqslant \|\mathbf{e}\|_H^2 \leqslant h_{\text{upper}}\|\mathbf{e}\|^2. \tag{21}$$

The error analysis follows [12], with some modifications appropriate for our case. Writing (16) for the projection of the exact solution, $\mathbf{v}$,

$$\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t} = -\left(\frac{a}{\Delta x}D_x + \frac{b}{\Delta y}D_y\right)\mathbf{v} - \frac{a}{\Delta x}\mathbf{g}_x - \frac{b}{\Delta y}\mathbf{g}_y - a\mathbf{t}_x - b\mathbf{t}_y, \tag{22}$$

an equation for the error, $\mathbf{e} = \mathbf{v} - \mathbf{u}$, may be derived according to

$$\frac{\mathrm{d}\mathbf{e}}{\mathrm{d}t} = -\left(\frac{a}{\Delta x}D_x + \frac{b}{\Delta y}D_y\right)\mathbf{e} + \mathbf{t}, \tag{23}$$

where $\mathbf{t} = -a\mathbf{t}_x - b\mathbf{t}_y$ contains the truncation error contributions from both directions. Taking the norm of this error in the sense just defined gives

$$\frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{e}\|_H^2 = \mathbf{e}^T\left[\left(-\frac{a}{\Delta x}D_x - \frac{b}{\Delta y}D_y\right)^T H + H\left(-\frac{a}{\Delta x}D_x - \frac{b}{\Delta y}D_y\right)\right]\mathbf{e} + 2(\mathbf{t}, H\mathbf{e}). \tag{24}$$

Writing the term in square brackets as

$$A = \left(\frac{a}{\Delta x}D_x + \frac{b}{\Delta y}D_y\right)^T H + H\left(\frac{a}{\Delta x}D_x + \frac{b}{\Delta y}D_y\right), \tag{25}$$

leads to

$$\frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{e}\|_H^2 = -\mathbf{e}^T A\mathbf{e} + 2(\mathbf{t}, H\mathbf{e}). \tag{26}$$

It is assumed that A can be diagonalized according to

$$A\mathbf{x}_i = \lambda_i\mathbf{x}_i, \tag{27}$$

where λ_i and $\mathbf{x}_i$ denote the eigenvalues and normalized eigenvectors of A, respectively. A positive definite matrix A, such that $\mathbf{e}^T A\mathbf{e} > 0$ for all $\mathbf{e} \neq 0$, implies $\mathrm{Re}(\lambda_i) > 0$. Expressing the error in this basis gives $\mathbf{e} = \sum \mathbf{x}_i c_i$, where c_i denote scalar coefficients, and leads to

Table 1
Test polynomial index pairs (z_1, z_2) for given derivative accuracy.

Derivative order	Index pairs
0th order	(0,0)
1st order	(1,0), (0,1)
2nd order	(2,0), (0,2), (1,1)
3rd order	(3,0), (0,3), (2,1), (1,2)
4th order	(4,0), (0,4), (3,1), (1,3), (2,2)

$$-\mathbf{e}^T A\mathbf{e} = -\sum c_i^2\lambda_i \leqslant -\lambda_{\min}\|\mathbf{e}\|^2 \leqslant -\frac{\lambda_{\min}}{h_{\text{upper}}}\|\mathbf{e}\|_H^2, \tag{28}$$

where $\lambda_{\min} = \min(\text{Re}(\lambda_i))$. Using the Cauchy–Schwarz inequality, $(\mathbf{t}, H\mathbf{e}) \leqslant \|\mathbf{e}\|_H\|\mathbf{t}\|_H$, the error rate

$$\frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{e}\|_H^2 \leqslant -\frac{\lambda_{\min}}{h_{\text{upper}}}\|\mathbf{e}\|_H^2 + 2\|\mathbf{e}\|_H\|\mathbf{t}\|_H, \tag{29}$$

reduces to

$$\frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{e}\|_H \leqslant -\frac{\lambda_{\min}}{2h_{\text{upper}}}\|\mathbf{e}\|_H + \|\mathbf{t}\|_H, \tag{30}$$

which may be integrated to give the error bound [12],

$$\|\mathbf{e}\|_H \leqslant \frac{2h_{\text{upper}}}{\lambda_{\min}}\left(\sup_{0\leqslant s\leqslant t}\|\mathbf{t}(s)\|_H\right)\left(1 - e^{-\frac{\lambda_{\min}t}{2h_{\text{upper}}}}\right). \tag{31}$$

In the case where A is only positive semidefinite, such that $\mathbf{e}^T A\mathbf{e} \geqslant 0$, the error bound is modified to become

$$\|\mathbf{e}\|_H \leqslant \left(\sup_{0\leqslant s\leqslant t}\|\mathbf{t}(s)\|_H\right)t, \tag{32}$$

which can be interpreted, with some care, as the limit of (31) when $\lambda_{\min} \to 0$. This shows that the scheme converges for $t < +\infty$ and suffers at most a linear growth in error with time owing to truncation error, thus exhibiting Lax (finite time) stability. To prove asymptotic stability requires a bound on $\|\mathbf{t}(s)\|_H$ for all time, which depends, in general, on the boundary and initial data, as well as on the order of the approximation. A demonstration of asymptotic stability for two particular grids is shown in Section 5.

The stability result depends on the condition

$$-\frac{a}{\Delta x}(HD_x + D_x^T H) - \frac{b}{\Delta y}(HD_y + D_y^T H) \leqslant 0. \tag{33}$$

Notice that the norm H is as yet undefined, except by the fact that it must be a positive definite matrix. For the boundaries, [12] proposes

$$H_b = P_x^{1/2}P_yP_x^{1/2}, \tag{34}$$

where H_b spans the set of nodes $\mathcal{I}^b \subset \mathcal{I}^c$ in the vicinity of $\partial\Omega$ and $D_x = P_x^{-1}Q_x$. This can be shown to satisfy (33) with $a > 0$ and $b > 0$. In a patch-refined domain with uniform boundary regions, the familiar 1-D SBP boundary scheme may be used, e.g., from [6], and the 2-D matrices P_x and P_y may be formed directly from the explicitly known P of the boundary scheme.

At grid interfaces, stability should be independent of the advection velocities so that the closure may be applied to systems with waves traveling in arbitrary directions. In the case of a uniform domain, the interior makes no contribution to the stability condition (33), and $\mathbf{e}^T A\mathbf{e}$ is dependent only on the boundary closure. For the patch-refined domain, in order to have the error bound (31) independent of the interface closure, this should remain true. This leads to the stronger conditions

$$\widehat{H}\widehat{D}_x + \widehat{D}_x^T\widehat{H} = 0, \tag{35}$$

$$\widehat{H}\widehat{D}_y + \widehat{D}_y^T\widehat{H} = 0, \tag{36}$$

for the interface region, where $\widehat{D}_x$ and $\widehat{D}_y$ are defined by Eqs. (17) and (18). The positive definite matrix $\widehat{H}$ spans $\mathcal{I}^i \times \mathcal{I}^i$, and is common to both conditions. It is to be determined along with the elements of $\widehat{D}_x$ and $\widehat{D}_y$ when constructing the stencil.

We make the following remarks:

1. For symmetric $\widehat{H}$, the equations are equivalent to requiring that the products $\widehat{H}\widehat{D}_x$ and $\widehat{H}\widehat{D}_y$ be antisymmetric matrices.
2. In the domain interior, away from interfaces and boundaries, $\mathcal{I} \setminus (\mathcal{I}^i \cup \mathcal{I}^b), D_x$ and D_y are naturally anti-symmetric by the centered difference scheme, automatically satisfying (35) and (36).
3. The full-domain matrix H has H_b in boundary regions, $\widehat{H}$ in interface regions and is diagonal in interior regions. Since all interior regions, including the interfaces, satisfy the stronger conditions (35) and (36), (33) holds throughout the domain and the estimate (31) is preserved.
4. The criteria (35) and (36) are independent of the cell aspect ratio. Although the refinement factor is fixed for a particular interface closure, any combination of Δx and Δy may be used.
5. The norm matrix $\widehat{H}$ is not necessary for computational implementation of the scheme, as it does not appear in (16), but it is critical in the derivation of the interface stencils because it links the x- and y-derivatives in a way that ensures stability.

