Journal of Computational Physics 230 (2011) 2952–2976

Contents lists available at ScienceDirect

Journal of Computational Physics

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

High-order, finite-volume methods in mapped coordinates

P. Colella [a,1], M.R. Dorr [b,2], J.A.F. Hittinger [b,*,2], D.F. Martin [a,1]

[a] Applied Numerical Algorithms Group, Lawrence Berkeley National Laboratory, One Cyclotron Road Mail Stop 50A-1148, Berkeley, CA 94720, United States
[b] Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, 7000 East Avenue L-561, Livermore, CA 94550, United States

ARTICLE INFO

Article history:
Received 9 February 2010
Received in revised form 23 December 2010
Accepted 25 December 2010
Available online 14 January 2011

Keywords:
Finite-volume method
High-order discretization
Mapped grids
Hyperbolic and elliptic partial differential equations

ABSTRACT

We present an approach for constructing finite-volume methods for flux-divergence forms to any order of accuracy defined as the image of a smooth mapping from a rectangular discretization of an abstract coordinate space. Our approach is based on two ideas. The first is that of using higher-order quadrature rules to compute the flux averages over faces that generalize a method developed for Cartesian grids to the case of mapped grids. The second is a method for computing the averages of the metric terms on faces such that freestream preservation is automatically satisfied. We derive detailed formulas for the cases of fourth-order accurate discretizations of linear elliptic and hyperbolic partial differential equations. For the latter case, we combine the method so derived with Runge–Kutta time discretization and demonstrate how to incorporate a high-order accurate limiter with the goal of obtaining a method that is robust in the presence of discontinuities and underresolved gradients. For both elliptic and hyperbolic problems, we demonstrate that the resulting methods are fourth-order accurate for smooth solutions.

1. Introduction

Finite-volume methods are a popular choice for the discretization of partial differential equations involving flux divergences, *e.g.*, conservation laws. In such approaches, the spatial domain is decomposed into a set of control volumes. The boundary of each volume is represented as a union of faces, with each face shared by exactly two control volumes. Based on this discretization of space, the average of the divergence of the flux function over each control volume is approximated by applying the divergence theorem to express the average in terms of averages of fluxes over the faces, which are then computed using some quadrature rule. The main advantage is that the resulting discretization satisfies a discrete form of the divergence theorem. This leads to a local conservation property holding for time-dependent problems and easily-checked solvability conditions for steady-state problems. Furthermore, this approach extends to a wide variety of grid systems: Cartesian, mapped, multiblock, and locally-refined structured grids, as well as unstructured grids. A limitation of these methods as developed to date is that they have typically been restricted to second-order accuracy [1–3]. The flux integrals are approximated using the midpoint rule, and the metric terms appearing in the quadrature computed using low-order geometric representations (*e.g.*, unions of triangles).

* Corresponding author.
E-mail addresses: pcolella@lbl.gov (P. Colella), dorr1@llnl.gov (M.R. Dorr), hittinger1@llnl.gov (J.A.F. Hittinger), dfmartin@llnl.gov (D.F. Martin).

[1] Research supported by the Office of Advanced Scientific Computing Research of the US Department of Energy under Contract Number DE-AC02-05CH11231.
[2] This work performed under the auspices of the US Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. LLNL-JRNL-422807.

0021-9991/$ - see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2010.12.044

In this paper, we present an approach for constructing finite-volume methods of any order of accuracy for control-volume discretizations of space defined as the image of a smooth mapping from a rectangular discretization of an abstract coordinate space. Our approach is based on two ideas. The first is that of using higher-order quadrature rules to compute the flux averages over faces that generalize the method described in [4] to the case of mapped grids. The second is a method for computing the averages of the metric terms on faces such that freestream preservation is automatically satisfied.

Freestream preservation is an important requirement for the discretization of conservation laws in mapped coordinates. This property ensures that a uniform flow is unaffected by the choice of mapping and discretization. As described in numerous works (*e.g.*, [1,5,6]), this goal is typically accomplished by the discrete enforcement of metric identities, which take the form of divergence-free conditions for products of the mapping Jacobian and gradients. Since cell faces are contractible (continuously deformable to a point), the Poincaré lemma guarantees that these products can be written as exterior derivatives. The form of these derivatives is not unique, however (see, *e.g.*, Section 4 of [6]), and the specific choice used for discretization is critical in achieving freestream preservation. In [5], it is observed that writing the derivatives in "conservative form" is sufficient to enable second-order central differencing to be applied in the exact enforcement of the metric identities. This result was more recently extended in [7] to higher-order (second-, fourth- and sixth-order) compact, finite difference operators. The equivalence of the central difference scheme used in [5] with a second-order, finite-volume method was also used to obtain an early result for this class of methods. For the higher-order finite-volume discretizations presented here, we describe how to take further advantage of the ability to express mapping metric products as exterior derivatives to achieve freestream preservation.

The paper is organized as follows. A formalism for computing a fourth-order accurate average of a flux divergence on a control volume in physical space in terms of fourth-order accurate face averages on a Cartesian computational grid is developed in Section 2. In Section 3, we describe the application of the mapped grid finite-volume formalism to obtain a fourth-order accurate discretization of a self-adjoint elliptic equation. In Section 4, we describe the application of the formalism to obtain a fourth-order accurate discretization of a scalar, linear hyperbolic equation. The spatial discretization is combined with Runge–Kutta time discretization, and we demonstrate the incorporation of a high-order accurate limiter.

2. High-order finite-volume methods

In the finite-volume approach, the spatial domain in $\mathbb{R}^D$ is discretized as a union of rectangular control volumes that covers the spatial domain. For Cartesian grid finite-volume methods, a control volume $V_{\mathbf{i}}$ takes the form

$$V_{\mathbf{i}} = \left[x_{i_1} - \frac{h}{2}, x_{i_1} + \frac{h}{2}\right] \times \left[x_{i_2} - \frac{h}{2}, x_{i_2} + \frac{h}{2}\right] \times \cdots \times \left[x_{i_D} - \frac{h}{2}, x_{i_D} + \frac{h}{2}\right], \tag{1}$$

where the multi-index $\mathbf{i} \equiv (i_1, i_2, \ldots, i_D) \in \mathbb{Z}^D$ is identified with the location of the control volume center and h is the grid spacing. A finite-volume method discretizes a partial differential equation by averaging that equation over control volumes and replacing the integrals that appear by quadratures. For operators that appear as the divergence of fluxes, the divergence theorem states that

$$\int_{V_{\mathbf{i}}} \nabla \cdot \mathbf{F}\, d\mathbf{x} = \sum_{\pm=+,-} \sum_{d=1}^{D} \pm \int_{A_d^{\pm}} F_d\, d\mathbf{A}, \tag{2}$$

where F_d is the dth component of $\mathbf{F}$ and the $A_d^{\pm}$ are the high and low faces bounding $V_{\mathbf{i}}$ with normals pointing the dth coordinate direction. In this case, the finite-volume approach computes the average of the divergence of the fluxes on the left-hand side of (2) with the sum of the integrals on the right-hand side, with the integrals approximated using quadratures. Such approximations are desirable because they lead to conserved quantities in the original PDE satisfying an analogous conservation law in the discretized system.

Most finite-volume methods use the midpoint rule to approximate the flux integrals in (2), leading to a second-order accurate method. We will develop higher-order methods (fourth-order or better) using the approach in [4]. The starting point for this approach is to replace the integrand in the right-hand side of (2) by a Taylor expansion about the center of the face:

$$\begin{aligned} &\int_{A_d} F_d\, d\mathbf{A} = \sum_{0 \leqslant |\mathbf{r}| < R} \frac{1}{\mathbf{r}!} \nabla^{\mathbf{r}} F_d|_{x=x_0} \int_{A_d} (\mathbf{x} - \mathbf{x}_0)^{\mathbf{r}} dA_x + O(h^{R+D-1}), \\ &\mathbf{r}! = r_1! \cdots r_D!, \quad \mathbf{q}^{\mathbf{r}} = q_1^{r_1} \cdots q_D^{r_D}. \end{aligned} \tag{3}$$

For example, if we take $R = 4$, we obtain

$$\frac{1}{h^{D-1}} \int_{A_d} F_d\, d\mathbf{A} = F_d(\mathbf{x}_0) + \frac{h^2}{24} \sum_{d' \neq d} \frac{\partial^2 F_d}{\partial x_{d'}^2}(\mathbf{x}_0) + O(h^4). \tag{4}$$

If we replace the derivatives by finite-difference approximations of a suitable order that are smooth functions of their inputs, the resulting approximation of the average of the flux divergence over a cell is $O(h^R)$.

2.1. Mapped grids

We can extend this formalism to the case of mapped grids. Assume that we have a smooth mapping $\mathbf{X}$ from some abstract coordinate space, say, the unit cube, into physical space:

$$\mathbf{X} = \mathbf{X}(\boldsymbol{\xi}), \quad \mathbf{X} : [0,1]^D \to \mathbb{R}^D. \tag{5}$$

Given this mapping, the divergence of a vector field in physical space can be written in terms of derivatives in the mapping space, that is,

$$\begin{aligned} &\nabla_{\mathbf{x}} \cdot \mathbf{F} = \frac{1}{J} \nabla_{\xi} \cdot (\mathbf{N}^T \mathbf{F}), \\ &J = \det(\nabla_{\xi} \mathbf{X}), \quad (\mathbf{N}^T)_{p,q} = \det\left(\mathbf{R}_p(\nabla_{\xi}\mathbf{X}, \mathbf{e}^q)\right), \end{aligned} \tag{6}$$

where $\mathbf{R}_p(\mathbf{A}, \mathbf{v})$ denotes the matrix obtained by replacing the pth row of the matrix $\mathbf{A}$ by the vector $\mathbf{v}$, and $\mathbf{e}^d$ denotes the unit vector in the dth coordinate direction. The relationship (6) is a consequence of the chain rule, Cramer's rule and (for $D > 2$) the equality of mixed partial derivatives.

If we define control volumes in physical space as the images $\mathbf{X}(V_{\mathbf{i}})$ of the cubic control volumes $V_{\mathbf{i}}$ in the mapped Cartesian grid space, the relationship corresponding to (2) for mapped grids is given as follows:

$$\int_{\mathbf{X}(V_{\mathbf{i}})} \nabla_{\mathbf{x}} \cdot \mathbf{F}\, d\mathbf{x} = \int_{V_{\mathbf{i}}} \nabla_{\xi} \cdot (\mathbf{N}^T \mathbf{F}) d\boldsymbol{\xi} = \sum_{\pm=+,-} \sum_{d=1}^{D} \pm \int_{A_d^{\pm}} (\mathbf{N}^T \mathbf{F})_d d\mathbf{A}_{\xi}. \tag{7}$$

To obtain a finite-volume method, the face integrals are replaced by quadratures, similar to what was done in (3). In the mapped-grid case, some care is required to obtain freestream preservation, that is, the property that the discrete divergence of a constant vector field is zero. To do that, we split each face integral into two pieces:

$$\int_{A_d} (\mathbf{N}^T \mathbf{F})_d dA_{\xi} = \left(\left(\int_{A_d} \mathbf{N}^T d\mathbf{A}_{\xi} \right) \mathbf{F}(\mathbf{x}_d) \right)_d + \int_{A_d} (\mathbf{N}^T (\mathbf{F} - \mathbf{F}(\mathbf{x}_d)))_d d\mathbf{A}_{\xi}, \tag{8}$$

where $\mathbf{x}_d$ is the image under the map of the center of the face in coordinate space. It is routine to derive a version of the Taylor expansion in (3) to approximate the second integrand on the right-hand side of (8) so that, if $\mathbf{F}$ is constant, the integral is identically zero. To obtain a fourth-order accurate discretization, we can use the following formulation:

$$\int_{A_d} (\mathbf{N}^T \mathbf{F})_d d\mathbf{A}_{\xi} = \left(\left(\int_{A_d} \mathbf{N}^T dA_{\xi} \right) \cdot \left(\int_{A_d} \mathbf{F}\, d\mathbf{A}_{\xi} \right) \right)_d + \frac{h^2}{12} \int_{A_d} \sum_{d' \neq d} \left(\frac{\partial}{\partial \xi_{d'}} (\mathbf{N}^T) \cdot \frac{\partial}{\partial \xi_{d'}} (\mathbf{F}) \right)_d d\mathbf{A}_{\xi} + O(h^4), \tag{9}$$

as $h \to 0$. In either case, we only need to derive quadrature formulas for $\int_{A_d} \mathbf{N}^T d\mathbf{A}_{\xi}$ so that the discrete divergence of a constant vector field given by (8) or (9) is zero.

The existence of such quadratures is a consequence of Stokes' theorem and the Poincaré lemma. The rows of the matrix $\mathbf{N}$, denoted by N^s, $s = 1, \ldots, D$, are divergence-free. This can be seen by a direct calculation, or inferred indirectly from applying (7) to constant vector fields. Then by the Poincaré lemma [8], there exist functions $\mathcal{N}^s_{d,d'}, d \neq d'$ such that

$$N^s_d = \sum_{d' \neq d} \frac{\partial \mathcal{N}^s_{d,d'}}{\partial \xi_{d'}}, \quad \mathcal{N}^s_{d,d'} = -\mathcal{N}^s_{d',d}. \tag{10}$$

Thus we have

$$\int_{A_d} N^s_d\, d\mathbf{A}_{\xi} = \sum_{\pm=+,-} \sum_{d' \neq d} \pm \int_{E^{\pm}_{d,d'}} \mathcal{N}^s_{d,d'}\, d\mathbf{E}_{\xi}, \tag{11}$$

where $E^{\pm}_{d,d'}$ are the (hyper) edges on the low and high sides of A_d in the d' direction. For each edge, the same integrals over the edge appear for the integral over each face adjacent to that edge, modulo signs. If we approximate the integrals over edges with the same quadrature formulas wherever they appear, then the freestream property

$$\sum_{d=1}^{D} \sum_{\pm=+,-} \pm \int_{A^{\pm}_d} N^s_d\, d\mathbf{A}_{\xi} = 0 \tag{12}$$

is satisfied. Furthermore, the quadrature formulas for the edge integrals can otherwise be chosen arbitrarily; in particular, they can be chosen so that (11) approximates the integral of N^s_d over the face to any order of accuracy. We note that this is a generalization to arbitrary dimensions and arbitrary orders of accuracy of standard methods to discretize electromagnetic fields so that discrete analogues of the various vector identities are satisfied identically [9].

