High-Order Finite-Volume Methods on Locally-Structured Grids

P. Colella, M. Dorr, J. Hittinger, P. McCorquodale, D. F. Martin

February 24, 2009

1 Introduction

For many problems in astrophysics and space sciences, it is desirable to compute solutions in a way that preserves spherical symmetry, so that the dynamics of small perturbations about the spherically symmetric case are not overwhelmed by numerical error. Traditionally, such calculations have been done by discretizing the equations expressed in spherical coordinates. This approach has significant numerical problems, due to the singularity in the spherical coordinate system at the poles. An alternative approach is based on the "cubed-sphere" representation (Figure 1). In this approach, a solid sphere is represented as the disjoint union of the images of smooth mappings of multiple rectangular blocks. Away from the block at the center of the sphere, the coordinate surfaces of the mappings coincide with spherical surfaces, thus providing a coordinate system that, when discretized, will preserve spherical symmetry, at least away from the center of the sphere.

At the block boundaries, the coordinate lines meet continuously, but are not smooth. In addition, the grids meet in a way that does not lead to a single logically rectangular coordinate transformation. These features are difficulties for standard finite-volume methods on structured grids. The accuracy of such methods depends strongly on the smoothness of the grid mapping. At places where the grid mapping fails to be smooth, there is a loss of one order of accuracy in the truncation error. For a second-order accurate method, this leads to a local truncation error that is first-order at block boundaries. Similar issues arise for finite-volume methods on locally-refined structured grids: at refinement boundaries, such methods lose one order of accuracy. For many problems, a such a reduction to first-order accuracy is unacceptable, even on a set of codimension one.

To address this issue, we we are developing a new class of finite-volume methods on locally-refined and mapped-multiblock grids. The central feature of these methods is that they are at least fourth-order accurate in regions where the solution is smooth; otherwise, we want to retain to as great an extent as possible the advantages of traditional finite-volume methods. In this paper, we discuss our approach to the design of such methods.

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 $V_{\boldsymbol{i}}$ takes the form

$$V_{\boldsymbol{i}} = [\boldsymbol{i}h, (\boldsymbol{i} + \boldsymbol{u})h] \ , \ \boldsymbol{i} \in \mathbb{Z}^D \ , \ \boldsymbol{u} = (1, 1, \ldots, 1),$$
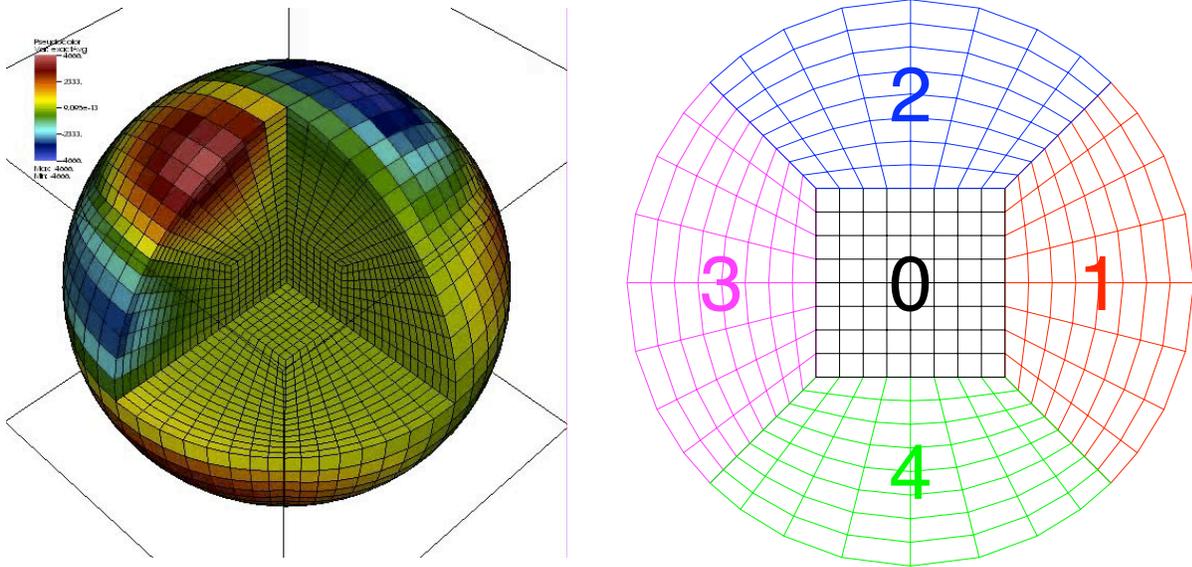
Figure 1: Examples of multiblock grids. Right: 3D cubed-sphere grid. Left: equatorial slice through a cubed-sphere grid, showing five logically-rectangular coordinate systems.

