UC Berkeley
UC Berkeley Previously Published Works

Title
High-order finite-volume methods on locally-structured grids

Permalink
https://escholarship.org/uc/item/3tz3m1k5

Journal
Discrete and Continuous Dynamical Systems, 36(8)

ISSN
1078-0947

Author
Colella, Phillip

Publication Date
2016-03-01

DOI
10.3934/dcds.2016.36.4247

Peer reviewed

HIGH-ORDER FINITE-VOLUME METHODS ON LOCALLY-STRUCTURED GRIDS

Phillip Colella [*]

Computational Research Division
Lawrence Berkeley National Laboratory
Berkeley, CA 94720, USA

Abstract. We present an approach to designing arbitrarily high-order finite-volume spatial discretizations on locally-rectangular grids. It is based on the use of a simple class of high-order quadratures for computing the average of fluxes over faces. This approach has the advantage of being a variation on widely-used second-order methods, so that the prior experience in engineering those methods carries over in the higher-order case. Among the issues discussed are the basic design principles for uniform grids, the extension to locally-refined nest grid hierarchies, and the treatment of complex geometries using mapped grids, multiblock grids, and cut-cell representations.

1. **Introduction.** It is often the case that one wants to compute solutions to partial differential equations containing terms of the form $\nabla \cdot \vec{F}$, $\vec{F} = (F^1, \ldots, F^D)$, $F^d = F^d(\boldsymbol{x})$. Classical examples include Poisson's equation

$$\nabla \cdot \vec{F} = \rho, \vec{F} = \nabla \phi, \tag{1}$$

and time-dependent systems of conservation laws

$$\frac{\partial U}{\partial t} + \nabla \cdot \vec{F} = 0, \vec{F} = \vec{F}(U, \nabla U) \; ; \; U, F^d \in \mathbb{R}^M. \tag{2}$$

If $\vec{F} = \vec{F}(U)$, the system is hyperbolic if $\nabla_U(\hat{n} \cdot \vec{F})$ has real eigenvalues and a complete set of eigenvectors for any constant unit vector $\hat{n}$.

One of the oldest and most widely-used spatial discretization methods is the finite-volume method on rectangular grids. In its simplest form, a rectangular spatial domain Ω on which the PDE is to be solved is discretized into a union of control volumes $V_{\boldsymbol{i}}$, $\boldsymbol{i} \in \mathbb{Z}^D$

$$V_{\boldsymbol{i}} = [\boldsymbol{i}h, (\boldsymbol{i} + \boldsymbol{u})h] \; , \; \boldsymbol{i} \in \mathbb{Z}^D \; , \boldsymbol{u} = (1, 1, ..., 1). \tag{3}$$

Averages over control volumes are denoted by $\langle \cdot \rangle_{\boldsymbol{i}}$

$$\langle q \rangle_{\boldsymbol{i}} = \frac{1}{h^D} \int\limits_{V_{\boldsymbol{i}}} q(\boldsymbol{x}) d\boldsymbol{x}. \tag{4}$$

By the divergence theorem, the average of $\nabla \cdot \vec{F}$ over a control volume is equal to the sum of differences of the averages of the normal component of $\vec{F}$ over the faces bounding the control volume

$$\langle \nabla \cdot \vec{F} \rangle_{\boldsymbol{i}} = \frac{1}{h} \sum_d \langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} - \langle F^d \rangle_{\boldsymbol{i}-\frac{1}{2}\boldsymbol{e}^d}, \tag{5}$$

$$\langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} \equiv \frac{1}{h^{D-1}} \int_{A^d_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}} F^d dA^d, A^d_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} = [(\boldsymbol{i}+\boldsymbol{e}^d)h, (\boldsymbol{i}+\boldsymbol{u})h], \tag{6}$$

where $\boldsymbol{e}^d$ is the unit vector in the dth coordinate direction. A numerical approximation to $\langle \nabla \cdot \vec{F} \rangle_{\boldsymbol{i}}$ is obtained by replacing the integrals of the fluxes on the boundary of the control volume with quadratures. The resulting discretization of $\nabla \cdot \vec{F}$ satisfies a discrete form of the divergence theorem: for any union of control volumes Υ, the sum of the volume-weighted averages of $\nabla \cdot \vec{F}$ over the control volumes are equal to the sum of the approximations of the boundary integrals.

$$h^D \sum_{\boldsymbol{i}:V_{\boldsymbol{i}} \subset \Upsilon} \langle \nabla \cdot \vec{F} \rangle_{\boldsymbol{i}} = h^{D-1} \sum_{A^d_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} \subset \partial \Upsilon} \pm \langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}, \tag{7}$$

where $\pm = -, +$ if the boundary face is on the low or high side, respectively of a control volume contained in Υ. To discretize solutions to time-dependent problems, we average (2) over control volumes

$$\frac{d\langle U \rangle_{\boldsymbol{i}}}{dt} = -\frac{1}{h} \sum_d \langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} - \langle F^d \rangle_{\boldsymbol{i}-\frac{1}{2}\boldsymbol{e}^d}, \tag{8}$$

with the numerical fluxes on the right-hand side computed as a function of the $\langle U \rangle$, and the resulting system of ODE discretized in time. Using such a method of lines approach, the fully-discrete system can be written in the form of a single-step update

$$\langle U \rangle_{\boldsymbol{i}}^{n+1} = \langle U \rangle_{\boldsymbol{i}}^n - \frac{\Delta t}{h} \sum_d \langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}^{n+\frac{1}{2}} - \langle F^d \rangle_{\boldsymbol{i}-\frac{1}{2}\boldsymbol{e}^d}^{n+\frac{1}{2}}, \tag{9}$$

where $\langle U \rangle_{\boldsymbol{i}}^n \approx \langle U \rangle_{\boldsymbol{i}}(t^n)$, and $\langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}^{n+\frac{1}{2}}$ is the approximation to

$$\frac{1}{\Delta t} \int_{t^n}^{t^n+\Delta t} \langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}(t) dt \tag{10}$$

implied by the time discretization of (8).

Finite-volume methods were first introduced by Lax in [25, 24] for the purpose of computing discontinuous solutions to hyperbolic conservation laws. He found that the discrete divergence property (7) leads to the discrete traveling waves that satisfy the Rankine-Hugoniot jump relations. This empirical observation was later supported by the theorem in [26]: if the discrete flux function is consistent and the discrete time evolution is given by a one-step discretization of the form (8), then the computed solution converges to a weak solution of (2), provided it converges at all.

In addition to its essential role in computing discontinuous solutions to conservation laws, the finite-volume approach is useful in at least two other settings. One

is the solution to marginally-resolved problems. For example, in low-Mach number combustion problems, it is possible to obtain temperatures that significantly exceed the adiabatic flame temperature if the thermodynamic energy and chemical species are not strictly conserved [35]. The second is in solving Poisson's equation with Neumann or periodic boundary conditions, for which there is a nontrivial null space consisting of constant functions. The resulting solvability condition is manifest at the discrete level in a finite-volume discretization by applying (7) to the entire computational domain.

For more than 60 years, finite volume methods have been a mainstay of applied numerical PDE, with application to complex multiphysics problems, accurate and robust treatment of discontinuities, and the use of elaborate gridding strategies, both for representing complex geometries and for solution adaptivity. However, up until about a decade ago, they were almost exclusively second-order accurate methods, at least for multidimensional problems. In the context of the rectangular grid discretization (5), this typically corresponds to the use of the midpoint rule as the quadrature method for approximating the average of the fluxes over the faces. There are a combination of software complexity and algorithmic reasons for this. One is that, for complex multiphysics problems with multiple time scales, models for different physical processes are coupled together using operator-splitting methods that are at most second-order accurate. This permits the use of different combinations of implicit or explicit time discretization methods, depending on degree of stiffness of the underlying submodel relative to the time scales of interest. Operator splitting also allows for a software design strategy that permits straightforward insertion and removal of software components to simulate different physical processes. A second issue is that the second-order spatial discretizations are robust: they lead to M-matrices for elliptic and parabolic problems; for hyperbolic problems, there are well-established strategies for suppressing oscillations at discontinuities and underresolved regions, while remaining second-order accurate in regions where the solution is smooth. More recently, this point of view has begun to change, for a number of reasons.

- In certain specialized fields such as fundamental studies of turbulent flows high-order methods (fourth- and higher-order) have been shown to be necessary in regimes in which scientists would like to simulate more general complex problems.
- There are new methods for time discretization [22, 9] that are higher-order accurate, while retaining the other desirable algorithmic and software properties of operator splitting.
- Finite-volume methods typically suffer a loss of one order of accuracy in truncation error at places where the mesh fails to be smooth, such as refinement boundaries in structured adaptive mesh refinement, block boundaries in multiblock methods, and near the irregular control volumes arising in cut-cell approaches to complex geometries. Dropping from second-order accuracy to first order accuracy, even on a set of codimension one, can lead to a significant increase in the error; the relative impact of the loss of one order of accuracy is less as the order of accuracy of the discretization increases.
- Processors used in high-performance computers have been undergoing disruptive changes in their architectures [31, 40]: the ratio of total memory capacity to aggregate floating point capability is decreasing, and the cost (in time and power consumption) of data movement is increasing relative to that of

4250   PHILLIP COLELLA

performing float point operations. Higher-order accurate methods could be an effective response to these changes. They typically perform more floating point operations per unit of data traffic, and they have the potential to produce a result with a given level of accuracy with fewer computational degrees of freedom, at nearly the same throughput as a low-order method [3].

In what follows, we describe one specific approach to designing arbitrarily high-order finite-volume spatial discretizations on locally-rectangular grids. It is based on the use of a particularly simple class of high-order quadratures for computing the average of fluxes over faces. This approach has the advantage of being a variation on widely-used second-order methods, so that the prior experience in engineering those methods carries over to the higher-order case. Some of the design issues include:

- High-order accurate methods on a Cartesian grid. Given averages of the solution over control volumes, how does one compute high-order accurate averages of nonlinear functions of the solution and / or its derivatives over faces ? For time-dependent hyperbolic problems, another set of issues are the development of limiters and other dissipation mechanisms to maintain stability, particularly at discontinuities and underresolved regions, while preserving high-order accuracy for resolved smooth solutions.
- Adaptive mesh refinement. Some aspects of adaptive mesh refinement on nested rectangular grids are made simpler by the fact that the averages (4), (6) on a coarse grid can be computed exactly by arithmetic averaging of the corresponding quantities defined on a nested refined grid. This allows the use of uniform-grid methods to interpolate boundary conditions at refinement boundaries by averaging the fine grid solution onto the coarse grid, while still maintaining high-order accuracy. In the case of time-dependent problems in which different time steps are used at different levels of refinement, it is essential to perform interpolation in time at refinement boundaries without having to keep the solution history at multiple time levels.
- Complex geometries. Mapped grids, multiblock grids, and cut-cell representations of geometry are all widely employed to handle various kinds of geometric complexity. Extending them to higher-order accuracy involves resolving a number of new conceptual and technical issues.

