Pure appl. geophys. 163 (2006) 1699–1735
0033–4553/06/081699–37
DOI 10.1007/s00024-006-0079-1

Pure and Applied Geophysics

A Godunov-Type Scheme for Atmospheric Flows on Unstructured Grids: Scalar Transport

Nash'at Ahmad,[1] Zafer Boybeyi,[2] Rainald Löhner,[2]
and Ananthakrishna Sarma[1]

Abstract—This is the first paper in a two-part series on the implementation of Godunov-type schemes on unstructured grids for atmospheric flow simulations. Construction of a high-resolution flow solver for the scalar transport equation is described in detail. Higher-order accuracy in space is achieved *via* a MUSCL-type gradient reconstruction after van Leer and the monotonicity of solution is enforced by slope limiters. Accuracy in time is maintained by implementing a multi-stage explicit Runge-Kutta time-marching algorithm. The scheme is conservative and exhibits minimal numerical dispersion and diffusion. Five different benchmark test cases are simulated for the validation of the numerical scheme.

1. Introduction

Atmospheric processes span a multitude of spatial and temporal scales. This wide range of scales requires accurate multiscale calculations in atmospheric models (Prusa and Smolarkiewicz, 2003). The current models used in atmospheric flow simulations, however, still remain scale-specific (Skamarock and Klemp, 1993). There is a general consensus that the simulation of the atmosphere at all relevant scales is an intractable problem. Nevertheless, atmospheric models should be capable of predicting three-dimensional, time-dependent mean flow and turbulence fields over complex terrain in an unsteady synoptic environment. In addition, the models should have a sufficient grid resolution to account for local scale phenomena, mean planetary boundary layer structures, and significant terrain and other land-use inhomogeneities. Consequently the eventual predictions from the models are closely related to the completeness of model physics, dynamics, resolution of the computational mesh, and the accuracy of the numerical techniques used.

Optimizing the degree of grid refinement and the choice of numerical techniques are usually constrained by CPU restrictions and by run-time expectations. This

[1] Center for Atmospheric Physics, Science Applications International Corporation, McLean, VA 22102, U.S.A. E-mail: ahmadn@saic.com
[2] School of Computational Sciences, George Mason University, 4031 University Drive, MS 5B2 Fairfax, VA 22030-444, U.S.A.

becomes critical for real-time flow predictions. For example, introducing fine spatial grid resolution throughout the simulation domain is not practical, since the size of the modeling domain, the interactions between the various atmospheric processes, and the computational cost of the numerical algorithms place restrictions on the grid resolution that can be used in current computers. These limitations prohibit the use of uniform high spatial resolution that is appropriate to resolve the smallest scales of interest. The alternative is then to develop methodologies capable of providing variable local refinement in regions of interest.

Different grid techniques have been applied in atmospheric simulations to deal with multiscale events in atmospheric motion. Among the most widely-used techniques, the grid nesting technique has been implemented extensively to provide increased spatial resolution in regions of interest without requiring a fine mesh throughout the entire domain (CLARK and FARLEY, 1984). This technique involves the sequential placement of multiple high-resolution meshes in the desired regions of the domain. Although the decision to spawn one or more submeshes is typically subjective and manually directed, many formulations have been developed to allow the submeshes to move with particular features in the flow (KURIHARA and BENDER, 1980). One major problem with this technique, however, is the interaction among multiple nested meshes, particularly the tendency of propagating waves to discontinuously change their speeds upon passing from one nest to the next and to reflect off the boundaries of each nested grid. In addition to the grid-nesting technique, different authors (e.g., STANIFORTH *et al.*, 1978; COTÉ *et al.*, 1993) have also used the variable grid resolution technique for atmospheric flow calculations to avoid the problems of wave reflections.

In recent years, unstructured grids have been used for three-dimensional atmospheric modeling (e.g., BACON *et al.*, 2000). Although, the unstructured grid technique is a relatively new method for the atmospheric science community, it has been widely used in other scientific disciplines (e.g., LÖHNER, 2001; LUO, *et al.*, 2003; LOTTATI and EIDELMAN, 1994). The primary benefit of the unstructured grid technique over a conventional structured grid lies in its ability to accurately discretize complex topologies with relative ease. Meshes for arbitrary surfaces and volumes in three dimensions can be generated. This capability is essential for resolving complex terrain features and shoreline boundaries for mesoscale and urban-scale atmospheric modeling. In addition, computational efficiency can be achieved by providing variable and continuous resolution throughout the computational domain, with a high mesh resolution only in regions of interest. This feature of the unstructured grid technique effectively removes the wave reflection problems that are common in grid-nesting techniques. Solution-adaptive techniques (i.e., dynamic grid adaptation) can also be implemented with relative ease to improve the solution by dynamically adapting the mesh to evolving physical features. Solution-adaptive methods can be helpful, for example, in predicting the trajectory of hazardous materials (AHMAD *et al.*, 1998; SARMA *et al.*, 1999; GHORAI *et al.*, 2000), simulations of convective weather systems

such as squall lines and mesoscale convective complexes, and for tracking hurricanes (GOPALAKRISHNAN *et al.*, 2002).

Traditionally, finite-difference discretizations of centered schemes such as the Leapfrog scheme have been favored for discretizing the atmospheric flow equation set. These types of schemes have large amounts of numerical dispersion errors (non-physical spurious oscillations), which can contaminate the numerical results (CARPENTER *et al.*, 1990). Furthermore, the scheme can introduce false negatives in important scalar microphysical quantities. To avoid false negatives either positive definite schemes (SMOLARKIEWICZ, 1984; BOTT, 1989) or Flux Corrected Transport (FCT)-type schemes (BORIS and BOOK, 1973; ZALESAK, 1979) are sometimes used to advect scalar quantities. In this study, high-resolution Godunov-type methods are explored for solving the equations arising in atmospheric flows. These finite volume discretizations are conservative and have the ability to resolve regions of steep gradients accurately, thus minimizing dispersion errors in the solution. CARPENTER *et al.* (1990) applied the method to atmospheric flows in two dimensions using an exact Riemann solver in conjunction with the Piecewise Parabolic Method (COLELLA and WOODWARD, 1984) to obtain higher-order spatial accuracy on structured grid. CARPENTER *et al.* (1990) gave a comprehensive review of the Riemann problem, its application in the Godunov's method to solve the atmospheric flow problems, and a comparison of the method with the MPDATA (SMOLARKIEWICZ, 1984) and the Leapfrog schemes. In recent years, different authors have explored the possible use of alternative finite volume schemes for atmospheric modeling (e.g., BOTTA *et al.*, 2004; HUBBARD and NIKIFORAKIS, 2003; HOURDIN and ARMENGAUD, 1999; PIETRZAK, 1998; LIN *et* al., 1994; MÜLLER, 1992).

This study aims at developing a Godunov-type scheme for atmospheric flows on unstructured meshes. In this paper, the construction of a high-resolution flow solver for the scalar transport equation is described in detail. Higher-order accuracy in space is achieved *via* a Monotone Upstream centered Scheme for Conservation Laws (MUSCL)-type gradient reconstruction after VAN LEER (1979) and the monotonicity of the solution is enforced with the help of slope limiters. The numerical scheme is then evaluated using the standard tests for benchmarking atmospheric advection solvers. The development and implementation of an approximate Riemann solver for the Euler equations governing atmospheric flows will follow in a later publication.

2. Unstructured Grids

The unstructured grid technique is a relatively new method for the atmospheric science community. As mentioned before, the primary benefit of the unstructured grid technique lies in its ability to discretize complex topologies with relative ease. In addition, computational efficiency can be achieved by providing variable and continuous resolution throughout the computational domain, with a high mesh resolution only in regions of interest. Solution-adaptation techniques are also

relatively easy to implement on unstructured grids. For example, BACON *et al.* (2000) have used unstructured meshes with both static and dynamic grid adaptation for atmospheric modeling in a fully compressible and nonhydrostatic model – the Operational Multiscale Environment model with Grid Adaptivity (OMEGA). The OMEGA model is based on unstructured prisms and has been applied to many atmospheric problems and validated extensively (e.g., BOYBEYI *et al.*, 2001; GOPALAKRISHNAN *et al.*, 2002). SARMA *et al.* (1999) have demonstrated significant improvement in solution accuracy by using dynamic grid adaptation for predicting chemical plume concentrations. VARVAYANNI *et al.* (1999) have used unstructured prisms in a diagnostic model, which reads in a flow field and interpolates it over the mesh to predict trajectory of tracers. In their case, they take advantage of the ability of the unstructured grid to resolve the underlying terrain in a more realistic and efficient manner. GHORAI *et al.* (2000) have used tetrahedral meshes to provide solution-adaptation in both horizontal and vertical for atmospheric dispersion calculations. BEHRENS *et al.* (2000) have implemented a semi-Lagrangian advection scheme on unstructured adaptive grids. These applications have shown the various inherent strengths of unstructured grids such as the better representation of topography, computational efficiency (via static or dynamic grid adaptation), and the flexibility of the grid to resolve multiple scales.

Solution-adaptive methods have also been implemented on structured grids. The implementation of solution-adaptive techniques (dynamic grid adapatation) on structured meshes for complex geometries, however, is non-trivial and so far the attempts for atmospheric applications have been limited to idealized problems (e.g., DIETACHMAYER and DROEGERMEIER, 1992; FIEDLER and TRAPP, 1993; ISELIN *et al.*, 2002). Another promising technique for providing variable mesh resolution is the Adaptive Mesh Refinement (AMR) method, which has been implemented for atmospheric applications by different authors (SKAMAROCK and KLEMP, 1993; HUBBARD and NIKIFORAKIS, 2003; JABLONOWSKI *et al.*, 2004).

In a structured mesh, the numbering of nodes and cells is ordered. For example, in a two-dimensional rectangular mesh, it is implicitly understood that the neighbors of node (i, j) are nodes $(i-1, j)$, $(i+1, j)$, $(i, j-1)$, and $(i, j+1)$. In an unstructured mesh on the other hand, the numbering is not ordered and therefore the connectivity information needs to be stored in arrays and requires linked lists. An unstructured triangular mesh can be defined by a list of cells holding the connectivity information (identification number for each node of the cell) and another list of positions for each node of the mesh. Usually more connectivity information is needed during the mesh-generation process and for flow-solvers, in which fluxes across edges of the control volume need to be calculated. These additional data structures can be derived from the basic mesh definition (Figs. 1 and 2 show some of the derived data structures). There are a number of ways in which data structures can be defined to represent an unstructured mesh. The choice of data structures may depend on the type of numerical discretization (e.g., node-based or cell-based control volume). One

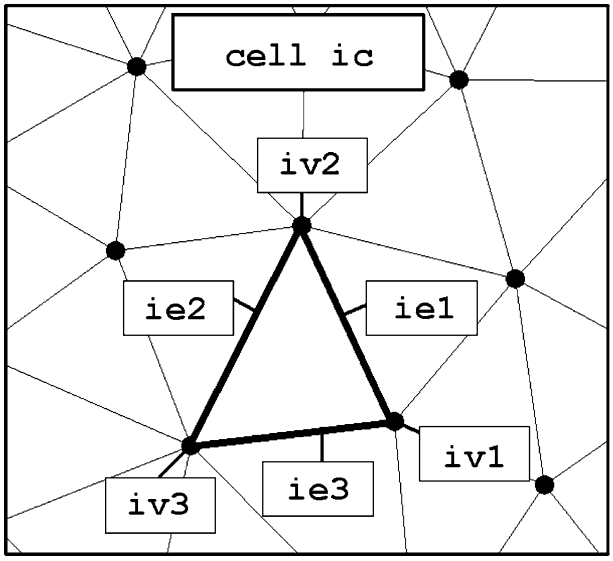


Figure 1
Cell connectivity information. Edges (*ie1*, *ie2*, and *ie3*) and nodes (*iv1*, *iv2* and *iv3*) of the cell *ic* are shown in the figure.

disadvantage in using unstructured meshes is the additional computational cost due to indirect accessing of data. It has been shown that this disadvantage can be minimized by using appropriate data structures and by implementing renumbering strategies (LÖHNER, 2001)—thus avoiding a large number of cache-misses. A detailed description of the data structures used in this study is given below:

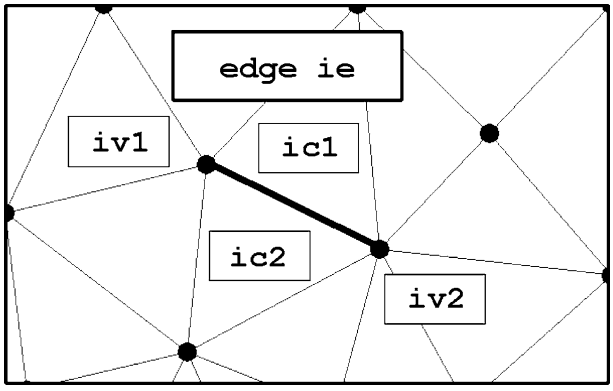


Figure 2
Edge-based data structure. The cells (*ic1* and *ic2*) on either side of the edge *ie* and the starting and ending points (*iv1* and *iv2*) of the edge *ie* are shown.

