

A Higher Order Tracer Transport Scheme for Icosahedral Hexagonal Grids

Maximo Menchaca

Academic Affiliation, Fall 2009: Junior, University of Illinois Urbana-Champaign

SOARS[®] Summer 2009

Science Research Mentor: William C. Skamarock

Writing and Communication Mentor: Arlene Laing

Community Mentor: Katherine Pohl

Peer Mentor: Alex O. Gonzalez

ABSTRACT

Weather and climate models require both efficient and accurate numerical methods to simulate tracer (e.g., moisture, salinity) advection. The distribution of the tracer used in transport equations is approximated by a Taylor expansion. A scheme is developed that builds upon a simpler, second-order convergent method. This original method describes tracer distributions with a first-order Taylor expansion, while the extension uses a second-order expansion to describe the distribution. The original method is conservative and defines a simple departure region, but violates monotonicity preservation. This scheme, due to its simplicity, is not very accurate with more complex tracer and velocity flow configurations. The extension of the method requires three specific modifications: Green's Theorem is used to calculate these next order terms and minimize the computational stencil, Gauss Quadrature is employed to calculate the tracer advected in a departure region, and the cell-averaged value is re-normalized to correct for the addition of these higher order terms. Two tests are run on a planar, perfect hexagonal grid: a solid body rotational case and a time-dependent deformational-flow case. The extension of the tracer distribution function shows marked improvements over the original method, and this extended scheme is third-order convergent for the solid-body rotation case. The improvements, however, are not as obvious when in the deformational-flow test. Nonetheless, the results indicate that the scheme warrants further testing. The successful application of a flux limiter shows that the method can be prepared further for possible implementation into weather and climate models.

The Significant Opportunities in Atmospheric Research and Science (SOARS) Program is managed by the University Corporation for Atmospheric Research (UCAR) with support from participating universities. SOARS is funded by the National Science Foundation, the National Oceanic and Atmospheric Administration (NOAA) Climate Program Office, the NOAA Oceans and Human Health Initiative, the Center for Multi-Scale Modeling of Atmospheric Processes at Colorado State University, and the Cooperative Institute for Research in Environmental Sciences. SOARS is a partner project with Research Experience in Solid Earth Science for Student (RESESS).

1. Introduction

The most accurate atmospheric model in the world might turn out to be useless, if its simulation time lagged behind weather's progression in real time. An accurate computer model could help provide useful insight in a field of study, but efficiency must be taken into consideration. It is this tradeoff between accuracy and efficiency that drives the development of new advection schemes. Some modicum of accuracy can be sacrificed if it vastly increases the speed at which a simulation runs, and vice versa. When it comes to atmospheric models, there are also other factors driving model development. Predictions produced by model simulations could have great consequences for policymaking and societal behavior. Thanks to technological innovations and other factors, there is also a growing body of data that can be gathered and implemented into a model. Developing effective and efficient models that can simulate past and future weather patterns would have a profound effect on a large number of people.

To simulate the changing patterns in atmospheric models, we begin with the transport equation. The transport equation are an integral part of simulations: they describe how certain properties are distributed through a fluid. In order to implement them into a numerical model, one must discretize them. We start with the continuous equation, describing the advection of a tracer density S at a velocity u ,

$$\frac{\partial S}{\partial t} + \nabla \cdot (S\vec{u}) = 0 \quad (1)$$

The next step is to take a finite volume integration within every grid cell, which leads to

$$\frac{\partial}{\partial t} \int_A S \, dA + \oint_A (S\vec{u}) \cdot \hat{n} \, dS = 0,$$

where A is the volume within the grid cell, and \hat{n} is a unit vector normal to a cell edge. The integral of the tracer density over the grid cell is just the cell averaged value multiplied by the cell area. We now have

$$\frac{\partial(\text{area}_{cell} * S_o)}{\partial t} + \oint_A S_t u \, dS = 0,$$

with S_o defined as the cell-averaged tracer value, and S_t defined as the tracer amount that moves across the cell face during the timestep. One eventually reaches

$$S_o^{t+\Delta t} - S_o^t + \frac{\sum_{sides} S_t * u * a * \Delta t}{\text{area}_{cell}} = 0. \quad (2)$$

Further discretization depends on how one wishes to calculate the factors in the final term on the left hand side of (2). This term contains the fluxes of tracer mass advected through the faces of the cells during a time-step. One of the factors in this flux

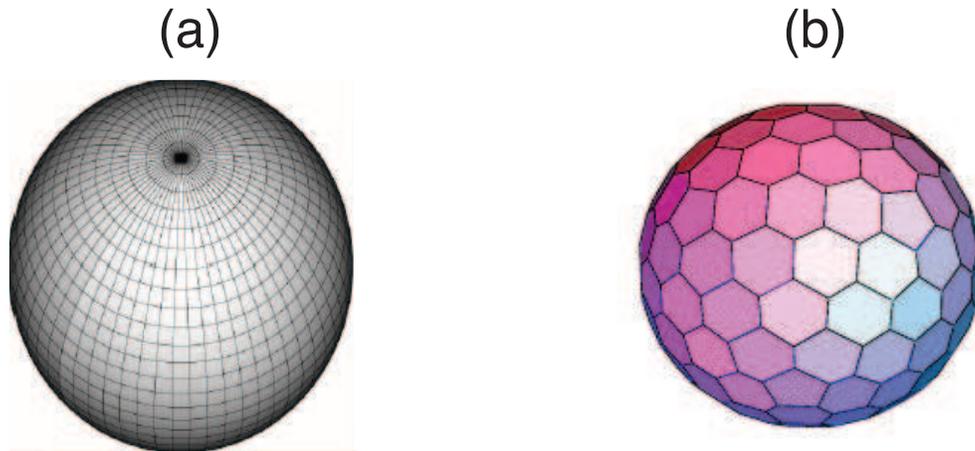


Fig. 1. The (a) traditional Latitude-Longitude Grid and (b) the Hexagonal Grid.