2. High-order quadratures for fluxes.

2.1. Cartesian grids. We will use time-dependent conservation laws of the form (2) as a motivating example. If we discretize first in space (8), we can use a high-order ODE solution algorithm to solve the resulting system of ODE, such as an explicit or semi-implicit Runge-Kutta method. In that case, it suffices to be able to compute a high-order approximation to the flux averages on faces, given the values of $\langle U \rangle$.

We first consider the case $\vec{F}(U) = \vec{C}U$, C^d each a constant $M \times M$ matrix. In one space dimension, we note [16, 43] that the averages over cells provide values for the indefinite integral of U evaluated at cell faces

$$\mathcal{U}(x) \equiv \int^x U(x)dx \to \mathcal{U}_{i+\frac{1}{2}} = h \sum_{i' \le i} \langle U \rangle_{i'}, \tag{11}$$

$$\frac{d\mathcal{U}}{dx} = U \to U_{i+\frac{1}{2}} = (\mathcal{D}(\mathcal{U}))_{i+\frac{1}{2}}, \tag{12}$$

Where $\mathcal{D}$ is a finite-difference approximation to the spatial derivative of any order of accuracy. For example, if $\mathcal{D}$ is the classical fourth-order centered-difference approximation to ∂_x, then

$$U_{i+\frac{1}{2}} = (\mathcal{D}\mathcal{U})_{i+\frac{1}{2}} = \frac{7}{12}(\langle U \rangle_i + \langle U \rangle_{i+1}) - \frac{1}{12}(\langle U \rangle_{i-1} + \langle U \rangle_{i+2}) + O(h^4), \quad (13)$$

$$\langle F^1 \rangle_{i+\frac{1}{2}} = C^1 \langle U \rangle_{i+\frac{1}{2}}. \quad (14)$$

This reasoning applies to constant coefficient systems in higher dimensions without modification. If $\mathcal{U}(x_d)$ is the indefinite integral of the average of U across rectangular patches aligned with a row of the faces $\{A^d_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} : i_{d'} = i^0_{d'}, d' \neq d\}$, then

$$\mathcal{U}(x_d) = \int^{x_d} U(x'_d)dx'_d \rightarrow \mathcal{U}_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} = h \sum_{\substack{\boldsymbol{i}:i'_d \leq i_d, \\ i'_{d'}=i^0_{d'}, d'\neq d}} \langle U \rangle_{\boldsymbol{i}'}, \quad (15)$$

$$\langle U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} = \frac{7}{12}(\langle U \rangle_{\boldsymbol{i}} + \langle U \rangle_{\boldsymbol{i}+\boldsymbol{e}^d}) - \frac{1}{12}(\langle U \rangle_{\boldsymbol{i}-\boldsymbol{e}^d} + \langle U \rangle_{\boldsymbol{i}+2\boldsymbol{e}^d}) + O(h^4), \quad (16)$$

$$\langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} = C^d \langle U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} + O(h^4). \quad (17)$$

We can use a similar approach to compute fluxes that are a linear combination with constant coefficients of $\langle \nabla U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$. For the derivative of U in the direction normal to the face, we use

$$\langle \partial_{x_d} U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} = \langle \partial^2_{x_d} \mathcal{U} \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}, \quad (18)$$

and approximate $\partial^2_{x_d}$ by a suitably high-order difference approximation. For the case of $\langle \partial_{x_{d'}} U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$, $d' \neq d$, we first compute $\langle U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ using (12). We then compute $\langle \partial_{x_{d'}} U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ using the exact relationship

$$\left\langle \frac{\partial U}{\partial x_{d'}} \right\rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} = \frac{1}{h}\big(\langle U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d+\frac{1}{2}\boldsymbol{e}^{d'}} - \langle U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d-\frac{1}{2}\boldsymbol{e}^{d'}}\big), \quad (19)$$

and approximate the averages of U over (d, d') edges of the grid on the right-hand side by applying (12) to the face averages of U.

For nonlinear problems, we have to do something more complicated, due to the fact that $\langle F(U) \rangle = F(\langle U \rangle) + O(h^2)$. However, applying a nonlinear function to U evaluated at a point has zero truncation error. So we need to be able to easily compute point values from average values, and conversely. This is easily done, to any order of accuracy. We denote by $q_{\boldsymbol{i}}$ the value of q at the center of a control volume. Then

$$q_{\boldsymbol{i}} = \langle q \rangle_{\boldsymbol{i}} - \frac{h^2}{24} \sum_d \frac{\partial^2 q}{\partial x_d^2} - \frac{h^4}{5760}\Big(3 \sum_d \frac{\partial^4 q}{\partial x_d^4} + 10 \sum_{d_1 > d_2} \frac{\partial^4 q}{\partial x_{d_1}^2 \partial x_{d_2}^2}\Big) + \dots \quad (20)$$

It is tedious, but straightforward, to apply the approach given above to compute the derivatives in the right hand side to the necessary order - the process can be automated using a symbolic algebra package. These formulas apply equally well to transform between face averages and point values at the centers of faces, except that the sums over the derivatives in the correction terms exclude the derivatives with respect to the direction normal to the face over which one is averaging.

A special case that often arises is the calculation of the divergence where the fluxes are the product of two spatially-varying functions, e.g. the advection operator $\nabla \cdot (\vec{u}q)$, $\vec{u} = \vec{u}(\boldsymbol{x})$. In that case, it is useful to express the average of the product

as the product of the averages plus higher order correction terms. For fourth-order accuracy, such a formula is given by

$$\langle fg \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} = \langle f \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} \langle g \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} + \tag{21}$$
$$\frac{h^2}{12} \sum_{d' \neq d} (D_{d'} \langle f \rangle)_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} (D_{d'} \langle g \rangle)_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} + O(h^4),$$

where $D_{d'}$ the two-point centered-difference approximation to $\partial_{d'}$ at $\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d$. There are further corrections that extend this to any order of accuracy.

To see how this works in practice, we outline how to apply this to equations in which the flux $\vec{F} = \vec{F}(W, \nabla W)$, where $W = W(U)$ is a nonlinear function of U. This is typical of nonlinear fluid equations, both in the hyperbolic case [28], where one may want to perform a change of variables to compute limiters; or in the presence of diffusive transport, in which case the diffusive contributions to the fluxes are given in terms of gradients of the derived quantities (velocity, temperature), rather than the conserved variables (momentum, total energy). Given $\langle U \rangle_{\boldsymbol{i}}$, we want to compute a fourth-order accurate approximation to $\langle F^d(W, \nabla W) \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$. We do so in the following steps.

1. Given $\langle U \rangle_{\boldsymbol{i}}$, compute $\langle W \rangle_{\boldsymbol{i}}$ using (20).

$$U_{\boldsymbol{i}} = \langle U \rangle_{\boldsymbol{i}} - \frac{h^2}{24} \sum_d (D_d^2 \langle U \rangle)_{\boldsymbol{i}}, \tag{22}$$
$$W_{\boldsymbol{i}} = W(U_{\boldsymbol{i}}), \overline{W}_{\boldsymbol{i}} = W(\langle U \rangle_{\boldsymbol{i}}),$$
$$\langle W \rangle_{\boldsymbol{i}} = W_{\boldsymbol{i}} + \frac{h^2}{24} \sum_d (D_d^2 \overline{W})_{\boldsymbol{i}},$$

   where D_d^2 is the three-point centered-difference approximation to $\partial_{x_d}^2$. Note that we use the average of U and W applied to the average of U to compute the $O(h^2)$ correction terms in this step. This keeps the size of the stencil to a minimum, while still maintaining fourth-order accuracy. It also guarantees that, if $W(U) = U$, then $\langle W \rangle = \langle U \rangle$.

2. Compute $\langle W \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$, $\langle \nabla W \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ given $\langle W \rangle_{\boldsymbol{i}}$, using the methods described above.

3. We compute the fluxes by evaluating the flux at the point value of W, then apply (20) again to obtain the average over the face. We use a similar second-order approximation to derivatives in the $O(h^2)$ correction in the final step as used in (22) to control the diameter of the stencil.

$$W_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} = \langle W \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} - \frac{h^2}{24} \sum_{d' \neq d} (D_{d'}^2 \langle W \rangle)_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}, \tag{23}$$

$$\nabla W_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} = \langle \nabla W \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} - \frac{h^2}{24} \sum_{d' \neq d} (D_{d'}^2 \langle \nabla W \rangle)_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}, \tag{24}$$

$$\langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} = F^d(W_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}, \nabla W_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}) + \frac{h^2}{24} \sum_{d' \neq d} (D_{d'}^2 \langle \nabla \overline{F}^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}), \tag{25}$$

$$\overline{F}^d = F^d(\langle W \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}, \langle \nabla W \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}). \tag{26}$$

Given that $\langle U \rangle_{\boldsymbol{i}}$ is the exact value for the average of U over the control volume, and that U is a smooth function of $\boldsymbol{x}$ and $\vec{F}$ is a smooth function of U, the error $\langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ as an approximation to the face average of the fluxes is $O(h^4)$, where the coefficient multiplying the leading-order term of the error is a smooth function of space. When this is substituted into the right-hand side of (7), we see that the truncation error of the divergence is $O(h^4)$, due to the usual cancellation of smoothly-varying errors on successive faces.

In the discussion above, we have focused mainly on the truncation error of these methods. With respect to stability, we can use discrete Fourier analysis to prove von Neumann stability in the constant coefficient case. For the higher-order extensions described here of the classical second-order methods for second-order elliptic operators with smooth coefficients, we obtain discretizations for which geometric multigrid converges in $O(1)$ iterations [2]. Boundary-value problems have the same well-posedness and solvability properties as for the second-order case. For hyperbolic problems, the interaction between accuracy and stability / robustness is more delicate, and is discussed in Section 2.3 below.