Given $N^s_d, d = 1, \ldots, D$, the family of functions $\mathcal{N}^s_{d,d'}$, $d' \neq d$, satisfying (10) is not unique. A particularly simple choice that is a local function of $\mathbf{X}$ and $\nabla_{\xi}\mathbf{X}$ is given by

$$\mathcal{N}^s_{d,d'} = \frac{1}{D-1} \det\left(\mathbf{R}_s(\mathbf{C}_{d'}(\nabla_{\xi}\mathbf{X}, \mathbf{X}), \mathbf{e}^d)\right), \tag{13}$$

where $\mathbf{C}_p(\mathbf{A},\mathbf{v})$ denotes the matrix obtained by replacing the pth column of the matrix $\mathbf{A}$ with $\mathbf{v}$. We note that the expression for $\mathcal{N}^s_{d,d'}$ given above only involves derivatives of $\mathbf{X}$ in directions tangent to $E_{d,d'}$. For example, the $\mathcal{N}^s_{d,d'}$ for the special case of $D = 3$ are given as follows:

$$\begin{aligned}
\mathcal{N}^1_{21} &= \frac{1}{2}\left(X_3\frac{\partial X_2}{\partial \xi_3} - X_2\frac{\partial X_3}{\partial \xi_3}\right), \quad \mathcal{N}^2_{21} = \frac{1}{2}\left(-X_3\frac{\partial X_1}{\partial \xi_3} + X_1\frac{\partial X_3}{\partial \xi_3}\right)\\
\mathcal{N}^3_{21} &= \frac{1}{2}\left(X_2\frac{\partial X_1}{\partial \xi_3} - X_1\frac{\partial X_2}{\partial \xi_3}\right), \quad \mathcal{N}^1_{31} = \frac{1}{2}\left(-X_3\frac{\partial X_2}{\partial \xi_2} + X_2\frac{\partial X_3}{\partial \xi_2}\right)\\
\mathcal{N}^2_{31} &= \frac{1}{2}\left(X_3\frac{\partial X_1}{\partial \xi_2} - X_1\frac{\partial X_3}{\partial \xi_2}\right), \quad \mathcal{N}^3_{31} = \frac{1}{2}\left(-X_2\frac{\partial X_1}{\partial \xi_2} + X_1\frac{\partial X_2}{\partial \xi_2}\right)\\
\mathcal{N}^1_{32} &= \frac{1}{2}\left(X_3\frac{\partial X_2}{\partial \xi_1} - X_2\frac{\partial X_3}{\partial \xi_1}\right), \quad \mathcal{N}^2_{32} = \frac{1}{2}\left(-X_3\frac{\partial X_1}{\partial \xi_1} + X_1\frac{\partial X_3}{\partial \xi_1}\right)\\
\mathcal{N}^3_{32} &= \frac{1}{2}\left(X_2\frac{\partial X_1}{\partial \xi_1} - X_1\frac{\partial X_2}{\partial \xi_1}\right).
\end{aligned} \tag{14}$$

The remaining $\mathcal{N}$'s are given by the antisymmetry condition $\mathcal{N}^s_{d,d'} = -\mathcal{N}^s_{d',d}$.

The proof that (13) satisfies (10) is a straightforward calculation. By Leibnitz' rule applied to determinants, we have

$$\begin{aligned}
\sum_{d'\neq d}\frac{\partial}{\partial \xi_{d'}}\left(\det\left(\mathbf{R}_s\left(\mathbf{C}_{d'}(\nabla_\xi\mathbf{X},\mathbf{X}),\mathbf{e}^d\right)\right)\right) &= \sum_{d'\neq d}\det\left(\mathbf{R}_s\left(\mathbf{C}_{d'}\left(\nabla_\xi\mathbf{X},\frac{\partial\mathbf{X}}{\partial\xi_{d'}}\right),\mathbf{e}^d\right)\right)\\
&\quad + \sum_{d'\neq d}\sum_{d''\neq d,d'}\det\left(\mathbf{R}_s\left(\mathbf{C}_{d''}\left(\mathbf{C}_{d'}(\nabla_\xi\mathbf{X},\mathbf{X}),\frac{\partial^2\mathbf{X}}{\partial\xi_{d'}\partial\xi_{d''}}\right),\mathbf{e}^d\right)\right).
\end{aligned} \tag{15}$$

Each summand in the first (single) sum is just N^s_d, so it suffices to show that the second (double) sum vanishes. However, for a given $d_1, d_2, d_1 \neq d_2$, summands in the double sum involving the mixed second partial $\frac{\partial^2\mathbf{X}}{\partial\xi_{d_1}\partial\xi_{d_2}}$ appear exactly twice, differing from one another only by the exchange of the d_1 and d_2 columns. By the antisymmetry of the determinant under column exchanges, the two summands cancel, and hence the entire second sum vanishes. Finally, we need to show the antisymmetry condition $\mathcal{N}^s_{d,d'} = -\mathcal{N}^s_{d',d}$. The following is a consequence of linearity of the determinant as a function of the d' column, plus the identity $\det(\mathbf{C}_q(\mathbf{A},\mathbf{e}^p)) = \det(\mathbf{R}^q(\mathbf{A},\mathbf{e}_p))$:

$$\det\left(\mathbf{R}_s\left(\mathbf{C}_{d'}(\nabla_\xi\mathbf{X},\mathbf{X}),\mathbf{e}^d\right)\right) = \sum_{s'\neq s}X_{s'}\det\left(\mathbf{R}_{s'}\left(\mathbf{R}_s(\nabla_\xi\mathbf{X},\mathbf{e}^d),\mathbf{e}^{d'}\right)\right). \tag{16}$$

The right-hand side of (16) is manifestly antisymmetric in d, d'.

2.2. Fourth-order mapped-grid finite-volume discretization

Following these ideas, we can specify the information required for a fourth-order accurate finite-volume discretization. Using a Taylor series, the integrals on the cell faces $A^{\pm}_d$ can be approximated using the following formula for the average of a product in terms of fourth-order accurate averages of each factor:

$$\langle fg\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \langle f\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}\langle g\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + \frac{h^2}{12}\mathbf{G}^{\perp,d}(\langle f\rangle)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}\cdot\mathbf{G}^{\perp,d}(\langle g\rangle)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + O(h^4). \tag{17}$$

Here, the operator $\langle\cdot\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$ denotes a fourth-order accurate average over the face centered at $\mathbf{i}+\frac{1}{2}\mathbf{e}^d$:

$$\langle q\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \frac{1}{h^{D-1}}\int_{A_d} q(\xi)d\mathbf{A}_\xi + O(h^4), \tag{18}$$

and $\mathbf{G}^{\perp,d} \approx \nabla_\xi - \mathbf{e}^d\frac{\partial}{\partial\xi_d}$ is a second-order accurate difference approximation to the components of the gradient operator orthogonal to the dth direction. Alternative expressions to (17) are obtained by replacing the averages $\langle f\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$ and/or $\langle g\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$ used in the transverse gradients $\mathbf{G}^{\perp,d}$ by corresponding fourth-order-accurate, face-centered pointwise values $f_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$ and/or $g_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$, respectively.

More generally, the transverse gradients can be replaced by any discretization that is second-order-accurate at the face center $\mathbf{i}+\frac{1}{2}\mathbf{e}^d$. For example,

$$\langle fg\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \langle f\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}\langle g\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + \frac{h^2}{12}\mathbf{G}^{\perp,d}(\langle f\rangle)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}\cdot\left(\nabla^{\perp,d}g\right)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + O(h^4), \tag{19}$$

where $(\nabla^{\perp,d}g)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$ is a second-order approximation of $(\nabla_\xi - \mathbf{e}^d\frac{\partial}{\partial\xi_d})g$ at $\mathbf{i}+\frac{1}{2}\mathbf{e}^d$. As employed in Section 3, this fact can be used to reduce the stencil size of the discretization that results from the application of (17) with factors g containing derivatives.

We thus approximate the divergence of a flux by

$$\int_{\mathbf{X}(V_\mathbf{i})} \nabla_\mathbf{x} \cdot \mathbf{F}\, d\mathbf{x} \approx h^{D-1} \sum_{d=1}^{D} \sum_{\pm=+,-} \pm F^d_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d}, \tag{20}$$

where

$$F^d_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \sum_{s=1}^{D} \langle N^s_d \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \langle F^s \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + \frac{h^2}{12} \sum_{s=1}^{D} \left(\mathbf{G}^{\perp,d}(\langle N^s_d \rangle)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \right) \cdot \left(\mathbf{G}^{\perp,d}(\langle F^s \rangle)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \right). \tag{21}$$

The column vectors $\{\langle N^s_d \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}, s = 1, \ldots, D\}$ are computed on each face using (11) and (13), with fourth-order accurate quadratures replacing the integrals in (11). The fourth-order average of $\mathbf{F}$ can be computed using (4).

We can apply this approach to compute a fourth-order accurate approximation to the cell volumes by taking $\mathbf{F}(\mathbf{x}) = \mathbf{x}$. In that case,

$$\int_{\mathbf{X}(V_\mathbf{i})} \nabla_\mathbf{x} \cdot \mathbf{F}\, d\mathbf{x} = D \times \text{Volume}(\mathbf{X}(V_\mathbf{i})),$$

and the volume of the cell can be written as the discrete divergence of fluxes. Such a flux form is convenient for maintaining conservation and freestream preservation for adaptive mesh refinement on mapped grids [10].

3. Application to elliptic equations

In this section, we apply the mapped grid, finite-volume formalism described above to obtain a fourth-order accurate finite-volume discretization of a self-adjoint equation

$$\nabla \cdot \mathbf{F}(\mathbf{x}) = \rho(\mathbf{x}), \quad \mathbf{x} \in \Omega \subset \mathbb{R}^2, \tag{22}$$

where

$$\mathbf{F}(\mathbf{x}) \equiv \mathbf{B}(\mathbf{x}) \nabla \Phi(\mathbf{x}), \tag{23}$$

and the matrix coefficient $\mathbf{B}$ is such that the second-order differential operator in (22) and (23) is elliptic. Assuming a mapping (5) of the physical domain Ω to a computational domain, we have in the latter using (6)

$$\mathbf{F} = \mathbf{B} \nabla_\mathbf{X} \xi \nabla_\xi \Phi \equiv J^{-1} \mathbf{B} \mathbf{N} \nabla_\xi \Phi. \tag{24}$$

Following (20) and (21), we therefore obtain

$$\int_{\mathbf{X}(V_\mathbf{i})} \nabla_\mathbf{x} \cdot \mathbf{F}\, d\mathbf{x} = h^2 \sum_{d=1}^{3} \sum_{\pm=+,-} \pm F^d_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d} + O(h^4), \tag{25}$$

where

$$F^d_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \equiv \langle \mathbf{F}_d \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \sum_{d'=1}^{3} \left\langle \tilde{b}_{dd'} \frac{\partial \Phi}{\partial \xi_{d'}} \right\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \sum_{d'=1}^{3} \left[\langle \tilde{b}_{dd'} \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \left\langle \frac{\partial \Phi}{\partial \xi_{d'}} \right\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + \frac{h^2}{12} \mathbf{G}^{\perp,d}\left(\langle \tilde{b}_{dd'} \rangle \right)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \cdot \left(\nabla^{\perp,d} \frac{\partial \Phi}{\partial \xi_{d'}} \right)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \right] \tag{26}$$

and

$$\tilde{b}_{dd'} \equiv (\tilde{\mathbf{B}})_{dd'} \equiv \left(J^{-1} \mathbf{N}^T \mathbf{B} \mathbf{N} \right)_{dd'}. \tag{27}$$

Face averages $\langle \tilde{b}_{dd'} \rangle$ can be computed to fourth order in terms of face averages of J^{-1} and the entries of the factor matrices $\mathbf{N}^T$, $\mathbf{B}$ and $\mathbf{N}$ using the product formula (17). We compute second-order accurate transverse gradients of the coefficients for $d' = 1,2,\ldots,D (d' \neq d)$ using

$$\left(\mathbf{G}^{\perp,d}\left(\langle \tilde{b}_{dd'} \rangle \right)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \right)_{d'} = \begin{cases} -\frac{1}{2h}\left(3\langle \tilde{b}_{dd'} \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} - 4\langle \tilde{b}_{dd'} \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+\mathbf{e}^{d'}} + 2\langle \tilde{b}_{dd'} \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+2\mathbf{e}^{d'}} \right), & i_{d'} - 1 < \min i_{d'} \\ \frac{1}{2h}\left(3\langle \tilde{b}_{dd'} \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} - 4\langle \tilde{b}_{dd'} \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-\mathbf{e}^{d'}} + 2\langle \tilde{b}_{dd'} \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-2\mathbf{e}^{d'}} \right), & i_{d'} + 1 < \max i_{d'} \\ \frac{1}{2h}\left(\langle \tilde{b}_{dd'} \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+\mathbf{e}^{d'}} - \langle \tilde{b}_{dd'} \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-\mathbf{e}^{d'}} \right), & \text{otherwise.} \end{cases} \tag{28}$$

The first two expressions correspond to second-order shifted stencils appropriate for the low and high domain boundaries in the d' direction, while the last is the standard second-order central difference approximation. What remains to specify is the discretization of the averages $\langle \partial\Phi/\partial\xi_{d'} \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$ and second-order approximations of the transverse gradients $(\nabla^{\perp,d} \frac{\partial \Phi}{\partial \xi_{d'}})_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$.

3.1. Discretization of $\langle \partial\Phi/\partial\xi_{d'}\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$ and $(\nabla^{\perp,d}\frac{\partial\Phi}{\partial\xi_{d'}})_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$

First consider the case where $d' = d$. We have

$$\left\langle \frac{\partial\Phi}{\partial\xi_d}\right\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \left(\frac{\partial\Phi}{\partial\xi_d} + \frac{h^2}{24}\Delta^{\perp,d}\frac{\partial\Phi}{\partial\xi_d}\right)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + O(h^4), \tag{29}$$

where $\Delta^{\perp,d}$ is the Laplacian in the directions transverse to the dth direction. Defining

$$\beta_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \equiv \frac{1}{24}\left[27(\Phi_{\mathbf{i}+\mathbf{e}^d} - \Phi_{\mathbf{i}}) - (\Phi_{\mathbf{i}+2\mathbf{e}^d} - \Phi_{\mathbf{i}-\mathbf{e}^d})\right], \tag{30}$$

where the $\Phi_{\mathbf{i}}$ denote pointwise values of Φ at cell centers, we have

$$\left.\frac{\partial\Phi}{\partial\xi_d}\right|_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \frac{\beta_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}}{h} + O(h^4), \tag{31}$$

$$\left.\Delta^{\perp,d}\frac{\partial\Phi}{\partial\xi_d}\right|_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \frac{1}{h^3}\sum_{d'\neq d}\left(\beta_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+\mathbf{e}^{d'}} + \beta_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-\mathbf{e}^{d'}} - 2\beta_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}\right) + O(h^2). \tag{32}$$

Furthermore, we make the approximation

$$\left(\nabla^{\perp,d}\frac{\partial\Phi}{\partial\xi_{d'}}\right)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \frac{1}{2h^2}\left(\beta_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+\mathbf{e}^{d'}} - \beta_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-\mathbf{e}^{d'}}\right) + O(h^2). \tag{33}$$

In Fig. 1, the centerings of the three β coefficients contributing to the calculation of (29) are indicated (filled circles), as well as the centerings of the Φ values required for their computation (empty circles).