term to consider is the grid cell one uses in a simulation. Traditionally, the latitude-longitude coordinate system (Fig. 1a) was implemented in models, but it provided some problems. The shapes of grid cells can vary widely, from near perfect squares near the equator, to oblong, distorted quadrilaterals as one approaches the poles. This means that the discretization of the transport equations becomes very difficult to implement, and processes and methods can become expensive to run.

An alternative to the latitude-longitude grid is a hexagonal grid (Fig. 1b). Hexagons are an attractive choice for dividing up a grid, as they provide highly isotropic cells. Having cells of near identical size and shape makes implementing discretizations relatively easy. Lipscomb and Ringler (2005) (hereafter LR) describe the use of a hexagonal grid on a sphere. They discussed the aforementioned drawbacks of the latitude-longitude system (e.g., polar singularities), and the advantages to isotropic grid cells. Next, they began to analyze various methods of numerical approximation of the shallow water equations, using the shallow water test cases to illustrate their results. It is desired that these numerical schemes preserve various properties of the continuous equations - among others, conservation properties and monotonicity (a property that helps to preserve physical results: e.g., results are not non-negative nor unbounded).

As described by LR (2005), there are many simple, inexpensive schemes that can be implemented into code that simulates fluid transport. The centered (leapfrog) scheme and the upwind scheme are two simple examples: the centered scheme averages values from the two cells that border an edge, and the upwind scheme approximates the transported flux from the donor, or upwind cell. However, these methods provide unphysical results, resulting from their excessively diffusive (in the case of the upwind scheme) and non-monotonic (in the case of the leapfrog scheme) natures. Another, higher-order method is flux-corrected transport (FCT), which attempts to correct these unphysical results by having a low-order flux, preserving monotonicity, and a higher-order, correc-

tive flux, to counteract diffusion (LR, 2005).

The approach LR (2005) took to simulate accurate fluid transport is to define an accurate departure region. They developed a method called incremental remapping. This scheme defines cell-averaged tracer values at cell centers, and vector quantities (such as velocity) at cell corners. By backward integrating the fluid velocities, one defines a departure region (Fig. 2). This region contains the material that will be transported into the cell at the next time step. Although LR (2005) acknowledged that their scheme has a higher startup time at the beginning of a timestep, it was less expensive than the FCT method when running a program with a high number of tracers. However, implementing this scheme into code can prove to be expensive as the departure region can fall into several grid cells.

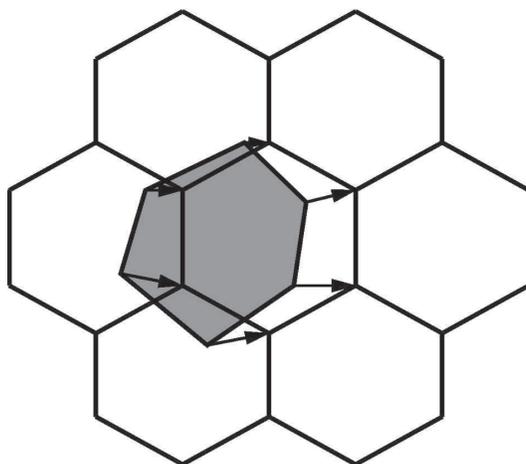


Fig. 2. Depiction of a sample departure region from Lipscomb and Ringler (2005).

Miura (2007) presented a much simpler method that avoids this weakness. Like LR (2005), Miura defined scalars at cell centers and vectors at cell corners. The corner-defined velocities are averaged to produce an edge-averaged velocity defined at the midpoint of a face. Backwards integrating this velocity will produce a parallelogram. Furthermore, Miura (2007) assumes that this parallelogram can exist in only one grid cell, making it much simpler to implement than the method provided by LR (2005). Miura (2007) showed that his method was just as accurate as the incremental remapping presented by LR (2005), through use of one of the shallow water test cases. However, while simple, the scheme breaks down in more complex, sophisticated cases.

We herein further develop the scheme introduced by Miura (2007). Its simplicity is attractive, and increasing its accuracy could make the method more effective and robust. While the approach LR (2005) took to increasing accuracy was to define a precise departure region, we take the simpler departure region presented by Miura (2007), and increase the accuracy by focusing on the distribution of the tracer function. Currently, both LR (2005) and Miura (2007) use a tracer distribution function accurate to

1st-order. Our expectation is that the efficient definition of Miura's departure region, coupled with the accuracy of a 2nd-order tracer distribution function, will provide an efficient and powerful method. The simpler scheme will not be expensive to run, but will still retain a high level of accuracy due to the higher-order distribution function.