2.2. **Mapped coordinates.** We assume that we want to compute solutions to conservation laws on a domain that is the image of a smooth mapping from the unit cube in an abstract coordinate space into physical space [14]

$$\boldsymbol{X}(\boldsymbol{\xi}) = \boldsymbol{X} : [0,1]^D \to \mathbb{R}^D. \tag{27}$$

In that case, the divergence of fluxes in physical space can be written in terms of a divergance in the mapping space.

$$\nabla_{\boldsymbol{x}} \cdot \vec{F} = \frac{1}{J} \nabla_{\boldsymbol{\xi}} \cdot (\boldsymbol{N}^T \vec{F}), \tag{28}$$

$$J = det(\nabla_{\xi} \boldsymbol{X}) = \nabla_{\boldsymbol{\xi}} \cdot (D^{-1} \boldsymbol{N}^T \boldsymbol{X}), \tag{29}$$

and $\boldsymbol{N}$ is the cofactor matrix of $\nabla_{\boldsymbol{\xi}} \boldsymbol{X}$. Conservation laws are discretized in terms of averages of $\nabla_{\boldsymbol{x}} \vec{F}$ over images of the control volumes in the coordinate space.

$$\frac{1}{h^D} \int\limits_{\boldsymbol{X}(V_{\boldsymbol{i}})} \nabla_{\boldsymbol{x}} \cdot \vec{F} d\boldsymbol{x} = \frac{1}{h^D} \int\limits_{V_{\boldsymbol{i}}} \nabla_{\boldsymbol{\xi}} \cdot (\boldsymbol{N}^T \vec{F}) d\boldsymbol{\xi} \tag{30}$$

$$= \frac{1}{h} \sum_d \langle (\boldsymbol{N}^T \vec{F})^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} - \langle (\boldsymbol{N}^T \vec{F})^d \rangle_{\boldsymbol{i}-\frac{1}{2}\boldsymbol{e}^d}, \tag{31}$$

where h is the mesh spacing in the coordinate space, and $\langle \cdot \rangle_{\boldsymbol{i}}$, $\langle \cdot \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ denote the averages over control volumes (4) and faces (6) in the coordinate space. Using the change of variables formula we have converted the average of $\nabla_{\boldsymbol{x}} \cdot \vec{F}$ into the average of a divergence of fluxes over cubic control volumes in the coordinate space, so that the quadratures in the previous section applies here. In particular, we can use product formulas such as (21) to compute $\langle (\boldsymbol{N}^T \vec{F})^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ to fourth- or higher-order accuracy

$$\langle (\boldsymbol{N}^T \vec{F}) \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}^d = (\langle \boldsymbol{N}^T \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} \langle \vec{F} \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d})^d + \frac{h^2}{12} \sum_{d' \neq d} D_{d'} (\langle \boldsymbol{N}^T \rangle D_{d'} \langle \vec{F} \rangle)_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}^d. \tag{32}$$

For time-dependent problems (2), we can use the change of variables formula to obtain an evolution equation for averages over the control volumes in the coordinate

space

$$\frac{d\langle JU \rangle_{\boldsymbol{i}}}{dt} = -\frac{1}{h} \sum_{d=1}^{D} \langle (\boldsymbol{N}^T \vec{F})^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} - \langle (\boldsymbol{N}^T \vec{F})^d \rangle_{\boldsymbol{i}-\frac{1}{2}\boldsymbol{e}^d}. \tag{33}$$

If $\vec{F}$ is identically equal to a constant as a function of space, then $\nabla_{\boldsymbol{x}} \cdot \vec{F} \equiv 0$. We want to preserve this property under discretization. This requirement is often referred to as freestream preservation. While this is trivially satisfied for the Cartesian discretization (5), it imposes a nontrivial constraint on the calculation of $\langle N \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ for the right-hand side of (31) to vanish. If we use (32) to compute the face-averaged fluxes, then the higher-order terms vanish if $\vec{F}$ is a constant, and (31) becomes a constraint on the quadrature rule used to compute $\langle N \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ that, if met, is sufficient to guarantee freestream preservation. By equality of mixed partials, the rows of $\boldsymbol{N}$ $\{N^s\}_{s=1\dots D}$ satisfy $\nabla_{\boldsymbol{\xi}} \cdot N^s = 0$. By the Poincare lemma, there exists $\mathcal{N}_{d,d'}^s$ such that

$$N_d^s = \sum_{d' \neq d} \frac{\partial \mathcal{N}_{d,d'}^s}{\partial \xi_{d'}}, \mathcal{N}_{d,d'}^s = -\mathcal{N}_{d',d}^s. \tag{34}$$

We can then use Stokes' theorem to compute the average of N^s over faces in terms of the integrals of $\mathcal{N}$ over edges

$$\langle N_d^s \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} = \frac{1}{h^{D-1}} \sum_{\pm=+,-} \sum_{d' \neq d} \pm \int_{E_{d,d'}^{\pm}} \mathcal{N}_{d,d'}^s dE_{\boldsymbol{\xi}}. \tag{35}$$

where $E_{d,d'}^{\pm}$ are the low $(-)$ and high $(+)$ edges of $A_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}^d$ corresponding to both ξ_d and $\xi_{d'}$ equal to constants. To obtain freestream preservation and p^{th}-order accuracy, we can compute $\langle N^s \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ using the right-hand side of (34) with the edge integrals replaced by any quadrature rule that is p^{th} order accurate, provided that they satisfy the antisymmetry condition corresponding to $\mathcal{N}_{d,d'}^s = -\mathcal{N}_{d',d}^s$. To complete the construction, we note that, while $\mathcal{N}$ is not unique, there is a distinguished choice that is a local function of $\boldsymbol{X}, \nabla_{\boldsymbol{\xi}}\boldsymbol{X}$.

Given (34), the extension of the approach used for Cartesian grids in the previous section to the mapped-grid setting is relatively straightforward. For example, to compute the right-hand side of (33) corresponding to (22)-(26), one can use a cell-centered version of the product formula (21) to make the transformation $\langle JU \rangle_{\boldsymbol{i}} \to \langle U \rangle_{\boldsymbol{i}}$; and compute $\langle U \rangle_{\boldsymbol{i}} \to \langle W \rangle_{\boldsymbol{i}} \to \langle W \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}, \langle \nabla_{\boldsymbol{x}} W \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} \to \langle \vec{F} \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$. All of the intermediate steps here use the Cartesian methods in (22)-(26), without modification, except for the calculation of $\langle \nabla_{\boldsymbol{x}} W \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$. In that case, we use the chain rule $\nabla_{\boldsymbol{x}} W = J^{-1} \boldsymbol{N} \nabla_{\boldsymbol{\xi}} W$ combined with the product formula (21). We then use the product formula (32) to compute the average of the fluxes at faces.

2.3. Dissipation mechanisms, limiters, and positivity preservation for hyperbolic problems.

Dissipative Methods. Methods such as (14) that use stencils that are centered around the face where the value is being computed correspond to even-order, centered-difference approximations. For constant-coefficient problems, as the wavelength of a Fourier mode approaches $2h$, the propagation speed of the mode approaches zero independent of the exact wave propagation velocity for the PDE, while

the amplitude of the Fourier mode remains unchanged. While these methods are still Fourier stable, the near-neutral stability and large phase error of short-wavelength modes can lead to various degrees of instability when applied to variable-coefficient and nonlinear problems. For this reason, it was suggested in [23] that such methods be modified so as to be dissipative: in the long-wavelength limit, the order of accuracy of the method remains unchanged, while reducing the amplitude of all nonconstant Fourier modes. This can be done in one of two ways. The approach proposed in [23] is to add to the flux in the d direction of a $2p^{th}$-order method a term of the form $Ch^{2p+1}\partial_{x_d}^{2p+1}U$, where $C = C(\boldsymbol{x}) = O(1)$ has units of velocity, with $(-1)^{p+1}C > 0$. This corresponds to adding a dissipation term to the right-hand side of (8). An alternative approach is to use an $(2p + 1)^{st}$-order method with a stencil for the flux at a face centered around the cell upwind to that face relative to the propagation speed. Such a one-point upstream-centered method has a truncation error that is instrinsically dissipative, damping all nonconstant Fourier modes, including the high-wavenumber ones. For systems of equations, one computes one-point upstream-centered face values centered on both cells adjacent to the face, and then uses an approximate Riemann solver to synthesize a upwind state relative to the different wave propagation speeds. Either approach provides a suitable mechanism for damping neutrally-stable high-wavenumber modes, at least in one dimension.

There are additional complications in more than one space dimension. For solutions that depend on only one space variable, it is possible for modes with wavenumbers perpendicular to that direction of propagation to be neutrally stable, and dissipation methods based on upwinding introduce no dissipation in the transverse directions. The complete lack of dissipation in those settings can lead to nonlinear instabilities, as observed, for example, in [41] for strong shocks aligned with one of the coordinate directions. This instability is eliminated by modifying the nonlinear diffusion term introduced near shocks so that the diffusion coefficient has the same magnitude in all of the coordinate directions. This has the effect of damping the nonlinear instability transverse to the shock, while maintaining the advertised level of accuracy in smooth regions [16, 28]. This suggests that any dissipation that is introduced to stabilize short-wavelength modes should introduce damping in all coordinate directions appropriate to the desired level of accuracy.