3.4. Conservation across interfaces

Conservation of the numerical method follows from the summation-by-parts condition [21]. For the general 2-D differential equation

$$\frac{\partial u}{\partial t}+\frac{\partial f}{\partial x}+\frac{\partial g}{\partial y}=0, \tag{37}$$

where $f(u)$ and $g(u)$ are arbitrary nonlinear functions, the weak form of the equation is given by

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega}\phi u\mathrm{d}x-\int_{\Omega}\left(u\frac{\partial\phi}{\partial t}+f\frac{\partial\phi}{\partial x}+g\frac{\partial\phi}{\partial y}\right)\mathrm{d}x=0, \tag{38}$$

where $\phi(x,y,t)$ is an arbitrary test function that vanishes at the boundaries. Discrete conservation follows if the numerical approximation to (37) has a weak form similar to (38), indicating that the conservation law is preserved by the discretization [21]. For the semidiscrete version of (37),

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t}+D_x\mathbf{f}+D_y\mathbf{g}=0, \tag{39}$$

multiplying by the vector $\Phi^T H$ and simplifying using the properties of the SBP operators, we obtain

$$\frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{u}^T H\Phi)-\left(\mathbf{u}^T H\frac{\mathrm{d}\Phi}{\mathrm{d}t}+\mathbf{f}^T HD_x\Phi+\mathbf{g}^T HD_y\Phi\right)=0. \tag{40}$$

This is clearly the discrete version of (38), and asymptotically approaches it in the limit of infinite resolution. The numerical solution therefore satisfies the weak form of the conservation law by the SBP condition, which by construction is unaffected by the presence of grid interfaces.

4. Interface closures

4.1. Mapping of interface schemes

Before considering the individual interface stencils, it is important to understand how the interface elements of Fig. 3 may be used to construct the interface scheme for an arbitrary domain. Consider the sketched edge and corner stencils of Fig. 4, shown in their default orientation (described relative to the fine region). Throughout this section, it is more convenient to map the global index set into a new, locally ordered set, $\mathcal{I}\mapsto\mathcal{I}'$. This local numbering is used to indicate the orientation of the stencil and the relative position of each node. The particular mapping between the global set, $\mathcal{I}$, and these local sets, $\mathcal{I}'$, depends on the global domain and constitutes an implementation issue.

Consider now a simple box refinement on a square grid, where the inner box is refined by a factor of two in each direction. To define the complete interface scheme for this grid, the stencils of Fig. 4 must be mapped to the four edge interfaces and four corner interfaces of this grid. There are two obvious mapping alternatives: reflection or rotation. In the former scheme, the upper edge is mapped from the original right-hand edge by reflection in the line $y=x$, and the remaining edges are obtained by reflection in $y=-x$. Examination of the corners, however, will reveal that the original upper-right corner cannot be simply mapped to the upper-left or the lower-right corners. Using a counterclockwise rotation mapping, though, all the geometries may be obtained from the original pair of Fig. 4, resulting in the numbering shown in Fig. 5. The original stencils are shaded and the transformed orientations are boxed.

The final step is to map the particular x- and y-derivatives to the edges and corners. For this purpose, consider 90° rotations of the derivatives, generating the sequence in Table 2, which shows how $\widehat{D}_x$ and $\widehat{D}_y$ in the original orientation are mapped to the x- and y-derivatives in each of the other orientations. With this mapping scheme, one solution for each interface type is sufficient for closure of the interfaces on an arbitrarily refined grid.

4.2. Interface scheme construction

Construction of the interface schemes has four steps:

1. Define interface matrices.
2. Form the accuracy equations from (17) and (18).

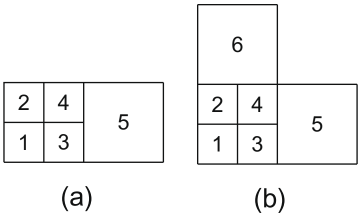


Fig. 4. Cartoon representation of interface elements: (a) right-hand edge, (b) upper-right corner.

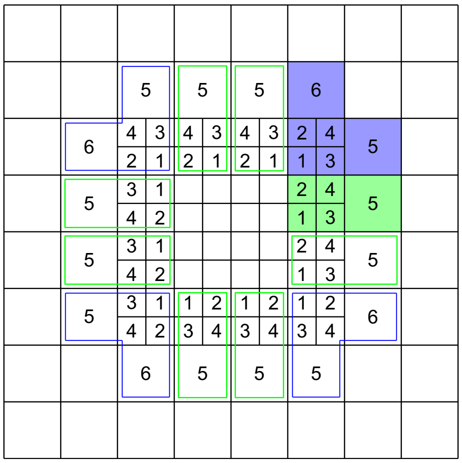


Fig. 5. Box-refined grid, with interfaces mapped to each grid location.

Table 2
Transformations of the derivative matrices from the default orientation to other orientations by rotation.

Edge orientation	Corner orientation	x-Derivative	y-Derivative
Right	Upper right	$\widehat{D}_x$	$\widehat{D}_y$
Upper	Upper left	$-\widehat{D}_y$	$\widehat{D}_x$
Left	Lower left	$-\widehat{D}_x$	$-\widehat{D}_y$
Lower	Lower right	$\widehat{D}_y$	$-\widehat{D}_x$

3. Form the stability Eqs. (35) and (36).
4. Solve the resulting system of equations.

The interface matrices are defined from the interface node set $\mathcal{I}^i$ and its dependence set $\mathcal{I}^d$ (which includes the interface nodes, and may not be the same for x- and y-derivatives). Dense $n \times m_x$ matrix $\widehat{D}_x$, $n \times m_y$ matrix $\widehat{D}_y$, and $n \times n$ matrix $\widehat{H}$ are assumed, the latter to contain the influence of the interface within $\mathcal{I}^d$. The accuracy equations are obtained from (17)–(19), resulting in the linear system of equations

$$\frac{\partial f_{(z_1,z_2)}}{\partial x}(x_k, y_k) = \frac{1}{\Delta x}\sum_{j\in\mathcal{I}^{dx}} \widehat{D}_{x,kj} f_{(z_1,z_2)}(x_j, y_j), \tag{41}$$

$$\frac{\partial f_{(z_1,z_2)}}{\partial y}(x_k, y_k) = \frac{1}{\Delta y}\sum_{j\in\mathcal{I}^{dy}} \widehat{D}_{y,kj} f_{(z_1,z_2)}(x_j, y_j), \tag{42}$$

formed for each $k \in \mathcal{I}^i$, where (x_k, y_k) is the location of the node globally indexed by k. For third-order accuracy, Table 1 shows that there are ten equations for each k for each derivative. The stability equations are the elements of the $m \times m$ matrices formed by the right-hand sides of (35) and (36). These are quadratic equations in the unknown elements of $\widehat{D}_x$, $\widehat{D}_y$ and $\widehat{H}$. The resulting system of quadratic equations is solved for the elements of the interface matrices, which is done numerically because of its very large size.