Next, for $d' \neq d$, we have

$$\left\langle \frac{\partial\Phi}{\partial\xi_{d'}}\right\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \left(\frac{\partial\Phi}{\partial\xi_{d'}} + \frac{h^2}{24}\frac{\partial^3\Phi}{\partial\xi_{d'}^3}\right)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + O(h^4). \tag{34}$$

Letting

$$\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \equiv \frac{1}{16}\left[9(\Phi_{\mathbf{i}+\mathbf{e}^d} + \Phi_{\mathbf{i}}) - (\Phi_{\mathbf{i}+2\mathbf{e}^d} + \Phi_{\mathbf{i}-\mathbf{e}^d})\right], \tag{35}$$

$$\gamma^{d',(4)}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \equiv \frac{1}{12}\left[8(\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+\mathbf{e}^{d'}} - \alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-\mathbf{e}^{d'}}) - (\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+2\mathbf{e}^{d'}} - \alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-2\mathbf{e}^{d'}})\right], \tag{36}$$

$$\gamma^{d',(2)}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \equiv \frac{1}{2}\left[-2(\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+\mathbf{e}^{d'}} - \alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-\mathbf{e}^{d'}}) + (\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+2\mathbf{e}^{d'}} - \alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-2\mathbf{e}^{d'}})\right], \tag{37}$$

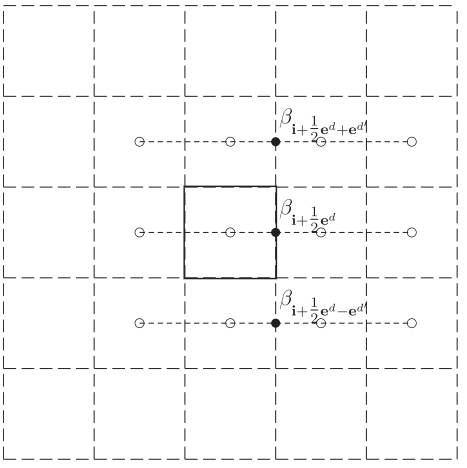


Fig. 1. Stencil for $\langle \partial\Phi/\partial\xi_d\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$. The values of Φ at the cell centers marked by the open circles determine the β values given by (30) at the cell faces marked by the solid circles.

we have

$$\left.\frac{\partial \Phi}{\partial \xi_{d'}}\right|_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \frac{\gamma^{d',(4)}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}}{h} + O(h^4), \tag{38}$$

$$\left.\frac{\partial^3 \Phi}{\partial \xi^3_{d'}}\right|_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \frac{\gamma^{d',(2)}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}}{h^3} + O(h^2). \tag{39}$$

Defining

$$\gamma^{d',(4),\pm}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \equiv \pm\frac{1}{12}\left[3\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d\pm 2\mathbf{e}^{d'}} + 10\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d\pm\mathbf{e}^{d'}} - 18\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + 6\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d\mp\mathbf{e}^{d'}} - \alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d\mp 2\mathbf{e}^{d'}}\right], \tag{40}$$

we furthermore approximate

$$\left(\nabla^{\perp,d}\frac{\partial \Phi}{\partial \xi_{d'}}\right)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \frac{1}{2h^2}\left(\gamma^{d',(4),+}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} - \gamma^{d',(4),-}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}\right) + O(h^2). \tag{41}$$

The stencil entries given by (40) yield fourth-order accurate first derivatives in the d' direction at the face centers $\mathbf{i}+\frac{1}{2}\mathbf{e}^d \pm \mathbf{e}^{d'}$. We employ these non-centered formulas to ensure that the resulting stencil is confined to a block of cells at most 5 cells wide centered on the cell in which the flux divergence average is being computed. In Fig. 2, the centerings of the α coefficients (filled circles) and γ coefficients ($\times$ symbols) contributing to the calculation of (34) are indicated, as well as the centerings of the Φ values required for their computation (empty circles).

3.2. Boundary conditions

For boundaries upon which a Dirichlet boundary condition is imposed, the face averages $\langle \partial\Phi/\partial\xi_{d'}\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$ used in (26) on faces contained in such boundaries can be computed using modified discretizations that incorporate the prescribed boundary values. Suppose that the cell face with center at $\mathbf{i}+\frac{1}{2}\mathbf{e}^d$ is one such face, such as the face centered on the point labeled A in Fig. 3. The averages $\langle \partial\Phi/\partial\xi_{d'}\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$ for the transverse coordinates $d' \neq d$ can presumably be computed directly from prescribed Dirichlet data to fourth-order accuracy. For the normal direction, the stencil describing $\langle \partial\Phi/\partial\xi_d\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$ can be modified by replacing the definition (30) by

$$\beta^{(4)}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \equiv \frac{1}{840}\left(2816\Phi_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} - 3675\Phi_{\mathbf{i}} + 1225\Phi_{\mathbf{i}-\mathbf{e}^d} - 441\Phi_{\mathbf{i}-2\mathbf{e}^d} + 75\Phi_{\mathbf{i}-3\mathbf{e}^d}\right), \tag{42}$$

where $\Phi_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$ is the prescribed boundary value at the center of the cell face. Although this formula yields a fourth order accurate approximation of the normal derivative, it results in a stencil extending beyond the 5-cell-wide block centered about the cell upon which the discrete divergence is being computed. To avoid using a larger stencil at the boundary than in the interior, we take advantage of the opportunity to reduce the discretization order at the boundary while still maintaining fourth-order accuracy overall due to elliptic regularity. In particular, instead of (42) we define

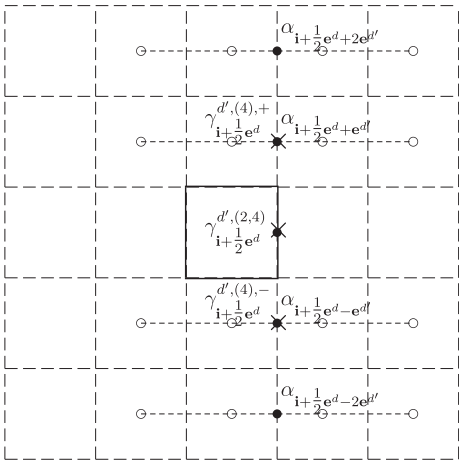


Fig. 2. Stencil for $\langle \partial\Phi/\partial\xi_{d'}\rangle_{\mathbf{i}}+\frac{1}{2}\mathbf{e}^d$ with $d' \neq d$. The values of Φ at the cell centers marked by the $\circ$ symbols determine the α values at the $\bullet$ symbols via (35), which in turn determine the γ values at the $\times$ symbols defined by (38)–(40).

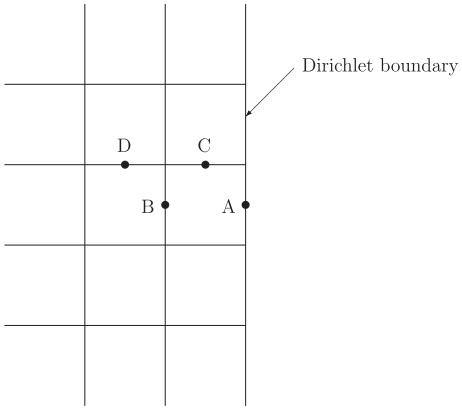


Fig. 3. Locations at or near a Dirichlet boundary requiring stencil modifications (see text).

$$\beta^{(4)}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \equiv \frac{1}{60}\left(184\Phi_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} - 225\Phi_{\mathbf{i}} + 50\Phi_{\mathbf{i}-\mathbf{e}^d} - 9\Phi_{\mathbf{i}-2\mathbf{e}^d}\right). \tag{43}$$

The same issue affects the normal and transverse derivatives on interior faces parallel to the boundary exactly one cell away, such as the face centered on the point labeled B in Fig. 3, *i.e.*, the use of a non-centered fourth-order discretization leads to a stencil that is not contained in a 5×5 block. However, we may use the same interpolating cubic polynomial used in finding (43) to obtain

$$\beta^{(4)}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \equiv \frac{1}{60}\left(-8\Phi_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + 75\Phi_{\mathbf{i}} - 70\Phi_{\mathbf{i}-\mathbf{e}^d} + 3\Phi_{\mathbf{i}-2\mathbf{e}^d}\right). \tag{44}$$

Similarly, (35) is replaced by

$$\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \equiv \frac{1}{20}\left(-4\Phi_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + 15\Phi_{\mathbf{i}} + 10\Phi_{\mathbf{i}-\mathbf{e}^d} - \Phi_{\mathbf{i}-2\mathbf{e}^d}\right). \tag{45}$$

Next, consider cell faces adjacent and normal to a Dirichlet boundary, such as the face centered on the point labeled C in Fig. 3. For the calculation of the average of the normal derivative ($d = d'$), the calculation of the transverse Laplacian (32) using centered differences can be shifted one cell away from the boundary with no loss of the required second-order accuracy. A non-centered, second-order accurate formula replaces (33):

$$\left(\nabla^{\perp,d}\frac{\partial\Phi}{\partial\xi_{d'}}\right)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \frac{1}{2h^2}\left(3\beta_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} - 4\beta_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-\mathbf{e}^{d'}} + \beta_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-2\mathbf{e}^{d'}}\right) + O(h^2). \tag{46}$$

For the average of the transverse derivative ($d \neq d'$), we replace (36) and (37) by

$$\gamma^{d',(4)}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \equiv \frac{1}{30}\left[32\Phi_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+\frac{1}{2}\mathbf{e}^{d'}} - 15\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} - 20\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-\mathbf{e}^{d'}} + 3\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-2\mathbf{e}^{d'}}\right], \tag{47}$$

$$\gamma^{d',(2)}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \equiv \frac{1}{5}\left[16\Phi_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+\frac{1}{2}\mathbf{e}^{d'}} - 30\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + 20\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-\mathbf{e}^{d'}} - 6\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-2\mathbf{e}^{d'}}\right], \tag{48}$$

where $\Phi_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+\frac{1}{2}\mathbf{e}^{d'}}$ is the prescribed boundary value. This yields a third-order accurate first derivative and first-order accurate third derivative, respectively. We furthermore approximate the transverse gradients by

$$\left(\nabla^{\perp,d}\frac{\partial\Phi}{\partial\xi_{d'}}\right)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \frac{1}{5h^2}\left[16\Phi_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+\frac{1}{2}\mathbf{e}^{d'}} - 25\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + 10\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-\mathbf{e}^{d'}} - \alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-2\mathbf{e}^{d'}}\right] + O(h^2). \tag{49}$$

Finally, consider cell faces normal to a Dirichlet boundary and one cell away, such as the face centered on the point labeled D in Fig. 3. The average of the normal derivatives ($d = d'$) is computed in the same manner as for the interior cells. For the average of the transverse derivative ($d \neq d'$), we replace (36) and (37) by

$$\gamma^{d',(4)}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \equiv \frac{1}{210}\left[-64\Phi_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+\frac{1}{2}\mathbf{e}^{d'}} + 210\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+\mathbf{e}^{d'}} - 35\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} - 126\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-\mathbf{e}^{d'}} + 15\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-2\mathbf{e}^{d'}}\right], \tag{50}$$

$$\gamma^{d',(2)}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \equiv \frac{1}{35}\left[64\Phi_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+\frac{1}{2}\mathbf{e}^{d'}} - 105\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+\mathbf{e}^{d'}} + 35\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + 21\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-\mathbf{e}^{d'}} - 15\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-2\mathbf{e}^{d'}}\right], \tag{51}$$

where again $\Phi_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+\frac{1}{2}\mathbf{e}^{d'}}$ is the prescribed boundary value. This yields a fourth-order accurate first derivative and second-order accurate third derivative, respectively. We approximate the transverse gradients by

$$\left(\nabla^{\perp,d}\frac{\partial\Phi}{\partial\xi_{d'}}\right)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \frac{1}{105h^2}\left[32\Phi_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+\frac{1}{2}\mathbf{e}^{d'}} + 35\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+\mathbf{e}^{d'}} - 140\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + 63\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-\mathbf{e}^{d'}} + 20\alpha_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-2\mathbf{e}^{d'}}\right] + O(h^2). \tag{52}$$

Stencil modifications corresponding to a Dirichlet condition at the lower boundary of a coordinate direction are obtained in the obvious way by permuting indices and negating the entries of stencils corresponding to odd order derivatives.

3.3. A numerical example

To test the discretization described in the preceding sections, we consider the solution of Poisson's equation in the "D"-shaped annular geometry Ω presented in Section IV of [11] and depicted in Fig. 4. The mapping $\mathbf{X}$ from computational coordinates $\boldsymbol{\xi} = (\xi_1, \xi_2)$ to physical coordinates $\mathbf{x} = (x_1, x_2)$ is given by

$$\begin{aligned} x_1 &= 1.7 + [0.074(2\xi_1 - 1) + 0.536]\cos\left[2\pi\xi_2 + \sin^{-1}(0.416)\sin(2\pi\xi_2)\right], \\ x_2 &= 1.66[0.074(2\xi_1 - 1) + 0.536]\sin(2\pi\xi_2), \end{aligned} \tag{53}$$

for $0 \leqslant \xi_1, \xi_2 \leqslant 1$. We seek the solution of

$$\nabla^2\Phi(\mathbf{x}) = \rho(\mathbf{x}), \quad \mathbf{x} \in \Omega, \tag{54}$$

satisfying homogeneous Dirichlet boundary conditions in the radial (ξ_1) direction

$$\Phi(\mathbf{X}(0,\xi_2)) = \Phi(\mathbf{X}(1,\xi_2)) = 0, \quad 0 \leqslant \xi_2 \leqslant 1, \tag{55}$$

and periodic boundary conditions in the azimuthal (ξ_2) direction. To test the accuracy of our discretization, we employ a manufactured solution procedure and compute a right-hand side ρ corresponding to a predetermined, analytically prescribed solution. After forming and solving the linear system resulting from our discretization using the right-hand side so obtained, we can compute the discretization error since the exact solution is known. In particular, we set $\Phi(\mathbf{x}) = \widetilde{\Phi}(\mathbf{X}^{-1}(\mathbf{x}))$, where

$$\widetilde{\Phi}(\xi_1, \xi_2) = 4\xi_1(1 - \xi_1)[1 + 0.1\sin(8\pi\xi_2)]. \tag{56}$$

Using (56), we apply the divergence theorem to compute the integral of ρ over a mapped grid cell $\mathbf{X}(V)$ as

$$\int_{\mathbf{X}(V)} \rho(\mathbf{x})d\mathbf{x} = \sum_{\pm=+,-}\sum_{d=1}^{2}\int_{A_d^{\pm}}\left(J^{-1}\mathbf{N}^T\mathbf{N}\nabla_{\xi}\widetilde{\Phi}\right)_d d\mathbf{A}_{\xi}. \tag{57}$$

Since the mapping (53) is prescribed using simple analytic formulas, the transformation matrix $\mathbf{N}$ and its Jacobian J are also explicitly available. The one-dimensional integrals in (57) are evaluated using the DQAG integrator from QUADPACK [12], which implements a globally adaptive Gauss–Kronrod quadrature to prescribed relative and absolute tolerances. For the

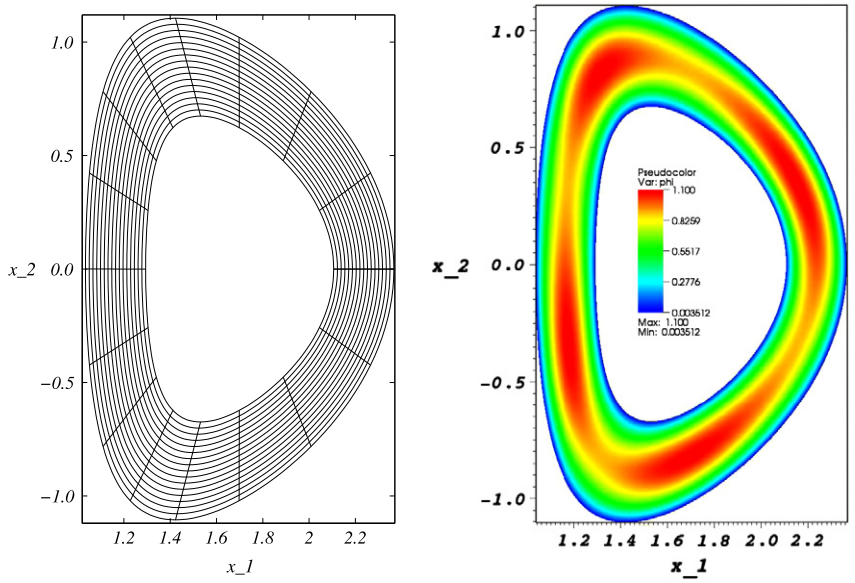


Fig. 4. Test problem geometry Ω with example $N = 16$ grid (left) and solution computed with $N = 512$ (right).