Limiters. For hyperbolic conservation laws, there is a fundamental problem to representing discontinuities as discrete traveling waves on a grid. In the neighborhood of the discontinuity, it is necessary to add a regularization term in the form of a numerical viscosity with a coefficient that is $O(h)$ in the neighborhood of the discontinuity [32, 24]. Such a term being applied everywhere leads to large errors in regions where the solution is smooth or for linearly degenerate discontinuities. This was recognized as a problem early on in the development of methods for discontinuities, and led to a variety of approaches for tuning the artificial viscosity so that it was sufficiently large at nonlinear discontinuities, and small in smooth regions. Early methods of this type required the setting of various parameters in the viscous terms that were problem-dependent. A more robust class of solutions, called flux limiters, were first introduced in the 1970's [8, 39, 42] to address this issue. The simplest version of this approach is the Flux-Corrected Transport (FCT) limiter in which the time-averaged flux in (9) is given by

$$\langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}^{n+\frac{1}{2}} = \langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}^{H} + \eta_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} \langle \delta F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}, \tag{36}$$

$$\langle \delta F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} \equiv \langle F^d \rangle^L_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} - \langle F^d \rangle^H_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d},$$

where $\langle F^d \rangle^H_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ is a time-averaged flux computed using a high-order method such as the one described above, $\langle F^d \rangle^L_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ is a flux computed using a stable first-order accurate method, such as donor-cell differencing, or the CTU method in [12, 36]. The quantity $\eta_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} \in [0,1]$ is a hybridization coefficient: if $\eta_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} = 0$, we are using the high-order accurate flux; if $\eta_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} = 1$, the first-order accurate flux. To understand the idea behind this approach, consider the case of a scalar equation, i.e., $U(\boldsymbol{x},t) \in \mathbb{R}$. We note that, by modified equation analysis, the correction term $\delta F^d \sim \alpha_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}(\langle U \rangle_{\boldsymbol{i}} - \langle U \rangle_{\boldsymbol{i}+\boldsymbol{e}^d})$, $\alpha_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} = O(1)$, $\alpha_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} \geq 0$, i.e. like an $O(h)$ diffusion term. The FCT approach in [42] is to compute η to be as small as possible subject to the conditions that the solution remains within prescribed bounds, e.g. the componentwise max and min of in the neighborhood of $\boldsymbol{i}$ after the low-order method has been applied for one time step, combined with the max and min of $\langle U \rangle^n$ in the same neighborhood.

Geometric limiting [39] is an approach that is specific to the interpolation-based evaluation of the fluxes. In the semi-discrete setting considered here, we apply the limiter at each evaluation of the right-hand side. For a scalar, we replace a single-valued calculation of $\langle U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ at the face with a double-valued one

$$\begin{aligned} \langle U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d, L} &= \langle U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} + \eta_{\boldsymbol{i}}(\langle U \rangle_{\boldsymbol{i}} - \langle U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}), \\ \langle U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d, R} &= \langle U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} + \eta_{\boldsymbol{i}+\boldsymbol{e}^d}(\langle U \rangle_{\boldsymbol{i}+\boldsymbol{e}^d} - \langle U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}), \end{aligned} \tag{37}$$

where $\langle U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ is computed using (11). The single-valued flux is computed as an approximate solution to the Riemann problem for the 1D equation obtained from assuming that the solution is constant in the coordinate directions $d' \neq d$ is constant. In the case where the sign of the wave speed is unambiguous, this simply corresponds to choosing the upwind value of (37). The prescription in [39, 16] for computing the hybridization coefficient $\eta_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ is based on assuming an interpolating polynomial derived from the $\langle U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ does not introduce any new extrema not already present in the values $\langle U \rangle_{\boldsymbol{i}}$. In the case where $\eta \equiv 1$, we are replacing the high-order method for computing the flux with a first-order upwind discretization in space, which introduces adequate dissipation at discontinuities. For both geometric limiting and FCT, additional conditions and dissipation mechanisms need to be applied, e.g. to enforce entropy conditions and obtain robust treatment of strong nonlinearities.

In order to obtain high-order accuracy for smooth solutions for h sufficiently small using either of these approaches, this choice of bounds assumes that the exact $\langle U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ can be bounded by local values of $\langle U \rangle$. This fails at smooth extrema, where it is possible for $\langle U \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}(t)$ to not be within the range of the local values of $\langle U \rangle$. If that occurs, $\eta > 0$ may be nonzero at some faces. This leads to a low-order update to the solution, and typically a method that has $O(h)$ errors in the solution at smooth extrema. This was recognized in [42], and a method for modifying such limiters to eliminate this loss of accuracy was proposed there. One way of summarizing this approach is given by the following [15, 28, 10].

- If there is no possibility of a smooth extremum, the solution is bounded by values of $\langle U \rangle$ in the neighborhood of the control volume, and and we use the limiters described above.

- In regions where there is a possibility of a smooth extremum, we use a different limiter, based on comparing various approximations to the *second* derivatives of the solution, and applying a limiter if they differ by $O(1)$, or change sign.

This is a restatement in algebraic terms of the extremum-preserving method in [42], in the spirit of [39]. In the latter work, the hybridization coefficient in one dimension is a function of the ratios of several finite difference approximations to first derivative. The coefficient η is nonzero if these ratios are not sufficiently close to 1 or the estimates of the first derivative do not have a consistent sign. Using constraints based on the relative sizes of first derivatives are reliable indicators of whether the solution is sufficiently well-resolved on the grid so that the use of higher-order methods can be used, except at locations where the first derivative vanishes and the solution is not discontinuous. In that case, we use the same strategy, except based on several finite-difference approximations to the second derivative. In one dimension, this is the generic case, in the sense that a function with extremum at which both of the first two derivatives vanish (e.g. a quartic) is transformed by small perturbations into a solution in which there are two non-degenerate extrema. Thus this approach to limiting using the second-derivative condition will preserve high-order accuracy for any order method. The extension to multiple dimensions is done by using the derivatives in each of the coordinate directions, with additional conditions where the solution varies in only one of the coordinate directions, or in the neighborhood of points where the solution is sufficiently close to a cubic [28]. In any case, the cost of applying these more elaborate limiters at extrema are small, since they are only computed at locations where there is a possibility of a smooth extremum, a condition which is inexpensive to compute, and occurs typically on a set of codimension one.

There are two possible approaches to extending the limiters described above to the case of systems of equations. The first is to apply the limiter componentwise, possibly after first performing a nonlinear change of variables, e.g. transforming from conserved variables to primitive variables in gas dynamics. This is a straightforward generalization, but can fail for more complex systems, such as magnetohydrodynamics. The second approach is to apply the limiters in characteristic variables. To do this, one expands either $\langle \delta F^d \rangle_{i + \frac{1}{2} e^d}$ in (36) or $(\langle U \rangle_i - \langle U \rangle_{i + \frac{1}{2} e^d})$ in (37) as a sum of right eigenvectors, compute a limiting coefficient for each amplitude, and resum the expansion with the corrected amplitudes to obtain the limited values. In the scalar case, these coefficients are obtained as nonlinear combinations of differences. For characteristic limiting, the limiter for the amplitude of the kth eigenvector is computed as the same nonlinear combination of the kth expansion coefficient of the same derivatives as used in the scalar case. While this is a sounder approach theoretically, it also must be done with care, due to the nonlinear dependence of the eigenvectors on the solution. All of the expansions in characteristic variables used to compute a given limiting coefficient must be done with respect to eigenvectors evaluated at the same value of the solution.

Positivity Preservation. Solutions to the advection equation remain nonnegative for all times if they are initially nonnegative. It is desirable that this property be preserved under discretization, for example, if other terms in a system of equation are not defined unless an advected quantity on which they depend is nonnegative. Such a property of a discretization is called positivity preservation. In the initial development of limiters in one dimension, positivity preservation was considered one of the design goals. In more than one dimension, this becomes problematic.

The only classical positivity-preserving linear low-order finite-volume method for advection by an arbitrary spatially-varying, discretely-divergence-free velocity field is donor-cell differencing, which has a CFL constraint that scales like D^{-1}. In high dimensions, such as for kinetic problems in 6D phase space, this is a significant time step penalty. A second issue is that positivity preservation constrains the design space for extremum-preserving limiters, even in one dimension: in order to allow for smooth extrema to be preserved, it is necessary to relax constraints on the max and min of the discrete solution so that small violations of the bounds are introduced. An alternative that has seen some success has been to use a separate post-processing step after each time step to preserve positivity (or other bounds satisfied by the PDE) [33, 11, 20]. At the end of every time step, one computes the increments in each control volume corresponding to the extent to which positivity has been violated. Those increments are then redistributed to nearby control volumes, with weights proportional to the mass of the quantity in that control volume, and with weights summing to one. If at the beginning of the time step, the averages were all nonnegative, there is some nearby control volumes that have a surplus of mass corresponding to the negative increment being redistributed. Thus the advected quantity is conserved, and positivity is preserved. Our experience is that, in the presence of discontinuities, redistribution cannot be used by itself, but must be applied to solutions to which limiting is also applied.

3. **Adaptive mesh refinement.** In the block-structured finite-volume adaptive mesh refinement algorithm in [5] the solution to a system of conservation laws is represented on a hierarchy of locally-refined grids. At each level of refinement $\ell = 0, \ldots, \ell_{max}$, a subset Ω^l of the spatial domain Ω is discretized into a union of control volumes of the form (3), which are in turn organized into a disjoint union of rectangles $\bigcup_k \Omega^{\ell,k} = \Omega^\ell$ (figure 1). The grids are assumed to be nested: $\Omega^\ell \subset \Omega^{\ell-1}$, $h^{\ell-1} = h^\ell n_{ref}^\ell$, where $n_{ref}^\ell \in \mathbb{N}_+$ is the refinement ratio. The control volumes defined by these grids are assumed to be nested, i.e. that any control volume in Ω^ℓ that intersects $\Omega^{\ell+1}$ is completely covered by $\Omega^{\ell+1}$. The additional requirement of proper nesting is usually imposed, i.e. that the distance between the boundary of $\Omega^{\ell+1}$ and any control volume in $\Omega^{\ell-1}$ not covered by Ω^ℓ is at least some integer multiple of h^ℓ. This proper nesting distance is chosen to make the calculation of $\langle \nabla \cdot \vec{F} \rangle$ on Ω^ℓ use only those values defined on $\Omega^{\{\ell,\ell\pm1\}}$.

For a Cartesian grid hierarchy, the calculation of $\langle \nabla \cdot \vec{F} \rangle$ on $\bigcup_\ell \Omega^\ell$, where $\vec{F} = \vec{F}(U, \nabla U)$, assuming $\langle U^\ell \rangle : \Omega^\ell \to \mathbb{R}^M$ is known, is performed as follows, with the computation being done on each level in the order $\ell = \ell_{max}, \ldots, 0$.