For a solution to exist at an interface, the norm H must have a specific form in the interior regions of the domain, where (6) or (7) applies. Let H_c define the part of H that corresponds to coarse interior nodes in $\mathcal{I}^c \setminus (\mathcal{I}^i \cup \mathcal{I}^b)$, and similarly let H_f correspond to the fine interior part $\mathcal{I}^f \setminus \mathcal{I}^i$. Then, we require

$$H_c = \operatorname{diag}(1) = I, \tag{43}$$

$$H_f = \operatorname{diag}\left(\frac{1}{r_x r_y}\right) = \frac{1}{r_x r_y} I, \tag{44}$$

i.e., H has 1 and $1/r_x r_y$ on the diagonal in coarse and fine regions, respectively. A physical intuition exists for this result: the norm, H, is effectively weighting the solution by the area occupied by each cell; for refinement ratios $r_x = r_y = 2$, four fine cells occupy the same area and have the same weight as one coarse cell. The interface solutions presented next also fixed the diagonal of $\widehat{H}$ to follow this scheme, although this is not essential to the existence of a solution.

For each of the following interface solutions, the particular difference matrices $\widehat{D}_x$ and $\widehat{D}_y$ used in Section 5 are too large to print but are available in the supplementary material published online or from the editorial office. These are sufficient for implementation of the interface scheme. Row and column numbering of each matrix follows the local node numbering presented in Sections 4.3–4.5. Note that each of the following systems involve many degrees of freedom and in some cases these solutions are not unique. Optimization of the solution families may be possible but has not been exhaustively investigated.

4.3. Edge interface

The edge interface is the simplest of the four interface geometries, and the default orientation is considered to be that shown in Fig. 3(a). It is clear that a special difference stencil is required for the x-derivative as it passes through the interface, but examination of the difference matrix for the y-derivative will show that it cannot satisfy (36) without a special stencil too. With a five-point interior stencil, the interface region must include at least the first two nodes on either side of the change in discretization. It was found that one additional point in each subrow was required in the refined region. Fig. 6 shows the fourth-order interface stencil thus derived, each interface node being formally third-order accurate. The dark-colored nodes (numbered above each node 1–8) comprise the interface set $\mathcal{I}^i$ for this stencil, and the gray nodes (together with the interface nodes, the full set numbered below each node 1–30 for D_x and 1–24 for D_y) comprise the dependence set, $\mathcal{I}^d$.

For the x-derivative, the dependence set includes the natural extensions in the x-direction of the stencils for nodes 1, 2 and 8. These adjacent noninterface nodes (9–12 and 21–22 of $\mathcal{I}^d$) have the interior dependence on nodes 1–4 on the fine side, and nodes 7 and 8 on the coarse side, linking the interface stencil to the interior regions. The y-derivative does not have a natural dependence on those nodes, and since the accuracy conditions at the interface may be resolved entirely within the stencil, its stencil has no need to include them. However, both stencils include dependence on the coarse rows above and below the row of interest, the x-derivative stencil needing the additional nodes to satisfy the accuracy conditions in the y-direction. The y-derivative at nodes 7 and 8 achieves third-order accuracy with essentially a three-point stencil in that direction.

As the most basic of the interface geometries, the edge stencil is expected to tesselate along interface edges and to be compatible with corner interfaces, as was assumed in Section 4.1. An arbitrary edge solution may not do either of these things; of all the edge solutions that exist, only a subset will have compatible corner solutions and, therefore, be usable (or useful) in practice. The solution presented here was derived simultaneously with the convex corner and corresponds to one particular member of the family of solutions. An alternative formulation of the edge interface is presented in Appendix A.

Alongside this fourth-order solution, variations with different interior schemes were also investigated, the critical condition for existence of a solution being that H has the interior form (43) and (44). A second-order edge solution (first-order at the interface) permits a smaller interface set comprising only nodes 3–7 of $\mathcal{I}^i$ from the fourth-order stencil because of the smaller three-point interior stencil. Dependence on the adjacent coarse rows is still required. A rational solution is easily found for this case, as $\widehat{H}$ has a particularly simple form. Similarly, a sixth-order solution exists that is fifth order at the interface, with an appropriately larger stencil.

4.4. Convex corner interface

The corner interface turns out to be a challenging problem in the summation-by-parts context. To satisfy the stability equations and thus ensure a stable and nondissipative interface treatment, a trade-off must be made that results in a loss of local accuracy at the corner cells compared to the interior accuracy. We favor stability over local accuracy as no zero-dissipation interface closures exist, and the SBP property has other desirable features that mitigate the local loss of accuracy. In lieu of a formal proof, experimentation with stencil sizes and interior schemes yielded the results shown in Table 3: with a

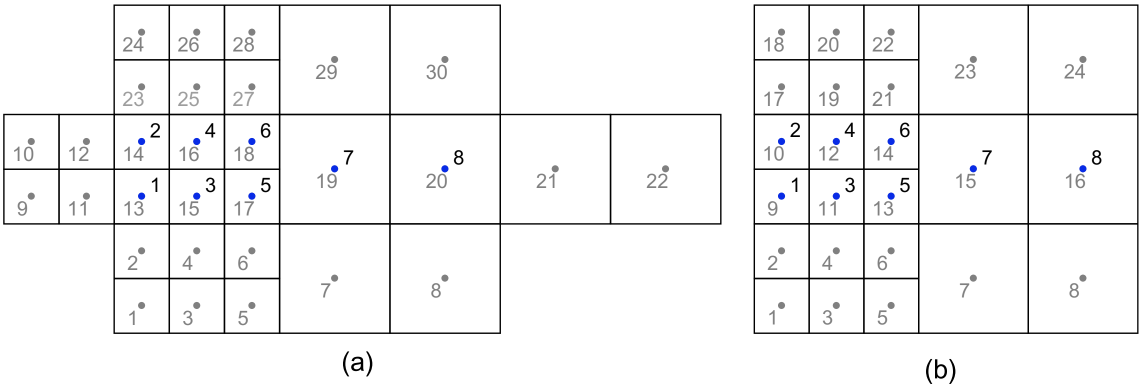


Fig. 6. Edge interface stencils for the fourth-order explicit scheme: (a) x-derivative, (b) y-derivative. Interface nodes are numbered 1–8 on each stencil and the dependence sets 1–30 (x-derivative) and 1–24 (y-derivative).

Table 3
Maximum accuracy achievable at every corner interface node with an SBP formulation.

Interior order	Edge order	Corner order
Second	First	Zeroth
Fourth	Third	First
Sixth	Fifth	Second

fourth-order interior scheme, the maximum accuracy achievable at every node in the stencil is just first order, though most nodes can satisfy accuracy up to third order.

Parameterization of the interior difference stencils revealed that it is the order of the interior scheme rather than the particular coefficients that limits accuracy at the corner interface. For a centered five-point stencil with coefficients $(-\alpha, -\beta, 0, \beta, \alpha)$, there is a family of second-order accurate solutions with $\beta = 1/2 - 2\alpha$, and a special case of $\alpha = -1/12$ that is fourth order. Leaving α as a parameter when solving the accuracy conditions at the corner, the maximum accuracy achievable at all interface nodes is zeroth order, except in the special case of $\alpha = -1/12$, i.e. fourth order. Similarly with a seven-point interior scheme: from the family of fourth-order stencils, only the special case of the sixth-order stencil gave a solution that was at least second order everywhere.

For the fourth-order stencil shown in Fig. 7, two nodes of $\mathcal{I}^i$ (14 and 15) are first order, three (8, 12 and 16) are second order, and the remainder are third-order accurate. The dependence set includes the interface nodes (numbered 1–50). The positions of the low-order nodes are not unique, but it is preferable to place them in the fine region where the truncation error is minimized by the smaller discretization. The most serious consequence of this result is that average fourth-order convergence can no longer be expected [20]; at worst, despite using a fourth-order interior scheme, only second-order convergence could be seen. Note, however, that since there exists no usable second-order solution (a zeroth-order-accurate derivative is meaningless), the fourth-order stencil represents the lowest-order interface solution that satisfies the stability criteria.