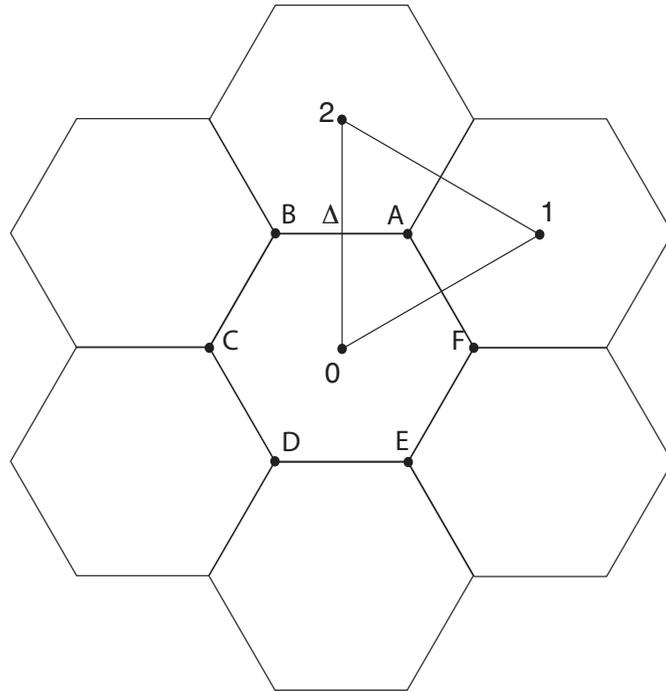


Fig. 3. A sample of our grid. Numbered points are hexagon centers, and lettered points are corners.

2. Methods

The grid mesh used in this study to implement and test both Miura's scheme and our extension is a planar and periodic perfect hexagonal grid (Fig. 3). Scalar, cell-averaged values are stored at the centers of grid cells (points 0, 1, 2, etc.), and fluid velocities are stored at the midpoints of cell faces. First, we implement Miura's method. To do this, we calculate the coefficients in the Taylor series expansion for the tracer distribution,

$$S(x, y) = S_o + S_x x + S_y y + \frac{1}{2}(S_{xx}x^2 + 2S_{xy}xy + S_{yy}y^2) + \Delta(x^3, y^3), \quad (3)$$

where subscripts x and y denote differentiation, and S_o denotes the cell-averaged value.

The process to calculate the linear terms of (3) is straightforward, as seen in (Fig. 3). For perfect planar hexagons

$$S_{xA} = \frac{S_1 - \frac{1}{2}(S_2 - S_0)}{\frac{\sqrt{3}}{2}\Delta}$$

and

$$S_{yA} = \frac{S_2 - S_0}{\Delta}.$$

This procedure is similar for the other corners. By averaging the values for each cell corner (A through F in Fig. 3), we obtain a value for each hexagon center. Then, we define a departure parallelogram, as presented by Miura (2007), and depicted in Fig. 4a. Point 0 is the center of the hexagon, R is the midpoint of the face, and P is the centroid of the departure parallelogram. Following Miura,

$$P = R - v_R^{t+\frac{\Delta t}{2}} * \frac{\Delta t}{2},$$

and we can then calculate the flux going through each face (2).

With this foundation, we implement a higher order scheme by finding the quadratic terms in the scalar distribution. Taking the first derivative values, defined at cell corners, we employ Green's Theorem (Fig. 5), to compute the cell-averaged values of the derivatives:

$$\oint_C M dx = \int \int_R \left(-\frac{\partial M}{\partial y}\right) dx dy, \quad (4)$$