1. For each patch $\Omega^{\ell,k}$, compute $\langle U \rangle$ for a slightly larger patch containing $\Omega^{\ell,k}$, copying values from other patches in Ω^ℓ if they exist, otherwise interpolating values of $\langle U \rangle^\ell$ from $\Omega^{\ell-1}$. Then compute the fluxes using the approach in (22) - (26).
2. Assuming that the fluxes $\langle F^d \rangle_{i+\frac{1}{2}e^d}$ on $\Omega^{\ell+1}$, have already been computed as in Step 1, replace the fluxes on the faces of the control volumes in Ω^ℓ covered by $\partial \Omega^{\ell+1}$ by the arithmetic average of the level $\ell + 1$ fluxes (figure 2).
3. Compute $\langle \nabla \cdot \vec{F} \rangle$ on Ω^ℓ using (5). For control volumes in Ω^ℓ covered by $\Omega^{\ell+1}$, replace the value of $\langle \nabla \cdot \vec{F} \rangle$ with the arithmetic average of the values from the level $\ell + 1$ control volumes that are contained in the covered level ℓ control volume.
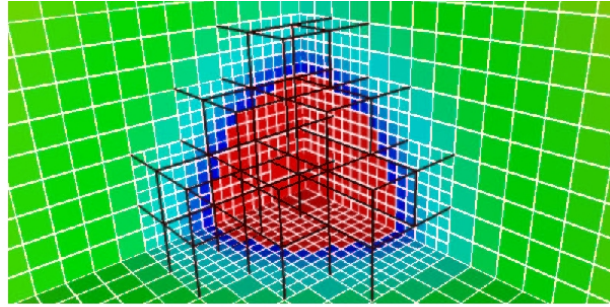
FIGURE 1. Example of block-structured locally-refined grids in 3D. The white lines outline the boundaries of the intersection of the individual control volumes with the green cut-planes, and the black lines show the organization of the refined region into rectangular blocks of control volumes.
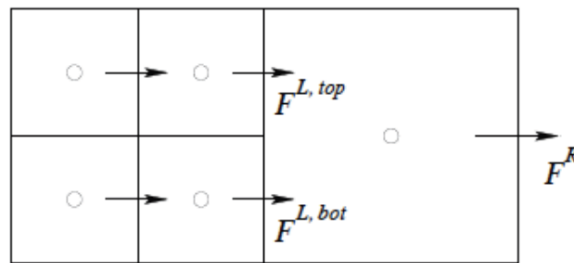

FIGURE 2. Computing fluxes at refinement boundaries. The flux into the coarse control volume to the right of the refined grid is given by the arithmetic average $\frac{1}{2}(F^{L,top} + F^{L,bot})$, which is also the flux out of the adjacent coarse control volume covered by the fine region. Since arithmetic averaging of fine face averages to obtain a coarse face average is exact, the accuracy of the flux on the refinement boundary is the same as the accuracy of the component fluxes.

This algorithm satisfies the natural generalization of the discrete conservation property (7) to nested grid hierarchies. This follows easily from the replacement of the fluxes in Step 2 with averages from the next finer level, combined with the replacement of values of $\langle \nabla \cdot F \rangle$ on the parts of Ω^ℓ covered by $\Omega^{\ell+1}$ by the arithmetic averages of $\langle \nabla \cdot F \rangle$ defined on the latter. A key observation is that arithmetic averaging from fine grids to coarser grids of averages over control volumes and over faces is exact in the sense of truncation error. This implies that if the truncation error of the single-level method is $O(h^p)$, then the truncation error in the calculation of the divergence on the hierarchy is $O(h^p)$ away from boundaries between different levels of refinement, and $O(h^{p-1})$ near refinement boundaries. For all faces, the error in the flux calculation is $O(h^p)$. Near refinement boundaries, the coefficient multiplying that error is not C^1, so the cancellation of error in the final flux divergence calculation does not hold, thus leading to a loss of one order of

accuracy. For classical PDE, this loss of accuracy is often less, even in max norm. For elliptic and parabolic PDE, the solution error is smoothed by the solution operator, typically leading to an $O(h^p)$ solution error. For hyperbolic PDE, the solution error can be $O(h^p)$ if the refinement boundary is non-characteristic. This is suggested by a modified equation analysis similar to that used in [13], and observed in practice [2, 28].

Formally, we view the solution to a system of conservation laws on a nested grid hierarchy as being defined only on the control volumes at each level that are not covered by the finer levels, i.e. on $\Omega^\ell - \Omega^{\ell+1}$. Furthermore, we need to interpolate values at control volumes contained in the stencils for operators applied to level ℓ control volumes that extend outside of Ω^ℓ. In the former case, we compute $\langle U \rangle$ in control volumes covered by the next finer level to be the arithmetic average of the values on the next finer level. Since arithmetic averaging of averages over control volumes is exact, the latter are p^{th}-order accurate, so is the average. To interpolate in space from the next coarser level, it is convenient to use a least-squares approach to define interpolation stencils [28]. This is similar to what is done for mapped-multiblock methods, so we will defer discussion to the next section. In addition, AMR algorithms for time-dependent problems often employ refinement in time, so that values of $\langle U \rangle$ at intermediate corresponding to the time steps at the next finer level are required. For the Runge-Kutta methods used here, this is done by means of dense output, i.e. reconstructing a polynomial in time from the intermediate stages of a pth-order Runge-Kutta method to obtain a pth-order accurate approximation to the solution at any intermediate time. This is done on for the coarse-grid values near the refinement boundary prior to performing interpolation in space.

For mapped grids, there is a potential additional complication in defining discretizations on the AMR hierarchy, in that the discretized grid mappings and the associated quadratures on faces may not match up exactly between refinement levels. In the case where the mapping is specified analytically, and in addition that the averages $\langle J \rangle_{\boldsymbol{i}}$ and $\langle N_d^s \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ can be computed analytically, arithmetic averaging between levels is exact, and there is no problem. In [6], this issue was addressed by constructing the grid at the finest possible level, then constructing the coarser levels by averaging these quantities down from the finer grids, so that the requisite consistency conditions are satisfied. This is not a practical solution in 3D, particularly when the grid hierarchy is changing as a function of time. Instead [19], we use (29) and (34) to construct values for $\langle J \rangle_{\boldsymbol{i}}$ and $\langle N_d^s \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ that are consistent across refinement boundaries, and, in the case of $\langle N_d^s \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$, satisfy freestream preservation. In the case of $\langle J \rangle$, this is just the calculation of $\langle \nabla_{\boldsymbol{\xi}} \cdot (D^{-1}\boldsymbol{N}^T\boldsymbol{X}) \rangle$ on the AMR grid hierarchy in the Cartesian mapping space. In that case, the discrete conservation property becomes conservation of volume independent of how the grid is refined. The calculation of the face averages uses the same idea, except one dimension lower. For all edges on level ℓ that are covered by edges on the boundary of level $\ell+1$, we replace the integrals in (35) over the coarse edges by the sum of the integrals of the fine edges (figure 3). By a similar argument to the divergence case, the resulting calculation of $\langle N_d^s \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ satisfies the freestream condition. There is no loss of accuracy of the edge integrals, since summing integrals is exact. however, there is a loss of one order of accuracy from differencing the edge integrals. Therefore, in order to obtain p^{th}-order order accuracy for the fluxes, the edge integrals in (35) must be computed with a $(p+1)^{st}$-order quadrature rule.
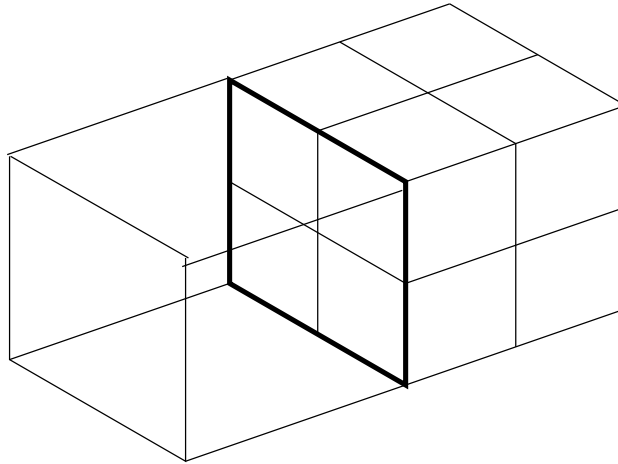
FIGURE 3. Correction of the face metrics at refinement boundaries. Along the dark coarse-grid edges, the $\langle \mathcal{N}_{d,d'} \rangle$ used to compute $\langle N_d \rangle$ on the coarse grid is obtained by averaging the fine grid $\langle \mathcal{N}_{d,d'} \rangle$ values onto the coarse lines. This changes the $\langle N_d \rangle$ on all the faces that intersect the refined-region edges.

4. **Complex geometries.** There are two geometric challenges related to generalizing the discretization methods described above that we address here. One is representing simple domains, but where it is desirable for reasons of efficiency and accuracy to align the grid with global anisotropies, while at the same time there are topological constraints that make it impossible to do so with a single smooth mesh mapping of the type described in Section 2.2. A simple example of this are problems on thin spherical shells, such as the earth's atmosphere. Large-scale motions can be shown to have large aspect ratios [18], and therefore the it is desirable to have the grid aligned with the normal and surface directions to the sphere, i.e. have the map be the tensor product of a map to the surface of the sphere and a 1D mapping in the radial direction, with the ratio of the grid spacing in those directions comparable to the aspect ratio. It is not possible to do that with a single mapping of the form (27) without introducing singularities, such as the pole singularities in classical spherical coordinates. A widely used alternative is the cubed-sphere grid, in which the surface of the sphere is represented by a mapping from six coordinate blocks of the form (27) that are continuous at boundaries, but cannot be represented in terms of a single smooth coordinate map (figure 4).

For highly complex geometries coming from CAD specifications of engineering devices, geophysical data, or image data, we use Cartesian grid embedded boundary (EB) / cut-cell methods [33, 11, 7, 4, 34, 21, 1], an irregular domain Ω is discretized by intersecting a rectangular grid with the domain to obtain a collection of control volumes. One of the principal advantages of EB methods is the ease of what corresponds to grid generation, i.e. the calculation of volume fractions, the area fractions, and the centroids. These can be computed easily for highly complex geometries, such as the exteriors of full aircraft configurations defined by surface triangulations of components [1], and microscale geometries of porous media obtained from image data [38].
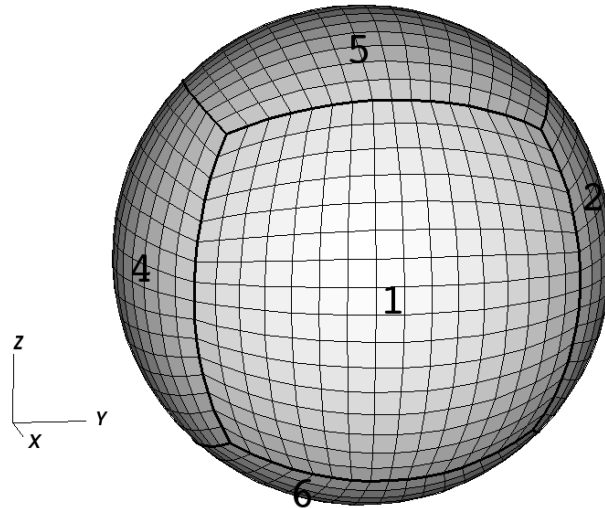
FIGURE 4. Mapped-multiblock gridding of the surface of a sphere. This uses the cubed-sphere map, which maps the surface of a cube onto the surface of a sphere. The map is smooth, except at boundaries between blocks (represented by the heavier dark lines) corresponding to the boundaries between different faces of the sphere, where the map has discontinuous derivatives in the direction normal to the boundary. The tensor-product of a cubed-sphere map with a radial coordinate can be used to represent a thin 3D spherical shell.