The extent of the convex corner interface is determined primarily by the extent of the edge stencil. The inclusion of nodes 1 and 2 in the edge stencil demands that the corner span at least two coarse rows/columns, so nodes 2–16 of the corner must be included to avoid any ambiguity in the stencils at those nodes. The differences between the x- and y-derivatives are lim-

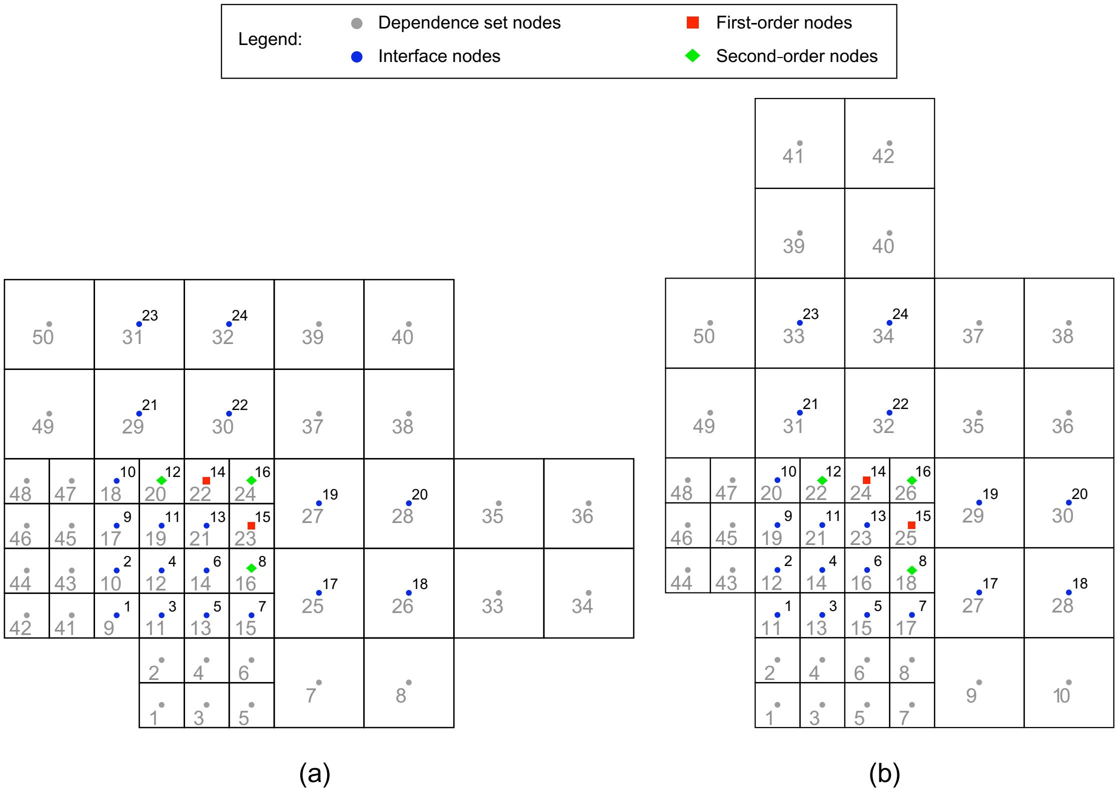


Fig. 7. Convex corner interface stencils for the fourth-order explicit scheme: (a) x-derivative, (b) y-derivative. Interface nodes are numbered 1–24 and the dependence set nodes are numbered 1–50.

ited to extensions of the stencils in the directions of the derivatives. Both stencils must be dependent on the adjacent edges for compatibility (both below and to the left of the interface, nodes 1–8 and 43–50 of $\mathcal{I}^d$), but node 1 requires another two nodes to its left (x) or below (y). The x-derivative stencil includes the interior nodes to the right of coarse nodes 17–20 on which they are naturally dependent (nodes 33–36); similarly, for the y-derivative, the coarse nodes above 21–24 are included (nodes 39–42).

The corner stencil has dense 24 × 50 derivative matrices $\widehat{D}_x$ and $\widehat{D}_y$ with many degrees of freedom, so again a family of solutions exist. With the current edge stencil, the dependence set may be customized so that all nodes need not be dependent on the full set. For example, nodes 17–20 need not be dependent on nodes 41–50, etc. The stencils may also be optimized to reduce the truncation error at the low-order nodes.

It should be noted that there exists a second-order corner interface that is locally first order, which consequently could be used as a meaningful derivative approximation. Unfortunately, this requires a modified stencil in the interior of the fine region: instead of the usual three-point second-order central difference scheme, the five-point second-order stencil with coefficients $(-1/4, 0, 0, 0, 1/4)$ is needed, which has two serious deficiencies. First, the grid resolution is not truly refined, as this modified stencil is equivalent to a staggered grid with the double (coarse) refinement, and second, multiple refinement with this scheme is not possible. Multiple refinement is dependent on any interface appearing locally identical, regardless of its refinement relative to a global scale. If the fine region requires a special stencil, then the same interface scheme cannot be used again in a second refinement of that region, resulting in the loss of proper nesting of the AMR grid hierarchy.

4.5. Concave corner interface

The concave corner geometry suffers similar difficulties to those of the convex corner. Again, for a fourth-order interior scheme, the maximum accuracy achievable at all nodes at the interface is first order. Fig. 8 shows the stencil thus derived, with four first-order nodes (nodes 5, 6, 17 and 18) and two second-order nodes (11 and 23). Note that for both corners, these first-order nodes do, in fact, satisfy the accuracy conditions for the second-order cross-derivative $\partial^2/\partial x\partial y$ in the Taylor series, as it is the second derivatives ($\partial^2/\partial x^2$ and $\partial^2/\partial y^2$) that present the closure problem. Further, at the concave corner, node 6 is second-order accurate in x in the x-derivative, and node 18 in y in the y-derivative. This arrangement of the low-order nodes was chosen to ensure symmetry of the corner and minimize the error at these points as far as is possible.

Once again, the extent of the concave corner interface is determined by the edge scheme, the potential ambiguity here occurring at the coarse nodes 25–28. The dependence set again includes the adjacent edge row/column (above and to the left of the interface set, nodes 1–8 and 51–58 of $\mathcal{I}^d$) and the natural stencil extensions in the x- and y-directions. The large 28 × 58 stencil is a consequence of this geometry being dominated by the fine region, but optimization and customization of the dependence sets for each individual node of the stencil can potentially reduce the density of the derivative matrices somewhat.