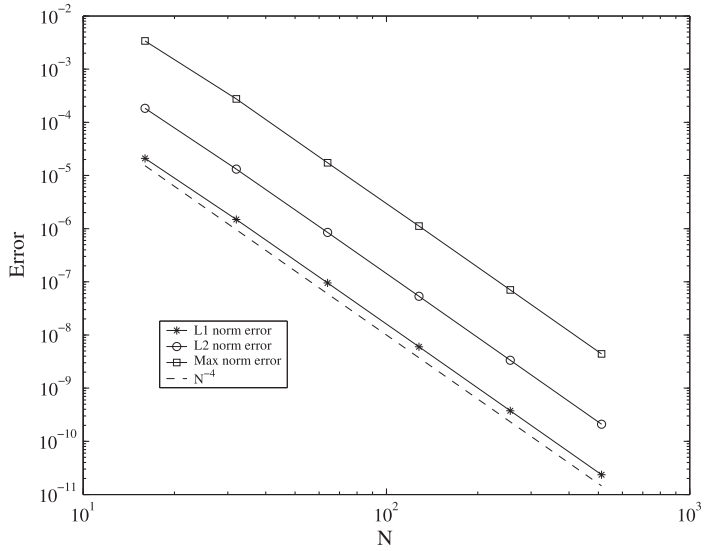


Fig. 5. Test problem convergence study.

Table 1
PCG convergence rates and residuals for $N = 512$.

Iteration number	Convergence rate	Residual norm
1	0.000223	2.232355e−04
2	0.209853	4.684667e−05
3	0.074080	3.470404e−06
4	0.120163	4.170155e−07
5	0.144002	6.005124e−08
6	0.075322	4.523166e−09
7	0.247301	1.118583e−09
8	0.107240	1.199568e−10
9	0.090233	1.082405e−11

convergence results presented Table 1, the relative and absolute tolerances were set to 10^{-12} to reduce the quadrature error well Table 1 the discretization errors being measured, yielding a nearly exact evaluation of the integral (57).

The discretization described above was implemented in a test code using the Chombo infrastructure [13]. The resulting linear system was solved using a preconditioned conjugate gradient (PCG) solver from the *hypre* library [14]. The *hypre* PCG solver accommodates the 5×5 stencil generated by the fourth-order discretization at each grid cell. Given the positive-definite, self-adjoint property of the linear operator, a possible alternative approach would have been to employ a multigrid solver. We do not, however, have a multigrid solver available that can accommodate a 5×5 stencil, including the multigrid solvers available in *hypre*. We therefore employ the *hypre* PFMG multigrid solver applied to a standard second-order discretization of (54) as a preconditioner for CG. Beyond the issue of solver availability, it is generally acknowledged that the combination of a Krylov method with multigrid preconditioning yields a more robust solver, especially with variable coefficients.

Fig. 5 shows the L_1, L_2 and max norm errors in the discrete solution of (54) using an $N \times N$ computational grid with $N = 16$, 32, 64, 128, 256 and 512, plotted against N^{-4}. By L_1 and L_2 norms, we mean the discrete norms computed using the solution error evaluated on cell averages, rather than continuous norms computed using a reconstruction of the discrete solution over grid cells. The fourth-order convergence rate is clearly observed. In each case the PCG iteration was performed to a tolerance of 10^{-10} on the relative L^2 residual norm (the L^2 norm of the residual relative to that of the right hand side). The multigrid preconditioner was solved to a relative tolerance of 10^{-2} in each iteration. Table 1 shows the relative residual at each iteration together with an estimate of the local convergence rate for the finest grid $N = 512$. A pseudocolor plot of the $N = 512$ solution is shown on the right-hand side of Fig. 4.

4. Application to scalar, linear hyperbolic equations

In this section, we apply the mapped grid formalism to obtain a fourth-order accurate finite-volume discretization of a scalar hyperbolic conservation law

$$\frac{\partial u}{\partial t} + \nabla_{\mathbf{x}} \cdot \mathbf{F} = 0 \quad \text{on } \mathbf{x} \in \Omega \subset \mathbb{R}^D, \ t \geqslant 0. \tag{58}$$

This equation expresses the evolution of the conserved scalar field $u : \Omega \times [0,\infty) \to \mathbb{R}$ under the action of the vector-valued flux function $\mathbf{F}(u,\mathbf{x},t)$, where $\mathbf{F} : \Omega \times [0,\infty) \to \mathbb{R}^D$. The vector whose entries consist of the wave speeds in each coordinate direction is given by

$$\mathbf{v}(u,\mathbf{x},t) \equiv \frac{\partial \mathbf{F}}{\partial u}, \tag{59}$$

where $\mathbf{v} : \Omega \times [0,\infty) \to \mathbb{R}^D$. In the simplest case, this velocity vector is a constant.

For our purposes here, we restrict consideration primarily to periodic domains Ω in order to focus on the base discretization and the incorporation of limiting. We thus leave the topic of boundary conditions for future work. Because ghost cells are a natural way to impose boundary conditions for explicit, hyperbolic schemes, we do not anticipate any severe complications or stencil modifications in order to enforce non-periodic boundary conditions for our mapped grid approach.

Because of the ease of formulation for higher-order, we adopt a method-of-lines discretization approach. As in (5), let $\mathbf{X}$ be a smooth mapping from some abstract Cartesian coordinate space ξ into physical space. We discretize on a uniform finite-volume grid (1) with grid spacing h; thus each control volume is $V_{\mathbf{i}} = h^D$. Integrating (58) over a cell $\mathbf{i}$ and applying (7) and (20) yields the semi-discrete system of ordinary differential equations

$$\frac{d}{dt}\left(\int_{V_{\mathbf{i}}} uJ\, d\xi\right) = \int_{\mathbf{X}(V_{\mathbf{i}})} \frac{\partial u}{\partial t} d\mathbf{x} = -\int_{\mathbf{X}(V_{\mathbf{i}})} \nabla_{\mathbf{x}} \cdot \mathbf{F}\, d\mathbf{x} \approx -h^{D-1} \sum_{d=1}^{D} \left(F^d_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} - F^d_{\mathbf{i}-\frac{1}{2}\mathbf{e}^d}\right). \tag{60}$$

Dividing by the uniform cell volume h^D, Eq. (60) becomes the fourth-order update formula

$$\frac{d\overline{(uJ)}_{\mathbf{i}}}{dt} = -\frac{1}{h} \sum_{d=1}^{D} \left(F^d_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} - F^d_{\mathbf{i}-\frac{1}{2}\mathbf{e}^d}\right) + O(h^4), \tag{61}$$

written in terms of the cell-averaged quantity $\overline{(uJ)}_{\mathbf{i}}$ on the computational grid. The cell average of the solution of the physical space grid is then

$$\tilde{u}_{\mathbf{i}} \equiv \left(\int_{\mathbf{X}(V_i)} d\mathbf{x}\right)^{-1} \int_{\mathbf{X}(V_i)} u(\mathbf{x}) d\mathbf{x} = \overline{(J)}_{\mathbf{i}}^{-1} \overline{(uJ)}_{\mathbf{i}}. \tag{62}$$

For discrete conservation, it is easiest to store and to update $\overline{(uJ)}_{\mathbf{i}}$ directly, converting to $\tilde{u}_{\mathbf{i}}$ only for output.

4.1. Face-averaged flux approximation

In the update (61), the face-averaged normal interface fluxes $F^d_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$ are approximated to fourth-order by (21). Thus, the problem is reduced to obtaining fourth-order accurate approximations to each component s of the face-averaged interface fluxes $\langle F^s \rangle_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d}$ $(s = 1,\ldots,D)$ averaged over the computational space cell faces with normals in the d-directions ($d = 1,\ldots,D$).

Taylor expansion of the integrand of (18) about the center of a dth face gives

$$\langle \mathbf{F} \rangle_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d} = \mathbf{F}(\xi_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d}) + \frac{h^2}{24} \sum_{d' \neq d} \left.\frac{\partial^2 \mathbf{F}}{\partial \xi^2_{d'}}\right|_{\xi=\xi_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d}} + O(h^4). \tag{63}$$

Specializing to a linear flux, we assume $\mathbf{F}(\xi) = \mathbf{v}(\xi)u(\xi)$. Then (63) becomes

$$\langle \mathbf{F} \rangle_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d} = \mathbf{v}_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d} u_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d} + \frac{h^2}{24} \sum_{d' \neq d} \left(u \frac{\partial^2 \mathbf{v}}{\partial \xi^2_{d'}} + 2 \frac{\partial \mathbf{v}}{\partial \xi_{d'}} \frac{\partial u}{\partial \xi_{d'}} + \mathbf{v} \frac{\partial^2 u}{\partial \xi^2_{d'}}\right)_{\xi=\xi_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d}} + O(h^4), \tag{64}$$

where we adopt the pointwise notation $q_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d} \equiv q(\xi_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d})$. Thus, the face-averaged flux is expressed in terms of pointwise values of $\mathbf{v}$, u, and their derivatives at the center of the face $\mathbf{i} \pm \frac{1}{2}\mathbf{e}^d$.

The expansion of the integral (63) also gives pointwise values expressed in terms of face-averaged values. Specifically, for pointwise values of u and $\mathbf{v}$, one can write

$$\mathbf{v}_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d} = \langle \mathbf{v} \rangle_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d} - \frac{h^2}{24} \sum_{d' \neq d} \left.\frac{\partial^2 \mathbf{v}}{\partial \xi^2_{d'}}\right|_{\xi=\xi_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d}} + O(h^4), \tag{65a}$$

$$u_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d} = \langle u \rangle_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d} - \frac{h^2}{24} \sum_{d' \neq d} \left.\frac{\partial^2 u}{\partial \xi^2_{d'}}\right|_{\xi=\xi_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d}} + O(h^4). \tag{65b}$$

Thus, the average interface flux is

$$\langle \mathbf{F} \rangle_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d} = \langle \mathbf{v} \rangle_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d} \langle u \rangle_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d} + \frac{h^2}{12} \sum_{d' \neq d} \left(\frac{\partial \mathbf{v}}{\partial \xi_{d'}} \frac{\partial u}{\partial \xi_{d'}} \right)_{\xi = \xi_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d}} + O(h^4), \tag{66}$$

written in terms of the face-averages $\langle u \rangle_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d}$ and $\langle \mathbf{v} \rangle_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d}$ and the pointwise values of the gradients of u and $\mathbf{v}$ at the center of the face.

In a finite-volume scheme, one works with cell-averaged values. Through the use of primitive functions [15], one can construct face-averages directly in terms of cell averages; at fourth-order on a uniform grid, this yields the centrally-differenced expression

$$\langle q \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \frac{7}{12} \left(\bar{q}_{\mathbf{i}} + \bar{q}_{\mathbf{i}+\mathbf{e}^d} \right) - \frac{1}{12} \left(\bar{q}_{\mathbf{i}+2\mathbf{e}^d} + \bar{q}_{\mathbf{i}-\mathbf{e}^d} \right) + O(h^4). \tag{67}$$

To approximate the pointwise transverse gradients, we first note that an $O(h^2)$ approximation is sufficient. In the d'th direction, a suitable centrally differenced approximation is

$$\left. \frac{\partial q}{\partial \xi_{d'}} \right|_{\xi = \xi_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}^d}} = \frac{1}{4h} \left(\bar{q}_{\mathbf{i}+\mathbf{e}^{d'}} + \bar{q}_{\mathbf{i}\pm\mathbf{e}^d+\mathbf{e}^{d'}} - \bar{q}_{\mathbf{i}-\mathbf{e}^{d'}} - \bar{q}_{\mathbf{i}\pm\mathbf{e}^d-\mathbf{e}^{d'}} \right) + O(h^2). \tag{68}$$

Expressions (67) and (68) provide the approximations necessary to evaluate the face-averaged fluxes (66) to fourth-order given the cell averages of u and $\mathbf{v}$ on the computational grid.

To obtain the average $\bar{u}_{\mathbf{i}}$ from the average $\overline{(uJ)}_{\mathbf{i}}$, we again appeal to Taylor series expansion of the integrand to express the average of products as the product of averages:

$$\overline{(uJ)}_{\mathbf{i}} = \bar{u}_{\mathbf{i}} \bar{J}_{\mathbf{i}} + \frac{h^2}{12} \left(\nabla_\xi u \cdot \nabla_\xi J \right)_{\mathbf{i}} + O(h^4). \tag{69}$$

Thus,

$$\bar{u}_{\mathbf{i}} \equiv h^{-D} \int_{V_i} u(\mathbf{x}(\xi)) d\xi = (\bar{J})_{\mathbf{i}}^{-1} \left[\overline{(uJ)}_{\mathbf{i}} - \frac{h^2}{12} \nabla_\xi u \cdot \nabla_\xi J + O(h^4) \right]. \tag{70}$$

We require at least second-order approximations of the gradients and choose the central differences

$$(\nabla_\xi u)_{\mathbf{i}}^d = \frac{1}{2h} \left[\frac{\overline{(uJ)}_{\mathbf{i}+\mathbf{e}^d}}{\bar{J}_{\mathbf{i}+\mathbf{e}^d}} - \frac{\overline{(uJ)}_{\mathbf{i}-\mathbf{e}^d}}{\bar{J}_{\mathbf{i}-\mathbf{e}^d}} \right] + O(h^2) \tag{71}$$

in each direction d. This choice is freestream preserving; the difference evaluates to zero (within roundoff) provided that the averages are initialized such that, for constant u, $\overline{uJ}_{\mathbf{i}} \equiv u \bar{J}_{\mathbf{i}}$.