For mapped multiblock methods, the basic discretization approach is that in Section 2.2, and the main issue is the definition of that discretization method near the boundaries between blocks that preserves high-order accuracy. For the embedded boundary approach, we have to define methods for computing high-order approximations to the average of $\nabla \cdot \vec{F}$ over irregular control volumes generated where Cartesian control volumes intersect the irregular domain, as well as for faces are sufficiently close to the boundary to require an irregular stencil. In both cases, the problem is one of generalizing the constructions used in Section 2 to compute fluxes to cases for which the unknowns are expressed in terms of averages of smooth functions over control volumes that fail to be smooth rectangular discretizations of space on a set of codimension one. The approach we have taken is based on constructing polynomials whose averages over all sufficiently-nearby control volumes are known. We use more control volumes than polynomial coefficients, leading to an overdetermined linear system for the coefficients or, equivalently, an underdetermined system for the stencil coefficients that express fluxes in terms of nearby averages. To solve this overdetermined system, we use least-squares methods, with the introduction of weighting in some cases to obtain stable discretizations.

4.1. **Mapped multiblock methods.** The method outlined here for computing $\langle \nabla \cdot \vec{F} \rangle$ on a mapped-multiblock grid is that given in [30]. In that approach, the method in Section 2.2 to computing fluxes can be used without modification for control volumes sufficiently far away from block boundaries. To compute the fluxes

near to the block boundaries, we assume that there is a smooth extension of each of the mappings beyond the boundary of the block. Using this extended mapping, we can define an extended rectangular grid and associated control volumes sufficient to compute the fluxes for all of the faces surrounding the control volumes inside the block. To compute the average of the solution at $\boldsymbol{i}$ for block s, we construct a polynomial approximation to the solution U near $\boldsymbol{i}$

$$U^{(\boldsymbol{i},s)}(\boldsymbol{x}) = \sum_{\boldsymbol{p} \in \mathbb{N}^D : ||\boldsymbol{p}||_1 < P} a_{\boldsymbol{p}} \boldsymbol{x}^{\boldsymbol{p}} + O(h^P). \tag{38}$$

To compute the coefficients $a_{\boldsymbol{p}}$, we use the conditions that the average of the polynomial over a collection of nearby valid control volumes is equal to the average of U over those control volumes.

$$\sum_{\boldsymbol{p} \in \mathbb{N}^D : ||\boldsymbol{p}||_1 < P} a_{\boldsymbol{p}} \langle \boldsymbol{x}^{\boldsymbol{p}} \rangle_{(\boldsymbol{i}',s')} = \langle U \rangle_{(\boldsymbol{i},s)}, (\boldsymbol{i}', s') \in \mathcal{N}_{(\boldsymbol{i},s)}, \tag{39}$$

where the moments $\langle \boldsymbol{x}^{\boldsymbol{p}} \rangle_{(\boldsymbol{i}',s')}$ are computed using quadratures. Given $(\boldsymbol{i}, s)$, we need to specify how to how to choose $\mathcal{N}_{(\boldsymbol{i},s)}$. For the case of fourth-order accuracy, we first find $(\boldsymbol{x}_0, s_0)$, the valid control volume containing the center of $(\boldsymbol{i}, s)$. Then we add to that set all of the valid control volumes that share a face, edge, or corner with $(\boldsymbol{x}_0, s_0)$. Finally, we add the valid control volumes that share a face opposite the shared faces from the previous set. Away from three-or-more block intersections, This stencil is a set of points of size $(3^D + 2D)$ (figure 5). At three-or-more-block intersections, the number of points can be larger or smaller, but in any case no less than the number of coefficients $\{a_{\boldsymbol{p}}\}$, and is sufficient to construct polynomials with $P = 4$. In two and three dimensions, (39) is in general an overdetermined system of equations. However, it has maximal rank, is well-conditioned, and a least-squares solution can be computed. This same approach is used to compute ghost values near domain boundaries, with slightly different heuristics [30] for choosing $\mathcal{N}_{(\boldsymbol{i},s)}$.

Once the coefficents have been constructed, the ghost value can be computed as

$$\langle U \rangle_{\boldsymbol{i},s} = \sum_{\boldsymbol{p} \in \mathbb{N}^D : ||\boldsymbol{p}||_1 < P} a_{\boldsymbol{p}} \langle \boldsymbol{x}^{\boldsymbol{p}} \rangle_{(\boldsymbol{i},s)}. \tag{40}$$

This provides us with the necessary ghost values to compute fluxes on all of the faces of each block. This leads to fluxes that are double-valued on faces at block boundaries. We can compute a single-valued flux either by averaging the two fluxes (appropriate for elliptic operators), or using some upwinding method for computing a single-valued flux (appropriate for hyperbolic operators).

4.2. **Embedded boundary methods.** The discretization of space given by the intersection of Cartesian control volumes with the irregular domain leads to a finite-volume discretization corresponding to (5)

$$\langle \nabla \cdot \vec{F} \rangle_{\boldsymbol{v}} = \frac{1}{\kappa h} \left(\langle \vec{F} \cdot \hat{n} \rangle_{EB} + \sum_d \langle F^d \rangle_{d,+} - \langle F^d \rangle_{d,-} \right), \tag{41}$$

$$\langle q \rangle_{\boldsymbol{v}} \equiv \frac{1}{\kappa h^D} \int_{\boldsymbol{v}} q(\boldsymbol{x}) d\boldsymbol{x}, \tag{42}$$

$$\langle F^d \rangle_{d,\pm} = \frac{1}{h^{D-1}} \int_{\boldsymbol{a}_{d,\pm}} F^d dA, \tag{43}$$
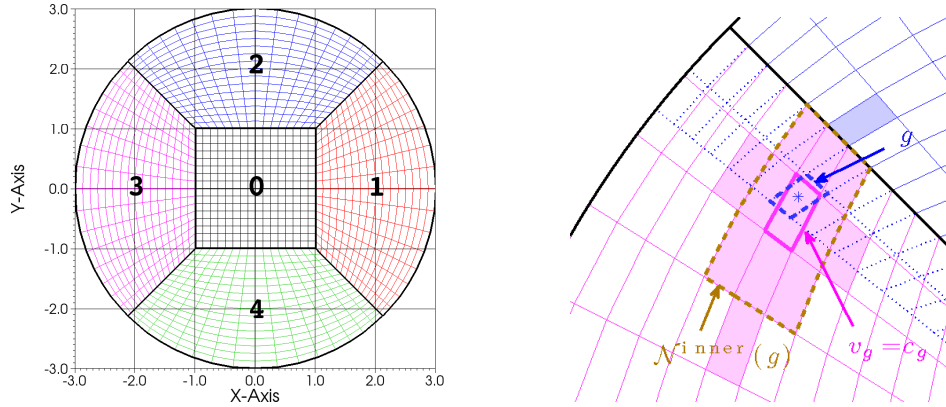
FIGURE 5. Left: mapped-multiblock coordinate system for a disk in 2D. The different blocks are colored, and numbered. Right: computing ghost values for the disk grids. The light dotted blue lines indicate the control volumes defined by the extension of the coordinate mapping for grid 2. For the ghost control volume outlined by the heavy dashed blue line: (i) find the valid control volume containing its center, outlined by the solid solid magenta line; the find all of the control volumes that touch the faces or corners of the control volume in (i), outlined by the dotted brown line; and (iii) add two points in each logical coordinate direction to obtain the stencil, indicated by the shaded valid control volumes.

$$\langle \hat{n} \cdot \vec{F} \rangle_{EB} = \frac{1}{h^{D-1}} \int_{\boldsymbol{a}_{EB}} \vec{F} \cdot \hat{n} dA, \tag{44}$$

(figure 6), where $\boldsymbol{v}$ is a connected component of $\Omega \bigcap V_{\boldsymbol{i}}$, $\boldsymbol{a}_d^{\pm} = \partial \boldsymbol{v} \bigcap A_{\boldsymbol{i} \pm \frac{1}{2} \boldsymbol{e}^d}$, and $\boldsymbol{a}_{EB} = \partial \boldsymbol{v} \bigcap \partial \Omega$. The quantity $\kappa = \kappa(\boldsymbol{v})$ is the volume fraction of $\boldsymbol{v}$, i.e. $Volume(\boldsymbol{v}) = \kappa h^D$. Globally, the geometry is represented as a graph: the nodes of the graph are the control volumes, and the arcs of the graph are the faces α_d connecting a pair of control volumes in adjacent cells. Since the $\boldsymbol{a}$ connect exactly two adjacent control volumes, the discretization (41) satisfies a discrete conservation principle analogous to (7): the volume-weighted sum of $\langle \nabla \cdot \vec{F} \rangle$ over a region defined as a collection of control volumes is equal to the area-weighted sum of the flux averages $\pm \langle F^d \rangle_{d,\pm}$, $\langle \vec{F} \cdot \hat{n} \rangle_{EB}$ summed over the boundary of the region. The relationship (41) is exact, and the most commonly used discretization is based on a second-order accurate approximation of the flux, combined with the analogue of the midpoint rule:

$$\langle \nabla \cdot \vec{F} \rangle_{\boldsymbol{v}} = \frac{1}{\kappa h} \Big(\alpha_{EB} \hat{n}(\boldsymbol{x}_{EB}) \cdot \vec{F}(\boldsymbol{x}_{EB}) + \tag{45}$$
$$\sum_d \alpha_{d,+} F^d(\boldsymbol{x}_{d,+}) - \alpha_{d,-} F^d(\boldsymbol{x}_{d,-}) \Big) + O\Big(\frac{h}{\kappa}\Big),$$