a. Cell Connectivity

The array `jcell` holds the connectivity information for each cell of the unstructured mesh. The information contains the numbering of nodes (*iv1*, *iv2* and *iv3*) and the numbering of edges (*ie1*, *ie2* and *ie3*) of each cell. A counter-clockwise numbering of nodes is assigned to a cell (Fig. 1).

$$\texttt{jcell}\ (\texttt{ic},1) = \texttt{iv1} - \texttt{identification number of node 1 of cell ic}$$
$$\texttt{jcell}\ (\texttt{ic},2) = \texttt{iv2} - \texttt{identification number of node 2 of cell ic}$$
$$\texttt{jcell}\ (\texttt{ic},3) = \texttt{iv3} - \texttt{identification number of node 3 of cell ic}$$
$$\texttt{jcell}\ (\texttt{ic},4) = \texttt{ie1} - \texttt{identification number of edge 1 of cell ic}$$
$$\texttt{jcell}\ (\texttt{ic},5) = \texttt{ie2} - \texttt{identification number of edge 2 of cell ic}$$
$$\texttt{jcell}\ (\texttt{ic},6) = \texttt{ie3} - \texttt{identification number of edge 3 of cell ic}$$

b. Edge Connectivity

The array `jedge` holds the connectivity information for each edge of the unstructured mesh. The information contains the numbering of nodes (*iv1* and *iv2*), the cells to the left and the right of each edge (*ic1* and *ic2*) and an identifier for the edge type (Fig. 2).

$$\texttt{jedge}\ (\texttt{ie},1) = \texttt{iv1} - \texttt{node 1} - \texttt{starting point of edge ie}$$
$$\texttt{jedge}\ (\texttt{ie},2) = \texttt{iv2} - \texttt{node 2} - \texttt{ending point of edge ie}$$
$$\texttt{jedge}\ (\texttt{ie},3) = \texttt{ic1} - \texttt{cell to the left of the edge ie}$$
$$\texttt{jedge}\ (\texttt{ie},4) = \texttt{ic2} - \texttt{cell to the right of the edge ie}$$
$$\texttt{jedge}\ (\texttt{ie},5) = \texttt{edge type}\ (\texttt{interior},\ \texttt{wall or far} - \texttt{field})$$

By defining a starting and an ending point for an edge, a sense of direction is given to each edge. Edge, *ie*, is shared by both cells *ic1* and *ic2*. If the counter-clockwise convention of node numbering for a cell is followed, then for one of the cells the edge direction *iv1* to *iv2* will be along the counter-clockwise direction (cell on the left). The cell on the right is the one with its numbering going against the edge direction. The cells surrounding a cell can also be found from this information. The edge-type is required, since the calculation of fluxes across the domain boundaries can differ from the calculation across interior edges. Three different edge-types are defined; 1) interior, 2) solid wall and 3) farfield. The farfield boundary can be either inflow or outflow depending on the direction of flow. The definition of edge-based data structures is essential for unstructured calculations. Several authors (e.g., LÖHNER, 2001) have shown that the edge-based flow solvers are much more efficient compared to cell-based solvers. It is a worthwhile effort to ensure that most calculation loops are over the edges. This results in an overall increase in speed of code execution.

c. Node Connectivity

The node connectivity information consists of linked lists, which identify the nodes surrounding a node and cells surrounding a node. These lists can easily be

generated from the basic mesh definition, using algorithms described by LÖHNER (2001).

d. Mesh Geometry

Once the data structures have been defined, basic geometric quantities of the mesh are calculated. The Cartesian coordinate system is used in the present study. For the two-dimensional unstructured triangular mesh the following quantities are required:

a) Node coordinates.
b) Cell areas.
c) Edge lengths.
d) Normals pointing outwards from each edge.
e) Distances between cell centers on either side of an edge.
f) Distances from the cell center to the midpoint of each of its edges.
g) Distances from the cell center to the point of intersection of edge and the line connecting the centers of cells on the either side of the edge. This distance is used in the calculation of the time step.

Over the last two decades the unstructured grid technology has seen major advances and there is an immense amount of literature on different techniques for generating grids for complex geometries. For a detailed review one can refer to LÖHNER (2001) and references therein.

3. Numerical Scheme

The 2-D scalar transport equation can be written in the conservative form as:

$$\frac{\partial Q}{\partial t}+\frac{\partial F}{\partial x}+\frac{\partial G}{\partial y}=0 \tag{1}$$

where,

$$Q=q, \qquad F=qu\,, \qquad G=qv \tag{2}$$

q is the conserved scalar quantity, u is the velocity component in the x-direction and v is the velocity component in the y-direction. The scalar transport equation is solved using a higher-order Godunov-type scheme (GODUNOV, 1959; VAN LEER, 1979) on unstructured meshes in two-dimensions. These finite volume discretizations are conservative and have the ability to resolve regions of steep gradients accurately, thus minimizing numerical dispersion errors in the solution. Godunov's method assumes piecewise constant data at cell centers. The Riemann problem is solved at each cell interface and the numerical flux is constructed from it. Thus, the global solution is a set of solutions of the local Riemann problems at each cell boundary, which is then evolved in time.

The advective fluxes are calculated by summing all the incoming and outgoing fluxes through each face of the control volume. In two dimensions, Eq. (1) can be written in the integral form as:

$$\frac{d}{dt}\int_{\Omega} Q\, d\Omega = -\oint_{\Gamma} (F, G).n\, d\Gamma, \tag{3}$$

where, n is the unit normal pointing out of the control surface Γ of the control volume Ω. Figure 3 shows the cell-centered control volume, Ω with each of its control surfaces and the unit normals pointing outwards from the control surfaces. Equation (3) can be approximated directly:

$$V_{cell}\frac{dQ_{cell}}{dt} + \sum_{faces} (F, G)\cdot s = 0, \tag{4}$$

where V_{cell} is the volume of the control volume (area of the triangle in the case of the two-dimensional triangular mesh), Q_{cell} is the cell-averaged value of the conserved variable Q at cell center and s is the control surface area (edge lengths of the triangle in case of the two-dimensional mesh). For the unstructured mesh, the implementation of the method is straightforward. The flux across each edge of the cell is calculated using Godunov's method using the exact Riemann solution (TORO, 1999). The values on either side of a cell edge form the initial conditions for

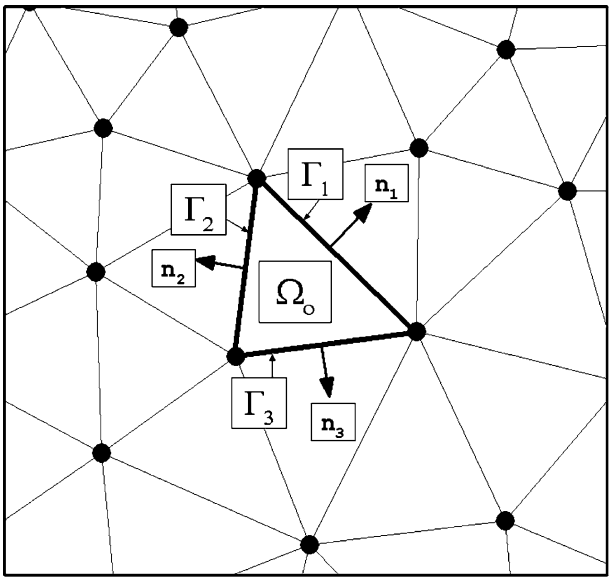


Figure 3
Cell-centered control volume Ω_o and its control surfaces, Γ_i.

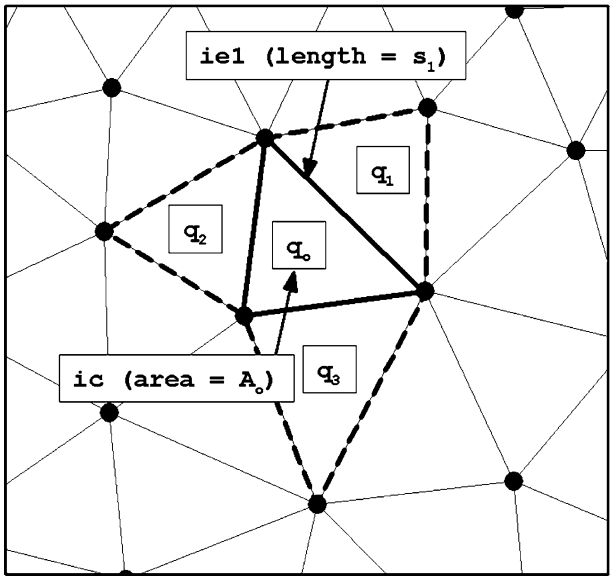


Figure 4
In a loop over edges the cell values on either side of the edge are used by the Riemann solver to calculate fluxes. Then in a loop over cells the fluxes are added to update the cell centered values.

the Riemann problem. In the loop over edges, the values of cells on either side of the edge (Fig. 4) are used to calculate the fluxes. Once the fluxes have been calculated, they are added to the cell-centered value in a loop over cells. For the second-order calculation gradient-limited extrapolated values are used in the Riemann solver instead of cell averages. The solution is marched in time using Eq. (4) within a multi-stage Runge-Kutta time marching scheme (JAMESON *et al.*, 1981). During the time integration, the local Riemann solution on one interface should not be allowed to interfere with the Riemann solution on another interface. If the interference of waves occurs, then the solution of the Riemann problem can no longer be considered local. This forms the basis of the Courant restriction on the Godunov method. The time step is calculated by finding the maximum wave speed in each cell:

$$\Delta t = CFL \cdot \frac{\Delta x}{abs(u)}, \tag{5}$$

where, u is the normal velocity at edge and Δx is the distance between the cell center and the point of intersection of the edge with the line connecting the cell centers on either sides of the edge (see Fig. 5). In the calculations presented in this paper, a CFL criterion of 1.0 was maintained for the first-order scheme and a CFL criterion of 0.9 was used for the higher-order scheme.

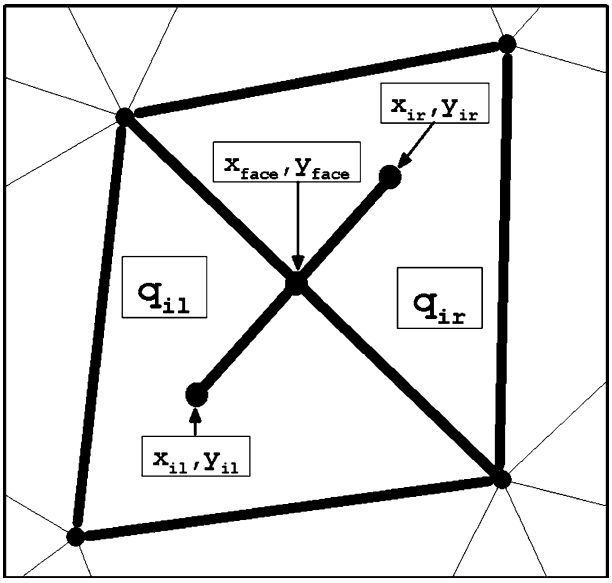


Figure 5
Cell-averaged data states to the left (q_{il}) and the right (q_{ir}) of the edge which are extrapolated to the edge. The gradient of q at the cell centers is also needed for the extrapolation.