$$\oint_C N dy = \int \int_R \frac{\partial N}{\partial x} dx dy. \quad (5)$$

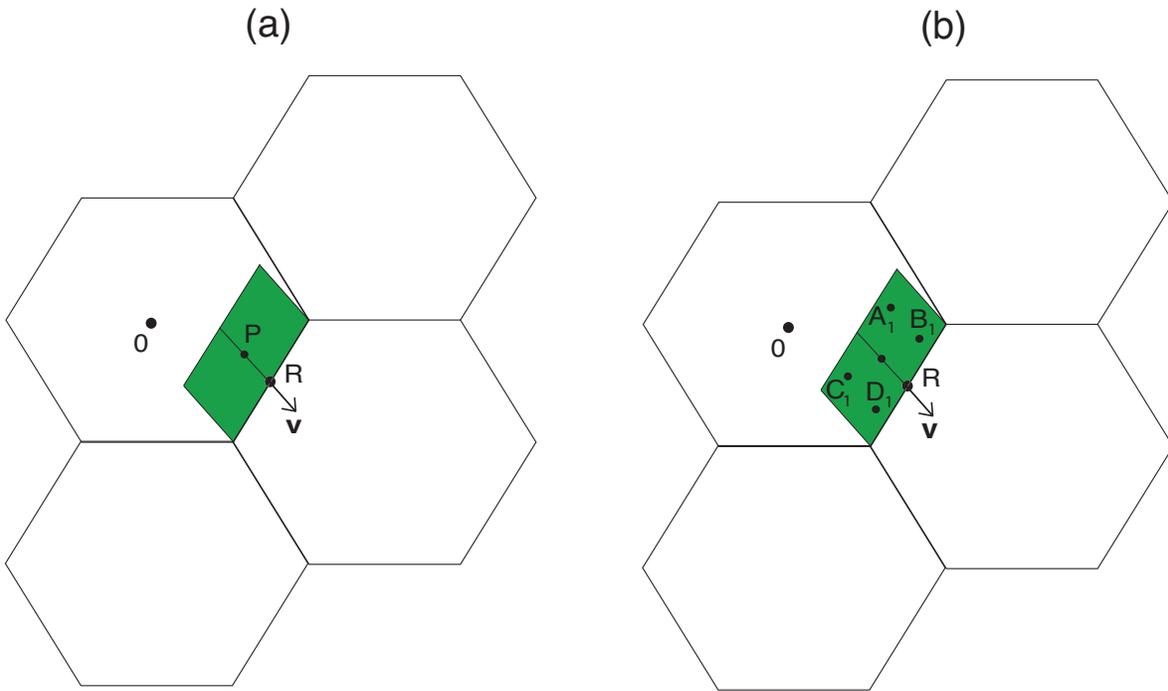


Fig. 4. The departure parallelogram for Miura's scheme (a) and the extended scheme (b).

By averaging the first-derivative values at each cell corner, we define a face-averaged value. We then use these values to compute the second derivative coefficients of this Taylor expansion. For example, to obtain S_{xx} , we choose $N = S_x$, and integrate the left hand side of (5) counter-clockwise around the hexagon to obtain $(S_{xx} \cdot \text{the cell area})$, the right hand side of (5). Numerically, this amounts to

$$S_{xx} = \frac{1}{\text{area}_{hex}} \left(S_{xAF} * \frac{\sqrt{3}}{2} a + S_{xBA} * 0 * a + S_{xCB} * \frac{-\sqrt{3}}{2} a + S_{xDC} * \frac{-\sqrt{3}}{2} a + S_{xED} * 0 * a + S_{xFE} * \frac{\sqrt{3}}{2} a \right).$$

The procedure is similar to find S_{yy} and the cross derivative terms S_{yx} and S_{xy} . The reason for using Green's Function to calculate these second-order terms is a matter of efficiency. Many models employ parallel computing during runs, and by minimizing the stencil of cells necessary to compute coefficients, the chances processors must exchange information is low, cutting down on computing time.

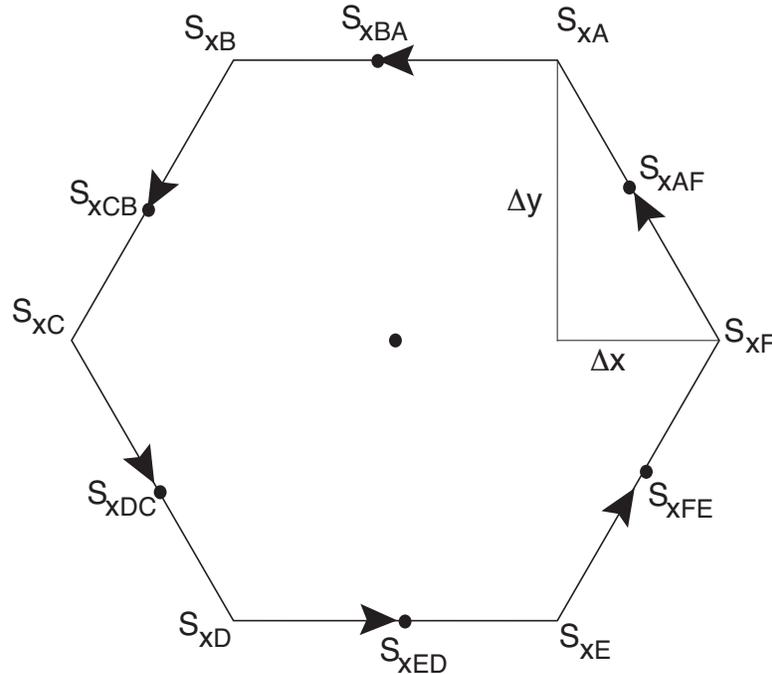


Fig. 5. Green's Theorem on the Hexagon. Values at adjacent corners are averaged to obtain edge-averaged values defined at the centers of every face.

In order to properly implement the scheme, this article uses Gauss quadrature in order to fit the quadratic function (Fig. 6). By evaluating and averaging the tracer distribution at the four quadrature points, we can obtain the flux through a face. ($flux = \mathbf{u} * \Delta t * a * S_{trans}$, $S_{trans} = \frac{1}{4} * (S(Q_{A1}) + S(Q_{B1}) + S(Q_{C1}) + S(Q_{D1}))$.) (Fig. 4b.) However, a correction must be made. To maintain accuracy, it is required that integrating the function for S returns just the constant term, because this value is the cell-averaged tracer

concentration defined in the code. Therefore, no higher-order terms must appear when integrating the tracer distribution in the departure region. The integral of the linear and cross terms vanish, but the integrals of S_{xx} and S_{yy} do not. As a result, we redefine S_o such that

$$\bar{S}_o^* = \bar{S}_o - \frac{1}{area} \frac{1}{2} \left(\int_A (S_{xx}x^2)dA + \int_A (S_{yy}y^2)dA \right).$$

This correction term allows the scheme to retain favorable properties of Miura's lower order method.