where the area fractions α are defined by $\alpha_{d,\pm} h^{D-1} = Area(\boldsymbol{a}_{d,\pm})$, $\alpha_{EB} h^{D-1} = Area(\boldsymbol{a}_{EB})$, and $\boldsymbol{x}_{d,\pm}$, $\boldsymbol{x}_{EB}$ are the centroids of the corresponding $\boldsymbol{a}$'s.
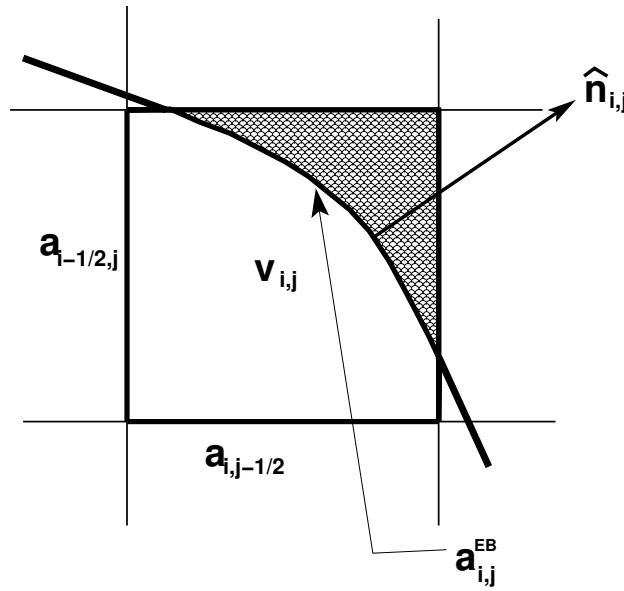
FIGURE 6. Example of an embedded-boundary control volume. The shaded region indicates the part of the control volume not contained in the domain. the control volume is indicated by $\boldsymbol{v}$, the faces through which fluxes are defined indicated by $\boldsymbol{a}$.

We note that there is a loss of one order in h in the truncation error relative to what we get from the midpoint rule in control volumes that do not intersect $\partial\Omega$. This is a loss of one order of accuracy on a set of codimension one, which results in second-order accuracy in max norm for elliptic problems, and at least in L^1 norm or possibly max norm or for hyperbolic problems, depending on whether or not the boundary is characteristic [13]. In addition, the presence of the volume fractions κ in denominators, which can take on arbitrarily small values, might be a cause for concern, both from a stability and an accuracy standpoint. This turns out not to be a problem. For elliptic problems, or parabolic problems that are discretized implicitly in time, the volume fractions in the denominator are eliminated by diagonal scaling [21, 29], while still leading to stable discretizations. For time-dependent hyperbolic problems, one can still use explicit time discretizations by hybridizing with a stable, non-conservative method, and redistribute the increments corresponding to the local loss of mass in each irregular control volume to nearby control volumes in a way that is stable overall [33, 11, 4].

We generalize the EB approach to higher order for domains Ω that are defined in terms of an implicit function: $\Omega = \{\boldsymbol{x} : \phi(\boldsymbol{x}) < 0\}$, where ϕ is assumed to be smooth for the purposes of constructing high-order accurate methods. This is a representation that arises naturally from image processing based on level-set methods [27], and is increasingly an approach used in CAD systems based on constructive solid geometry. In that case, $\partial\Omega = \phi^{-1}(0)$, and

$$\hat{n}_{EB} = \frac{\nabla\phi}{||\nabla\phi||}. \tag{46}$$

We take the Taylor expansion of of the fluxes inside the flux integrals (43) - (44).

$$\langle \nabla \cdot \vec{F} \rangle_{\boldsymbol{v}} = \frac{1}{\kappa h} \sum_{0 \leq \boldsymbol{q} < Q} \frac{1}{\boldsymbol{q}!} \Big(\sum_d \Big((\nabla^{\boldsymbol{q}} F^d)(\boldsymbol{x}_{d,+}) m_{d,+}^{(\boldsymbol{q})} - (\nabla^{\boldsymbol{q}} F^d)(\boldsymbol{x}_{d,-}) m_{d,-}^{(\boldsymbol{q})} \Big)$$

$$+ (\nabla^{\boldsymbol{q}} \vec{F})(\boldsymbol{x}_{EB}) \cdot \vec{m}_{EB}^{(\boldsymbol{q})} \Big) + O\Big(\frac{h^{Q-1}}{\kappa} \Big), \tag{47}$$

$$m_{d,\pm}^{(\boldsymbol{q})} = \frac{1}{h^{D-1}} \int_{\boldsymbol{a}_{d,\pm}} (\boldsymbol{x} - \boldsymbol{x}_{d,\pm})^{\boldsymbol{q}} dA_{\boldsymbol{x}}, \tag{48}$$

$$\vec{m}_{EB}^{(\boldsymbol{q})} = \sum_{0 \leq ||\boldsymbol{s}||_1 < Q - ||\boldsymbol{q}||_1} \frac{1}{\boldsymbol{s}!} (\nabla^{\boldsymbol{s}} \hat{n}_{EB})(\boldsymbol{x}_{EB}) \Big(\frac{1}{h^{D-1}} \int_{\boldsymbol{a}_{EB}} (\boldsymbol{x} - \boldsymbol{x}_{EB})^{\boldsymbol{q}+\boldsymbol{s}} dA_{\boldsymbol{x}} \Big). \tag{49}$$

where we use the representation (46) to define the Taylor expansion of $\hat{n}$ and the points $\boldsymbol{x}_{d,\pm}$, $\boldsymbol{x}_{EB}$ are points on the Cartesian face containing $\boldsymbol{a}_{d,\pm}$, and in the Cartesian control volume containing $\boldsymbol{v}$, respectively. Thus grid generation in this extension of EB to higher order reduces to computing the moments over $\boldsymbol{a}_{d,\pm}$, $\boldsymbol{a}_{EB}$ in (48), (49), along with moments over the volume fractions

$$M_{\boldsymbol{v}}^{(\boldsymbol{q})} = \frac{1}{h^D} \int_{\boldsymbol{v}} \boldsymbol{x}^{\boldsymbol{q}} d\boldsymbol{x}. \tag{50}$$

Once the moments are computed, the computation of $\langle \nabla \cdot \vec{F} \rangle_{\boldsymbol{v}}$ is reduced to finding suitable approximations to the $\nabla^{\boldsymbol{q}} F^d$, or equivalently, a polynomial approximation to F^d of the appropriate order.

Following [37], we compute suitable approximations to these moments by applying (41) to fluxes of the form $\vec{F}(\boldsymbol{x}) = \boldsymbol{x}^{\boldsymbol{r}} \boldsymbol{e}^d$

$$\int_{\boldsymbol{v}} r_d \boldsymbol{x}^{\boldsymbol{r} - \boldsymbol{e}^d} d\boldsymbol{x} - \hat{n}_{EB,d}(\boldsymbol{0}) \int_{\boldsymbol{a}_{EB}} \boldsymbol{x}^{\boldsymbol{r}} dA_{\boldsymbol{x}} = \tag{51}$$

$$\sum_d \Big(\int_{\boldsymbol{a}_{d,+}} \boldsymbol{x}^{\boldsymbol{r}} dA_{\boldsymbol{x}} - \int_{\boldsymbol{a}_{d,-}} \boldsymbol{x}^{\boldsymbol{r}} dA_{\boldsymbol{x}} \Big) +$$

$$\sum_{\boldsymbol{s} \in \mathbb{N}^D : 0 < ||\boldsymbol{s}||_1 < Q - ||\boldsymbol{r}||_1} \frac{1}{\boldsymbol{s}!} (\nabla^{\boldsymbol{s}} \hat{n}_{EB,d})(\boldsymbol{0}) \int_{\boldsymbol{a}_{EB}} \boldsymbol{x}^{\boldsymbol{r}+\boldsymbol{s}} dA_{\boldsymbol{x}} +$$

$$O\Big(h^{Q+D-1} \Big).$$

The error term in this expansion is sufficient to lead the asymptotic error term in (49) unchanged. For each $R \geq 0$, the equations of the form (51) for $\{\boldsymbol{r} : ||\boldsymbol{r}||_1 = R, d = 1, \ldots, D\}$ form a well-conditioned overdetermined system of maximal rank for the moments over $\boldsymbol{v}$ and $\boldsymbol{a}_{EB}$ on the left-hand side, where the right-hand side consists of higher-order moments over $\boldsymbol{a}_{EB}$ or moments of the form (50) for dimension $D' = D - 1$. The hierarchy in R terminates at $R = Q$, for which the contribution to the right-hand side of integral over $\boldsymbol{a}_{EB}$ vanishes, so that, if one can compute the moments over $\boldsymbol{a}_{d,\pm}$, we obtain a closed hierarchy of least-squares problems for the moments which can then be solved, starting at the highest moment. We can apply this argument recursively in dimensionality, until we get to the case $D = 1$, for which the moments can be computed analytically by finding the intersection of coordinate lines with $\partial\Omega$.

There are two cases in which the above construction must be modified. One is in the case where a $D' < D$ calculation involves a zero set that grazes a D' coordinate surface; the second is where thickness of the region $\{\boldsymbol{x} : \phi(\boldsymbol{x}) > 0\}$ is comparable to h (figure 7). In both cases, the solution is the same, i.e. to locally refine the grid in the process of generating the control volumes and moments. In the latter case, this will also lead to multiple control volumes per Cartesian cell, and a graph representation of the embedded boundary grid that is not a subgraph of the lattice $\mathbb{Z}^D$.

FIGURE 7. Example of an underresolved collection of control volumes in 2D. The region above the heavy curve is contained in the domain. However, a simple representation of the geometry and topology of the embedded boundary grid based on computing a single intersection of the coordinate line with the face is inadequate: there are two disconnected components to that face that connect the single volume on top to a pair of disconnected control volumes below. We deal with this problem by locally refining the grid as indicated here to represent accurately the geometry, as indicated by the dotted lines.

The main problem for the higher-order EB algorithm formulation is finding approximations to the flux and its derivatives that are stable. There is no rigorous theory for stability, so the goal is to find a methodology and a set of heuristics for designing stable methods. One example of such a combination is contained in [17] for Poisson's equation. In that case, a least-squares algorithm similar to that in used to compute ghost cells at block boundaries in Section 4.1 is used to compute a polynomial interpolant to the flux on the faces. The essential new heuristic that leads to a stable discretization is the use of weighted least-squares, in which the

control volumes nearest the face are weighted the most heavily. This has the effect of more heavily weighting the diagonal of the resulting matrix, while still providing enough degrees of freedom to obtain higher-order accuracy.