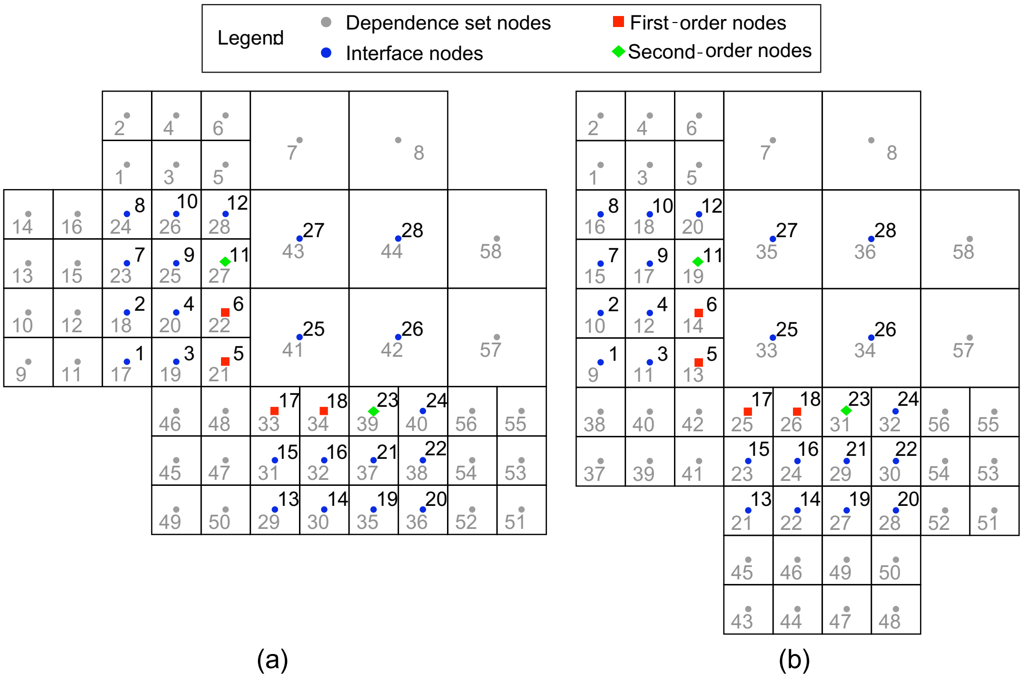


Fig. 8. Concave corner interface stencils for the fourth-order explicit scheme: (a) x-derivative, (b) y-derivative. Interface nodes are numbered 1–28 and the dependence set nodes are numbered 1–58.

While the convex corner interface solution was obtained simultaneously with the edge interface to ensure compatibility, it is not necessary to do the same with the concave corner: a solution was found using the previously derived edge solution. This keeps each stencil derivation problem to a size that is feasible to solve on a modern desktop computer.

4.6. Refinement region topology

Having now defined the full set of interior interface stencils, it is clear that there are a few rules that must be imposed on a refinement scheme such that only these stencils are required to close the interfaces. The basic principle is to avoid interference between corners caused by overlapping dependence sets, such that the elements of $HD + D^T H$ are the same as those in (35) and (36) from the original definitions of the individual stencils. Thus in a refined region, any two corners must be separated by at least two coarse cells, and since each corner itself occupies two coarse cells in each direction, the minimum length of any side of a refinement region is six coarse cells. Similarly, the minimum separation between two parallel refined regions is eight coarse cells. For multiple levels of refinement, to preserve the independence of each interface, at least seven intermediate cells are required between consecutive refinement interfaces.

Near boundaries, the dependence sets of the interface and boundary closures are permitted to overlap, because the contribution of the boundary stencil to (35) and (36) will be the same as the interior stencil expected by the interface. For a four-point boundary stencil with a six-point dependence set, the minimum separation between the domain boundary and an interface is eight cells. With periodic boundary conditions, the interface may extend to the boundary as long as the refinement scheme is periodic also.

5. Test examples

5.1. Advecting wave

For the first test example, consider the scalar advection Eq. (1) with $a = b = 1$, given by

$$\begin{aligned}
&\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} = 0, \qquad (45)\\
&u(0, y, t) = \sin[\omega(y - 2t)],\\
&u(x, 0, t) = \sin[\omega(x - 2t)],\\
&u(x, y, 0) = \sin[\omega(x + y)],
\end{aligned}$$

with a frequency parameter $\omega = 2\pi$. This has an analytic solution

$$u(x, y, t) = \sin[\omega(x + y - 2t)], \qquad (46)$$

which is used to evaluate the accuracy of the numerical method.

To show that the interface schemes do possess the claimed stability properties, we examine the spectrum of the semidiscrete form of (45),

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = \left(-\frac{1}{\Delta x} D_x - \frac{1}{\Delta y} D_y\right)\mathbf{u} - \frac{\mathbf{g}_x}{\Delta x} - \frac{\mathbf{g}_y}{\Delta y}. \qquad (47)$$

Time stability of this equation (with $\Delta x = \Delta y$) demands that the matrix $(-D_x - D_y)$ have eigenvalues with nonpositive real parts. This matrix has a structure strongly dependent on the grid and its refinement. Two refinement schemes are considered to demonstrate by example that the interface schemes are stable. For the first case of a box refinement, the central third of the domain is refined in each direction by the usual factor of two. The resulting grid has only edge and convex corner interfaces. The second grid of a cross refinement, has a "+"-shaped refined region in the center of the domain, which is three-fifths the width and height of the domain and occupies one-fifth of the total area. It has eight convex corner interfaces and four concave, with edges in between, providing a test of the three interface solutions together. Both are shown schematically in Fig. 9.

For our demonstration, each grid is divided into blocks of eight coarse cells per side. The box refinement has three blocks in each direction, resulting in a 24 × 24 grid before refinement. The cross refinement has five blocks and a 40 × 40 coarse grid. With the refinements described, the grids have 768 and 2560 nodes, respectively. Fig. 10 shows the eigenvalues of $(-D_x - D_y)$ in each case, both using the fourth-order explicit boundary scheme from [6] and SAT parameter $\tau = 2$ at the boundary. Note that it is the presence of boundaries alone that results in the spectra having any nonzero real part; the equivalent spectra for domains with periodic boundaries are purely imaginary. This is consistent with the assertion that the interface schemes introduce no additional numerical dissipation.

Two features of Fig. 10 stand out. First, there appear to be a significant number of eigenvalues on or near the vertical axis. Closer inspection reveals that the maximum real part in each example is very small $O(10^{-13})$ but still clearly nonpositive, so the stability criteria is satisfied with all eigenvalues indeed lying in the left half plane. Second, the maximum imaginary part is large relative to the minimum real part, especially compared to the spectrum for a similar uniform grid. This number

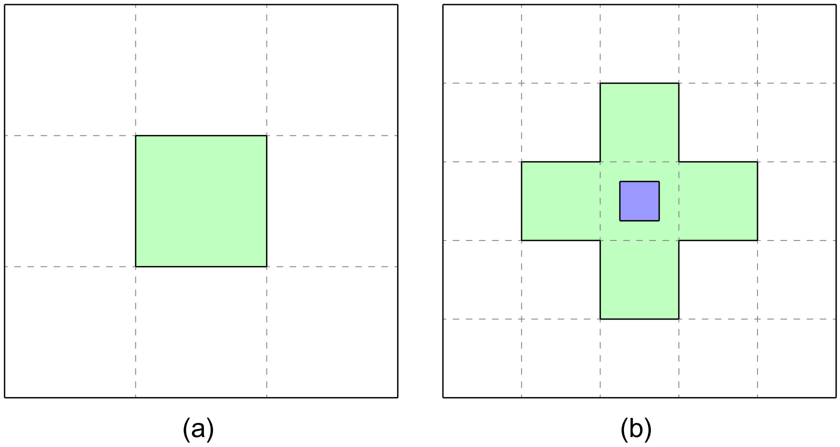


Fig. 9. Diagrams of the grids used for the validation problems: (a) box refinement, (b) cross/cross + box refinement. Shaded regions are refined by a factor of two in each direction compared to the base grid and dashed lines indicate the division of the domain into blocks. The darker shaded region on (b) shows the location of the second level of refinement for the cross + box grid.