where h is the grid spacing.

A finite volume discretization of a partial differential equation is based on averaging that equation over control volumes, applying the divergence theorem to replace volume integrals by interals over the boundary of the control volume, and approximating the boundary integrals by quadratures. For example, for time-dependent problems of the form

$$\frac{\partial U}{\partial t} + \nabla \cdot \vec{F} = 0 \tag{1}$$

The discretized solution in space is the average of U over a control volume.

$$\langle U \rangle_{\boldsymbol{i}}(t) = \frac{1}{h^D} \int\limits_{V_{\boldsymbol{i}}} U(\boldsymbol{x}, t) d\boldsymbol{x} \tag{2}$$

In that case, we can compute the evolution of the spatially discretized system by a method-of-lines approach.

$$\frac{d\langle U \rangle_{\boldsymbol{i}}}{dt} = -\frac{1}{h^D} \int\limits_{V_{\boldsymbol{i}}} \nabla \cdot \vec{F} dx = -\frac{1}{h} \sum_{d=1}^{D} \langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} - \langle F^d \rangle_{\boldsymbol{i}-\frac{1}{2}\boldsymbol{e}^d} \tag{3}$$

$$\langle F^d \rangle_{\boldsymbol{i}\pm\frac{1}{2}\boldsymbol{e}^d} = \frac{1}{h^{D-1}} \int\limits_{A_d^\pm} F^d dA, \tag{4}$$

where $A_d^\pm$ are the high and low faces bounding $V_{\boldsymbol{i}}$ with normals pointing in the $\boldsymbol{e}^d$ direction. In this case, the finite volume approach computes the average of the divergence of the fluxes on the left-hand side of (4) with the sum of the integrals over faces on the right-hand side, with the latter

approximated using some quadrature rule. 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 (4), leading to a second-order accurate method. We will be developing higher-order methods (fourth-order or better) using the approach in [1]. The starting point for this approach is to replace the integrand in the right-hand side of (4) by a Taylor expansion about the center of the face:

$$\int_{A_d} F^d dA = \sum_{0 \le |\boldsymbol{r}| < R} \frac{1}{\boldsymbol{r}!} \nabla^{\boldsymbol{r}} F_d|_{\boldsymbol{x}=\boldsymbol{x}_0} \int_{A_d} (\boldsymbol{x} - \boldsymbol{x}_0)^{\boldsymbol{r}} dA_{\boldsymbol{x}} + O(h^{R+D-1}), \tag{5}$$

$$\boldsymbol{r}! = r_1! \dots r_D! \; , \; \boldsymbol{q}^{\boldsymbol{r}} = q_1^{r_1} \dots q_D^{r_D}. \tag{6}$$

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

$$\langle F^d \rangle = F^d(\boldsymbol{x}_0) + \frac{h^2}{24} \sum_{d' \ne d} \frac{\partial^2 F_d}{\partial x_{d'}^2} + O(h^4). \tag{7}$$

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 $O(h^R)$. In this paper, we will be concerned with the case $R = 4$.