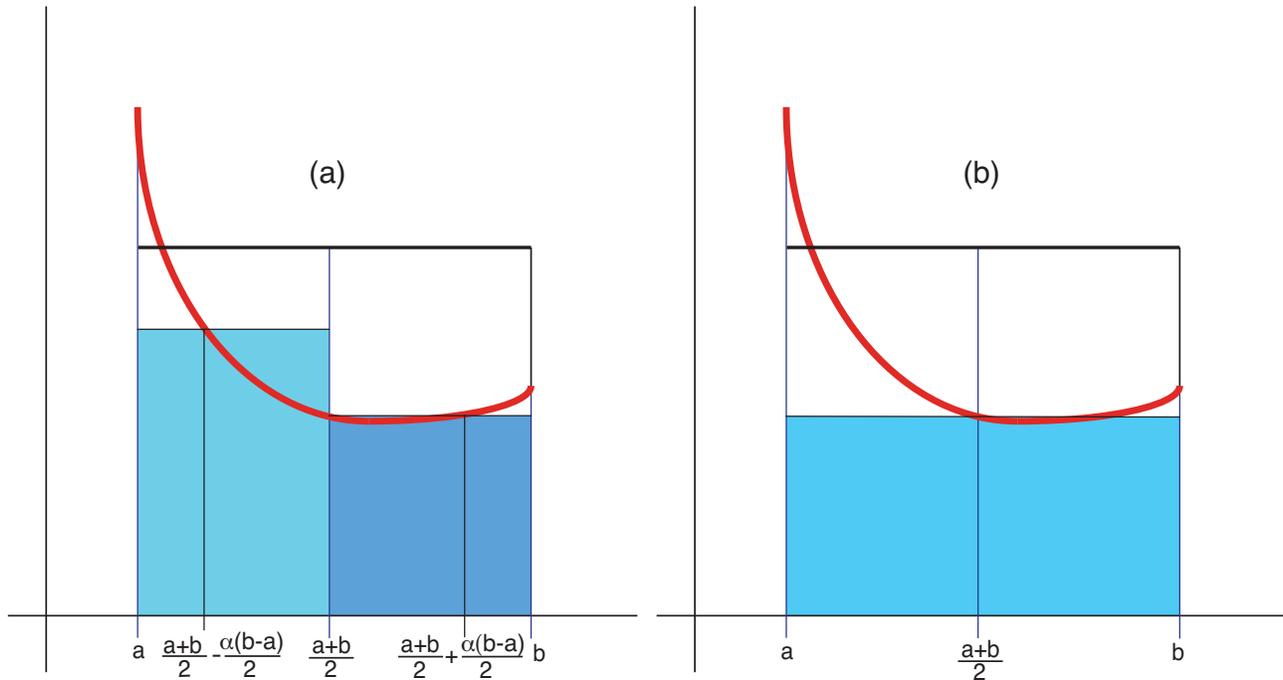


Fig. 6. Gauss Quadrature (a) implemented on a one-dimensional quadratic function. Note that it is much more accurate than the midpoint method on the right (b).

3. Test Cases

In order to compare our extension to Miura's scheme, this article analyzes two test cases. The first test case is simple solid body rotation (Fig. 7). The velocity field is linearly dependent on r , the distance from the origin of the grid. To ensure a divergence-free velocity field, we define a streamfunction ψ , such that

$$\psi = \frac{4\pi}{T} \left(\frac{r^2}{2} \right),$$

where T is the total integration time. This function is implemented by defining it at the center of every grid cell, and taking advantage of the properties $\psi_y = u, \psi_x =$

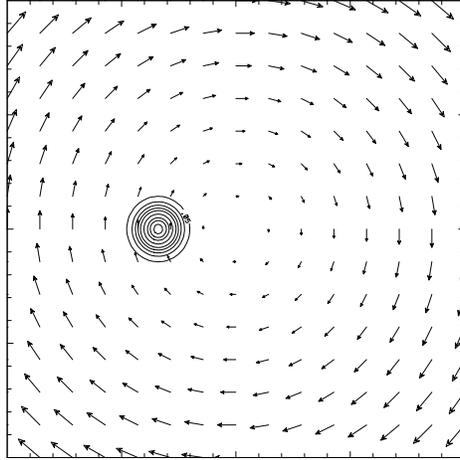


Fig. 7. The velocity field for solid body rotation, with tracer configuration.

–*v*. The other test case run is an extension of solid body rotation, developed by Blossey and Durran (2008), and colloquially referred to as "Washing Machine" rotation. This introduces a new term into the streamfunction:

$$\psi = \frac{4\pi}{T} \left(\frac{r^2}{2} + \cos\left(\frac{2\pi t}{T}\right) \left[\frac{r^2}{2} + \frac{1}{96} \ln(1 - 16r^2 + 256r^4) - \frac{1}{48} \ln(1 + 16r^2) - \frac{\sqrt{3}}{48} \tan^{-1}\left(\frac{-1 + 32r^2}{\sqrt{3}}\right) \right] \right).$$

The scalar contours are perfectly circular at time 0. At $T/4$, the bell has been distorted into a crescent, with the highest concentrations of tracer at the tail end. By $T/2$, the bell has been returned to the initial spot; the analytical solution should exactly match the initial conditions. The scalar concentration at $3T/4$ is similar to the concentration at $T/4$, but the highest concentrations of tracer are now at the head of the bell. At the end of the integration, the solution should match the initial conditions (Fig. 8).