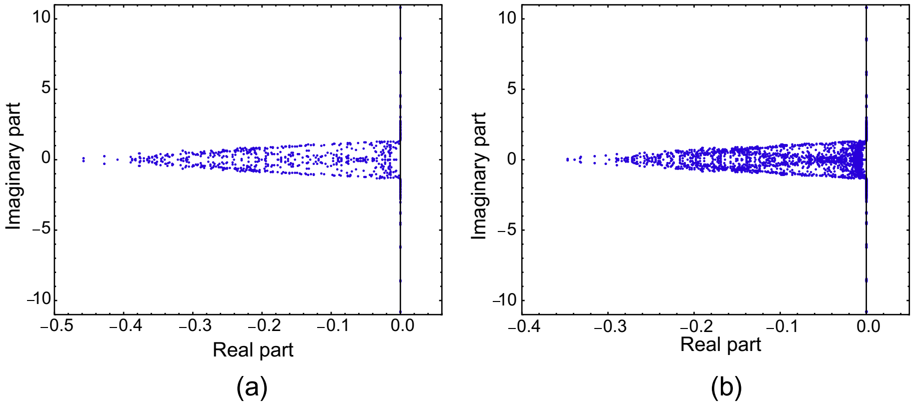


Fig. 10. Spectra of the matrices $(-D_x - D_y)$ for the test domains: (a) box refinement, (b) cross refinement. These include boundaries using the SAT scheme with $\tau = 2$.

affects the stability limit of the time-marching scheme used to implement (45): for a third-order Runge–Kutta scheme (RK32 from [22]) and advection velocities $a = b = 1$, we find the time discretization limit (based on the refined scale Δx)

$$\Delta t \lessapprox 0.16\Delta x. \tag{48}$$

This agrees with the stability limit observed in practice. Note that this limit may be extended by using a fourth-order Runge–Kutta scheme, and is dependent on the value of τ at the boundaries.

For the convergence study, a third grid with an additional level of refinement is considered. The cross + box refinement is based on the cross refinement previously described, but with the middle quarters of the subblock at the center of the cross refined in each direction (one quarter of its area), for a total refinement factor of 4 across the grid and demonstrating the interface scheme across multiple levels of refinement. All simulations use the fourth-order interface schemes described in Section 4. Table 4 shows the results from the convergence study on the cross and cross + box refinements, both run to a time $t = 1$ (two periods) with a CFL number of 1/8. The twice-refined grid is stable at this CFL when based on its finest grid scale.

It is immediately clear from these results that a fourth-order convergence rate is not achieved. Average convergence (in the L_2-norm) is approximately third order, but the point-wise L_∞-norm converges at closer to a second-order rate. This could be expected with the presence of first-order nodes in the domain, but it appears that in the averaged sense, the rate is a little better. Note that the number of first-order (and second-order) nodes is fixed, a function of the geometry of the refinement rather than the resolution itself. In the convergence study on the cross refinement, there are 32 first-order points; at the coarse resolution, this represents 1.25% of the grid, while at the finest resolution (coarse $\Delta x = 1/160$) this is just 0.08%.

There is a computational payoff for the ability to refine the grid locally to mitigate the lost convergence rate and accuracy. A simple comparison was made between the cross + box grid and a uniform grid of equivalent resolution. For the twice-re-

Table 4
Convergence results for the advection equation.

Coarse Δx	Cross refinement				Cross + box refinement			
	$\log_{10}(L_2)$	Rate	$\log_{10}(L_\infty)$	Rate	$\log_{10}(L_2)$	Rate	$\log_{10}(L_\infty)$	Rate
1/40	−2.004		−1.173		−2.011		−1.060	
1/60	−2.489	2.75	−1.584	2.34	−2.518	2.88	−1.314	1.44
1/80	−2.817	2.62	−1.791	1.65	−2.819	2.41	−1.609	2.37
1/120	−3.304	2.76	−2.145	2.01	−3.299	2.72	−2.109	2.84
1/160	−3.642	2.71	−2.350	1.64	−3.641	2.74	−2.403	2.36
Average rate		2.72		1.94		2.69		2.31

fined grid, a coarse discretization of $1/40$ gives $\Delta x = 1/160$ in the most refined region and a total of 2752 nodes. The appropriate comparison is therefore to a uniform grid with that resolution having 25,600 nodes. In testing, without optimization of the interface scheme implementation, the uniform grid took 10–13 times longer to reach $t = 1$ with the same time step. This offers some saving beyond the factor of ~ 9.3 reduction in the problem size with the locally refined grid, despite the cost of the matrix multiplication operations at interfaces.

5.2. Inviscid compressible vortex

For the second example, a nonlinear problem is considered to demonstrate more generally the stability of the interface schemes. The dimensionless compressible Euler equations, with parameter $\beta = (\gamma - 1)/2$,

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x}(\rho u) + \frac{\partial}{\partial y}(\rho v) = 0, \tag{49}$$

$$\frac{\partial}{\partial t}(\rho u) + \frac{\partial}{\partial x}\left(\rho u^2 + \frac{p}{\gamma M_0^2}\right) + \frac{\partial}{\partial y}(\rho u v) = 0, \tag{50}$$

$$\frac{\partial}{\partial t}(\rho v) + \frac{\partial}{\partial x}(\rho u v) + \frac{\partial}{\partial y}(\rho v^2 + \frac{p}{\gamma M_0^2}) = 0, \tag{51}$$

$$\frac{\partial}{\partial t}\left(\frac{p}{\gamma M_0^2} + \beta\rho(u^2 + v^2)\right) + \frac{\partial}{\partial x}\left(\frac{pu}{M_0^2} + \beta\rho u(u^2 + v^2)\right) + \frac{\partial}{\partial y}\left(\frac{pv}{M_0^2} + \beta\rho v(u^2 + v^2)\right) = 0, \tag{52}$$

are implemented with a normalization in terms of far-field values, ρ_0 and p_0, and a velocity scale, u_0, which are incorporated into the Mach number parameter, M_0. An analytic solution exists for a constant-entropy (Lamb–Oseen) vortex with the tangential velocity profile

$$\frac{u_\theta}{u_0} = \frac{r_0}{r}\left(1 - e^{-\frac{r^2}{r_0^2}}\right), \tag{53}$$

where $r = \sqrt{x^2 + y^2}$, and r_0 is the core radius of the vortex. The corresponding pressure distribution is given by

$$\frac{p}{p_0} = \left(1 - (\gamma - 1)M_0^2\left[\frac{r_0^2}{2r^2}\left(1 - e^{-\frac{r^2}{r_0^2}}\right)^2 + \mathrm{Ei}\left(-2\frac{r^2}{r_0^2}\right) - \mathrm{Ei}\left(-\frac{r^2}{r_0^2}\right)\right]\right)^{\frac{\gamma}{\gamma-1}}, \tag{54}$$

where $\mathrm{Ei}(z)$ is the exponential integral function, and the density is related to pressure by $p/p_0 = (\rho/\rho_0)^\gamma$. Results are presented for $\gamma = 1.4, M_0 = 1.2$ and $r_0 = 4/25$.

The Lamb–Oseen vortex is preferred for this work over the more familiar Taylor vortex, despite having algebraic $(1/r)$ rather than exponential decay of the core strength, because it is stable to axisymmetric perturbation while the Taylor vortex is not. With the lack of numerical dissipation in these interface schemes, numerical error can provide sufficient perturbation to induce this instability. Because of this algebraic decay, in order to keep the domain to a side length of 4, exact boundary conditions are imposed using the analytic solution.

Both stationary and convecting vortex cases are considered, the latter having a convection velocity of (0.2,0.3) across the domain. In both cases the vortex is initially located at the center of the domain, as shown in Fig. 11(a). Fig. 11(b) shows the final location of the convecting vortex at time $t = 1$. Dashed lines indicate the boundaries of the refined region for the box refinement (a) and the cross + box refinement (b). These two grids, along with the cross refinement, are as described for the advection problem, but scaled to the larger domain of this problem.

Table 5 shows the results of the convergence study with the convecting vortex. Here, a CFL number of 1/10 was used for the cross grid and 1/12 for the cross + box grid. For the nonlinear problem, the same third-order average (L_2) convergence is seen, while uniform convergence is similar or marginally better than the linear problem, nearing third order in some cases. This confirms the observation that at least one order of convergence is lost from the interior scheme when corner interfaces are present.