A commonly-occuring case is where the flux is a product of two other variables; for example, in advection, the flux is a product of the normal component of velocity and the advected quantity. Similarly, it is sometimes useful to compute average of products over control volumes. In these cases, we can use the approximations

$$\langle fg \rangle_{\boldsymbol{i}} = \langle f \rangle_{\boldsymbol{i}} \langle g \rangle_{\boldsymbol{i}} + \frac{h^2}{12} \nabla f \cdot \nabla g + O(h^4) \tag{8}$$

$$\langle fg \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} = \langle f \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} \langle g \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} + \frac{h^2}{12} \sum_{d' \ne d} \frac{\partial f}{\partial x_{d'}} \frac{\partial g}{\partial x_{d'}} + O(h^4) \tag{9}$$

with the first derivative terms approximated by second-order accurate centered differences.

We apply the approach described above to the case of a scalar advection equation

$$\frac{\partial s}{\partial t} + \nabla \cdot (\vec{u} s) = 0 \tag{10}$$

where s is the advected scalar, and $\vec{u} = \vec{u}(\boldsymbol{x})$ is an advection velocity. Given the solution $\langle s \rangle_{\boldsymbol{i}}^n \approx \langle s \rangle_{\boldsymbol{i}}(t^n)$, we want to compute $\langle s \rangle_{\boldsymbol{i}}^{n+1}$ using a fourth-order accurate method based on the semi-discrete formulation (2), using a fourth-order Runge-Kutta method as our time-stepping scheme. We assume that we know $\langle u_d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$, the average of the normal components of the discrete velocity field over cell faces. To compute the right-hand side as a function of the discretized data, we use (4) and (9), with the standard deconvolution algorithm to compute fourth-order accurate face averages from cell averages [4].

$$\langle s \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} = \frac{7}{12}(\langle s \rangle_{\boldsymbol{i}} + \langle s \rangle_{\boldsymbol{i}+\boldsymbol{e}^d}) - \frac{1}{12}(\langle s \rangle_{\boldsymbol{i}-\boldsymbol{e}^d} + \langle s \rangle_{\boldsymbol{i}+2\boldsymbol{e}^d}) \tag{11}$$

For hyperbolic problems, it is also necessary to introduce limiters to suppress oscillations in the presence of discontinuities and underresolved gradients. To do this, we apply an extension of the PPM limiter [3,4] to the values obtained from the deconvolution formula (11). In this approach, we

3
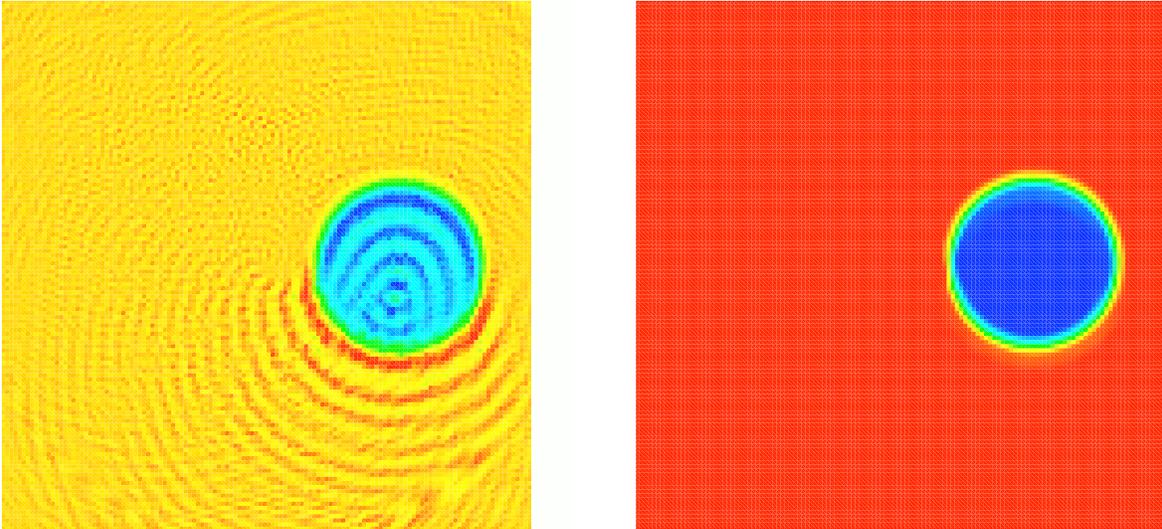
Figure 2: Advection of a piecewise-constant scalar field. Left: fourth-order accurate method, no limiter. Right: fourth-order accurate method, with limiters from [3, 6].