5. **Conclusions.** The principal focus of the discussion here has been to lay out an approach for extending classical finite-volume methods for (1), (2) to obtain discretization methods of any order of accuracy in the truncation error. It is based on the use of uniform or smoothly-varying grids to define a rectangular lattice of control volumes. Because of the grid smoothness, the truncation error in the discretization of the operator is $O(h^p)$ provided that the face averages of fluxes are also computed to $O(h^p)$, and that the truncation error is a smooth function of its inputs and location in space. This leads to a loss of one order of accuracy in truncation error when the latter condition is not satisfied, e.g. at domain boundaries, at boundaries between different levels of refinement in AMR, at block boundaries for mapped-multiblock methods, and at control volumes near the irregular domain boundary in embedded boundary methods. For classical PDE, this loss of accuracy in the truncation error is often smoothed out by the solution operator, leading to a solution error that can recover $O(h^p)$ accuracy, even in max norm. The approach has been generalized to discretizations that are approriate for representing a variety problems in complex domains.

Within the context of the approach given here, there are still some possible extensions that would increase the geometric flexibility of the methods, specifically combinations of mapped and mapped-multiblock methods with embedded boundary geometries. This could be useful, for example, in representing orography in atmospheric modeling, or CFD in internal combustion engines. In these cases, a mapped-multiblock background grid would be used to represent some global feature: the thin-layer anisotropy in atmospheric modeling, or in case of the combustion problem, alignment of the grid with the cylinder wall in order to efficiently represent the large-scale viscous boundary layer. Then the embedded boundary approach can be used to represent the detailed geometry of variations in the surface of the earth, or of valves, injectors, piston geometries, and other detailed engineering features in the cylinder. Finally, for moving or free boundaries, the methods described in Section 4.2 can be applied in space-time to obtain the moment information required to derive finite-volume discretizations for the space-time divergence of $(U, \vec{F})$ to generalize the methods in [11, 4] to any order of accuracy.

REFERENCES

[1] M. Aftosmis, M. Berger and J. Melton, Robust and efficient Cartesian mesh generation for component-based geometry, *AIAA Journal*, **6** (1998), 952–960.

[2] M. Barad and P. Colella, A fourth-order accurate local refinement method for Poisson's equation, *Journal of Computational Physics*, **209** (2005), 1–18.

[3] P. Basu, M. Hall, S. Williams, B. Van Straalen, L. Oliker and P. Colella, Compiler-directed transformation for higher-order stencils, in *Proceedings of the Parallel and Distributed Processing Symposium (IPDPS)*, Institute for Electrical and Electronics Engineers (2015), 313–323.

[4] J. B. Bell, P. Colella and M. Welcome, A conservative front-tracking for inviscid compressible flow, in *Proceedings of the Tenth AIAA Computational Fluid Dynamics Conference*, American Institute for Aeronautics and Astronautics, (1991), 814–822.

[5] M. J. Berger and P. Colella, Local adaptive mesh refinement for shock hydrodynamics, *Journal of Computational Physics*, **82** (1989), 64–84.

[6] M. J. Berger and A. Jameson, Automatic adaptive grid refinement for the Euler equations, *AIAA Journal*, **23** (1985), 561–568.

[7] M. J. Berger and R. J. LeVeque, An adaptive Cartesian mesh algorithm for the euler equations in arbitrary geometries, in *Proceedings of the AIAA 9th Computational Fluid Dynamics Conference*, American Institute for Aeronautics and Astronautics, (1989), 1–7.

[8] J. P. Boris and D. L. Book, Flux-corrected transport. I. SHASTA, a fluid transport algorithm that works, *Journal of Computational Physics*, **11** (1973), 38–69.

[9] A. Bourlioux, A. T. Layton and M. L. Minion, Higher-order multi-implicit spectral deferred correction methods for problems of reacting flow, *Journal of Computational Physics*, **189** (2003), 651–675.

[10] C. Chaplin and P. Colella, A single stage flux-corrected transport algorithm for high-order finite-volume methods, preprint, `arXiv:1506.02999`.

[11] I.-L. Chern and P. Colella, *A conservative front tracking method for hyperbolic conservation laws*, Technical Report UCRL-97200, Lawrence Livermore National Laboratory, 1987.

[12] P. Colella, Multidimensional upwind methods for hyperbolic conservation laws, *Journal of Computational Physics*, **87** (1990), 171–200.

[13] P. Colella, Volume-of-fluid methods for partial differential equations, In *Godunov Methods: Theory and Applications*, pages 161–177. Kluwer, 2001.

[14] P. Colella, M.R. Dorr, J. A. F. Hittinger and D. F. Martin, High-order, finite-volume methods in mapped coordinates, *Journal of Computational Physics*, **230** (2011), 2952–2976.

[15] P. Colella and M. D. Sekora, A limiter for PPM that preserves accuracy at smooth extrema, *Journal of Computational Physics*, **227** (2008), 7069–7076.

[16] P. Colella and P. R. Woodward, The piecewise parabolic method (PPM) for gas-dynamical simulations, *Journal of Computational Physics*, **54** (1989), 174–201.

[17] D. Devendran, D. T. Graves and H. Johansen, A higher-order finite-volume discretization method for Poisson's equation in cut cell geometries, preprint, `arXiv:1411.4283`.

[18] C. Gatti-Bono and P. Colella, An anelastic allspeed projection method for gravitationally stratified flows, *Journal of Computational Physics*, **216** (2006), 589–615.

[19] S. M. Guzik, X. Gao, L. D. Owen, P. McCorquodale and P. Colella, A freestream-preserving fourth-order finite-volume method in mapped coordinates with adaptive mesh refinement, *Computers and Fluids*, **123** (2015), 202–217.

[20] J. Hilditch and P. Colella, *A Projection Method for Low Mach Number Fast Chemistry Reacting Flow*, Technical Report AIAA-97-0263, American Institute of Aeronautics and Astronautics, 1997.

[21] H. Johansen and P. Colella, A Cartesian grid embedded boundary method for Poisson's equation on irregular domains, *Journal of Computational Physics*, **147** (1998), 60–85.

[22] C. A. Kennedy and M. H. Carpenter, Additive Runge-Kutta schemes for convection-diffusion-reaction equations, *Applied Numerical Mathematics*, **44** (2003), 139–181.

[23] H.-O. Kreiss and J. Oliger, Comparison of accurate methods for the integration of hyperbolic equations, *Tellus*, **24** (1972), 199–215.

[24] P. D. Lax, Weak solutions of nonlinear hyperbolic equations and their numerical computation, *Communications on Pure and Applied Mathematics* **7** (1954), 159–193.

[25] P. D. Lax, *On Discontinuous Initial-Value Problems and Finite-Difference Schemes*, Technical Report LAMS-1332, Los Alamos Scientific Laboratory, 1952.

[26] P. D. Lax and B. Wendroff, Systems of conservation laws, *Communications on Pure and Applied Mathematics*, **13** (1960), 217–237.

[27] R. Malladi, J. A. Sethian and B. C. Vemuri, Shape modeling with front propagation: A level set approach, *IEEE Transactions on Pattern Anal. Machine Intell*, **17** (1995), 158–175.

[28] P. McCorquodale and P. Colella, A high-order finite-volume method for conservation laws on locally refined grids, *Communications in Applied Mathematics and Computational Science* **6** (2011), 1–25.

[29] P. McCorquodale, P. Colella and H. Johansen, A Cartesian grid embedded boundary method for the heat equation on irregular domains, *Journal of Computational Physics*, **173** (2001), 620–635.

[30] P. McCorquodale, M. R. Dorr, J. A. F. Hittinger and P. Colella, High-order finite-volume methods for hyperbolic conservation laws on mapped multiblock grids, *Journal of Computational Physics*, **288** (2015), 181–195.

[31] L. I. Millett and S. H. Fuller, et al., *The Future of Computing Performance: Game Over or Next Level?*, National Academies Press, 2011.

[32] J. von Neumann and R. D. Richtmyer, A method for the numerical calculation of hydrodynamic shocks, *Journal of Applied Physics*, **21** (1950), 232–237.

[33] W. F. Noh, CEL: A time-dependent, two-space-dimensional, coupled Eulerian - Lagrangian code, *Methods in Computational Physics*, **3** (1964), 117–180.

[34] R. B. Pember, J. B. Bell, P. Colella, W. Y. Crutchfield and M. L. Welcome, An adaptive Cartesian grid method for unsteady compressible flow in irregular regions, *Journal of Computational Physics*, **120** (1995), 278–304.

[35] R. B. Pember, L. H. Howell, J. B. Bell, P. Colella, W. Y. Crutchfield, W. A. Fiveland and J. P. Jessee, An adaptive projection method for unsteady, low-Mach-number combustion, *Combustion Science and Technology*, **140** (1998), 123–168.

[36] J. S. Saltzman, An unsplit 3D upwind method for hyperbolic conservation laws, *Journal of Computational Physics*, **115** (1994), 153–168.

[37] P. Schwartz, J. Percelay, T. Ligocki, H. Johansen, D. Graves, D. Devendran, P. Colella and E. Ateljevich, High-accuracy embedded boundary grid generation using the divergence theorem, *Communications in Applied Mathematics and Computational Science*, **10** (2015), 83–96.

[38] D. Trebotich, M. F. Adams, S. Molins, C. I. Steefel and C. Shen, High-resolution simulation of pore-scale reactive transport processes associated with carbon sequestration, *Computing in Science and Engineering*, **16** (2014), 22–31.

[39] B. van Leer, Towards the ultimate conservative differences scheme IV: a new approach to numerical convection, *Journal of Computational Physics*, **23** (1977), 263–275.

[40] S. Williams, A. Waterman and D. Patterson, Roofline: an insightful visual performance model for multicore architectures, *Communications of the ACM*, **52** (2009), 65–76.

[41] P. R. Woodward and P. Colella, The numerical simulation of two-dimensional fluid flow with strong shocks, *Journal of Computational Physics*, **54** (1984), 115–173.

[42] S. T. Zalesak, Fully multidimensional flux-corrected transport algorithms for fluids, *Journal of Computational Physics*, **31** (1979), 335–362.

[43] S. T. Zalesak, A physical interpretation of the Richtmyer two-step Lax-Wendroff scheme and its generalization to higher spatial order, in *Advances in Computer Methods for Partial Differential Equations*, IMACS, (1984), 19–21.

E-mail address: pcolella@lbl.gov