The comparison is done by calculating L_2 and L_{inf} error norms. These values compare the analytic solution to the numerical solution. The L_2 norm is summed over every grid cell, and is defined to be

$$L_2 = \sqrt{\sum_{cells} \frac{(S - S_{true})^2}{Number\ of\ Cells}}, \quad (6)$$

where S_{true} is the analytic solution. The L_{inf} norm is calculated as the maximum (positive) difference between the numerical and analytic solution,

$$L_{inf} = \max(abs(S - S_{true})). \quad (7)$$

Convergence tests were run with (6) and (7) on grids of various densities. The grids are rectangular, and the densities were varied by factors of two. Densities of $nx = 43$,

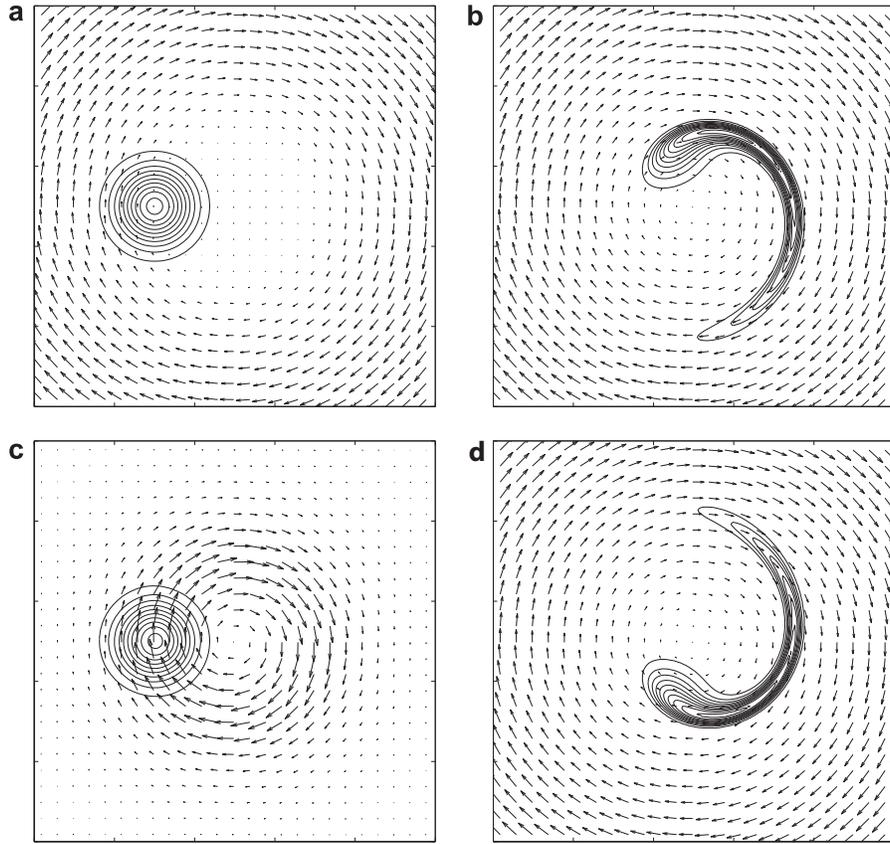


Fig. 8. The velocity field and tracer configuration for Blossey and Durran (2008) rotation. Configurations shown are at (a) $t=0$, (b) $t=T/4$, (c) $t=T/2$, and (d) $t=3T/4$.

85, 169, 337, and 673 were tested, where nx is the number of cells lengthwise on the grid. For the purposes of stability, the time step was also varied, in order to keep the Courant number (a measure of the stability of a numerical scheme) constant.

$$C = \frac{u\Delta t}{\Delta x}$$

Since nx is inversely proportional to the width of a grid cell, a constant Courant number would require that an increase in nx corresponds to a decrease in the timestep.

4. Results

Final results for the Miura scheme and the extended scheme are displayed for $nx = 169$ in Fig. 9 (Solid Body Rotation) and Fig. 10 (Washing Machine Rotation). Our extension of Miura's scheme removes the trailing edge apparent in Miura's scheme in solid

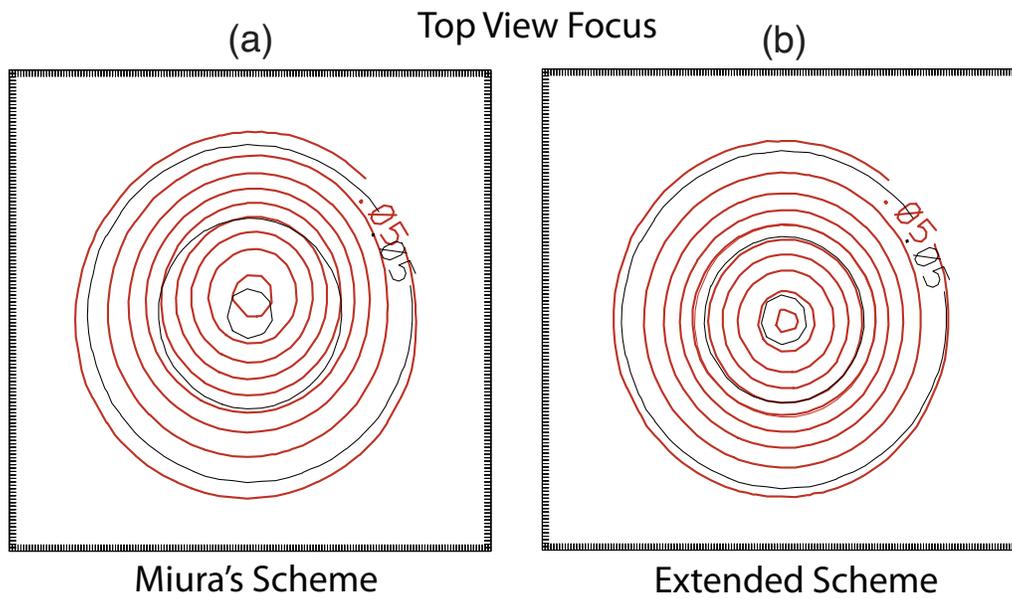


Fig. 9. Final results for $n_x=169$ of (a) Miura's Scheme and (b) the Extended Scheme for solid body rotation. The exact solution, outlined in black, is overlaid over the numerical solution, in red.