view the PPM limiter as producing limited, and possibly double-valued face averages of s. In the latter case, we choose the upwind value at the face to compute the flux. The use of the limiter [3] leads to an algorithm that preserves fourth-order accuracy in the presence of smooth extrema. If, in addition, there is a constraint that s remains non-negative, we can use a version of the Zalesak FCT limiter in [6] to impose that constraint, with the low-order method given by the CTU scheme in [2]. Since we do not use the FCT limiter to impose any bounds other than positivity, the method should remain fourth-order accurate for smooth data. This is confirmed in Table 2, in which we see fourth-order convergence for a Gaussian profile in two dimensions advected by a rigid-body rotation. On the other hand, in figure 2, we see that a circular blob is advected by this method without any apparent oscillations.

$\frac{1}{h_0}$	no limiter	rate	limiter	rate	Zalesak	rate
16	80.0384	–	79.152	–	79.152	–
32	45.2754	0.822	37.4879	1.078	37.5237	1.076
64	7.54183	2.59	7.46333	2.328	7.46309	2.329
128	0.51193	3.88	0.510228	3.87	0.510228	3.870
256	0.0324131	3.98	0.0324245	3.98	0.0324245	3.975
512	0.00205496	3.98	0.0020557	3.979	0.0020557	3.979

Table 1: Max norm error as a function of mesh spacing for advection of a Gaussian profile by a rigid-body rotation in two dimensions. The table shows errors and convergence rates for, respectively, the method without any limiter, with only the extremum-preserving limiter in [3], and with the extremum-preserving limiter and the Zalesak limiter in [6] used only to preserve positivity.

3   Mapped Grids and Freestream Preservation

We can extend this approach to the case of mapped grids. We assume that we have a smooth mapping $\boldsymbol{X}$ from some abstract coordinate space into physical space:

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

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

$$\nabla_{\boldsymbol{x}} \cdot \vec{F} = \frac{1}{J} \nabla_{\boldsymbol{\xi}} \cdot (\boldsymbol{N}^T \vec{F}), \tag{12}$$

$$J = det(\nabla_{\boldsymbol{\xi}} \boldsymbol{X}) \ , \ \boldsymbol{N}_{p,q}^T = det((\nabla_{\boldsymbol{\xi}} \boldsymbol{X})(p|\boldsymbol{e}^q)) \tag{13}$$

where $\boldsymbol{A}(p|\boldsymbol{v})$ denotes the matrix obtained by replacing the pth row of the matrix $\boldsymbol{A}$ by the vector $\boldsymbol{v}$. The relationship (12) is an easy consequence of the chain rule, equality of mixed partials, and Cramer's rule. We can also rewrite (1) in terms of the independent variable $\boldsymbol{\xi}$.

$$\frac{\partial(JU)}{\partial t} + \nabla_{\boldsymbol{\xi}} \cdot (\boldsymbol{N}^T \vec{F}) = 0 \tag{14}$$