4.2. Time discretization

As in [7], we discretize the semi-discrete system of ordinary differential equations (61) using the explicit, four-stage, fourth-order classical Runge–Kutta scheme [16]. Consider the variable-coefficient problem

$$\frac{d\mathbf{y}}{dt} = \mathbf{A}(\mathbf{x}) \mathbf{y}, \tag{72}$$

where $\mathbf{y}$ is the vector of N unknowns $\overline{(uJ)}_{\mathbf{i}}$ and $\mathbf{A}(\mathbf{x})$ is an $N \times N$ spatially-varying variable-coefficient matrix. For all explicit, four-stage, fourth-order Runge–Kutta temporal discretizations, the characteristic polynomial is

$$P(z_j) = 1 + z_j + \frac{z_j^2}{2} + \frac{z_j^3}{6} + \frac{z_j^4}{24}, \tag{73}$$

where $z_j = \Delta t \lambda_j$ and the $\lambda_j \in \mathbb{C}$ are the N eigenvalues of $\mathbf{A}$. The constant-coefficient stability constraint is

$$|P(z_j)| \leqslant 1, \quad \forall j \in [1, N]. \tag{74}$$

Alternatively, for all j, let $z_j = x_j + i y_j$ with $x_j, y_j \in \mathbb{R}$, then the amplification factor g_j of the fully discrete scheme has real part

$$\mathrm{Re}\, g_j = \left(1 + x_j + \frac{x_j^2}{2} + \frac{x_j^3}{6} + \frac{x_j^4}{24} \right) - \frac{y_j^2}{2} \left(1 + x_j + \frac{x_j^2}{2} \right) + \frac{y_j^4}{24} \tag{75}$$

and imaginary part

$$\mathrm{Im}\, g_j = y_j\left(1 + x_j + \frac{x_j^2}{2} + \frac{x_j^3}{6}\right) - \frac{y_j^3}{6}(1 + x_j), \tag{76}$$

and our notion of stability implies that $|g_j| \leqslant 1$ for all j. If one can estimate the eigenvalues of the spatial operator λ_j, one then has a means of selecting a stable timestep.

Analytically, the constant-coefficient problem reveals a potential shortcoming of our full discretization. Define the shift operator and its inverse

$$T_d u_{\mathbf{i}} = u_{\mathbf{i}+\mathbf{e}^d} \quad \text{and} \quad T_d^{-1} u_{\mathbf{i}} = u_{\mathbf{i}-\mathbf{e}^d}.$$

The semi-discrete system of ordinary differential equations (61) reduces to

$$\frac{d\overline{(uJ)}_{\mathbf{i}}}{dt} = -\frac{1}{h}\left(\sum_{d=1}^{D} \mathbf{v}\cdot\mathbf{e}^d\left[\frac{2}{3}\left(T_d - T_d^{-1}\right) - \frac{1}{12}\left(T_d^2 - T_d^{-2}\right)\right]\right)\overline{(uJ)}_{\mathbf{i}}. \tag{77}$$

On a periodic domain, the eigenvalues are

$$\lambda_{\mathbf{i}} = -\frac{i}{3h}\sum_{d=1}^{D} \mathbf{v}\cdot\mathbf{e}^d \sin\theta_{k_d}\left[4 - \cos\theta_{k_d}\right], \tag{78}$$

where the discrete phase angles are $\theta_{k_d} = 2\pi k_d/n,\ k_d = 0, \pm 1, \pm 2, \ldots, \pm n/2$. Because of the central spatial discretization, the eigenvalues are all pure imaginary, that is, the spatial discretization contributes no numerical dissipation.

To keep the purely imaginary eigenvalues of the linear scheme within the Runge–Kutta stability region, the eigenvalue of maximum magnitude must satisfy $|\lambda_i|\Delta t \leqslant 2\sqrt{2}$. Each dimensional term in (78) is maximized near $\theta \approx 1.8$ with a value near 1.3722, and so the stability constraint becomes

$$\frac{\Delta t}{h}\sum_{d=1}^{D} |\mathbf{v}\cdot\mathbf{e}^d| \lessapprox 2.06. \tag{79}$$

Of course, this constraint is merely an estimate; in addition to assuming constant-coefficient advection, this analysis does not take into account contributions from the transverse gradients.

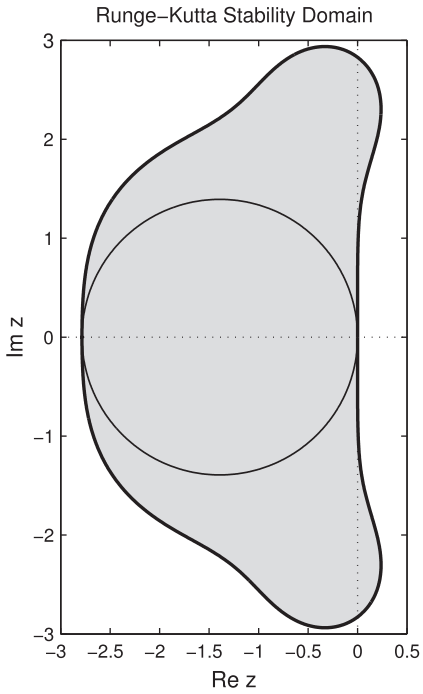


Fig. 6. Stability region of the classical Runge–Kutta scheme. The enclosed circle is the locus of eigenvalues for maximum CFL number for first-order upwind fluxes.

For mapped grids in physical space, we apply the stability conditions in computational space using the computational-space velocity vector $\mathbf{w} = J^{-1}\mathbf{N}^T\mathbf{v}$. Because the computational-space velocity varies spatially even for constant $\mathbf{v}$, we apply the condition (79) locally,

$$\frac{\Delta t}{h} \max_{\mathbf{i}} \left(\sum_{d=1}^{D} |\mathbf{w}_{\mathbf{i}} \cdot \mathbf{e}^d| \right) \lessapprox 2.06 \tag{80}$$

to obtain a global limit on the time step.

The magnitude of the amplification factor as a function of continuous phase angles is

$$|g| = \sqrt{1 - \frac{y^6}{72}\left(1 - \frac{y^2}{8}\right)}. \tag{81}$$

For $D = 1$, this amplification factor is plotted in Fig. 7. Similarly, the relative phase speed of the one-dimensional scheme,

$$\frac{\alpha(\theta)}{a} = -\frac{1}{\sigma\theta}\frac{\mathrm{Im}\, g(\theta)}{\mathrm{Re}\, g(\theta)}, \tag{82}$$

where $\sigma = a\Delta t/h$, is also plotted in Fig. 7. We see that the Runge–Kutta scheme adds a small amount of dissipation and that, as $\theta \to \pm\pi$, this dissipation vanishes. At the same time, we see that these high-wavenumber modes (grid modes) do not propagate. For variable-coefficient and nonlinear problems, these undamped grid modes can pollute the solutions, if not cause instability.

4.3. Limiting

One approach to stabilize a high-order, no- or low-dissipation scheme for variable-coefficient or nonlinear hyperbolic problems is to add artificial dissipation. For example, adding an artificial dissipation of the form

$$\mu(-1)^{r-1}h^{2r-1}(T-1)^r(1-T^{-1})^r\bar{u}_{\mathbf{i}}, \tag{83}$$

to the discrete scheme in computational space gives a scheme dissipative of order $2r$ in the sense of Kreiss [17,18]. However, for $r > 1$, the discrete higher-order derivatives will introduce new numerical difficulties in the presence of discontinuous solutions or poorly resolved gradients.

An alternative approach from the shock-capturing literature is to use a nonlinear limiting scheme. For linear, variable-coefficient problems, genuinely nonlinear discontinuities (shocks) do not occur. However, velocity gradients can cause slopes in the solution to steepen, and initial and boundary conditions can introduce linear discontinuities. We therefore will use nonlinear flux limiting for robust handling of under-resolved gradients and discontinuities. A disadvantage of this approach is that it locally reduces the order of convergence of the scheme, but for smooth problems, this should only occur in a very small subset of cells, if at all. Thus, the maximum pointwise error may not converge at fourth-order, but the errors should converge near fourth-order almost everywhere.

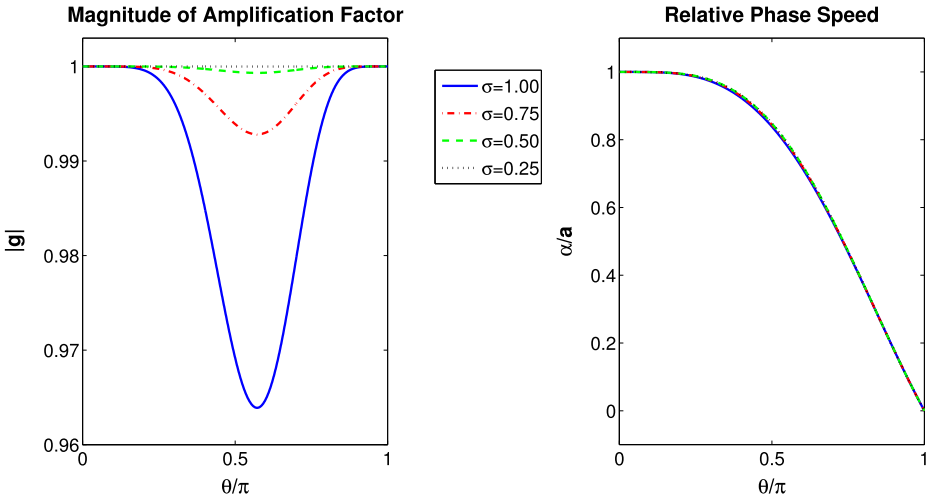


Fig. 7. Variation of the magnitude of the amplification factor $|g|$ and the relative phase speed α/a with phase angle θ for several values of σ. Note that as $\theta \to \pm\pi$, the damping vanishes and the modes do not propagate. Furthermore, the phase error is effectively independent of σ.

In the mapped-grid formalism, we propose to apply a limiter scheme to the cell-averaged solution on the computational grid, $\bar{u}_{\mathbf{i}}$. As a specific example, we implement a method-of-lines variant of the extremum-preserving, piecewise-parabolic limiter developed in [19] with some modifications. Unfortunately, we discovered that this limiter turned out to be sensitive to round-off error and to be overly aggressive near certain cubic extrema in multidimensional problems. We present here a modification intended to eliminate these problems, but our results will demonstrate that these modifications decrease the effectiveness of the limiter for discontinuous initial data. Recently, a more robust improvement of the limiter that operates correctly on discontinuities was formulated in [20].

4.3.1. Limiting face values

We compute limited face-averaged values as in [19], starting with the face-averaged values $\langle u\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$ given by (67). At each face $\mathbf{i}+\frac{1}{2}\mathbf{e}^d$, we compute the left and right undivided differences corresponding to the local slope:

$$\delta u_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,L} = \langle u\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} - \bar{u}_{\mathbf{i}}, \tag{84a}$$

$$\delta u_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,R} = \bar{u}_{\mathbf{i}+\mathbf{e}^d} - \langle u\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}. \tag{84b}$$

If $\delta u_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,L}\delta u_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,R} < 0$, then this is a local extremum, and we will limit the face value if it is not a genuine extremum. We compute three undivided differences corresponding to the curvature at the face $\mathbf{i}+\frac{1}{2}\mathbf{e}^d$:

$$\delta^2 u_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,L} = (\bar{u}_{\mathbf{i}-\mathbf{e}^d} - 2\bar{u}_{\mathbf{i}} + \bar{u}_{\mathbf{i}+\mathbf{e}^d})/2, \tag{85a}$$

$$\delta^2 u_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,C} = 3\left(\bar{u}_{\mathbf{i}} - 2\langle u\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + \bar{u}_{\mathbf{i}+\mathbf{e}^d}\right), \tag{85b}$$

$$\delta^2 u_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,R} = (\bar{u}_{\mathbf{i}} - 2\bar{u}_{\mathbf{i}+\mathbf{e}^d} + \bar{u}_{\mathbf{i}+2\mathbf{e}^d})/2. \tag{85c}$$

We now include a check that deviates from [19]: we limit the face values only if the three curvature estimates (85) are not monotonic, that is,

$$\left(\delta^2 u_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,R} - \delta^2 u_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,C}\right)\left(\delta^2 u_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,C} - \delta^2 u_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,L}\right) < 0. \tag{86}$$

When (86) is true, provided that all three curvature estimates (85) are of the same sign, we define the limited curvature to be

$$\delta^2\tilde{u}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = s_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}\min\left(C\delta^2 u_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,L}, C\delta^2 u_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,R}, \delta^2 u_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,C}\right), \tag{87}$$

where $C = 1.25$ and $s_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \mathrm{sign}(\delta^2 u_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,C})$. Otherwise, $\delta^2\tilde{u}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = 0$. The limited face values are then

$$\langle\tilde{u}\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} = \frac{1}{2}(\bar{u}_{\mathbf{i}} + \bar{u}_{\mathbf{i}+\mathbf{e}^d}) - \frac{1}{6}\delta^2\tilde{u}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}, \tag{88}$$

where $\delta^2\tilde{u}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$ is the limited local curvature estimate. We note that the corresponding Eq. (19) in [19] has an incorrect factor of 1/3 multiplying the final term.

4.3.2. Constructing the parabolic interpolant

In the space–time PPM formulation [15], a limited parabolic interpolant is formed in each cell and advanced in time. In our method-of-lines case, we only require the instantaneous end points of the limited, parabolic interpolant, $\langle\tilde{u}\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,\pm}$. It is in this step that we depart substantially from the procedure described in [19].