4. Higher-Order Spatial Accuracy and Monotonicity Constraints

To increase the solution accuracy in space, VAN LEER (1979) suggested using piecewise-linear data instead of piecewise-constant data to construct fluxes from the Riemann calculation. The left and right data states are constructed from extrapolated values from cell centers to cell interfaces and then used in the solution of the Riemann problem. The reconstruction of gradients and the enforcement of monotonicity constraints are discussed below.

a. Green-Gauss Reconstruction

The Green-Gauss gradient reconstruction follows from the Gauss theorem:

$$\int_{\Omega} \nabla \cdot q \, d\Omega = \int_{\Gamma} q \cdot \vec{n} \, d\Gamma. \tag{6}$$

Numerically, the method is a combination of a *gather-scatter* process (BARTH and JESPERSON, 1989). First the cell-centered values are averaged to obtain values on edge centers. The edge values are used to calculate the integral in Eq. (6). The gradients are calculated by integrating over the edges of the control volume. Finally the gradients are scaled by area in a loop over cells. The calculation can easily be implemented following edge-based data structures for computational efficiency.

b. Linear Least-squares Reconstruction

Consider the cell-averaged value q_o and assume that the solution varies linearly over the adjacent neighbors of the cell. Then the solution can be represented as a piecewise linear polynomial (BARTH and JESPERSON, 1989):

$$q_o(x,y) = q_o + a(x - x_o) + b(y - y_o), \tag{7}$$

where x_o and y_o are the coordinates of the cell center (the baricenters of each triangle in case of the two-dimensional triangular mesh) and $(a,b)^T$ represents the solution gradient in x and y-direction, respectively. Using the weighted least-squares method the following linear system of equations can be written:

$$\begin{pmatrix} w_1(x_1 - x_o) & w_1(y_1 - y_o) \\ w_2(x_2 - x_o) & w_2(y_2 - y_o) \\ \vdots & \vdots \\ w_n(x_n - x_o) & w_n(y_n - y_o) \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} w_1(q_1 - q_o) \\ w_2(q_2 - q_o) \\ \vdots \\ w_n(q_n - q_o) \end{pmatrix}, \tag{8}$$

where, n is the number of neighbors and each equation has been multiplied by an arbitrary weighting factor, w_i. This system of equation is overdetermined and can be solved in the least-squares sense by multiplying with the matrix transpose to yield a symmetric 2×2 system:

$$\begin{pmatrix} \sum_i w_i^2 \Delta x_i^2 & \sum_i w_i^2 \Delta x_i \Delta y_i \\ \sum_i w_i^2 \Delta x_i \Delta y_i & \sum_i w_i^2 \Delta y_i^2 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} \sum_i w_i^2 \Delta x_i \Delta q_i \\ \sum_i w_i^2 \Delta y_i \Delta q_i \end{pmatrix}, \tag{9}$$

where,

$$\begin{aligned} \Delta x_i &= x_i - x_o \\ \Delta y_i &= y_i - y_o \\ \Delta q_i &= q_i - q_o \end{aligned} . \tag{10}$$

For a mesh, which does not change in time, these geometric quantities can be calculated once at the beginning of the simulation. The weights w_i are defined as:

$$w_i = \frac{1}{\sqrt{(x_i - x_o)^2 + (y_i - y_o)^2}} . \tag{11}$$

Although the method is computationally more expensive, for distorted meshes the least-squares gradient reconstruction gives better results than the Green-Gauss reconstruction. A detailed discussion on some of the limitations of least-squares method for highly stretched meshes is given in MAVRIPLIS (2003). In this study, only piecewise linear reconstructions have been used to achieve higher-order spatial accuracy. Schemes with piecewise quadratic reconstructions have also been implemented (e.g., COLLELA and WOODWARD, 1984; CARPENTER *et al.*, 1990; MITCHELL, 1994).

c. Limiters

The Godunov scheme uses cell averages for the solution of the Riemann problem across cell interfaces. The second-order accuracy in space can be achieved by constructing piecewise linear data from cell averages:

$$q_{left} = q_{il} + (x_{face} - x_{il})L_{face}\frac{\partial q_{il}}{\partial x} + (y_{face} - y_{il})L_{face}\frac{\partial q_{il}}{\partial y}, \tag{12a}$$

$$q_{rite} = q_{ir} + (x_{face} - x_{ir})L_{face}\frac{\partial q_{ir}}{\partial x} + (y_{face} - y_{ir})L_{face}\frac{\partial q_{ir}}{\partial y}, \tag{12b}$$

where, q_{left} is the extrapolated value of the conserved quantity q on the left side of the edge, the subscript il is used for the cell center quantities in the cell on the left of the edge, e.g., q_{il} is the cell-averaged value of the conserved quantity q and (x_{il},y_{il}) are the cell center coordinates of the cell on the left. Similarly, q_{rite} is the extrapolated value of the conserved quantity q on the right side of the edge, the subscript ir is used for the cell center quantities in the cell on the right of the edge, e.g., q_{ir} is the cell-averaged value of the conserved quantity q and (x_{ir},y_{ir}) are the cell center coordinates of the cell on the right (see Fig. 5). The point of intersection between the edge and the line connecting the centers of the two cells on either side of the edge is denoted by (x_{face}, y_{face}). L_{face} is the limiter on the gradient to ensure a monotonic solution.

The concept of monotonicity was introduced by GODUNOV (1959) for scalar conservation law. Godunov showed that the linear schemes which are monotone can only be first-order accurate at the most. This implies that if fluxes are computed using VAN LEER's reconstruction technique, then monotonicity cannot be enforced without the introduction of some form of nonlinearity (limiters). A scheme is monotone (HARTEN, 1983; SPEKREIJSE, 1987), if the extrapolation to the face is bounded by:

$$\min(q_i, q_{i+1}) \le q_{i+\frac{1}{2}} \le \max(q_i, q_{i+1}), \tag{13}$$

where, $q_{i+1/2}$ is the value of q at the face between cells i and $i+1$. On a structured grid the extension to higher dimensions is intuitive. The gradients can be limited in each direction by enforcing the following conditions:

$$\min(q_{i,j}, q_{i+1,j}) \le q_{i+\frac{1}{2},j} \le \max(q_{i,j}, q_{i+1,j}),$$

$$\min(q_{i,j}, q_{i,j+1}) \le q_{i,j+\frac{1}{2}} \le \max(q_{i,j}, q_{i,j+1}),$$

where, i and j are the indices for the x and y directions, respectively. BARTH and JESPERSON (1989) extended this definition to unstructured grids by using the maximum and minimum values over the cell and its neighbors to bound the extrapolated value. First the maximum and minimum values of cell and its neighbors are calculated:

$$q_j^{\min} = \min_{i \in N_j}(q_j, q_i) \tag{14}$$

and

$$q_j^{\max} = \max_{i \in N_j}(q_j, q_i), \tag{15}$$

where, N_j are the neighbors of the cell j. The piecewise linear reconstruction of data is bounded by enforcing the following condition,

$$q_j^{\min} \leq q(x,y)_j \leq q_j^{\max}. \tag{16}$$

The limiter L_{face} is determined by:

$$L_{face} = \begin{cases} \min(1, \frac{q_j^{\max} - q_o}{q^{face} - q_o}) & \text{if} \quad q^{face} - q_o > 0 \\ \min(1, \frac{q_j^{\min} - q_o}{q^{face} - q_o}) & \text{if} \quad q^{face} - q_o < 0 \\ 1 & \text{if} \quad q^{face} - q_o = 0 \end{cases}, \tag{17}$$

where, q_o is the cell-averaged value and q^{face} is the extrapolated value on the face of the cell. Three values of L_{face} are obtained for each cell (one for each edge) from Eq. (17) and the minimum of the three is used to limit the gradient at the cell center. The design and implementation of limiters and especially multidimensional limiters (e.g., HUBBARD, 1999) is an active field of research.

5. Results

In this section, the results from five different test cases (rotating cone case, Smolarkiewicz's deformational flow case, Doswell's frontogenesis case, a solution-adaptation case and the Noye-Tan test) are presented.

a. Rotating Cone Test

Rotating cone tests were performed to demonstrate the higher-order accuracy of the scheme. The domain was bounded within [0,100] × [0,100]. The cone was centered at (x_c, y_c) = (50,75) with a maximum height of 0.975 unit and a radius of 10 units. The rotational flow field was defined as follows:

$$u(x,y) = -\omega(y - y_o), \tag{18}$$

$$v(x,y) = \omega(x - x_o), \tag{19}$$

where, $u(x,y)$ and $v(x,y)$ are the velocities in the x and y direction, respectively ω = 0.4 is the constant angular velocity and (x_o, y_o) = (50,50) is the center of the mesh. The simulation was run for 15.7079 s (time taken by the cone to complete one revolution = $2\pi/\omega$). The unstructured mesh was defined in terms of boundary edges

►

Figure 6
Rotating Cone Test. First-order Godunov solution after one revolution (top), and higher-order solution after one revolution (bottom). Time = 15.7079 s.

(100 edges on each side). The resulting mesh consisted of 39,386 triangles with the edge lengths ranging from 0.35 m to 1.24 m. The boundary conditions are defined with the help of *ghost* cells (LEVEQUE, 2002) which are mirrors of the boundary cells. Transmissive boundary conditions (LEVEQUE, 2002) were used in the calculations. The solution was marched in time within a four-stage explicit Runge-Kutta scheme (JAMESON *et al.*, 1981). The linear least-squares gradient-reconstruction was used to achieve higher-order spatial accuracy.

Figure 6 shows the concentration contours after one revolution for the first- and the higher-order schemes. The comparison between the high resolution Godunov scheme and the first-order scheme is shown in Figure 7 along with the exact solution. The figure shows concentration profiles at $y = 75$ for x between 25 and 75. The first-order Godunov scheme exhibits large amounts of numerical diffusion. The higher-order scheme however, is not only monotonic and shape-preserving but it also retains around 83% of the initial tracer maxima compared to approximately 31% for the first-order scheme. Furthermore, it minimizes the numerical dispersion errors in the solution, which are commonly associated with central finite difference schemes such as the Leapfrog scheme.

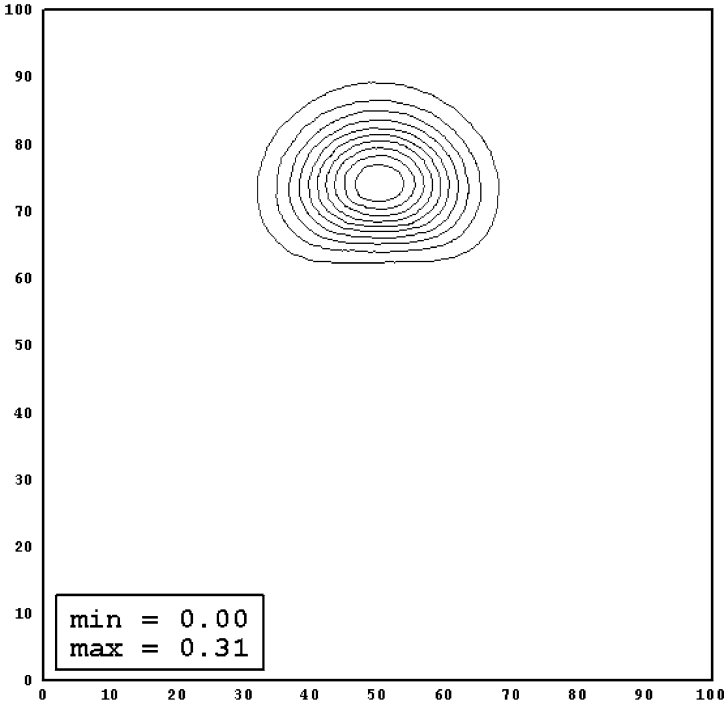
b. Smolarkiewicz's Deformational Flow Test

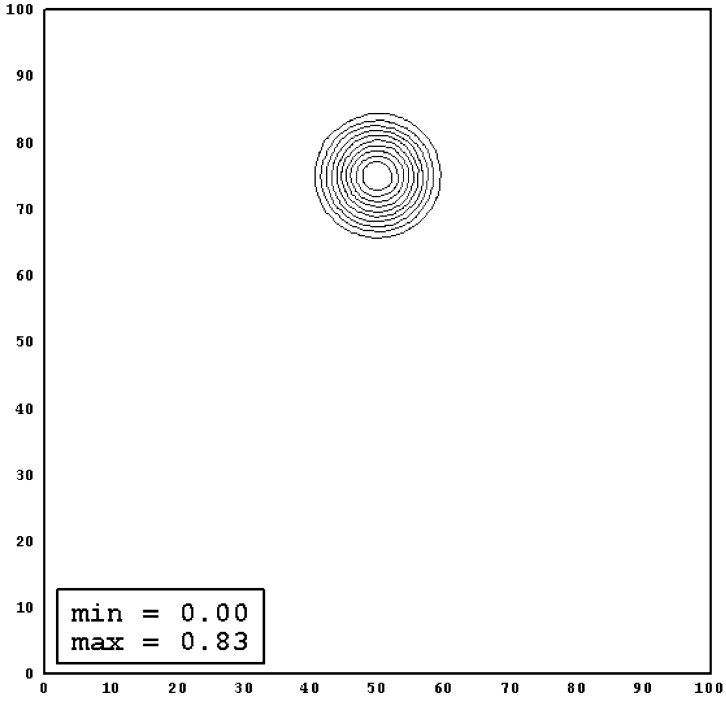
Smolarkiewicz's deformational flow (SMOLARKIEWICZ, 1982; STANIFORTH *et al.*, 1987; SYKES and HENN, 1995) is often used for a qualitative evaluation of advection schemes for atmospheric flow simulations. The flow field for the deformation test consists of sets of symmetrical vortices and is given by:

$$u(x, y) = Ak \sin kx \sin ky, \tag{20}$$

$$v(x, y) = Ak \cos kx \cos ky, \tag{21}$$

where, $u(x,y)$ and $v(x,y)$ are the velocities in the x and y direction respectively, $k = 4\pi/L$, $A = 8$ and $L = 100$ units. The domain was bounded within [0,100] × [0,100]. A tracer cone with a height of 1 unit and radius of 15 units was initialized in the middle of the domain. The mesh, boundary conditions, gradient reconstruction technique, and the time-marching scheme, were the same as in the rotating cone test. Figures 8 and 9 show the tracer distribution at time = 0, T/100; T/50 and T/25 (T = 2637.6 seconds is the final time of integration used in SMOLARKIEWICZ, 1982). Figures 10–12 show the comparison with Staniforth's analytical solution for tracer values between $x = 25$ and $x = 50$, for $y = 50$. The profiles of the computed tracer field are generated by interpolating the data from the cell centers closest to points on