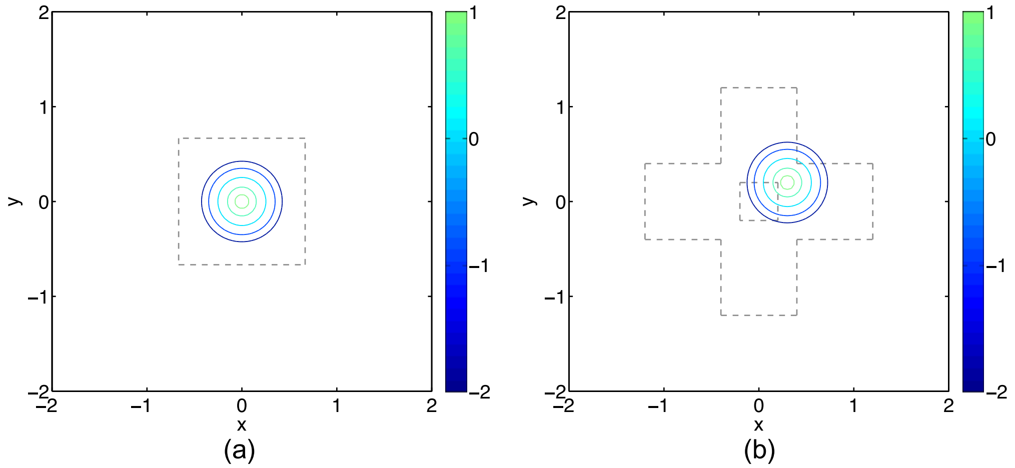


Fig. 11. Contours of vorticity for the convecting Lamb–Oseen vortex: (a) initial vortex position, (b) final vortex position at $t = 1$. Color scale is $\log_{10}(\omega)$. Grid interfaces are indicated by dashed lines, (a) showing box refinement interfaces, (b) showing cross + box refinement. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 5
Convergence results for the convecting vortex problem.

Coarse Δx	Cross refinement				Cross + box refinement			
	$\log_{10}(L_2)$	Rate	$\log_{10}(L_\infty)$	Rate	$\log_{10}(L_2)$	Rate	$\log_{10}(L_\infty)$	Rate
1/10	−1.858		−0.934		−1.750		−0.859	
1/15	−2.325	2.65	−1.320	2.19	−2.255	2.87	−1.332	2.69
1/20	−2.680	2.83	−1.678	2.87	−2.618	2.90	−1.678	2.77
1/30	−3.190	2.90	−2.158	2.72	−3.126	2.89	−2.125	2.54
1/40	−3.538	2.78	−2.404	1.97	−3.465	2.72	−2.398	2.18
Average rate		2.81		2.51		2.86		2.57

Table 6
Convergence results for the grid with a central box refinement region.

N	Advection problem				Stationary vortex problem			
	$\log_{10}(L_2)$	Rate	$\log_{10}(L_\infty)$	Rate	$\log_{10}(L_2)$	Rate	$\log_{10}(L_\infty)$	Rate
24	−1.766		−0.957		−1.757		−1.444	
30	−2.006	2.48	−1.102	1.49	−2.179	4.35	−1.895	4.65
48	−2.568	2.75	−1.446	1.69	−3.041	4.22	−2.661	3.75
60	−2.808	2.48	−1.595	1.53	−3.427	3.99	−2.910	2.57
120	−3.605	2.65	−2.226	2.10	−4.418	3.29	−3.403	1.64
Average rate		2.64		1.81		3.82		2.80

A better convergence result is obtained for the case of the vortex on the box refinement grid. As shown in Table 6, near-fourth-order convergence is seen, at least in the L_2-norm. For comparison, results for the advection problem on the same grid are shown alongside, where near-third-order L_2 convergence is seen as in Table 4. Promisingly for future adaptive use of the interface scheme, the vortex problem on the box refinement grid is the only case where the refinement conformed to the feature of interest (see the relative positions of the vortex core and the box in Fig. 11). The advection examples represent something of a worst case, where refinement occurs independently of the discretized field. In practice, near-fourth-order convergence is achievable with the present scheme when an adaptive refinement strategy is used, i.e., by moving the regions of refinement to always cover the steepest gradients of the solution. Doing so minimizes the truncation error at the interfaces and leads to convergence performance that should be similar to or better than that seen in the test problems. This is, in fact, the original idea behind AMR and it is the only reasonable manner in which to use the present schemes.

6. Conclusions

Nondissipative energy-stable interface schemes have been developed for the three distinct interior geometries encountered in a 2-D patch-refined grid. By construction, each satisfies the summation-by-parts condition, which was shown to guarantee stability of the semidiscrete form of the partial differential equation in the linear case. For a fourth-order explicit interior finite-difference scheme, a locally third-order edge interface closure was presented, while at the convex and concave corners, closure was achieved only by the presence of first and second order-accurate nodes in the stencils.

Implementation of the interface closure requires only the difference matrices $\widehat{D}_x$ and $\widehat{D}_y$, and these are available in the supplementary material for each geometry. Numerical experiments using these interface schemes on three different grids show that the low-order points reduce the convergence rate by approximately one order compared to the interior accuracy. Similar results are seen with a nonlinear vortex problem using the Euler equations. This decrease in convergence rate is the cost of achieving a stable scheme without artificial dissipation. The cost is much reduced, however, if the refinement conforms to the numerical solution, as would be the case in a practical implementation.

The ultimate goal of this effort is to develop a stable interface treatment for fully three-dimensional patch-refined grids, but extension of the 2-D results is not necessarily straightforward. The hanging node topology of the 3-D grid will include faces, edges and corners, each of which requires a separate solution. We speculate that further reductions in order of accuracy may result at three-dimensional edges and corners, though the actual order reduction is unknown at this point. The 3-D stencils are also likely to be large and computationally demanding to derive.

Acknowledgments

This work was supported by the ASC program of the Department of Energy under Subcontract No. B341492 of DOE Contract W-7405-ENG-48. The authors would like to thank the reviewers for valuable and constructive comments, in particular for requesting clarifications with respect to discrete conservation.

Appendix A. Alternative edge interface formulation

For the edge interface, an alternative formulation to that shown in Section 4.3 is available. Consider solving (1) now on a grid of infinite extent with an interface at $x = 0$, such that the left-half plane is refined by ratios r_x and r_y in the x- and y-directions, respectively, relative to the right-half plane. Thus $\Omega^f = \{(x,y) \in x < 0\}$, and $\Omega^c = \{(x,y) \in x > 0\}$, and a grid is obtained that contains only the edge interface geometry from Fig. 3. Note that each cell row of the grid is identical in this case, so nodes may be identified by a global index pair (i,j), where $i \in \mathbb{N}$ identifies the node position in the coarse row j, with the indexing chosen such that $(0,j)$ is the coarse node closest to the interface at $x = 0$. A convenient choice for the numbering within the fine part of each row is the "N"-counting scheme, shown in Fig. 12. With the familiar refinement ratios $r_x = r_y = 2$, the node locations $(x,y)_{i,j}$ are given by

$$(x,y)_{i,j} = \begin{cases} ([i/2]\Delta x, [2j + 1/2]\Delta y), & \text{if } i < 0 \text{ and odd,} \\ ([i/2 + 1/2]\Delta x, [2j - 1/2]\Delta y), & \text{if } i < 0 \text{ and even,} \\ ([2i + 1]\Delta x, 2j\Delta y), & i \geqslant 0, \end{cases} \tag{55}$$

where Δx and Δy are the refined grid discretizations.