We first determine if we have a genuine extremum. For each cell $\mathbf{i}$, we first compute a pair of undivided differences:

$$\delta u_{\mathbf{i},L} = \bar{u}_{\mathbf{i}} - \langle\tilde{u}\rangle_{\mathbf{i}-\frac{1}{2}\mathbf{e}^d}, \tag{89a}$$

$$\delta u_{\mathbf{i},R} = \langle\tilde{u}\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} - \bar{u}_{\mathbf{i}}. \tag{89b}$$

If $\delta u_{\mathbf{i},L}\delta u_{\mathbf{i},R} < 0$, the cell contains a local extremum. The parabolic interpolant in the cell may also contain a local extremum if

$$|\delta u_{\mathbf{i},L}| > 2|\delta u_{\mathbf{i},R}| \quad \text{or } |\delta u_{\mathbf{i},R}| > 2|\delta u_{\mathbf{i},L}|, \tag{90}$$

so we check to see if the differences in values adjacent to the cell also change sign. We do this for both cell- and face-averaged values:

$$\delta u^{\mathrm{face}}_{\mathbf{i},L} = \langle\tilde{u}\rangle_{\mathbf{i}-\frac{1}{2}\mathbf{e}^d} - \langle\tilde{u}\rangle_{\mathbf{i}-\frac{3}{2}\mathbf{e}^d}, \tag{91a}$$

$$\delta u^{\mathrm{face}}_{\mathbf{i},R} = \langle\tilde{u}\rangle_{\mathbf{i}+\frac{3}{2}\mathbf{e}^d} - \langle\tilde{u}\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}, \tag{91b}$$

$$\delta u^{\mathrm{face}}_{\mathbf{i},\mathrm{minmod}} = \min\left(\left|\delta u^{\mathrm{face}}_{\mathbf{i},L}\right|, \left|\delta u^{\mathrm{face}}_{\mathbf{i},R}\right|\right), \tag{91c}$$

$$\delta u^{\mathrm{cell}}_{\mathbf{i},L} = \bar{u}_{\mathbf{i}} - \bar{u}_{\mathbf{i}-\mathbf{e}}, \tag{92a}$$

$$\delta u^{\mathrm{cell}}_{\mathbf{i},R} = \bar{u}_{\mathbf{i}+\mathbf{e}} - \bar{u}_{\mathbf{i}}, \tag{92b}$$

$$\delta u^{\mathrm{cell}}_{\mathbf{i},\mathrm{minmod}} = \min\left(\left|\delta u^{\mathrm{cell}}_{\mathbf{i},L}\right|, \left|\delta u^{\mathrm{cell}}_{\mathbf{i},R}\right|\right), \tag{92c}$$

and choose the larger of the two so as to avoid issues with round-off error. If

$$\delta u^{\text{face}}_{\mathbf{i},\text{minmod}} \geqslant \delta u^{\text{cell}}_{\mathbf{i},\text{minmod}} \quad \text{and} \quad \delta u^{\text{face}}_{\mathbf{i},L}\delta u^{\text{face}}_{\mathbf{i},R} < 0 \tag{93}$$

or if

$$\delta u^{\text{cell}}_{\mathbf{i},\text{minmod}} \geqslant \delta u^{\text{face}}_{\mathbf{i},\text{minmod}} \quad \text{and} \quad \delta u^{\text{cell}}_{\mathbf{i},L}\delta u^{\text{cell}}_{\mathbf{i},R} < 0, \tag{94}$$

we determine that the interpolant contains an local extremum.

Now, for cells with local extrema, we follow the procedure in [19] and construct a limited undivided difference that corresponds to the curvature in the cell:

$$\delta^2 u_{\mathbf{i}} = -2\left(6\bar{u}_{\mathbf{i}} - 3\left[\langle\tilde{u}\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + \langle\tilde{u}\rangle_{\mathbf{i}-\frac{1}{2}\mathbf{e}^d}\right]\right), \tag{95a}$$

$$\delta^2 u_{\mathbf{i},L} = (\bar{u}_{\mathbf{i}-2\mathbf{e}^d} - 2\bar{u}_{\mathbf{i}-\mathbf{e}^d} + \bar{u}_{\mathbf{i}})/2, \tag{95b}$$

$$\delta^2 u_{\mathbf{i},C} = (\bar{u}_{\mathbf{i}-\mathbf{e}^d} - 2\bar{u}_{\mathbf{i}} + \bar{u}_{\mathbf{i}+\mathbf{e}^d})/2, \tag{95c}$$

$$\delta^2 u_{\mathbf{i},R} = (\bar{u}_{\mathbf{i}} - 2\bar{u}_{\mathbf{i}+\mathbf{e}^d} + \bar{u}_{\mathbf{i}+2\mathbf{e}^d})/2. \tag{95d}$$

If all the signs agree, we define

$$\delta^2\tilde{u}_{\mathbf{i}} = s_{\mathbf{i}}\min\left(\delta^2 u_{\mathbf{i}}, C\delta^2 u_{\mathbf{i},L}, C\delta^2 u_{\mathbf{i},R}, C\delta^2 u_{\mathbf{i},C}\right), \tag{96}$$

where $C = 1.25$ and $s_{\mathbf{i}} = \text{sign}(\delta^2 u_{\mathbf{i}})$. Otherwise, $\delta^2\tilde{u}_{\mathbf{i}} = 0$. If the cell does not contain an extremum but (90) holds, we limit the difference $\delta u_{\mathbf{i},L}$ or $\delta u_{\mathbf{i},R}$ that is greater in magnitude by twice the other:

$$\delta\tilde{u}_{\mathbf{i},L} = \begin{cases} 2\delta u_{\mathbf{i},R}, & \text{if} \quad |\delta u_{\mathbf{i},L}| > 2|\delta u_{\mathbf{i},R}|, \\ \delta u_{\mathbf{i},L}, & \text{otherwise}, \end{cases} \tag{97a}$$

$$\delta\tilde{u}_{\mathbf{i},R} = \begin{cases} 2\delta u_{\mathbf{i},L}, & \text{if} \quad |\delta u_{\mathbf{i},R}| > 2|\delta u_{\mathbf{i},L}|, \\ \delta u_{\mathbf{i},R}, & \text{otherwise}; \end{cases} \tag{97b}$$

note that at this point in the logic, we are guaranteed that $\delta u_{\mathbf{i},L}\delta u_{\mathbf{i},R} \geqslant 0$.

At cell interface $\mathbf{i}+\frac{1}{2}\mathbf{e}^d$, we then construct the left and and right values:

$$\langle u\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,-} = \bar{u}_{\mathbf{i}} + \frac{\delta^2\tilde{u}_{\mathbf{i}}}{\delta^2 u_{\mathbf{i}}}\delta\tilde{u}_{\mathbf{i},R}, \tag{98a}$$

$$\langle u\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,+} = \bar{u}_{\mathbf{i}+\mathbf{e}^d} - \frac{\delta^2\tilde{u}_{\mathbf{i}+\mathbf{e}^d}}{\delta^2 u_{\mathbf{i}+\mathbf{e}^d}}\delta\tilde{u}_{\mathbf{i}+\mathbf{e}^d,L}. \tag{98b}$$

We then choose the interface state to be the upwind state. For our scalar problem, this is simply

$$\langle u\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,\text{upwind}} = \begin{cases} \langle u\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,-}, & \text{if } \mathbf{w}\cdot\mathbf{e}^d \geqslant 0, \\ \langle u\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d,+}, & \text{otherwise}, \end{cases} \tag{99}$$

where the flow is smooth, these two values will be equivalent to the original fourth-order, face-averaged value $\langle u\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$.

4.3.3. Limited scheme stability constraints

Finally, we note that as the limiter engages, the limited fluxes may be anywhere between first- and fourth-order. This changes the stability constraints for the algorithm. Specifically, consider the case where all fluxes for a cell are first-order upwind fluxes. Assuming that $\mathbf{v}\cdot\mathbf{e}^d > 0$ for all d, the eigenvalues are

$$\lambda_{\mathbf{i}} = -\frac{1}{h}\sum_{d=1}^{D}\mathbf{v}\cdot\mathbf{e}^d\left[1 - \cos\theta_{k_d} + i\sin\theta_{k_d}\right], \tag{100}$$

which is the weighted sum of circles centered on $z = -1$ with radius $r = 1$ in the complex plane. The weights scale the center and radius of each dimensional contribution. As Fig. 6 demonstrates, eigenvalues on these circles will have maximum magnitude on the real axis ($\theta = \pm\pi$), and the condition that the eigenvalues remain within the Runge–Kutta stability region is $|\lambda_i|\Delta t \lessapprox 2.785$. Thus, the stability constraint for first-order upwind fluxes is

$$\frac{\Delta t}{h}\sum_{d=1}^{D}\left|\mathbf{v}\cdot\mathbf{e}^d\right| \lessapprox 1.3925, \tag{101}$$

which is more restrictive than the constraint (79) on fourth-order fluxes. We have investigated several other combinations of low- and higher-order fluxes, but the constraint of purely first-order fluxes appears to be the most severe. As before, we extend the stability constraint to mapped grids by considering locally the computational-space velocity $\mathbf{w}_{\mathbf{i}}$ in place of $\mathbf{v}$.

4.4. Example problems

To demonstrate the behavior of the hyperbolic mapped finite-volume scheme, we investigate several initial value test problems. On a periodic domain, we consider uniform advection of constant, sinusoidal, and compactly supported initial data. We also consider the standard limiter test problem of rotational advection of a slotted cylinder. For each of these problems, we compute on a uniform, Cartesian mesh and a nonlinearly deformed mesh.

For each smooth problem, a grid convergence study is conducted. We compute on a sequence of six meshes with $N_{\text{cells}} = \{16, 32, 64, 128, 256, 512\}$. The error in the cell-averaged numerical solution $\widetilde{u}_{\mathbf{i}}$ is computed by differencing with a cell-averaged reference solution. The exact solutions for the test problems are easily expressed pointwise, but to compute the cell-averaged reference solution on the mapped grid analytically is difficult. Instead, for our reference solution, we compute cell-centered pointwise values $(uJ)_{\mathbf{i}'}$ on a uniform computational grid with $h = 1/(8N_{\text{cells}})$, and then use the relationships (4) and (62) to obtain approximate values of $\tilde{u}_{\mathbf{i}}$ on the original, coarser grid. Thus, our reference solution is a fourth-order approximation of the cell-averaged exact solution, but for the problems we consider, this error in the reference solution is several orders of magnitude smaller than the error between the computed solution and the reference solution.

4.4.1. Mappings

For reference, we compute on a uniform, Cartesian mesh, which corresponds to the identity mapping $\mathbf{x}(\xi) = \xi$. This demonstrates that the metric computations reduce to the correct trivial relationships and provides a baseline against which to compare the results of simulations with less trivial mappings.

The nonlinear mapping we consider is a specialization of the mapping used in [21]. This mapping is generated by perturbing a uniform Cartesian mesh by a Cartesian sinusoidal product, specifically,

$$x_d = \xi_d + c_d \prod_{p=1}^{D} \sin(2\pi\xi_p), \quad d = 1, \ldots, D. \tag{102}$$

To ensure that the perturbed mesh does not tangle, it is sufficient to take $\forall d$, $0 \leqslant 2\pi c_d \leqslant 1$. We use $c_d = 0.1$, $d = 1, 2$ in our example calculations.

This mapping cannot be inverted analytically, however, noting that

$$\xi_d = x_d + \frac{c_d}{c_{d'}}(\xi_{d'} - x_{d'}), \quad d = 1, \ldots, D, \ d \neq d', \tag{103}$$

for $c_{d'} \neq 0$, the mapping can be inverted numerically using a fixed-point iteration on the scalar equation (102) for $\xi_{d'}$. Note that for this mapping, the Jacobian is not constant. A depiction of the mapped grid for $N_{\text{cells}} = 64$ in each direction is plotted in Fig. 8.

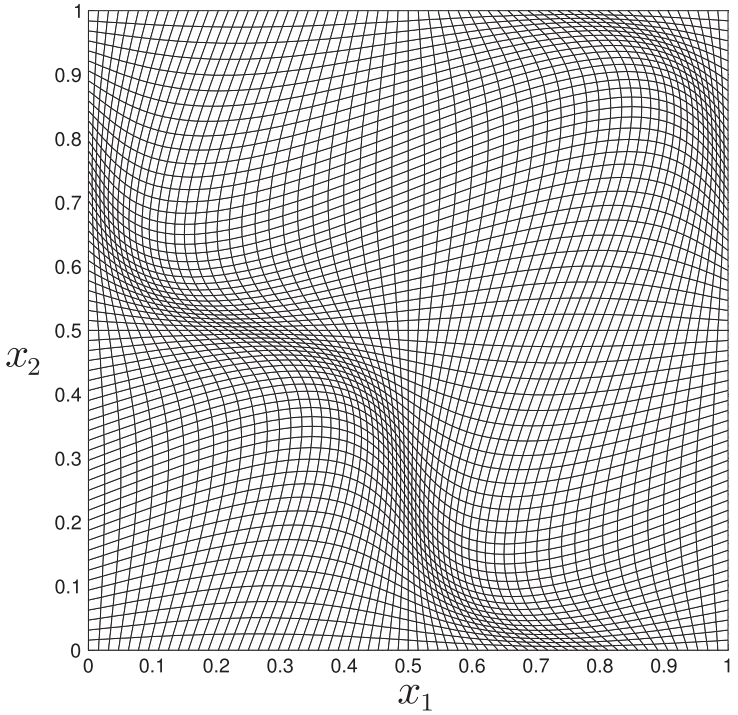


Fig. 8. The deformed mesh (102) used in the example problems, shown with $\mathbf{c} = [0.1, 0.1]$ and $N_{\text{cells}} = 64$ in each direction.

4.4.2. Uniform advection of constant data

We consider uniform advection of constant initial data on a periodic domain $\mathbf{x} \in [0,1)^2$. The initial conditions are

$$u_0(\mathbf{x}) = 1, \tag{IC1}$$

and the uniform velocity vector is

$$\mathbf{v}(\mathbf{x}) = (1, 0.5). \tag{104}$$

The exact solution is $u(\mathbf{x},t) = 1$ at all times, and we integrate to a final time of $t = 2$. While this problem appears to be trivial, it is an important demonstration of the freestream preservation capability of the discretization.

For all runs, we take $\Delta t\, N_{\text{cells}} = 4/15$ to be constant. Doing so ensures that, for a given resolution, the same integer number of time steps are taken regardless of the mapping and that, across resolutions for a particular mapping, the CFL number is fixed. The value of $\Delta t\, N_{\text{cells}}$ is set by the stability constraint for the limited scheme on the deformed mesh, for which $\max_{\mathbf{i}} \sum_d |\mathbf{w_i} \cdot \mathbf{e}^d| \approx 4.88$; the corresponding CFL number is 1.30. On the Cartesian mesh, $\sum_d |\mathbf{v} \cdot \mathbf{e}^d| = 1.50$, so the CFL number is 0.40.

Results of grid convergence studies are presented in Tables 2 and 3. Freestream preservation is demonstrated by the fact that the maximum pointwise error in the computed solution in both cases is at most dominated by round-off error. Both with and without the limiter, the method is shown to be freestream preserving.

We note that the results differ with and without the limiter because the limiter scheme operates on data even when the solution is constant. As such, the method does a number of finite-precision operations that can introduce additional round-off error. This is particularly true on the deformed mesh (102), where the non-constant Jacobian for this mapping also contributes to round-off error. Since the number of time steps, and thus the number of floating-point operations, increase with resolution, it is expected to see a slight growth in the magnitude of the roundoff error.

4.4.3. Uniform advection of sinusoidal data

We consider uniform advection of Cartesian-product sinusoidal initial data on a periodic domain $\mathbf{x} \in [0,1)^2$. The initial conditions are

$$u_0(\mathbf{x}) = \cos(2\pi x_1)\cos(2\pi x_2), \tag{IC2}$$

and the uniform velocity vector is again (104). The exact solution is

$$u(\mathbf{x},t) = u_0(\mathbf{x} - \mathbf{v}t), \tag{105}$$

where u_0 is by definition periodic in each direction with unit period. As before, the final time is $t = 2$, and we take a fixed CFL number of 1.30 on both meshes.