The semi-discrete form of (14) corresponding to (2), (4)

$$\frac{d}{dt}\langle JU \rangle_{\boldsymbol{i}} = -\frac{1}{h^D} \int_{V_{\boldsymbol{i}}} \nabla \cdot (\boldsymbol{N}^T \vec{F}) d\boldsymbol{\xi} = -\frac{1}{h} \sum_{d=1}^{D} \langle (\boldsymbol{N}^T \vec{F})_d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} - \langle (\boldsymbol{N}^T \vec{F})_d \rangle_{\boldsymbol{i}-\frac{1}{2}\boldsymbol{e}^d} \tag{15}$$

$$\langle (\boldsymbol{N}^T \vec{F})_d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} = \frac{1}{h^{D-1}} \int_{A_d} (\boldsymbol{N}^T \vec{F})_d dA, \tag{16}$$

can be interpreted as a finite-volume discretization in physical space, where the control volumes are given by the image of the cells $V_{\boldsymbol{i}}$ under the mapping $\boldsymbol{X}$.

$$\frac{1}{h^D} \int_{\boldsymbol{X}(V_i)} U d\boldsymbol{x} = \langle JU \rangle_{\boldsymbol{i}} \tag{17}$$

$$\frac{1}{h^D} \int_{\boldsymbol{X}(V_i)} \nabla_{\boldsymbol{x}} \cdot \vec{F} d\boldsymbol{x} = \frac{1}{h^D} \int_{V_i} \nabla_{\boldsymbol{\xi}} \cdot (\boldsymbol{N}^T \vec{F}) d\boldsymbol{\xi} = \frac{1}{h} \sum_{d=1}^{D} \langle (\boldsymbol{N}^T \vec{F})_d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} - \langle (\boldsymbol{N}^T \vec{F})_d \rangle_{\boldsymbol{i}-\frac{1}{2}\boldsymbol{e}^d} \tag{18}$$

We can use (9) to compute the face integrals, thus obtaining a fourth-order accurate discretization.

$$\langle (\boldsymbol{N}^T \vec{F})_d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} = \left(\langle \boldsymbol{N}^T \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} \langle \vec{F} \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} \right)_d + \frac{h^2}{12} \sum_{d' \neq d} \left(\frac{\partial}{\partial \xi_{d'}}(\boldsymbol{N}^T) \cdot \frac{\partial}{\partial \xi_{d'}}(\vec{F}) \right)_d + O(h^4) \tag{19}$$

An important property of finite-volume methods on mapped grids is free-stream preservation, i.e. the divergence of a constant vector field $\vec{F}$ is identically zero. Thus, we need to derive quadrature formulas for $\int_{A_d} \boldsymbol{N}^T dA_{\boldsymbol{\xi}}$ so that the discrete divergence of a constant vector field given by (18), (19) is zero.

The existence of such quadratures is a consequence of Stokes' theorem and the Poincare lemma. The rows of the matrix $\boldsymbol{N}$, denoted by $N^s, s = 1, \ldots, D$ satisfy $\nabla_{\boldsymbol{\xi}} \cdot N^s = 0$. This can be seen by a

direct calculation, or inferred indirectly from applying (18) to constant vector fields. Then by the Poincare lemma [5], there exist functions $\mathcal{N}_{d,d'}^s, d \neq d'$ such that

$$N_d^s = \sum_{d' \neq d} \frac{\partial \mathcal{N}_{d,d'}^s}{\partial \xi_{d'}} \ , \ \mathcal{N}_{d,d'}^s = -\mathcal{N}_{d',d}^s. \tag{20}$$

Thus we have

$$\int_{A_d} N_d^s dA_{\boldsymbol{\xi}} = \sum_{\pm=+,-} \sum_{d' \neq d} \pm \int_{E_{d,d'}^{\pm}} \mathcal{N}_{d,d'}^s dE_{\boldsymbol{\xi}}, \tag{21}$$

where $E_{d,d'}^{\pm}$ 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

$$\sum_{d=1}^{D} \sum_{\pm=+,-} \pm \int_{A_d^{\pm}} N_d^s dA_{\boldsymbol{\xi}} = 0$$

is satisfied. Furthermore, the quadrature formulas for the edge integrals can otherwise be chosen arbitrarily; in particular, they can be chosen so that (21) approximates the integral of N_d^s over the face to any order of accuracy. We note that this is a generalization to arbitrary dimensions and arbitrary orders of accuracy of the staggered-grid methods used to discretize electromagnetic fields so that discrete analogues of the various vector identities are satisfied identically.