100
90
80
70
60
50
40
30
20
10
0
0 10 20 30 40 50 60 70 80 90 100
min = 0.00
max = 0.31


100
90
80
70
60
50
40
30
20
10
0
0 10 20 30 40 50 60 70 80 90 100
min = 0.00
max = 0.83

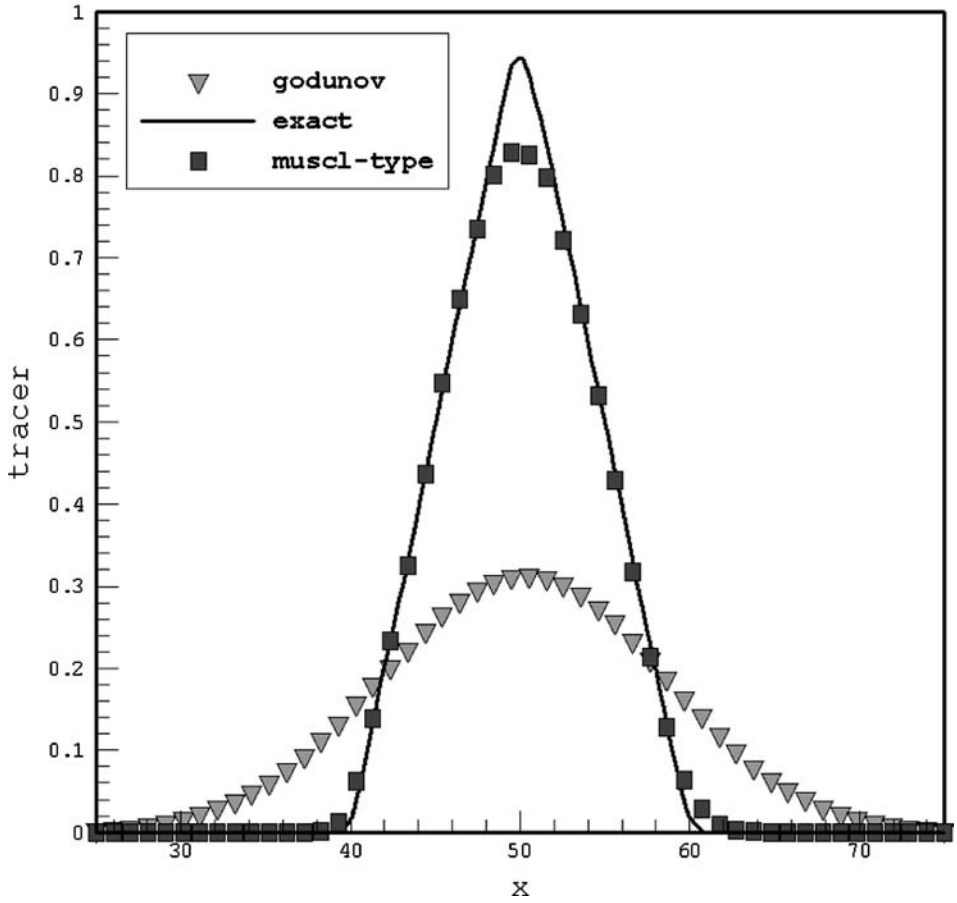
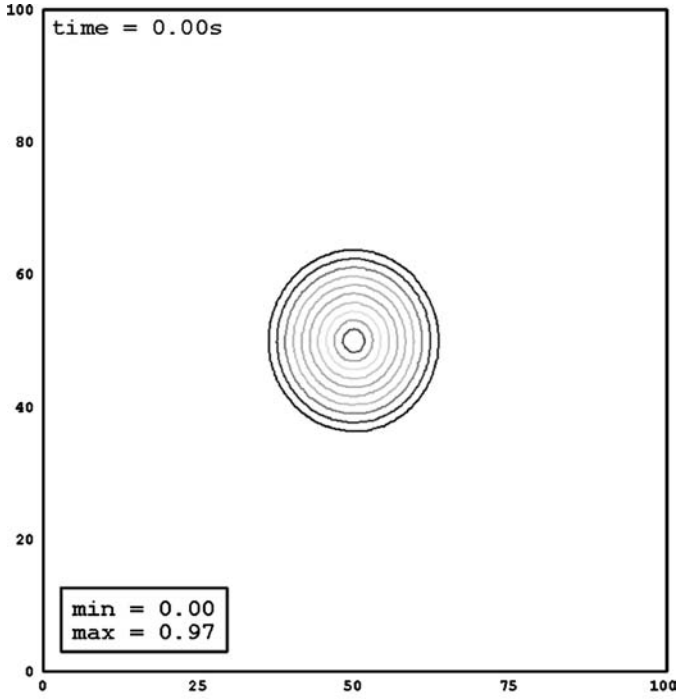


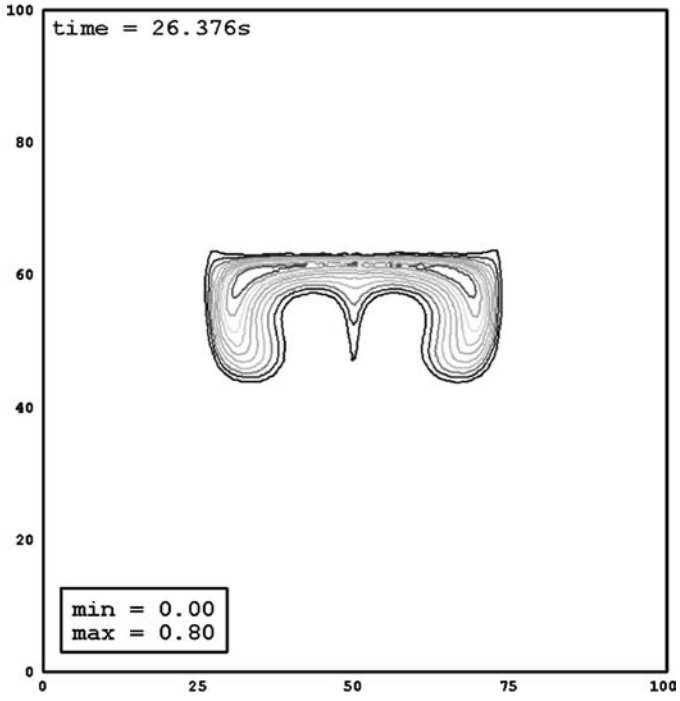
Figure 7
Rotating Cone Test. Comparison of the first- and higher-order Godunov schemes with the exact solution after one revolution. The profiles shown are at $y = 75$ for x varying from 25 to 75. Time = 15.7079 s.

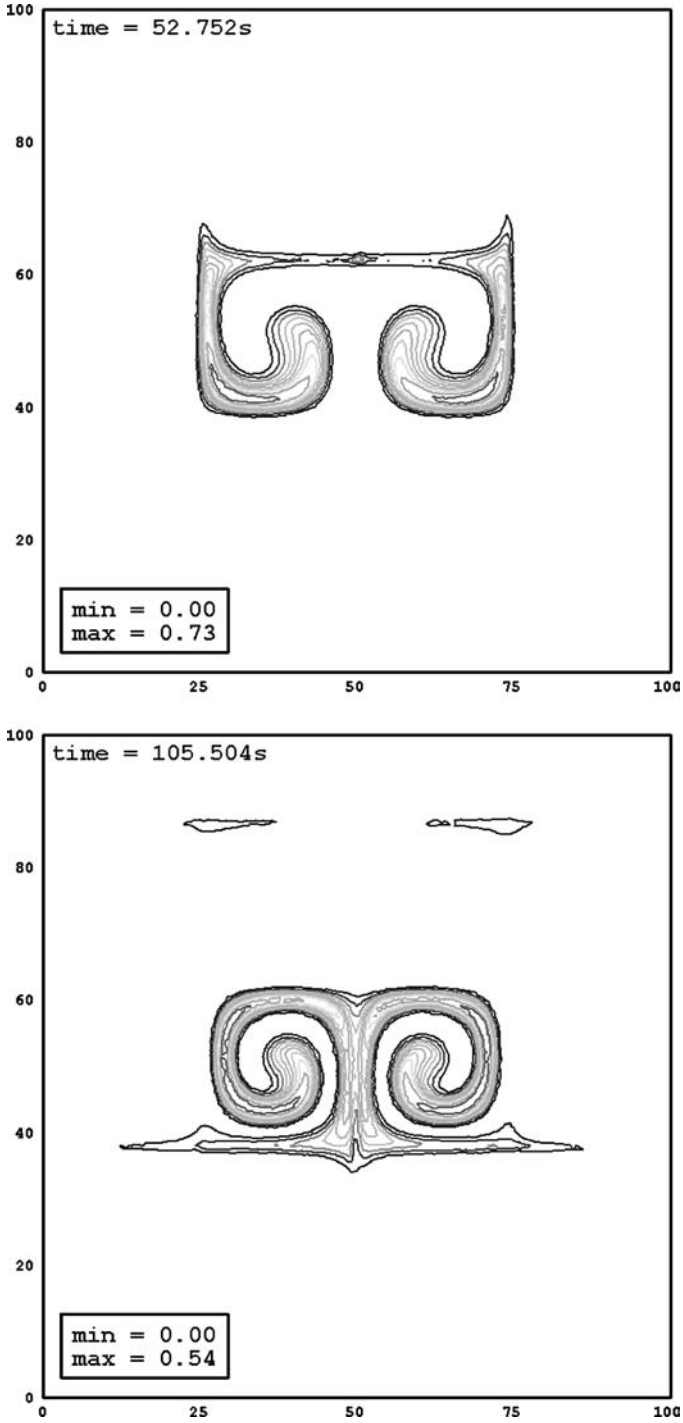
the line (25,50)–(50,50). The Staniforth solution is computed numerically and requires an input of sampling interval. For the comparison shown in Figures 10–12, a sampling interval of 0.1 was used. STANIFORTH *et al.* (1987) have discussed this test case in detail. They point out that for a mesh resolution of 1 used in SMOLARKIEWICZ (1982), the numerical solution is valid only for time ≤ T/50. After time > T/50 the features of the tracer field become too small to be effectively captured by a mesh

►

Figure 8
Smolarkiewicz's Deformational Flow. Initial tracer concentration (top), and the tracer field at time = T/100 = 26.376 s into the simulation (bottom). T = 2637.6 s.


time = 0.00s
min = 0.00
max = 0.97
0
20
40
60
80
100
0
25
50
75
100


time = 26.376s
min = 0.00
max = 0.80
0
20
40
60
80
100
0
25
50
75
100


time = 52.752s
min = 0.00
max = 0.73
time = 105.504s
min = 0.00
max = 0.54
0
20
40
60
80
100
0
25
50
75
100

◀

Figure 9
Smolarkiewicz's Deformational Flow. Tracer field at time = T/50 = 52.752 s (top), and time = T/25 = 105.504 s into the simulation (bottom). T = 2637.6 s.

resolution of 1, normally used for this test. Small amounts of numerical diffusion can be seen in the comparison with the exact solution for time = T/100 and T/50. The solution, however becomes more diffusive at time = T/25 which can be attributed to the fact that by this time the tracer field has evolved beyond the mesh's ability to resolve it.

c. Doswell's Frontogenesis Test

The simulation of Doswell's frontogenesis problem (DOSWELL, 1984; HÓLM, 1995) is presented in this section. Doswell's idealized model describes the interaction of a nondivergent vortex with an initially straight frontal zone. An exact solution is readily available for this case, which makes it ideal for a quantitative as well as qualitative validation of a numerical scheme. The flow field was defined as follows:

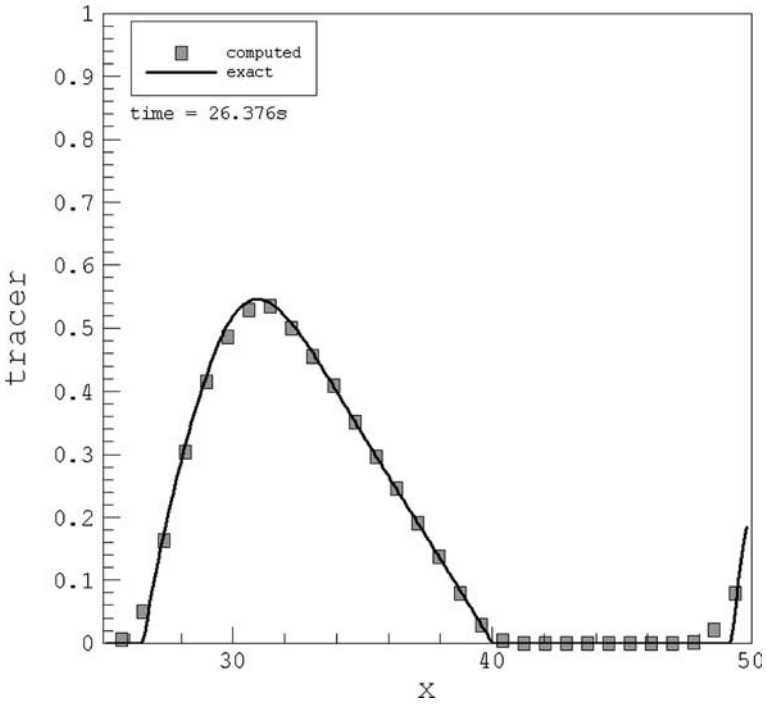


Figure 10
Smolarkiewicz's Deformational Flow. Solution at time = T/100 = 26.376 s. The values between $x = 25$ and $x = 50$ are shown for $y = 50$. The sampling interval was set to 0.1 for the analytical solution. T = 2637.6 s.