Results of grid convergence studies are plotted in Figs. 9 and 10. Fourth-order convergence is seen for both the base and limited schemes on both meshes with sufficient resolution. On the Cartesian mesh, the error norms of both schemes become indistinguishable at $N_{\text{cells}} = 32$. On the deformed mesh, the L_1 error norms of both schemes become indistinguishable at $N_{\text{cells}} = 64$, but the L_∞ error norms are slightly different. This behavior is consistent with the limiter 'turning off' once the solution is sufficiently resolved. Convergence rates above fourth-order are artifacts of the transition into the asymptotic regime of convergence.

4.4.4. Uniform advection of compactly-supported data

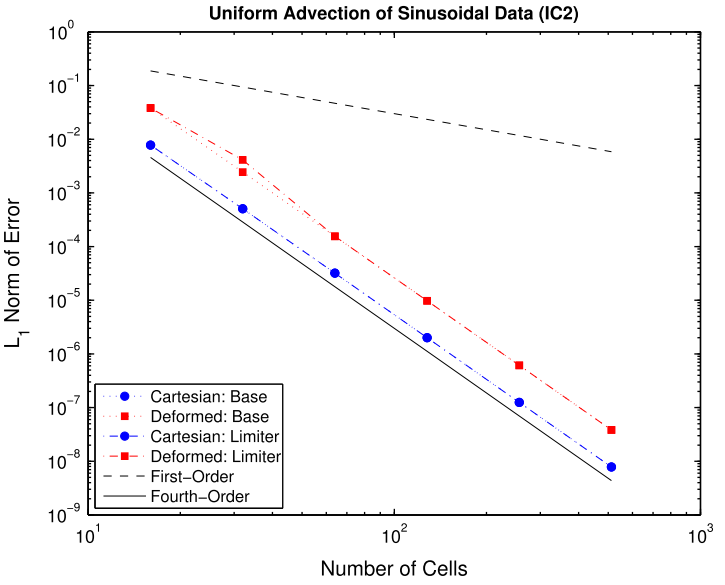
We consider uniform advection of compactly-supported smooth initial pulse on a periodic domain $\mathbf{x} \in [0,1)^2$. The initial conditions are

Table 2
L_∞ error norm of cell averages for freestream preservation test at six resolutions for the base scheme.

N_{cells}	16	32	64	128	256	512
Cartesian	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00
Deformed	0.00e+00	0.00e+00	0.00e+00	0.00e+00	4.44e−16	2.22e−16

Table 3
L_∞ error norm of cell averages for freestream preservation test at six resolutions for the base scheme with limiter.

N_{cells}	16	32	64	128	256	512
Cartesian	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00
Deformed	1.11e−16	2.22e−16	3.33e−16	3.33e−16	4.44e−16	6.66e−16



L_1 Error Norm for Base Scheme

N_{cells}		16	32	64	128	256	512
Cartesian	error	7.77e-03	5.07e-04	3.19e-05	2.00e-06	1.25e-07	7.82e-09
	rate	-	3.94	3.99	4.00	4.00	4.00
Deformed	error	3.82e-02	2.45e-03	1.56e-04	9.78e-06	6.12e-07	3.83e-08
	rate	-	3.96	3.98	3.99	4.00	4.00

L_1 Error Norm for Base Scheme with Limiter

N_{cells}		16	32	64	128	256	512
Cartesian	error	7.78e-03	5.07e-04	3.19e-05	2.00e-06	1.25e-07	7.82e-09
	rate	-	3.94	3.99	4.00	4.00	4.00
Deformed	error	3.81e-02	4.10e-03	1.56e-04	9.78e-06	6.12e-07	3.83e-08
	rate	-	3.22	4.72	3.99	4.00	4.00

Fig. 9. Convergence of L_1 error norm of cell averages with grid refinement for uniform advection (104) of sinusoidal data (IC2) for both meshes.

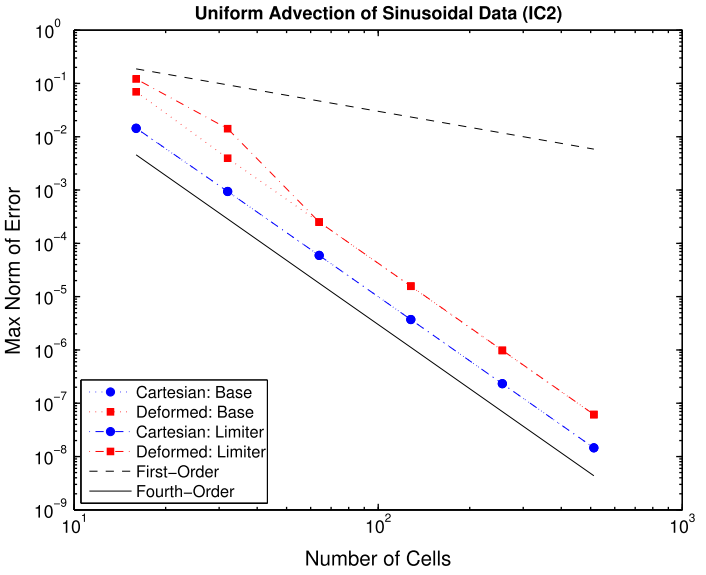
$$u_0(\mathbf{x}) = \begin{cases} \cos^8\left(\frac{\pi r}{2R}\right), & 0 \leqslant r \leqslant R, \\ 0, & \text{otherwise}, \end{cases} \tag{IC3}$$

$r = |\mathbf{x} - \mathbf{x}^*|$, and the uniform velocity vector is again (104). We take $R = 0.25$ and $\mathbf{x}^* = (0.75, 0.5)$. The exact solution is again given by (105), with $u(\mathbf{x} + \mathbf{e}^d, t) = u(\mathbf{x}, t)$ for $d = 1, 2$. As before, the final time is $t = 2$, and the fixed CFL number on both meshes is 1.30.

Results of the grid convergence study for the base and the limited scheme are plotted in Figs. 11 and 12. For both schemes and both meshes, the compact function is poorly represented at coarse resolutions; at $N_{\text{cells}} = 16$, there are only four cells in each direction across the non-zero portion of the function. By around $N_{\text{cells}} = 256$, the approximate solution enters the asymptotic convergence regime in both norms; a fourth-order convergence rate is observed for both schemes and on both meshes. On both meshes, the error norms of both schemes become effectively the same at $N_{\text{cells}} = 128$.

4.4.5. Circular advection of a slotted cylinder

A standard problem used to test the multidimensional performance of limiting schemes is the circular advection of a slotted cylinder. The rotational velocity vector for rotation about $\mathbf{x}^c = (0.5, 0.5)$ is



L_∞ Error Norm for Base Scheme

N_{cells}		16	32	64	128	256	512
Cartesian	error	1.43e-02	9.39e-04	5.94e-05	3.73e-06	2.33e-07	1.46e-08
	rate	-	3.93	3.98	4.00	4.00	4.00
Deformed	error	6.97e-02	3.94e-03	2.49e-04	1.56e-05	9.76e-07	6.10e-08
	rate	-	4.14	3.98	4.00	4.00	4.00

L_∞ Error Norm for Base Scheme with Limiter

N_{cells}		16	32	64	128	256	512
Cartesian	error	1.45e-02	9.39e-04	5.94e-05	3.73e-06	2.33e-07	1.46e-08
	rate	-	3.95	3.98	4.00	4.00	4.00
Deformed	error	1.21e-01	1.41e-02	2.51e-02	1.58e-05	9.86e-06	6.16e-08
	rate	-	3.10	5.81	3.99	4.00	4.00

Fig. 10. Convergence of L_∞ error norm of cell-averages with grid refinement for uniform advection (104) of sinusoidal data (IC2) for both meshes.

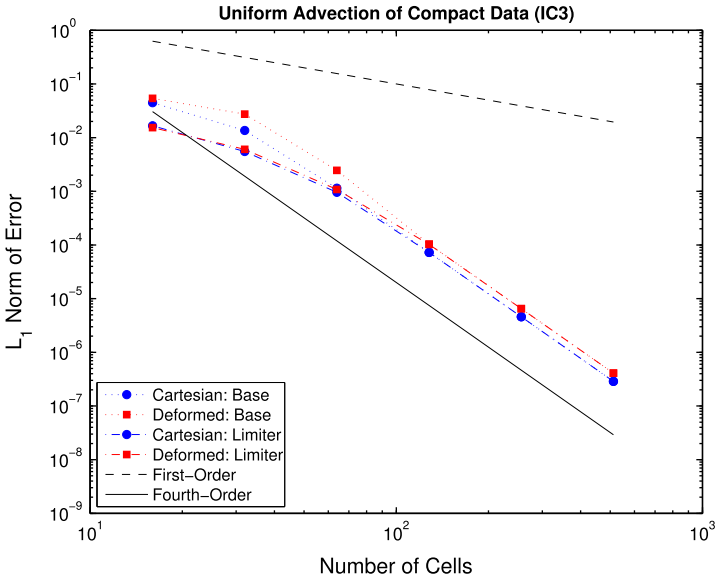
$$\mathbf{v}(\mathbf{x}) = 2\pi\omega(-(x_2 - x_2^c), (x_1 - x_1^c)), \tag{V3}$$

where we take $\omega = 1$. We define a slotted cylinder of radius $R = 0.15$, slot width $W = 0.05$, and slot height $H = 0.25$ centered on $\mathbf{x}^* = (0.5, 0.75)$ for the initial conditions:

$$u_0(\mathbf{x}) = \begin{cases} 0, & 0 \leqslant R < r, \\ 0, & |2z_1| < W \text{ and } 0 < z_2 + R < H, \\ 1, & \text{otherwise}, \end{cases}$$

where $\mathbf{z} = \mathbf{x} - \mathbf{x}^*$ and $r = |\mathbf{z}|$. We simulate on the truncated domain $\mathbf{x} \in [0,1]^2$ with $N_{\text{cells}} = 100$ cells in each direction. We use analytically prescribed boundary conditions for this problem; the solution near and beyond the artificial boundary of the domain is zero, and this is what we prescribe in the ghost cells. The mappings are periodic in their definition and so extend periodically beyond the simulation domain. The exact solution at time $t = 1$ is the initial condition, and we use a fixed CFL number of 1.33, which corresponds to $\Delta t = 1/472$ on the Cartesian mesh and $\Delta t = 1/800$ on the deformed mesh.

The initial and final solution for simulation on a Cartesian mesh are plotted in Fig. 13. The basic shape of the cylinder is preserved, albeit with some spreading of the initially sharp edges of the cylinder. The slot has some fill-in, and the



L_1 Error Norm for Base Scheme

N_{cells}		16	32	64	128	256	512
Cartesian	error	4.48e-02	1.36e-02	1.14e-03	7.28e-05	4.59e-06	2.89e-07
	rate	-	1.72	3.58	3.97	3.99	4.00
Deformed	error	5.38e-02	2.75e-02	2.45e-03	1.04e-04	6.52e-06	4.09e-07
	rate	-	0.97	3.49	4.56	3.99	3.99

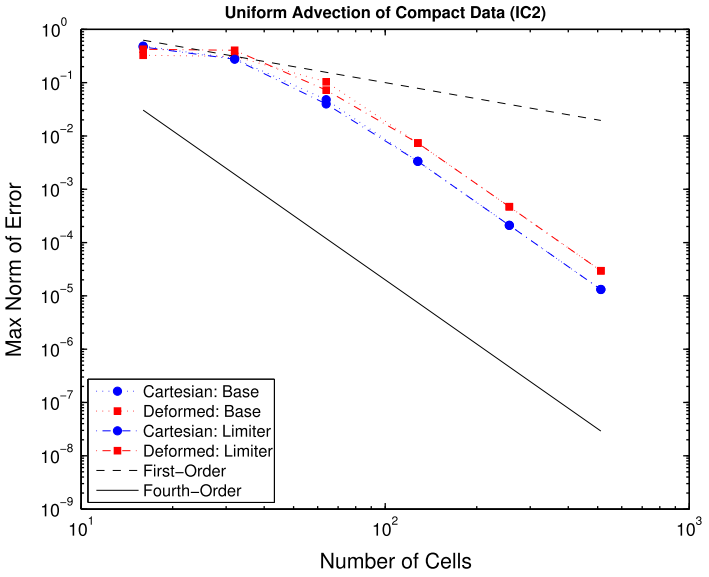
L_1 Error Norm for Base Scheme with Limiter

N_{cells}		16	32	64	128	256	512
Cartesian	error	1.66e-02	5.52e-03	9.57e-04	7.27e-05	4.59e-06	2.87e-07
	rate	-	1.59	2.53	3.72	3.98	4.00
Deformed	error	1.54e-02	6.06e-03	1.08e-03	1.02e-04	6.51e-06	4.10e-07
	rate	-	1.34	2.49	3.40	3.97	3.99

Fig. 11. Convergence of L_1 error norm of cell averages with grid refinement for the base and limited schemes for uniform advection (104) of compactly-supported data (IC3) for both meshes.

bridge slumps. Nevertheless, the shape and size of the slot is fairly well preserved. We see a maximum overshoot of 8.8% that occurs near the lead edges of the cylinder. The overshoot indicates that our modified limiter fails to reduce fully to first-order in the vicinity of the discontinuous cylinder edges. Clearly, the limiter requires further development, but we do note that, despite the overshoot, the limiter is doing a good job of preserving the shape and suppressing oscillations. This can be seen by direct comparison with Fig. 14 in which the solution for the unlimited scheme is plotted. Overall, this is a good approximation for such a coarse resolution, particularly with only only five cells across the slot.

The initial and final solution for simulation on the mapped grid (102) are plotted in Fig. 15. Again, the basic shape of the cylinder is preserved, although there is more smearing than for the Cartesian case. The slot has more fill-in and the bridge is more eroded than on the Cartesian mesh. Of course, the cylinder has traveled through some regions of stretched mesh that make representation of these finer features more difficult. The symmetry of the cylinder is still relatively good, despite the mesh distortions. On the mapped grid, the maximum overshoot is 9.9%, which is slightly higher than for the Cartesian case, but consistent with the limiter behavior on the Cartesian mesh. Thus, while the limiter itself requires improvement to eliminate overshoots at discontinuities, we have successfully shown that such a limiter can be incorporated into our mapped grid formulation.



L_∞ Error Norm for Base Scheme

N_{cells}		16	32	64	128	256	512
Cartesian	error	4.86e-01	2.84e-01	4.80e-02	3.34e-03	2.11e-04	1.32e-05
	rate	-	0.77	2.57	3.85	3.98	4.00
Deformed	error	3.26e-01	3.11e-01	1.04e-01	7.37e-03	4.70e-04	2.96e-05
	rate	-	0.07	1.59	3.81	3.97	3.99

L_∞ Error Norm for Base Scheme with Limiter

N_{cells}		16	32	64	128	256	512
Cartesian	error	4.67e-01	2.75e-01	3.97e-02	3.34e-03	2.11e-04	1.32e-05
	rate	-	0.77	2.79	3.57	3.98	4.00
Deformed	error	4.23e-01	4.04e-01	7.24e-02	7.37e-03	4.70e-04	2.96e-05
	rate	-	0.07	2.48	3.30	3.97	3.99

Fig. 12. Convergence of L_∞ error norm of cell averages with grid refinement for the base and limiter schemes for uniform advection (104) of compactly-supported data (IC3) on both meshes.