Given $N_d^s, d = 1, \ldots, D$, the family of functions $\mathcal{N}_{d,d'}^s, d' \neq d$ satisfying (20) is not unique. A particularly simple choice that is a local function of $\boldsymbol{X}$ and $\nabla_{\boldsymbol{\xi}} \boldsymbol{X}$ is given by

$$\mathcal{N}_{d,d'}^s = \frac{1}{D-1} det\big((\nabla_{\boldsymbol{\xi}} \boldsymbol{X}) (\boldsymbol{X}|d')(s|e^d) \big) \tag{22}$$

where $\boldsymbol{A}(\boldsymbol{v}|p) = \boldsymbol{A}(p|\boldsymbol{v})^T$. We note that the expression for $\mathcal{N}_{d,d}^s$ given above only involves derivatives of $\boldsymbol{X}$ in directions tangent to $E_{d,d'}$.

The proof that (22) satisfies (20) is a straightforward calculation. For the special case of $D = 3$, the $\mathcal{N}_{d,d}^s$ are given as follows.

$$\mathcal{N}_{21}^1 = \frac{1}{2}\left(X_3 \frac{\partial X_2}{\partial \xi_3} - X_2 \frac{\partial X_3}{\partial \xi_3} \right) \quad \mathcal{N}_{21}^2 = \frac{1}{2}\left(-X_3 \frac{\partial X_1}{\partial \xi_3} + X_1 \frac{\partial X_3}{\partial \xi_3} \right) \quad \mathcal{N}_{21}^3 = \frac{1}{2}\left(X_2 \frac{\partial X_1}{\partial \xi_3} - X_1 \frac{\partial X_2}{\partial \xi_3} \right)$$

$$\mathcal{N}_{31}^1 = \frac{1}{2}\left(-X_3 \frac{\partial X_2}{\partial \xi_2} + X_2 \frac{\partial X_3}{\partial \xi_2} \right) \quad \mathcal{N}_{31}^2 = \frac{1}{2}\left(X_3 \frac{\partial X_1}{\partial \xi_2} - X_1 \frac{\partial X_3}{\partial \xi_2} \right) \quad \mathcal{N}_{31}^3 = \frac{1}{2}\left(-X_2 \frac{\partial X_1}{\partial \xi_2} + X_1 \frac{\partial X_2}{\partial \xi_2} \right)$$

$$\mathcal{N}_{32}^1 = \frac{1}{2}\left(X_3 \frac{\partial X_2}{\partial \xi_1} - X_2 \frac{\partial X_3}{\partial \xi_1} \right) \quad \mathcal{N}_{32}^2 = \frac{1}{2}\left(-X_3 \frac{\partial X_1}{\partial \xi_1} + X_1 \frac{\partial X_3}{\partial \xi_1} \right) \quad \mathcal{N}_{32}^3 = \frac{1}{2}\left(X_2 \frac{\partial X_1}{\partial \xi_1} - X_1 \frac{\partial X_2}{\partial \xi_1} \right)$$