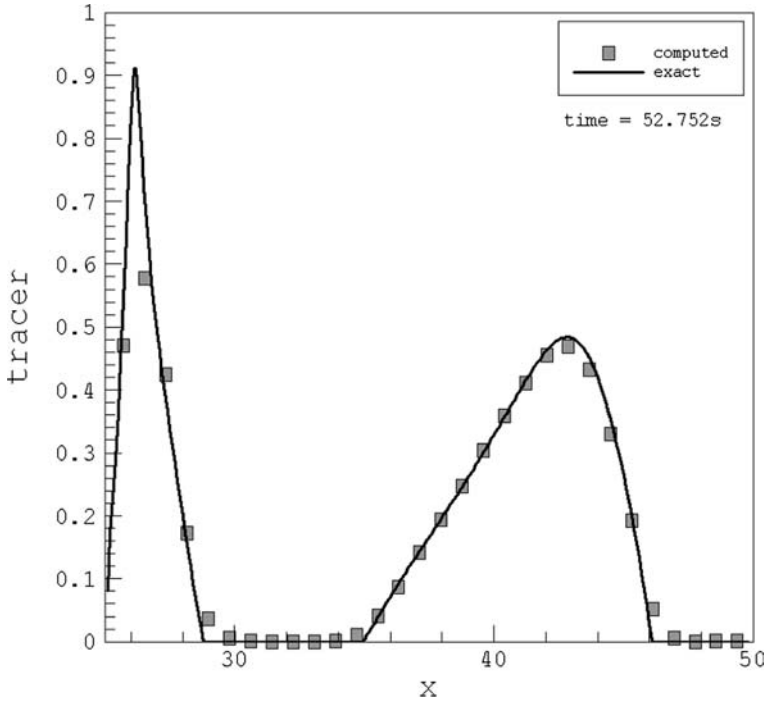


Figure 11
Smolarkiewicz's Deformational Flow. Solution at time = T/50 = 52.752 s. The values between $x = 25$ and $x = 50$ are shown for $y = 50$. The sampling interval was set to 0.1 for the analytical solution. T = 2637.6 s.

$$u(x,y) = -\frac{y}{r}\frac{f_t}{f_{\max}}; \quad v(x,y) = \frac{x}{r}\frac{f_t}{f_{\max}}, \tag{22}$$

where, $u(x,y)$ and $v(x,y)$ are the velocities in the x and y direction, respectively, r is the distance from any given point to the origin of the coordinate system, $f_{\max} = 0.385$ is the maximum tangential velocity and f_t is given by:

$$f_t = \frac{\tanh(r)}{\cosh^2(r)}. \tag{23}$$

The domain was bounded within $[-4, 4] \times [-4, 4]$. The boundary conditions and the time-marching scheme were the same as in the rotating cone test. The simulation was run for $t = 4$ seconds. The evolution of tracer field in time t, is given by the exact solution:

$$q(x,y,t) = -\tanh\left[\frac{y}{\delta}\cos(f \cdot t) - \frac{x}{\delta}\sin(f \cdot t)\right], \tag{24}$$

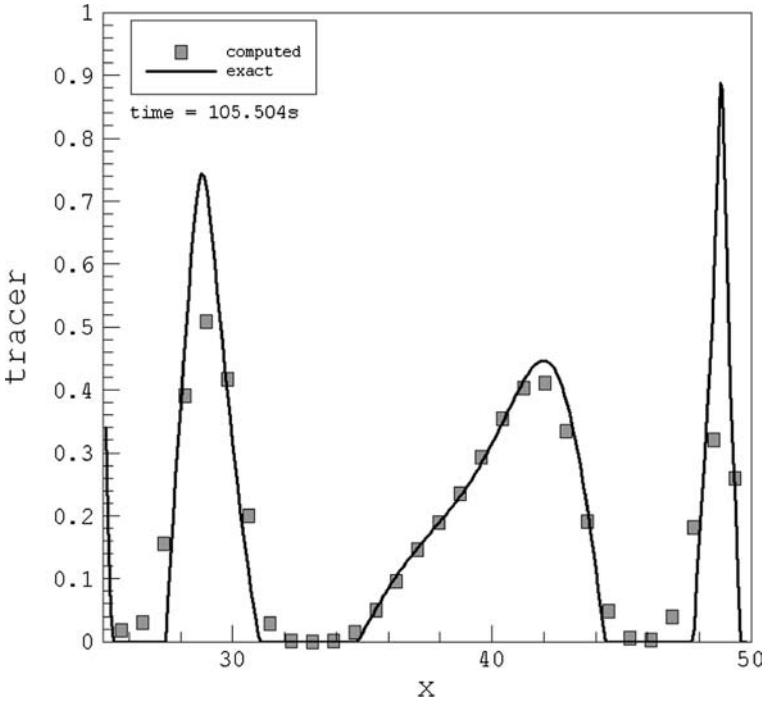


Figure 12
Smolarkiewicz's Deformational Flow. Solution at time = T/25 = 105.504 s. The values between $x = 25$ and $x = 50$ are shown for $y = 50$. The sampling interval was set to 0.1 for the analytical solution. T = 2637.6 s.

where, $f = 1/r\ f_t/f_{\max}$ and δ is set to 2 for a smooth frontogenesis. The initial tracer field can be obtained from Eq. (24) by setting $t = 0$:

$$q(x, y, 0) = -\tanh\left(\frac{y}{\delta}\right). \tag{25}$$

Figure 13 shows the initial conditions for the Doswell, and Fig. 14 shows a comparison between the exact and the numerical solutions. The simulation results are in good agreement with the exact solution. In addition several simulations were conducted to compare solutions on different types of meshes using different gradient-reconstruction techniques. Figure 15 shows the computed results for time = 4 seconds on different meshes. Only a zoomed view of the mesh center is shown in the figure. Mesh 1 consists of right angle triangles; Mesh 2 is a standard unstructured mesh in which an effort has been made to ensure good quality of triangles. Mesh 3 is similar to Mesh 2 except that smoothing is not performed. Mesh 3 is the worst of the three in terms of quality. The first-order solution on Mesh 2 is also shown for comparison. The error in the solution was defined as:

▶

Figure 13
Doswell's Frontogenesis Case. Initial frontal zone (top) and the vortex defining the flow field (bottom). Variable shown in the bottom panel is the wind speed.

$$error = \frac{\sqrt{\sum_{i=1}^{ncells} (q_i^{exact} - q_i^{computed})^2}}{ncells} \tag{26}$$

where, *ncells* is the total number of cells in the mesh. The error in the solution compared to the exact solution is presented in Figure 16. Although the error in the higher-order solution is low for all three meshes, the differences in accuracy due to mesh quality and the type of reconstruction technique used are clearly demonstrated. Mesh 2 (best quality mesh) performs much better than the other two meshes and the linear least-squares technique produces more accurate results than the Green-Gauss reconstruction technique. It is interesting to note that the Green-Gauss for Mesh 1 gives the worst results. This is due to the fact that the Green-Gauss technique is second-order accurate only for equilateral triangles and Mesh 1 consists of only right angle triangles.

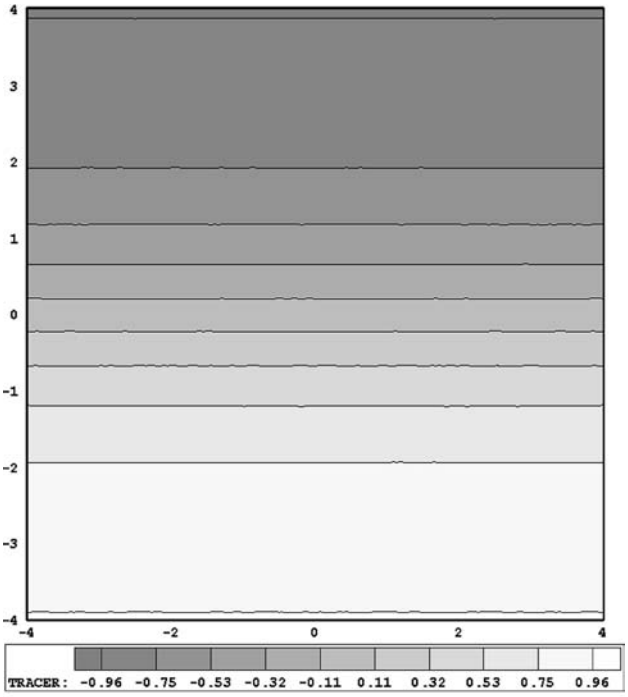
A convergence study was performed to determine the accuracy of the high-resolution Godunov scheme. The L_2 error was defined as follows (BURG *et al.*, 2002):

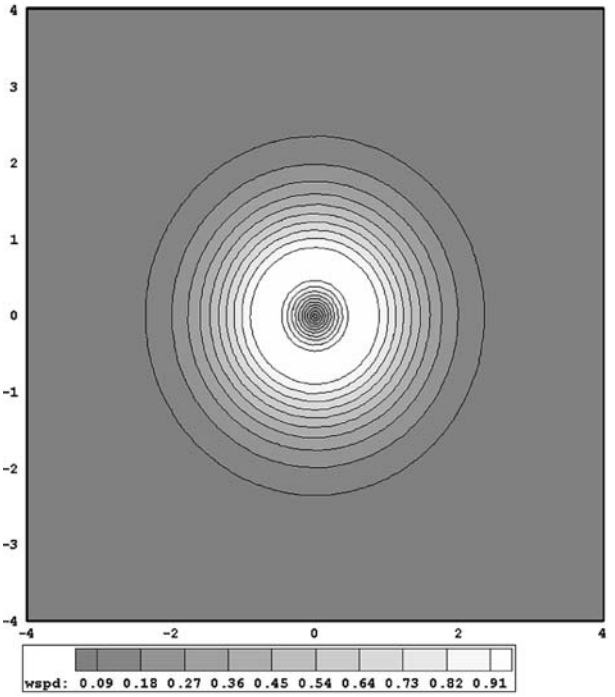
$$error(L2) = \sqrt{\sum_{i=1}^{ncells} (q_i^{exact} - q_i^{computed})^2 A_i}, \tag{27}$$

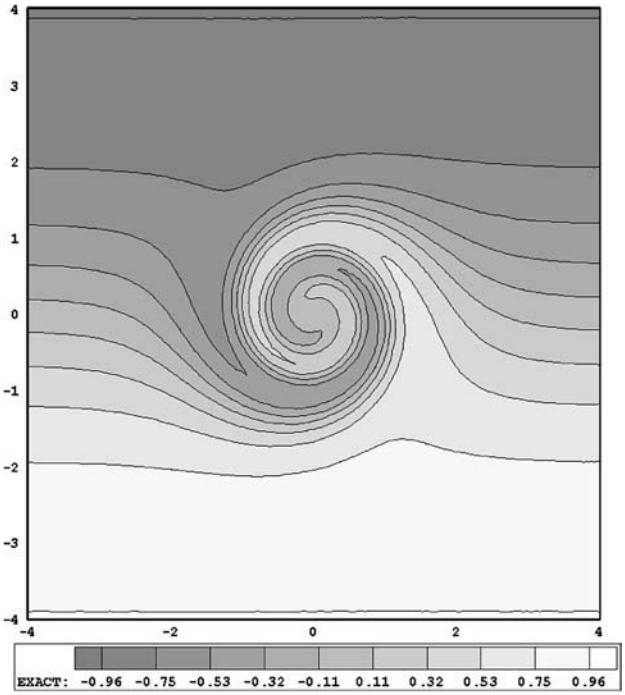
where, A_i is the area of each cell and *ncells* is the total number of cells in the mesh. The L_2 error is plotted in Figure 17. The log-log plot shows a slope of 1.67 for the high-resolution scheme. A maximum slope of 1.67 indicates that the higher-order scheme is not formally second-order accurate. This impression can be misleading, because in the presence of discontinuities, the monotonic schemes revert to first-order locally, which can degrade the overall order of accuracy for the scheme and not necessarily the accuracy of the scheme itself (LEVEQUE, 2002). It has been shown that even in the absence of discontinuities, for calculating smooth solutions, the monotonic schemes may or may not formally indicate a convergence rate of 2. VENKATAKRISHNAN (1993) reports an order of accuracy of 1.65 for various monotonic schemes which are formally second-order accurate.