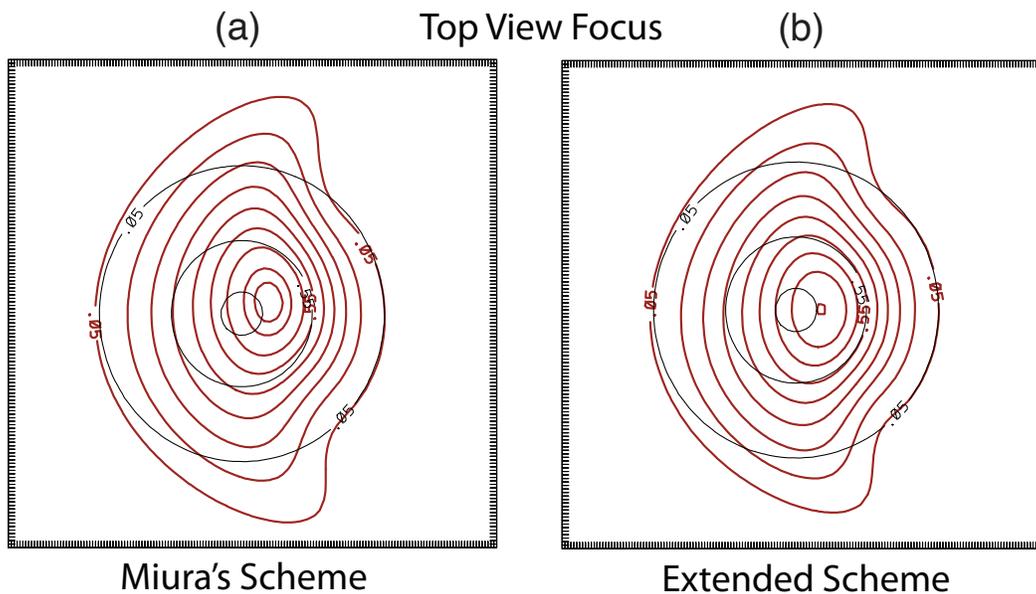


Fig. 10. Final results for $n_x=169$ of (a) Miura's Scheme and (b) the Extended Scheme for time dependent, Blossey and Durran (2008) rotation. The exact solution, outlined in black, is overlaid over the numerical solution, in red.

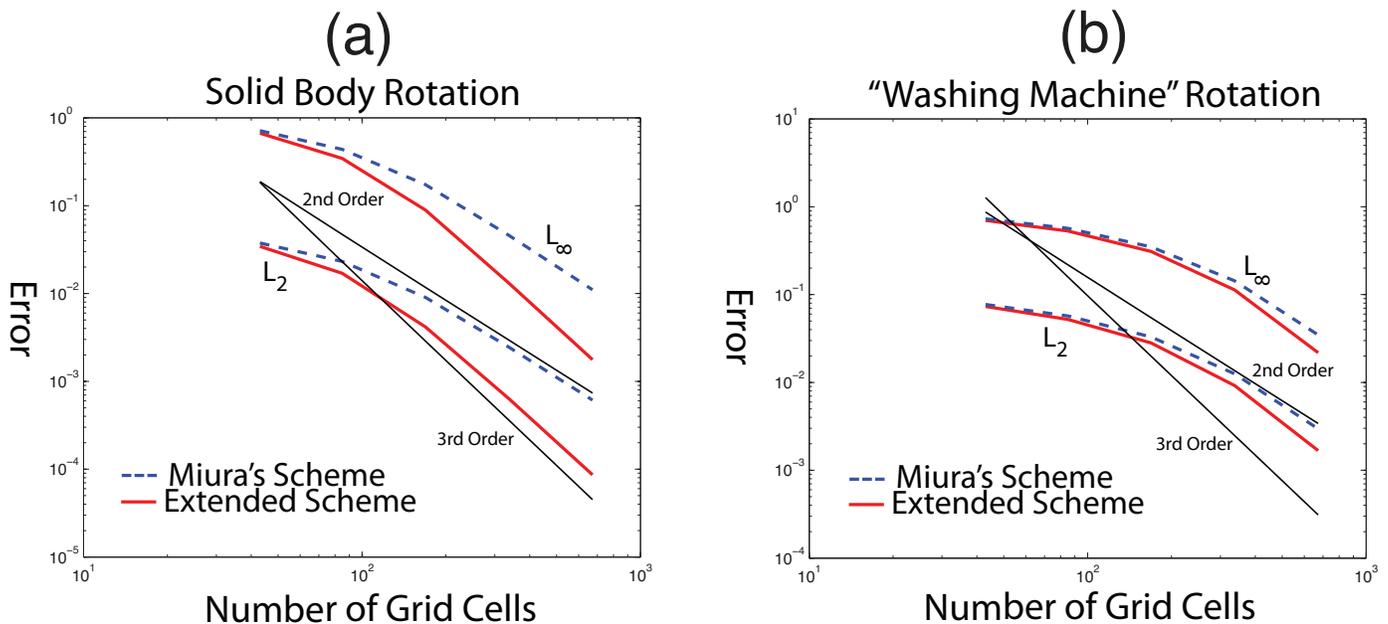


Fig. 11. Convergence results for (a) solid body rotation and (b) Blossey and Durran (2008) rotation. The thin solid black lines display second and third order convergence.