5. Summary

We have presented a strategy for the construction of high-order, finite-volume discretizations of flux divergences in mapped coordinates. The approach is based on the systematic development of sufficiently accurate cell face averages, including the computation of product averages in terms of factor averages and/or face-centered values. Among the face averages to be computed are those of the coordinate mapping metric factors, whose calculation affects not only the overall accuracy of the scheme but also freestream preservation. The latter is automatically achieved to machine roundoff by representing the metric factors as exterior derivatives, whose face averages are in turn reduced to quadratures on face hyperedges. The quadratures can be performed by any convenient method of sufficient accuracy.

To demonstrate the approach, we developed fourth-order discretizations of prototypical elliptic and hyperbolic problems. In addition to testing fourth-order accuracy, the elliptic example displayed the use of an operator based on a second-order finite-volume discretization as a preconditioner in a conjugate gradient iteration. Such strategies can be important in reducing the solver cost of the larger stencils that inevitably accompany high-order discretizations. In the hyperbolic examples, we demonstrated how one can include a nonlinear limiter. Our results indicate that our limited method can achieve a fourth-order convergence rate for smooth data on mapped grids and can suppress oscillations for discontinuous data on mapped grids. However, we were not able to demonstrate a limiter that prevented all overshoots at discontinuities. Development

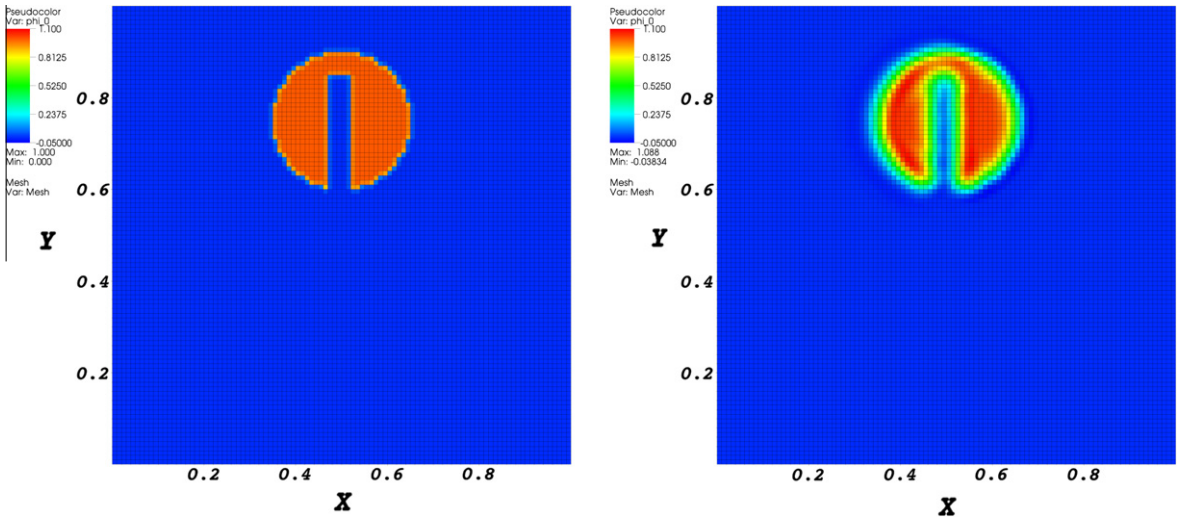


Fig. 13. Circular advection of a slotted cylinder on a 100 × 100 Cartesian grid. The left is the initial condition (and reference solution), and the right is the computed solution after one revolution. The slot and bridge do fill in and slump, respectively, but the overall agreement is good.

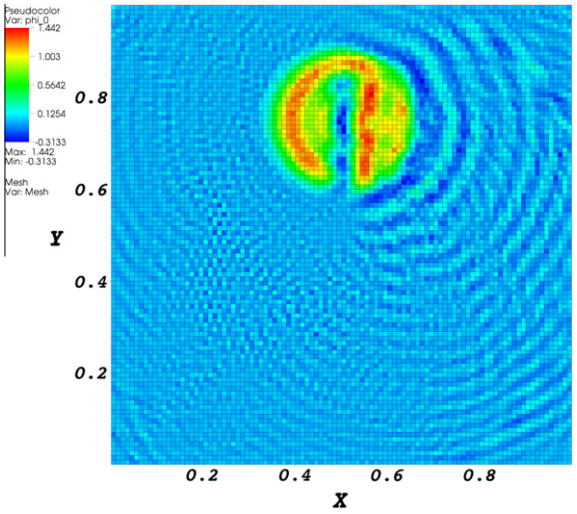


Fig. 14. Circular advection of a slotted cylinder on a 100 × 100 Cartesian grid after one revolution with no limiter. Comparison with limited results indicates the success of the limiter in the preservation of symmetry and the suppression of unphysical oscillations.

of a more robust limiter for fourth-order schemes on mapped grids is beyond the scope of this paper. Recent advances can be found elsewhere [20].

There are a number of possible extensions to the work described here. Two of the most immediate ones, that of the extension to nonlinear hyperbolic systems [20] and to advection–diffusion equations [22], have been carried out for the case of Cartesian grids. In the former case, care is required to make the necessary nonlinear transformations between conserved variables and the fluxes in a way that preserves higher-order accuracy. This is done by transforming average values to point values, applying the nonlinear transformation, and averaging, using formulas similar to Eq. (4). In the latter case, to preserve fourth-order accuracy in time, we use the additive Runge–Kutta method in [23]. On a single grid, both of these approaches can be combined in a straightforward fashion with the mapped approach described here.

In both of the applications [20,22], the methods have been constructed for nested, locally-refined grids. However, the extension to local refinement of our mapped-grid method requires some care. The difficulty is that the geometry of the control volumes changes depending on the refinement level. One possible solution is to generalize to higher order the method in [13], in which the description of the control volumes between levels is forced to be consistent by averaging down the geometric information from the locally finest level. The averaging is done so that accuracy and freestream conditions are simultaneously preserved, particularly on regridding.

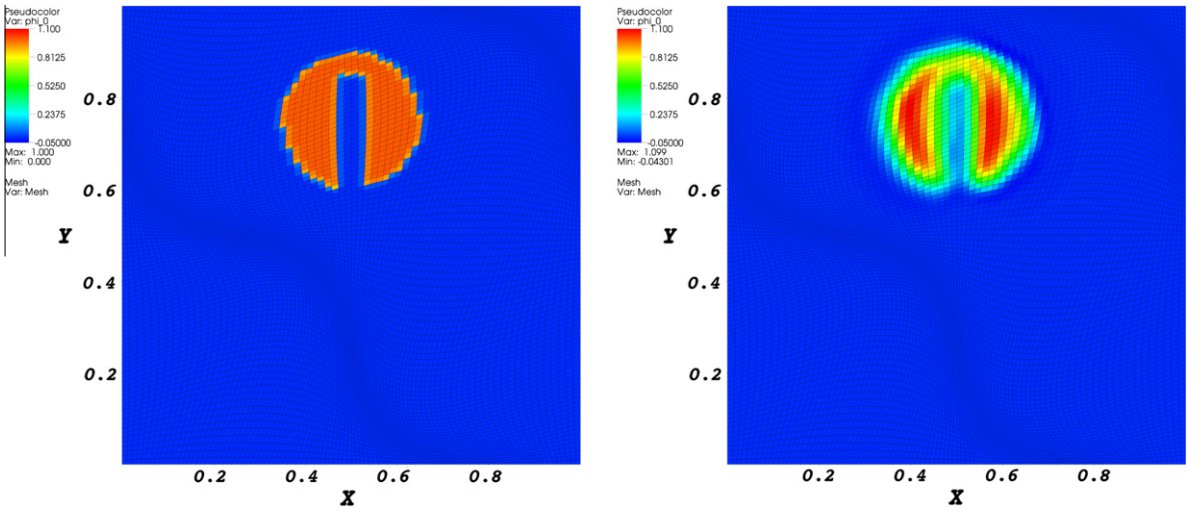


Fig. 15. Circular advection of a slotted cylinder on a 100×100 deformed grid (102). The left is the initial condition (and reference solution), and the right is the computed solution after one revolution. The slot and bridge do fill in and slump, respectively, but the overall agreement is good.

Another possible extension of our approach is to mapped-multiblock grids. In this application, the computational domain in physical space is represented as the disjoint union of images of mappings that are conforming, *i.e.*, aligned at common boundaries in such a way that when the maps are discretized, the individual faces of control volumes at those boundaries coincide. The conforming property makes the imposition of conservation straightforward; however, the maps typically are not smooth at the boundaries, so some care is required in constructing stencils that preserve the high-order accuracy of the fluxes [24].

Moving coordinate systems are yet another possible extension of our approach. The formalism described here applies to any number of space dimensions, so in particular can be used to derive a finite-volume discretization for $\nabla_{(t,\mathbf{x})} \cdot (U, \vec{F}) = 0$ in space–time. If the grid motion is an extrinsically-specified function of time, this presents no obvious difficulties. However, if the grid motion is computed from the solution, it will be necessary to extract from the solution sufficient time-derivative information to construct a smooth mapping in space–time.

Finally, there is the issue of the cost of this approach, relative, for example, to a finite-difference method. We have not made any effort to perform single-processor optimization of the implementations that produced the results here, and therefore it would be inappropriate to attempt to draw any conclusions from these numerical results. We can make some inferences from applications to which we are in the process of applying this method: Euler's equations for compressible flow in 3D and gyrokinetic plasma modeling in two space and two velocity dimensions. In these cases, operation counts indicate the cost of evaluating the averages of the $\mathcal{N}$'s along edges and the other geometric information, even to sixth-order accuracy, is small relative to a single flux evaluation. In the case of compressible flow, this is due to the cost of evaluating the high-order differences and nonlinear fluxes. In the gyrokinetic plasma case, only two of the coordinate directions (out of four) are subject to a nontrivial mapping, and the cost of solving the elliptic gyrokinetic Poisson equation in 2D, which is much less than the advection equation solve in 4D, overwhelms the cost of computing the metric information. Our conclusion supports the notion that, even if the grid moves, the cost of recomputing the metric information will remain a small fraction of the overall cost of the calculation for realistic problems. Although we have not yet designed or implemented a method based on these ideas for accuracy greater than fourth-order, we believe that the cost of computing the metric information will increase proportionally with the cost of computing the high-order fluxes, since the required order of accuracy for the quadratures in both cases increases in lockstep.

References

[1] M. Vinokur, An analysis of finite-difference and finite-volume formulations of conservation laws, J. Comput. Phys. 81 (1989) 1–52.
[2] D. Calhoun, R.J. LeVeque, An accuracy study of mesh refinement on mapped grids, in: T. Plewa (Ed.), Adaptive Mesh Refinement – Theory And Applications: Proceedings of the Chicago Workshop On Adaptive Mesh Refinement Methods, Lecture Notes in Computational Science and Engineering, vol. 41, Springer-Verlag, 2003, pp. 91–102.
[3] D.A. Calhoun, C. Helzel, R.J. Leveque, Logically rectangular grids and finite volume methods for PDEs in circular and spherical domains, SIAM Rev. 50 (4) (2008) 723–752.
[4] M. Barad, P. Colella, A fourth-order accurate local refinement method for Poisson's equation, J. Comput. Phys. 209 (2005) 1–18.
[5] P.D. Thomas, C.K. Lombard, Geometric conservation law and its application to flow computations on moving grids, AIAA J. 17 (10) (1979) 1030–1037.
[6] D.A. Kopriva, Metric identities and the discontinuous spectral element method on curvilinear meshes, SIAM J. Sci. Comput. 26 (3) (2006) 301–327.
[7] M.R. Visbal, D.V. Gaitonde, On the use of high-order finite-difference schemes on curvilinear and deforming meshes, J. Comput. Phys. 181 (1) (2002) 155–185.
[8] M. Spivak, Calculus on Manifolds, W.A. Benjamin, Inc., New York, NY, 1965.

[9] C.H. Evans, J.H. Hawley, Simulation of magnetohydrodynamic flows – a constrained transport method, Astrophys. J. 332 (1988) 659.
[10] J.B. Bell, P. Colella, J.A. Trangenstein, M. Welcome, Adaptive mesh refinement on moving quadrilateral grids, in: Proceedings of the 9th AIAA Computational Fluid Dynamics Conference, Buffalo, NY, 1989, pp. 471–479, aIAA-1989-1979.
[11] R.L. Miller, M.S. Chu, J.M. Greene, Y.R. Lin-Liu, R.E. Waltz, Noncircular, finite aspect ratio, local equilibrium model, Phys. Plasmas 5 (4) (1998) 973–978.
[12] http://www.netlib.org/quadpack/.
[13] P. Colella, D.T. Graves, T.J. Ligocki, D.F. Martin, D. Modiano, D.B. Serafini, B.V. Straalen, Chombo Software Package for AMR Applications – Design Document, <http://seesar.lbl.gov/anag/chombo>.
[14] http://computation.llnl.gov/casc/hypre.
[15] P. Colella, P.R. Woodward, The Piecewise Parabolic Method (PPM) for gas-dynamical simulations, J. Comput. Phys. 54 (1) (1984) 174–201.
[16] J.D. Lambert, Numerical Methods for Ordinary Differential Systems, John Wiley & Sons, New York, 1991.
[17] R.D. Richtmyer, K.W. Morton, Difference Methods for Initial-Value Problems, second ed., John Wiley & Sons, New York, 1967.
[18] B. Gustafsson, H.-O. Kreiss, J. Oliger, Time Dependent Problems and Difference Methods, Pure and Applied Mathematics, John Wiley & Sons, New York, 1995.
[19] P. Colella, M.D. Sekora, A limiter for PPM that preserves accuracy at smooth extrema, J. Comput. Phys. 227 (2008) 7069–7076.
[20] P. McCorquodale, P. Colella, A high-order finite-volume method for hyperbolic conservation laws on locally-refined grids, Comm. Appl. Math. Comput. Sci., in press.
[21] P.-O. Persson, J. Bonet, J. Peraire, Discontinuous Galerkin solution of the Navier–Stokes equations on deformable domains, Comput. Methods Appl. Mech. Eng. 198 (2009) 1585–1595.
[22] Q. Zhang, H. Johansen, P. Colella, A fourth-order accurate finite-volume method with structured adaptive mesh refinement for solving the advection–diffusion equation, J. Comput. Phys., submitted to publication.
[23] C.A. Kennedy, M.H. Carpenter, Additive Runge–Kutta schemes for convection–diffusion–reaction equations, Appl. Numer. Math. 44 (1–2) (2003) 139–181.
[24] P. Colella, M. Dorr, J. Hittinger, P. McCorquodale, D.F. Martin, High-order finite-volume methods on locally-structured grids, in: Proceedings of the Astronum-2008 Meeting, Astronomical Society of the Pacific Conference Series, vol. 406, 2008.