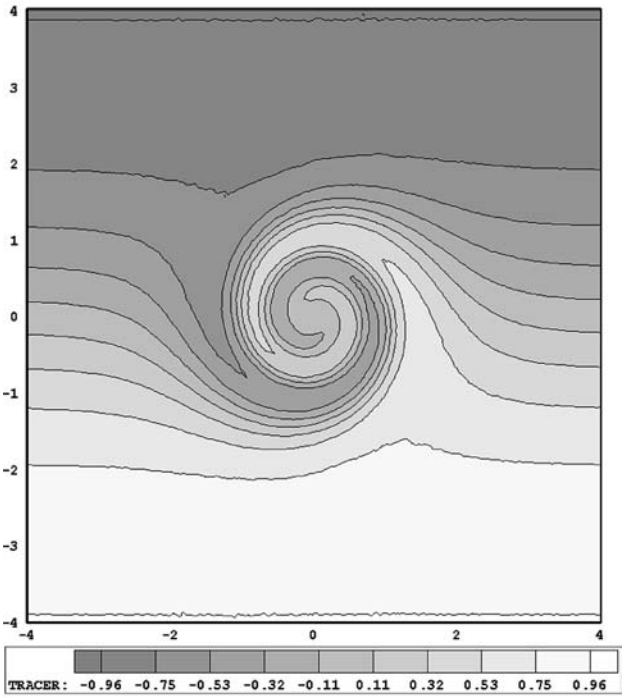
d. Solution-adaptation Test

The basic idea behind adaptive mesh refinement is to distribute the error equally over a computational mesh. The regions, where numerical error is large, are refined to provide greater spatial accuracy. In the current study, adaptation was achieved via


4
3
2
1
0
-1
-2
-3
-4
-4
-2
0
2
4
TRACER: -0.96 -0.75 -0.53 -0.32 -0.11 0.11 0.32 0.53 0.75 0.96


4
3
2
1
0
-1
-2
-3
-4
-4
-2
0
2
4
wspd: 0.09 0.18 0.27 0.36 0.45 0.54 0.64 0.73 0.82 0.91


4
3
2
1
0
-1
-2
-3
-4
-4
-2
0
2
4
EXACT: -0.96 -0.75 -0.53 -0.32 -0.11 0.11 0.32 0.53 0.75 0.96


4
3
2
1
0
-1
-2
-3
-4
-4
-2
0
2
4
TRACER: -0.96 -0.75 -0.53 -0.32 -0.11 0.11 0.32 0.53 0.75 0.96

◀

Figure 14
Doswell's Frontogenesis Case. Exact solution at time = 4 s (top) and the corresponding simulation result using the higher-order Godunov scheme (bottom).

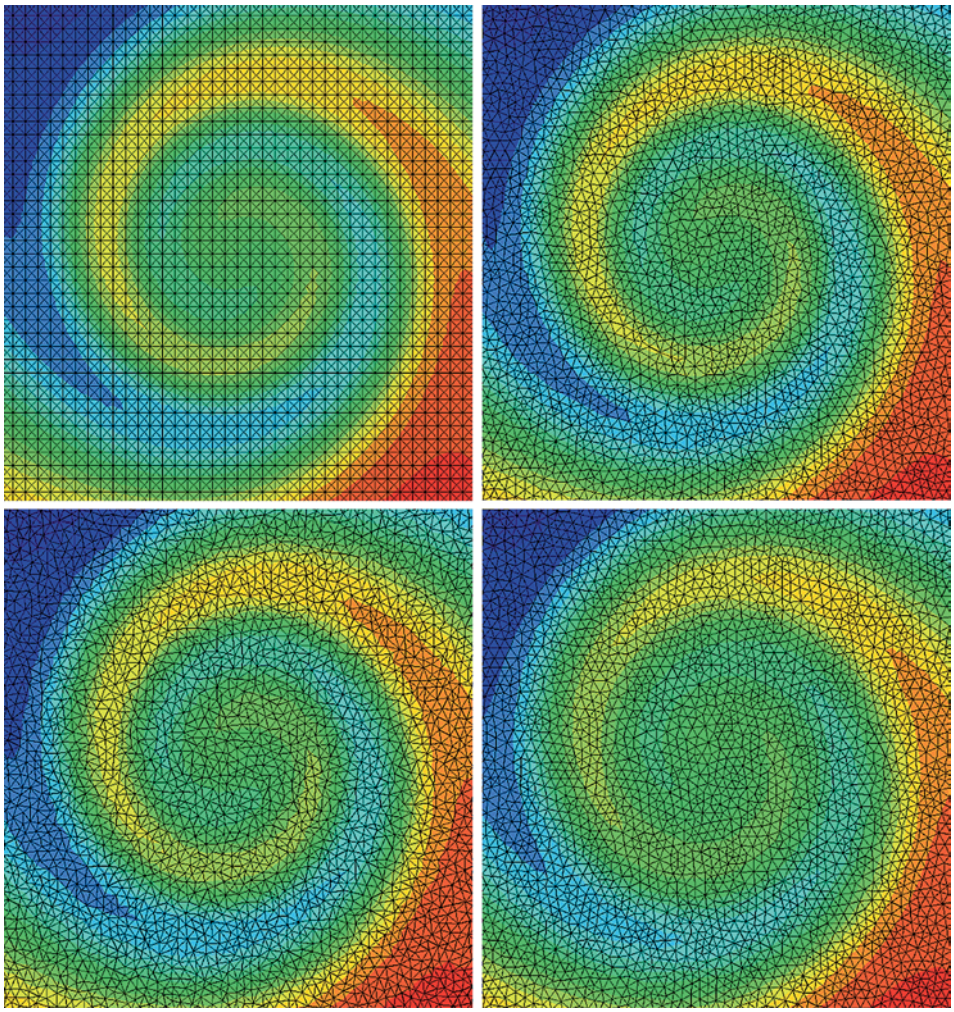


Figure 15
Doswell's Frontogenesis Case. Higher-order solution on Mesh 1 that consists of right angle triangles (top left), higher-order solution on Mesh 2 that consists of good aspect ratio triangles (top right), higher-order solution on Mesh 3 that is similar to Mesh 2 except that smoothing is not performed (bottom left), and first-order solution on Mesh 2 (bottom right). Each mesh consists of approximately 40000 cells.

h-refinement in which, the conservation of quantities is easier to maintain during interpolation and computational overhead is also smaller compared to remeshing (e.g., LÖHNER, 2001).

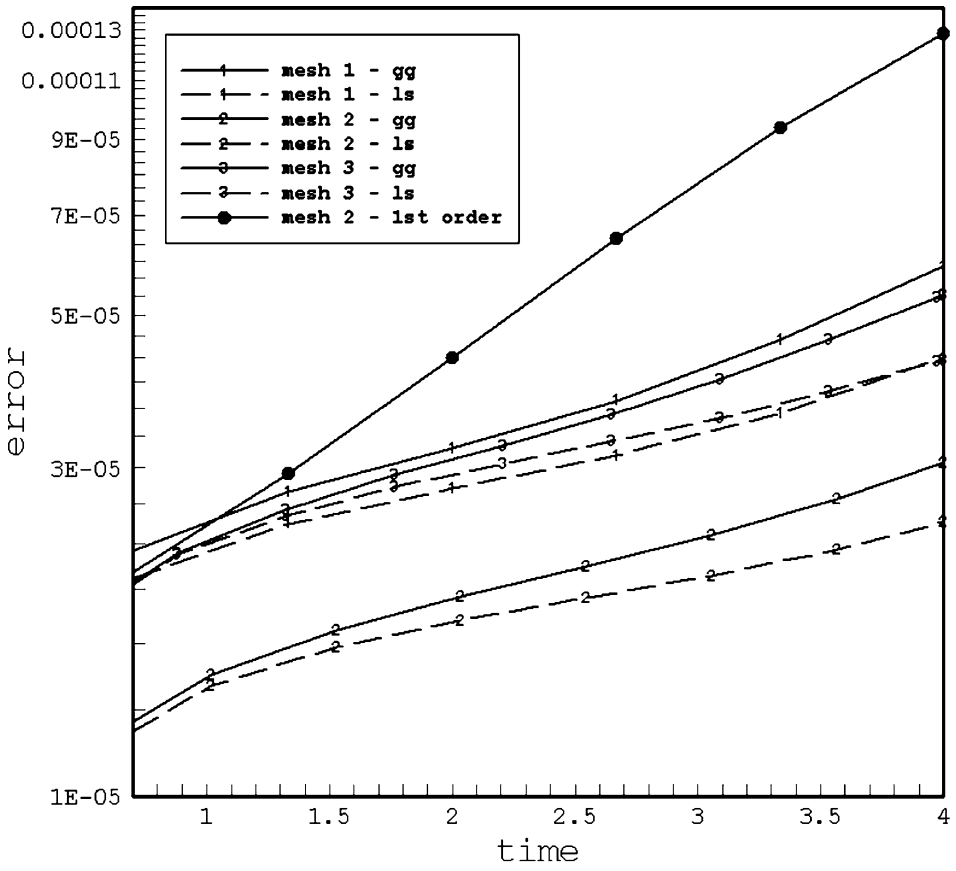


Figure 16
Doswell's Frontogenesis Case. Error in solution for different meshes, different reconstruction methods, and different order of numerical schemes. (gg = green-Gauss; ls = linear least-squares).

The rotating-cone test was used to demonstrate the advantages of solution-adaptation technique. The problem setup (initial and boundary conditions, etc.) is the same as in the previous simulation (described in *Section a*). The unstructured mesh was defined in terms of boundary edges (100 edges on each side for the globally refined mesh and 25 edges on each side for the adaptive mesh). The adaptive mesh started with a minimum edge length of 0.337 and a maximum edge length of 4.667. The mesh-adaptation algorithms used in this study are described in AHMAD *et al.*, (1998).

If the exact solution is known then the error-indicator (adaptation criteria) can easily be defined in terms of relative error or a similar quantity. In practice the exact solution is not known *a priori*. The regions of large errors, however, usually coincide with regions of high gradients. There are various ways in which one can define the

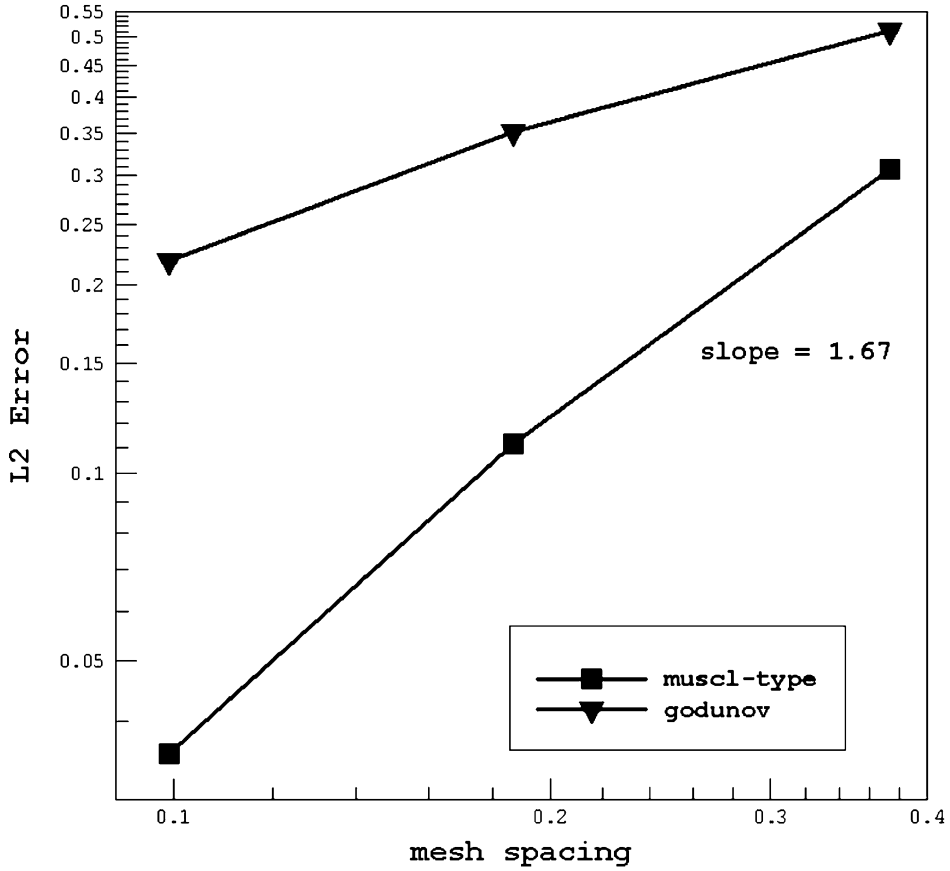


Figure 17
Doswell's Frontogenesis Case. Convergence study.

adaptation criteria depending on the problem. AHMAD *et al.* (1998) identify cells for refinement based on a Gaussian function around Lagrangian particles (for atmospheric dispersion simulations). The adaptation criteria proposed by LÖHNER (2001) is a function of the Laplacian, first derivatives and differences (for tracking shock-wave propagation, fluid-structure interactions, etc.). GHORAI *et al.* (2000) have based their error-indicator on the difference between the first and second-order solutions (for Eulerian transport simulations).