body rotation (Fig. 9). We also note that the extension is less diffusive than Miura's scheme, and does not over shoot the analytical solution. However, there are less discernible differences between the two schemes when running the washing machine rotation (Fig. 10).

Results from the convergence graph results for both Miura's scheme and this higher-order scheme are displayed in Fig. 11. Fig. 11a depicts the results from the solid body rotation case, while Fig. 11a displays results from the washing machine rotation. For solid body rotation, the disparities between Miura's scheme and the extension are readily apparent (Fig. 11a), While Miura's scheme is second-order convergent, our extended scheme quickly approaches third-order convergence. These differences are less dramatic in the rotational case from Blossey and Durran (2008) (Fig. 11b); the extension does not approach third-order convergence as quickly as in the solid body rotation case.

Nonetheless, this new scheme is indeed third-order convergent. By expanding Miura's scheme to include second-order tracer terms, the solutions are more accurate than solutions produced by the Miura (2007) scheme. However, as seen in the convergence tests (Fig. 11), with more complex configurations of the tracer and velocity flow, the improvements over the Miura scheme are not as pronounced. This possibly suggests that the assumptions Miura makes in defining the departure region are too large. It is possible that by assuming the departure region is a parallelogram, it limits the level of accuracy that can be achieved.

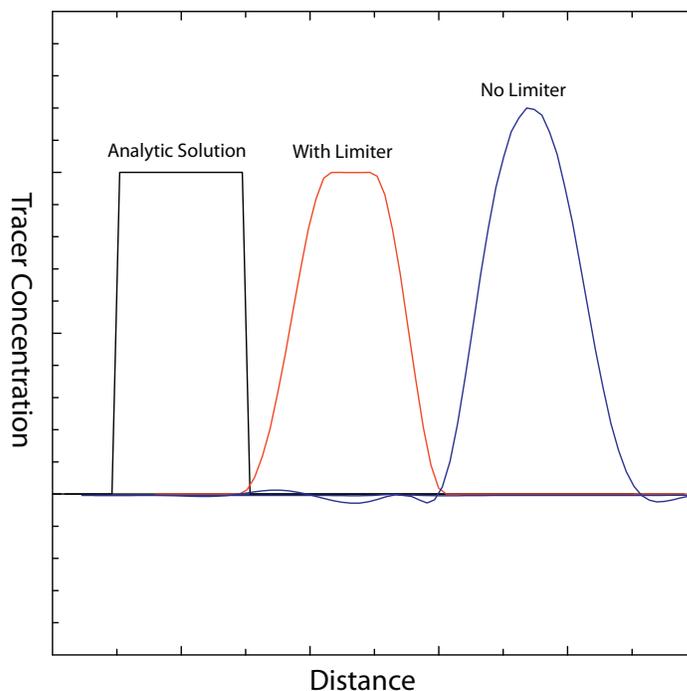


Fig. 12. Results from the flux limiter. The analytic solution is on the left, the middle solution is with the flux limiter, and the right hand solution removes the flux limiter.

Besides extending the Taylor expansion, further improvements can be added to the scheme to extend it and make it more robust. A flux limiter constraint was added to the scheme by imposing a minimum and maximum on the tracer concentration. If $S^{t+\Delta t}$ is greater than (less than) S_{max} (S_{min}), the fluxes are scaled so that no cell contains more (less) tracer than the maximum (minimum) value. Results from a solid body rotation test are depicted in Fig. 12.

The scheme developed by Miura (2007) is a simple one that is easy to implement into code. This present study developed a method that extended Miura's scheme by developing the Taylor expansion. So far, the results have shown this extension is a viable one that is both accurate and efficient, and worthy of improvements. Based on these encouraging results, next we must implement the scheme on a hexagonal grid over a sphere. This is the first step towards preparing this method for use in weather and atmospheric models used by the scientific community and beyond. Provided as an option in these models, it is possible that this balance between accuracy and efficiency could help to make these models more attractive to a wider audience. As the next generation of atmospheric models are developed, we hope that this new scheme proves to be an important and valuable method.

REFERENCES

Blossey, P. N., and Durran, D. R., 2008: Selective Monotonicity Preservation in Scalar Advection. *Journal of Computational Physics*, **227**, 5160-5183

Lipscomb, W.H., and Ringler, T.D., 2005: An Incremental Remapping Transport Scheme on a Spherical Geodesic Grid. *Monthly Weather Review*, **133**, 2335-2350.

Miura, H., 2007: An Upwind-Biased Conservative Advection Scheme for Spherical Hexagonal-Pentagonal Grids. *Monthly Weather Review*, **135**, 4038-4044