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

We can apply this approach to solve the advection equation (10) in a mapped coordinate system. The method for computing the fluxes is the same as in the Cartesian grid case, except that it is necessary to compute a fourth-order accurate approximation of $\langle s \rangle_{\boldsymbol{i}}$ from $\langle Js \rangle_{\boldsymbol{i}}$. We do that by
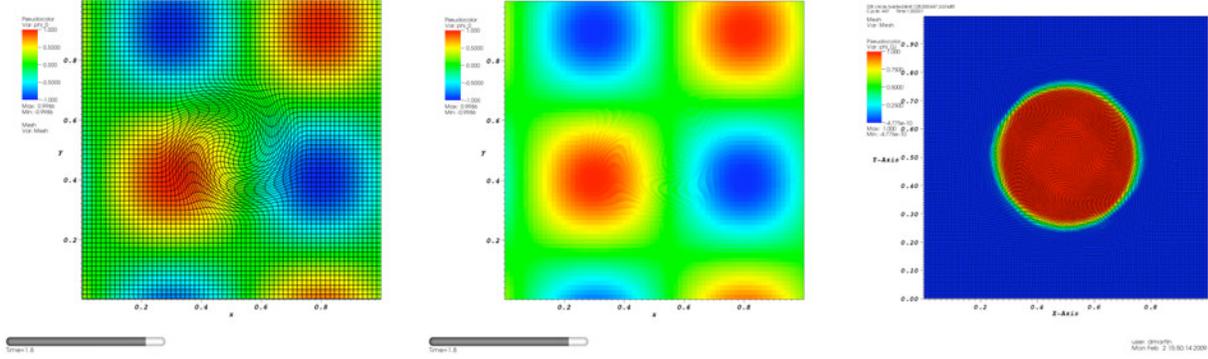
Figure 3: Advection on mapped grids. Left, center: advection of product of trigonomietric functions on a grid with a twist in the center. Advection velocity is a constant pointing in the $(1,1)$ direction, with periodic boundary conditions. Right: same grid, velocity, and boundary conditions as on the left, but with a piecewise-constant circular patch.

repeated applications of (8).

$$\langle s \rangle_{\boldsymbol{i}} \approx \langle Js \rangle_{\boldsymbol{i}} \left\langle \frac{1}{J} \right\rangle_{\boldsymbol{i}} + \frac{h^2}{12} \nabla (Js) \cdot \nabla \left(\frac{1}{J} \right)$$

$$\left\langle \frac{1}{J} \right\rangle_{\boldsymbol{i}} \approx \frac{1}{\langle J \rangle_{\boldsymbol{i}}} \left(1 + \frac{h^2}{12J^2} |\nabla J|^2 \right)$$

In Figure 3, we show the advection of a product of trigonometric functions by a constant velocity field in a grid that has a smoothly-twisted region in the center. As before, we obtain fourth-order accuracy as the mesh spacing becomes finer. We also show the advection of a circular patch by the same velocity field, and on the same mesh. The limiters are effective in suppressing oscillations at the discontinuity.

4 Ghost Cell Interpolation at Grid Boundaries

To extend the methods described above to multiblock and locally-refined meshes, we will apply the various stencil operations by computing ghost-cell values on grids that are smooth extensions of the original block or union of rectangles at a level (Figure 4), together with a mechanism for obtaining single-valued fluxes at block boundaries.

To compute these ghost-cell values, we use a least-squares approach that allows us to obtain high-order accuracy independent of the degree of smoothness of the mesh. We compute a polynomial interpolant in the neighborhood of a ghost cell of the form

$$\varphi(\boldsymbol{x}) \approx \sum_{p_d \geq 0; p_1 + \cdots + p_D \leq P-1} a_{\boldsymbol{p}} \boldsymbol{x}^{\boldsymbol{p}} \ , \ \boldsymbol{p} = (p_1, \ldots, p_D) \ , \ \boldsymbol{x}^{\boldsymbol{p}} = x_1^{p_1} \ldots x_d^{p_d} \tag{23}$$

We will assume that we know the conserved quantities in a collection of control volumes $\boldsymbol{v} \in \mathcal{V}$. In that case, we impose the conditions

$$\langle \varphi \rangle_{\boldsymbol{v}} = \sum_{\boldsymbol{p} \geq 0 : |\boldsymbol{p}| \leq P-1} a_{\boldsymbol{p}} \langle \boldsymbol{x}^{\boldsymbol{p}} \rangle_{\boldsymbol{v}} \ , \ \boldsymbol{v} \in \mathcal{V} \tag{24}$$