In the current study a simple error-indicator was used. Three radii were defined – R_{con} was set to the radius of the cone; R_{ref} was 2 units larger than R_{con} and R_{cor} was defined as 4 units larger than R_{con} (see Fig. 18). The maximum and minimum allowable edge lengths were also specified. The cells were tagged for refinement if a cell with large edge lengths was found between R_{con} and R_{ref} and cells were tagged for

deletion if a cell with small edge lengths was found outside the circle defined by R_{cor}. The refinement cycle was invoked every 15 iterations and the coarsening cycle was invoked every 150 iterations. The adaptation criteria used in this study is simple, but has direct relevance to atmospheric modeling. An area of influence, *e.g.*, can be defined around the puff centers as the adaptation criteria for dispersion modeling applications.

Figure 19 shows the initial concentration contours, tracer field at an intermediate stage and after one revolution for the solution-adaptive run. The comparison with exact solution is shown in Figure 20. The figure shows concentration profiles at $y = 75$ for x between 25 and 75. Errors and timings for the solution-adaptive run and the simulation on a globally refined mesh are tabulated in Table 1. The timing was obtained by using the Linux *time* command. Calculations were made on a P4 Dell Laptop running RedHat Linux 7.3. The phase error was defined as follows (ISELIN *et al.*, 2002):

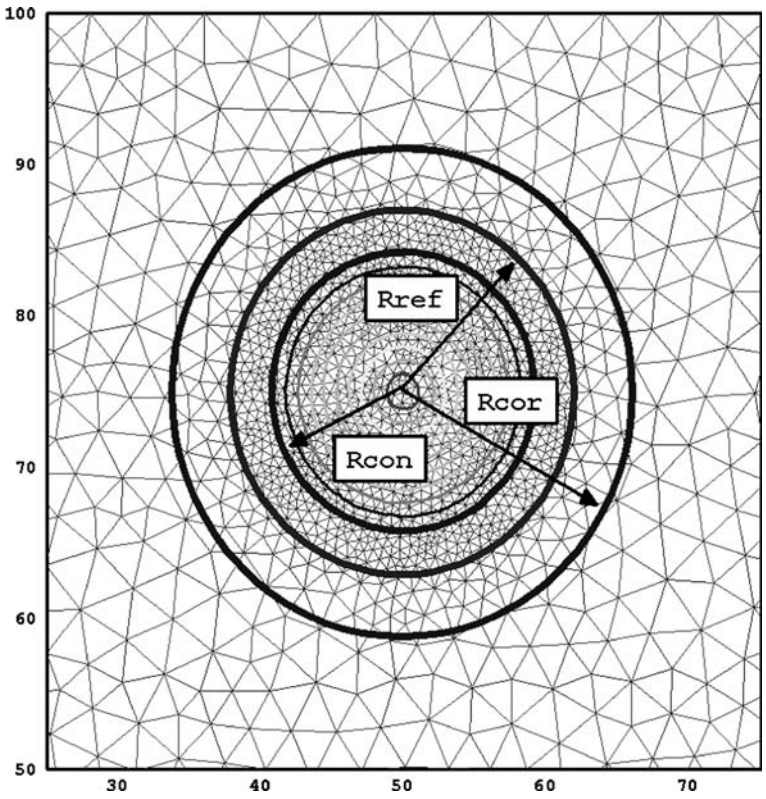


Figure 18
Solution-adaptation Test. Adaptation criteria.

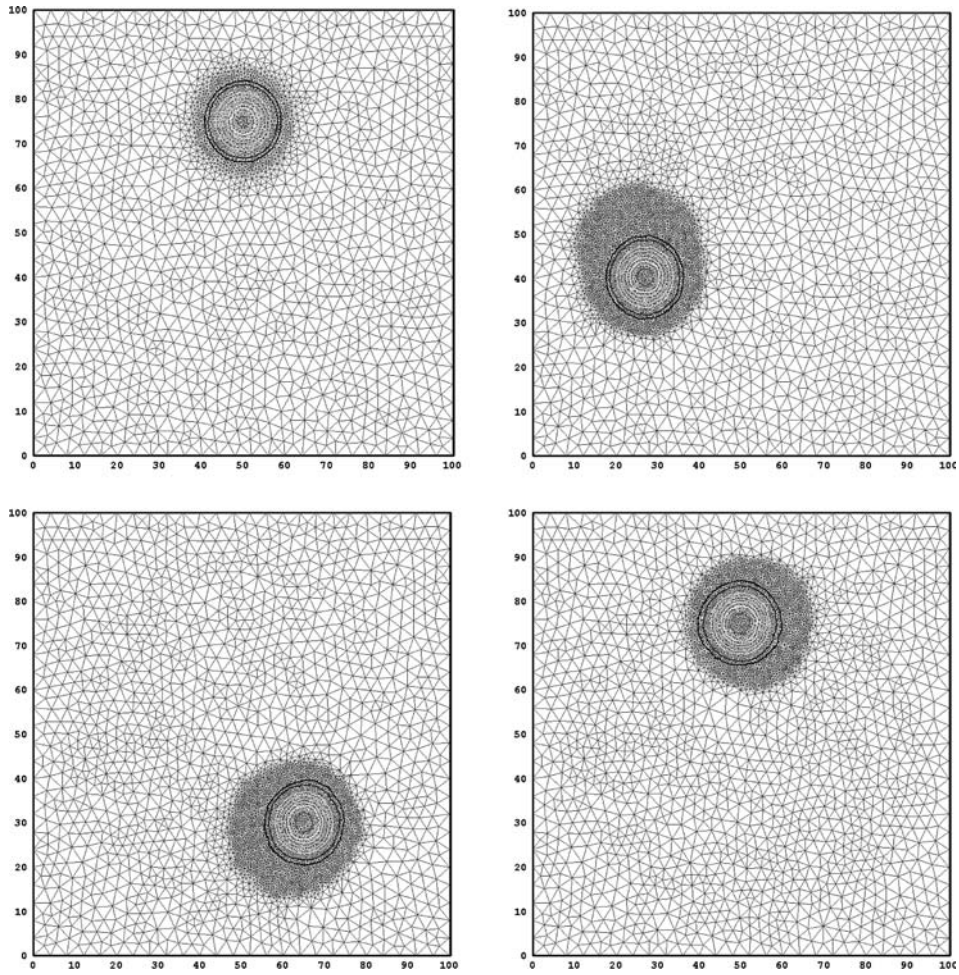


Figure 19
Solution-adaptation Test. Initial conditions for the adaptive run (top left), solution at an intermediate stage (top right), at an intermediate stage (bottom left) and after one complete revolution (bottom right).

$$error(phase) = \sqrt{(x^{exact} - x^{computed})^2 + (y^{exact} - y^{computed})^2} \tag{28}$$

where, x^{exact} and y^{exact} are the coordinates of cell in which the tracer *maxima* lies for the exact solution and $x^{computed}$ and $y^{computed}$ are the coordinates of the cell in which the *maxima* lies for the computed solution. The diffusion error was found by subtracting the computed tracer *maxima* from the exact value of the tracer *maxima* (ISELIN *et al.*, 2002):

$$error(diffusion) = \max(q^{exact}) - \max(q^{computed}). \tag{29}$$

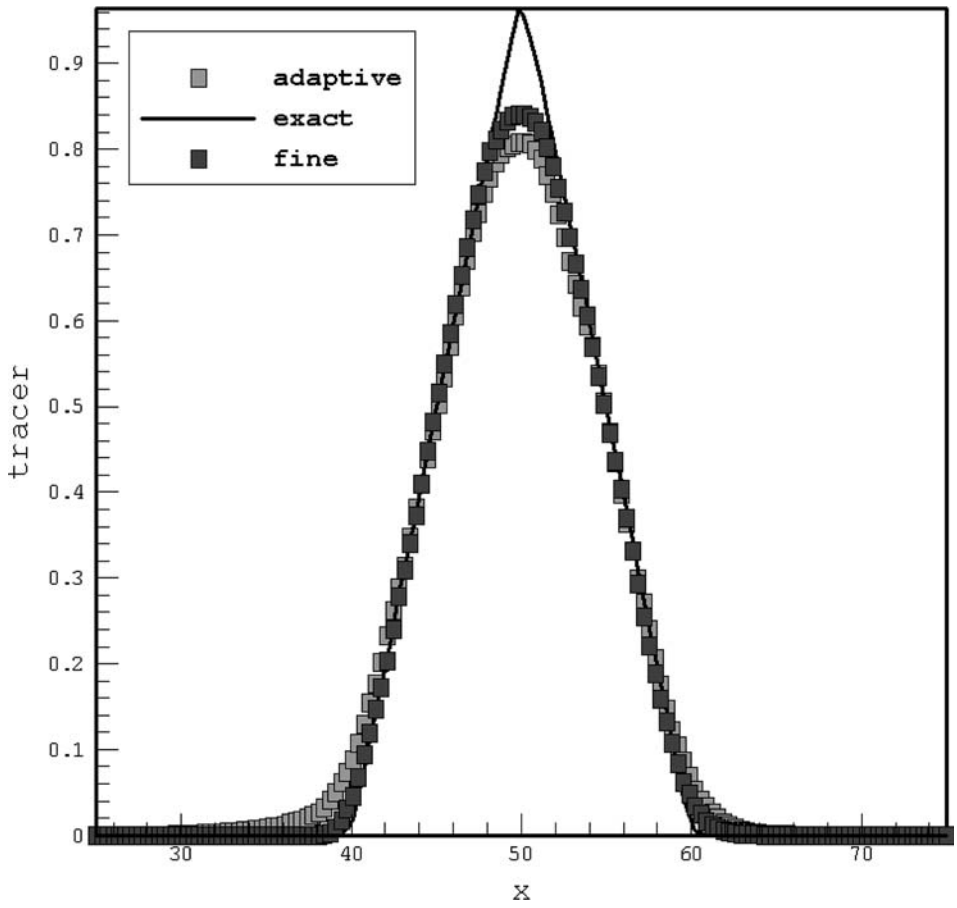


Figure 20
Solution-adaptation Test. Comparison of the solution-adaptive and globally refined mesh simulations with exact solution.

The adaptive mesh reproduces comparable results to the globally refined mesh at reduced computational cost (adaptive run is approximately 4 times faster). A further reduction in timing may be achieved by code optimization.

e. Noye-Tan Test

In this section the results of the test case for the advection-diffusion equation proposed by NOYE and TAN (1989) are presented. The test case describes the diffusion of an initial Gaussian pulse as it is advected along a straight line. In this test case no source terms are included which simplifies the problem. In the presence of diffusion, Eq. (4) can be rewritten as:

Table 1

Mesh Parameters for Solution-Adaptive Run

	fine	adaptive
$\mathbf{E_{L2}}$	0.35297	0.79073
$\mathbf{E_{phase}}$	0.29900	0.56245
$\mathbf{E_{diffusion}}$	0.13924	0.16537
real time	˜130 min	˜35 min
max edge	1.27941	4.93815
min edge	0.21593	0.22007
$\mathbf{ncells_{initial}}$	48366	4830
$\mathbf{ncells_{final}}$	48366	7872

$$V_{cell}\frac{dQ_{cell}}{dt}+\sum_{faces}(F,G)\cdot s=\sum_{faces}(J)\cdot s, \tag{30}$$

where, $J=k\nabla\cdot Q_{cell}$ and k is the diffusion coefficient. The diffusive flux across the edge is calculated by finding the gradients on the faces (the edges of each triangle in the case of a two-dimensional triangular mesh):

$$J=k\left(\frac{\partial q}{\partial x}\right)_{face}n_x+k\left(\frac{\partial q}{\partial y}\right)_{face}n_y, \tag{31}$$

where, n_x and n_y are the normal unit vectors in x and y direction, respectively. It is tempting to use an average of cell gradients on either side of the edge, but that can lead to loss of accuracy and may generate instabilities in the solution (Weiss *et al.*, 1999). Therefore, a correction term is included in the calculation:

$$\left(\frac{\partial q}{\partial x}\right)_{face}=\left(\frac{\partial q}{\partial x}\right)_{ave}-\frac{\partial q}{\partial c}c_x, \tag{32}$$

$$\left(\frac{\partial q}{\partial y}\right)_{face}=\left(\frac{\partial q}{\partial y}\right)_{ave}-\frac{\partial q}{\partial c}c_y, \tag{33}$$

where, c denotes the line segment connecting cell centers on either side of an edge (line connecting points (x_{il}, y_{il}) and (x_{ir}, y_{ir}) in Fig. 5) and c_x and c_y are the normal unit vectors for the line segment c. The subscript *ave* denotes the average value on the cell faces. The correction term is given by:

$$\frac{\partial q}{\partial c}=\left(\frac{\partial q}{\partial x}\right)_{ave}c_x+\left(\frac{\partial q}{\partial y}\right)_{ave}c_y-\frac{q_{il}-q_{ir}}{\vec{c}}, \tag{34}$$

where, q_{il} and q_{ir} are the scalar quantities in the cell to the left and the right of the edge and $\vec{c}$ is the length of the line segment connecting the cell centers on either side of the edge. The average gradient on the cell faces is calculated as follows (the