This node-numbering scheme leads naturally to a formulation of the interface problem in terms of fourth-order tensors. Here, using the difference tensor $\mathbf{D}_{x,ijkl}$ as an example, the first pair of indices relate to the rows of the matrix D_x (the independent indices), and the second pair to the columns of D_x (the dependent indices). Within each index pair, the first index refers to the node number (i) and the second to the row (j). This can be done only because each row of this grid is identical; for any refinement where $\partial\Omega^f$ has corners, there is implicitly a change in the form of the rows and this formulation cannot be used.

For the interface scheme from Section 4.3, for a given grid row j of the edge-interface grid, the dependence set extends only to rows $\{j-1, j, j+1\}$. Consequently, the matrices for the interface stencil may be written as

$$\widehat{\mathbf{D}}_{x,ijkl} = \widehat{D}^{-1}_{x,ik}\delta_{j-1,l} + \widehat{D}^{0}_{x,ik}\delta_{j,l} + \widehat{D}^{+1}_{x,ik}\delta_{j+1,l}, \tag{56}$$

$$\widehat{\mathbf{D}}_{y,ijkl} = \widehat{D}^{-1}_{y,ik}\delta_{j-1,l} + \widehat{D}^{0}_{y,ik}\delta_{j,l} + \widehat{D}^{+1}_{y,ik}\delta_{j+1,l}, \tag{57}$$

$$\widehat{\mathbf{H}}_{ijkl} = \widehat{H}^{0}_{ik}\delta_{j,l}, \tag{58}$$

where $\delta_{j,l}$ is the Kronecker delta, and each of $\widehat{D}^{-1}_x$, $\widehat{D}^{0}_x$, $\widehat{D}^{+1}_x$, $\widehat{D}^{-1}_y$, $\widehat{D}^{0}_y$, $\widehat{D}^{+1}_y$ and $\widehat{H}^0$ are second-order matrices of finite extent. In the case of the stencil from Fig. 6, all are 8×8 matrices, except $\widehat{D}^{0}_{x}$, which is 8×14. The accuracy conditions (41) and (42) are constructed as before, in this case over the full matrices $\widehat{\mathbf{D}}_x$ and $\widehat{\mathbf{D}}_y$, and the stability conditions (35) and (36) expand to

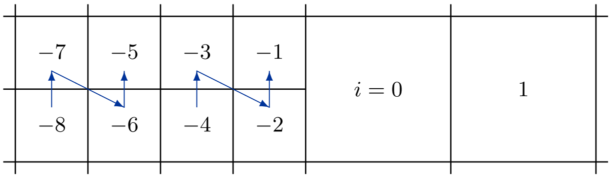


Fig. 12. The "N"-counting scheme for refined regions of an infinite grid with an interface at $x = 0$, for a row j.

$$\widehat{H}^0_{im}\widehat{D}^{-1}_{x,mk} + \widehat{H}^0_{mk}\widehat{D}^{+1}_{x,mi} = 0, \tag{59}$$
$$\widehat{H}^0_{im}\widehat{D}^{0}_{x,mk} + \widehat{H}^0_{mk}\widehat{D}^{0}_{x,mi} = 0, \tag{60}$$
$$\widehat{H}^0_{im}\widehat{D}^{+1}_{x,mk} + \widehat{H}^0_{mk}\widehat{D}^{-1}_{x,mi} = 0, \tag{61}$$

in x, and similarly in y. The resulting edge solution is identical to that of Section 4.3.

Appendix B. Supplementary material

Supplementary data associated with this article can be found, in the online version, at doi:10.1016/j.jcp.2009.04.010.

References

[1] S. Li, L. Petzgold, Y. Ren, Stability of moving mesh systems of partial differential equations, SIAM J. Sci. Comput. 20 (2) (1998) 719–738.
[2] L. Jameson, AMR vs high order schemes, J. Sci. Comp. 18 (1) (2003) 1–24.
[3] M.J. Berger, P. Colella, Local adaptive mesh refinement for shock hydrodynamics, J. Comput. Phys. 82 (1) (1989) 64–84.
[4] D.-I. Choi, J.D. Brown, B. Imbiriba, J. Centrella, P. MacNeice, Interface conditions for wave propagation through mesh refinement boundaries, J. Comput. Phys. 193 (2) (2004) 398–425.
[5] C. Pantano, R. Deiterding, D.J. Hill, D.I. Pullin, A low numerical dissipation patch-based adaptive mesh refinement method for large-eddy simulation of compressible flows, J. Comput. Phys. 221 (1) (2007) 63–87.
[6] R.M.J. Kramer, C. Pantano, D.I. Pullin, A class of energy stable high-order finite-difference interface schemes suitable for adaptive mesh refinement of hyperbolic problems, J. Comput. Phys. 226 (2007) 1458–1484.
[7] P. Lötstedt, S. Söderberg, A. Ramage, L. Hemmingsson-Frändén, Implicit solution of hyperbolic equations with space-time adaptivity, BIT 42 (1) (2002) 134–158.
[8] L. Ferm, P. Lötstedt, Accurate and stable grid interfaces for finite volume methods, Appl. Numer. Math. 49 (2004).
[9] L.N. Trefethen, Stability of finite-difference models containing two boundaries or interfaces, Math. Comput. 45 (172) (1985) 279–300.
[10] H.-O. Kreiss, G. Scherer, Finite element and finite difference methods for hyperbolic partial differential equations, in: Mathematical Aspects of Finite Elements in Partial Differential Equations, Academic Press, New York, 1974.
[11] M.H. Carpenter, D. Gottlieb, S. Abarbanel, Time-stable boundary conditions for finite difference schemes solving hyperbolic systems: methodology and application to high-order compact schemes, J. Comput. Phys. 111 (2) (1994) 220–236.
[12] S.S. Abarbanel, A.E. Chertock, Strict stability of high-order compact implicit finite-difference schemes: the role of boundary conditions for hyperbolic PDEs I, J. Comput. Phys. 160 (2000) 42–66.
[13] B. Gustafsson, H.-O. Kreiss, A. Sundström, Stability theory of difference approximations for mixed initial boundary value problems II, Math. Comput. 26 (119) (1972) 649–686.
[14] M.H. Carpenter, D. Gottlieb, S. Abarbanel, The stability of numerical boundary treatments for compact high-order finite-difference schemes, J. Comput. Phys. 108 (1993) 272–295.
[15] J. Nordström, M.H. Carpenter, Boundary and interface conditions for high-order finite-difference methods applied to the Euler and Navier–Stokes equations, J. Comput. Phys. 148 (1999) 621–645.
[16] J. Nordström, R. Gustafsson, High order finite difference approximations of electromagnetic wave propagation close to material discontinuities, J. Sci. Comp. 18 (2) (2003) 215–234.
[17] B. Strand, Summation by parts for finite difference approximations for d/dx, J. Comput. Phys. 110 (1) (1994) 47–67.
[18] M. Gerritsen, P. Olsson, Designing and efficient solution strategy for fluid flows: II. Stable high-order central finite difference schemes on composite adaptive grids with sharp shock resolution, J. Comput. Phys. 147 (2) (1998) 293–317.
[19] K. Sebastian, C.-W. Shu, Multidomain WENO finite difference method with interpolation at subdomain interfaces, J. Sci. Comp. 19 (1) (2003) 405–438.
[20] B. Gustafsson, The convergence rate for difference approximations to mixed initial boundary value problems, Math. Comput. 29 (130) (1975) 396–406.
[21] M.H. Carpenter, J. Nordström, D. Gottlieb, A stable and conservative interface treatment of arbitrary spatial accuracy, J. Comput. Phys. 148 (1999) 341–365.
[22] J.C. Butcher, Numerical Methods for Ordinary Differential Equations, J. Wiley, Hoboken, NJ, 2003. pp. 86–88.