7
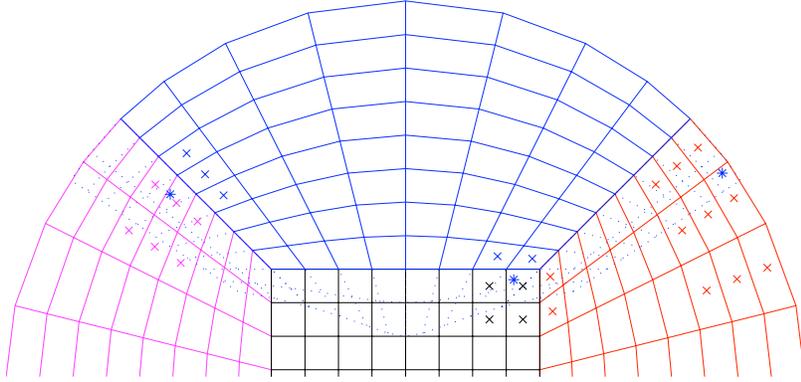
Figure 4: Interpolation of ghost cells on smoothly-extended grid. $*$'s indicate ghost control volumes for which values are to be interpolated, while $\times$'s indicate the nearby control volumes that will be used in the least-squares calculation.

h	$|\epsilon_h|_{max}/|f|_{max}$	p	$(\kappa_h)_{max}$
$1/8$	6.2787×10^{-3}		1.3640×10^{3}
$1/16$	3.5566×10^{-4}	4.14	1.3696×10^{3}
$1/32$	1.6911×10^{-5}	4.39	1.5594×10^{3}
$1/64$	9.1544×10^{-7}	4.21	1.8096×10^{3}

Table 2: Errors in a fourth-order accurate least-squares computation of ghost-cell values. The second column shows relative max norm errors, and the fourth column shows the maximum condition number of the least-squares calculations done using the QR algorithm from LAPack.

Since the conserved quantities are known, and we can compute the moments $\langle x^{p} \rangle_{v}$ from the grid geometry, this constitutes a system of linear equations for the interpolation coefficients a_{p}. Generally, we choose the number of equations to be greater than the number of unknowns in such a way that the resulting overdetermined system has maximal rank, so that it can be solved using least squares. In the case where we are computing an interpolant onto a finer grid from a coarser one in a locally-refined mesh calculation, we can impose the conservation condition as a linear constraint. Table 2 shows the convergence and conditioning properties for a cubic polynomial least-squares interpolation of ghost-cell values for the cubed-sphere grid.

5 Conclusions

In this paper, we have shown some of the major features of a class of higher-order methods for locally-structured grids. Central to this approach is to systematically distinguish between cell- and face- averages and point values, and providing appropriate quadrature and deconvolution methods for converting one to the other. We are currently implementing these methods for coupled hyperbolic and elliptic systems arising in gyrokinetic models of plasmas, and for nonlinear hyperbolic conservation laws.

References

[1] M. Barad and P. Colella. A fourth-order accurate local refinement method for Poisson's equation. *J. Comput. Phys.*, 209:1–18, 2005.

[2] P. Colella. Multidimensional upwind methods for hyperbolic conservation laws. *J. Comput. Phys.*, 87(1):171–200, 1990.

[3] P. Colella and M. D. Sekora. A limiter for PPM that preserves accuracy at smooth extrema. *J. Comput. Phys.*, 227:7069–7076, 2008.

[4] P. Colella and P. R. Woodward. The piecewise-parabolic method (PPM) for gas-dynamical simulations. *J. Comput. Phys.*, 54:174–201, 1984.

[5] M. Spivak. *Calculus on Manifolds*. W. A. Benjamin, Inc., New York, NY, 1965.

[6] S. T. Zalesak. Fully multidimensional flux-corrected transport algorithms for fluids. *J. Comput. Phys.*, 31:335–362, 1979.