►

Figure 21
Noye-Tan Test. Initial conditions (top) and solution at $t = 1.25$ s (bottom).

subscripts *il* and *ir* denote the cell-centered values in the cells to the left and the right of the face/edge, respectively):

$$\left(\frac{\partial q}{\partial x}\right)_{ave} = 0.5\left(\left(\frac{\partial q}{\partial x}\right)_{il} + \left(\frac{\partial q}{\partial x}\right)_{ir}\right), \tag{35}$$

$$\left(\frac{\partial q}{\partial y}\right)_{ave} = 0.5\left(\left(\frac{\partial q}{\partial y}\right)_{il} + \left(\frac{\partial q}{\partial y}\right)_{ir}\right). \tag{36}$$

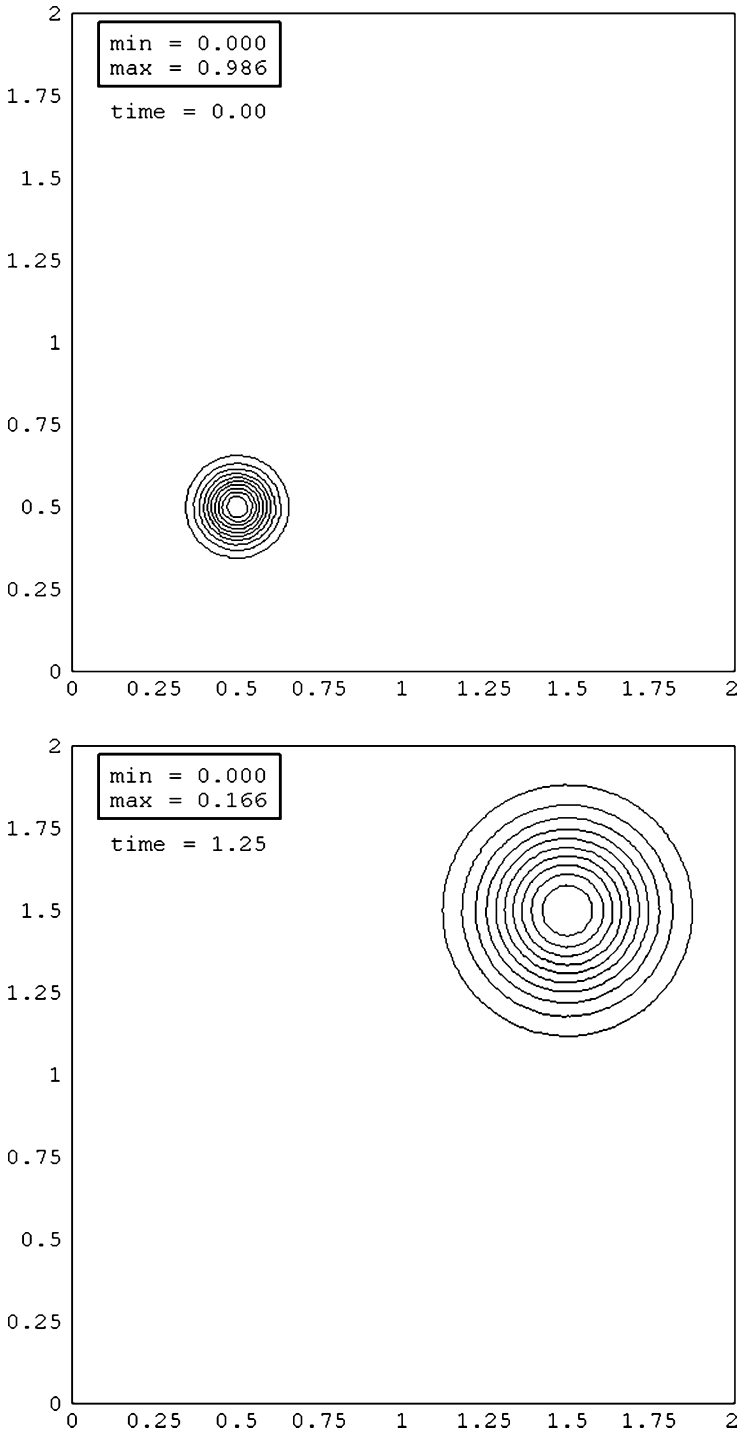
The computational domain was bounded within $[0, 2] \times [0, 2]$. The mesh was defined in terms of boundary edges (100 edges on each side of the computational domain). The resulting mesh consisted of 39,386 cells. The analytical solution of the unsteady advection-diffusion for the Noye-Tan test case is given by:

$$q(x, y, t) = \frac{1}{4t + 1} \exp\left[-\frac{(x - ut - x_o)^2}{k_x(4t + 1)} - \frac{(y - vt - y_o)^2}{k_y(4t + 1)}\right], \tag{37}$$

where, $k_x = k_y = 0.01$ are the diffusion coefficients and $u = v = 0.8$ are the velocities in the x and y direction, respectively. $x_o = y_o = 0.5$ is the center of the initial tracer distribution. The initial conditions and the Dirichlet boundary conditions can be obtained from the analytical solution by setting $t = 0$. The final time of the simulation is set to 1.25 seconds in NOYE and TAN (1989). Solution-adaptation was not used in this simulation. The initial conditions and the computed solution at $t = 1.25$ seconds are shown in Figure 21. A comparison between the exact solution and the computed solution along the mesh diagonal is shown in Figure 22. The computed solution matches well with the exact solution in the current study.

6. Conclusions

A higher-order Godunov-type scheme is implemented on unstructured meshes for atmospheric flow calculations. The method is validated against five different benchmark cases and the results are encouraging. The scheme is conservative and exhibits minimal numerical diffusion and dispersion errors. The computational efficiency is achieved by implementing a solution-adaptation technique (i.e., dynamic grid adaptation) that provided variable and continuous resolution throughout the computational domain, with a high mesh resolution only in regions of interest. The use of unstructured grids in the atmospheric modeling community is relatively new and more research is needed for the technology to mature for atmospheric applications.


min = 0.000
max = 0.986
time = 0.00
min = 0.000
max = 0.166
time = 1.25
0
0.25
0.5
0.75
1
1.25
1.5
1.75
2

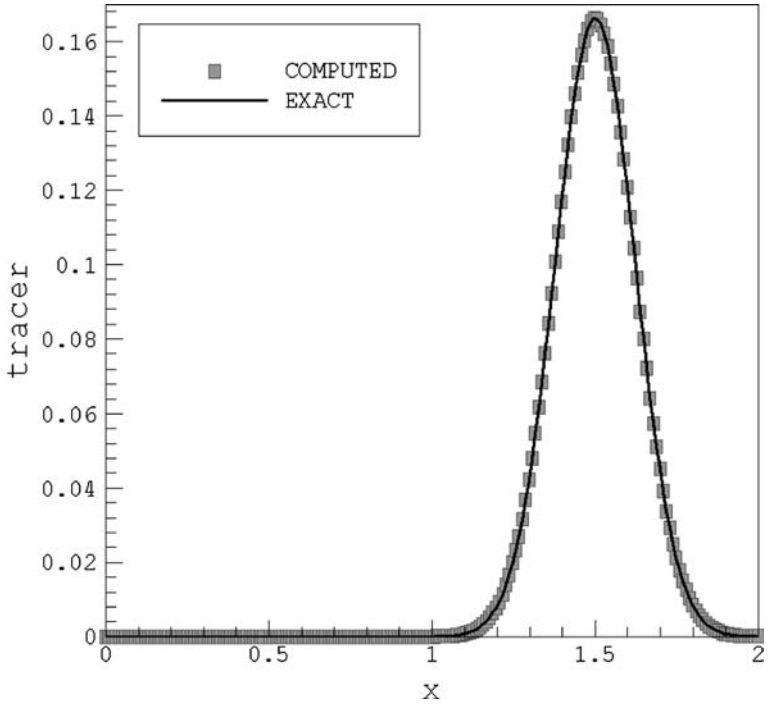


Figure 22
Noye-Tan Test. Comparison between the exact solution and the computed solution along the mesh diagonal at time = 1.25 s.

Acknowledgement

Many thanks to Drs. Doug Henn and Ian Sykes for providing the analytical solution of the deformational flow test. We have greatly benefited from the helpful comments and suggestions from an anonymous reviewer.

The FORTRAN code **ADVEC2D** used for making the calculations presented in this paper can be obtained by writing to the corresponding author. Please note that part of the code related to solution-adaptation is proprietary and therefore cannot be distributed.

References

Ahmad, N., Bacon, D., Boybeyi, Z., Dunn, T., Hall, M., Lee, P., Mays, D. Sarma, A. and Turner, M. (1998), *A solution-adaptive grid generation scheme for atmospheric flow simulations. In Numerical Grid Generation in Computational Field Simulations*, Proceedings of the 6th International Conference held at University of Greenwich. pp. 327–335.

BACON, D.P., AHMAD, N.N., BOYBEYI, Z., DUNN, T.J., HALL, M.S., LEE, P.C.S., SARMA, R.A., TURNER, M.D., WAIGHT, K., YOUNG, S. and ZACK, J. (2000), *A dynamically adapting weather and dispersion model: The operational multiscale environment model with grid adaptivity (OMEGA)*, Mon. Wea. Rev. *128*, 2044–2076.

BARTH, T.J. and JESPERSON, D.C. (1989), *The design and application of upwind schemes on unstructured meshes*, AIAA Paper 1989–0366.

BEHRENS, J., DETHLOFF, K., HILLER, W. and RINKE, A. (2000), *Evolution of small-scale filaments in an adaptive advection model for idealized tracer transport,* Mon. Wea. Rev. *128*, 2976–2982.

BORIS, J. and BOOK, D.L. (1973), *Flux-corrected transport. I. SHASTA, a fluid transport algorithm that works,* J. Comp. Phys. *11*, 38–69.

BOTT, A. (1989), *A positive definite advection scheme obtained by nonlinear renormalization of the advective fluxes,* Mon. Wea. Rev. *117*, 1006–1015.

BOTTA, N., Klein, R., LANGENBERG, S. and LUTZENKIRCHEN, S. (2004), *Well balanced finite volume methods for nearly hydrostatic flows,* J. Comp. Phys. *196*, 539–565.

BOYBEYI, Z., AHMAD, N., BACON, D., DUNN, T., HALL, M., LEE, P., SARMA, A. and WAIT, T. (2001), *Evaluation of the operational multiscale environment model with grid adaptivity against the european tracer experiment,* J. App. Met. *40*, 1541–1558.

BURG, C., SREENIVAS, K. HYAMS, D. and MITCHELL, B. (2002), *Unstructured nonlinear free surface flow simulations: Validation and verification*, AIAA Paper 2002–2977.

CLARK, T.L. and FARLEY, R.D. (1984), *Severe downslope windstorm calculations in two and three spatial dimensions using anelastic interactive grid nesting: A possible mechanism for gustiness*, J. Atmos. Sci. *41*, 329–350.

CARPENTER, R.L., DROEGEMEIER, K.K., WOODWARD, P.R. and HANE, C.E. (1990), *Application of the Piecewise Parabolic Method (PPM) to Meteorological Modeling*, Mon. Wea. Rev. *118*, 586–612.

COLLELA, P. and WOODWARD, P.R. (1984), *The Piecewise Parabolic Method (PPM) for Gas-Dynamical Simulations*, J. Comp. Phys. *54*, 174–201.

COTÉ, J., ROCH, M., STANIFORTH, A. and FILLION, L. (1993), *A variable-resolution semi-lagrangian finite-element global model of the shallow-water equations*, Mon. Wea. Rev. *121*, 231–243.

DIETACHMAYER, G.S. and DROEGERMEIER, K.K. (1992), *Application of continous dynamic grid adaptation techniques to meteorological modeling*, Mon. Wea. Rev. *120*, 1675–1706.

DOSWELL, C.A. (1984), *A kinematic analysis of frontogenesis associated with a nondivergent vortex*, J. Atmos. Sci. *41*, 1242–1248.

FIEDLER, B.H. and TRAPP, R.J. (1993), *A fast dynamic grid adaption scheme for meteorological flows,* Mon. Wea. Rev. *121*, 2879–2888.

GHORAI, S., TOMLIN, A.S. and BERZINS, M. (2000), *Resolution of pollutant concentrations in the boundary layer using a fully 3D adaptive gridding technique*, Atmospheric Environment, *34*, 2851–2863.

GODUNOV, S.K. (1959), *A finite difference method for the computation of discontinous solutions of the equations of fluid dynamics*, Mat. Sb. *47*, 357–393.

GOPALAKRISHNAN, S.G., BACON, D., AHMAD, N., BOYBEYI, Z., DUNN, T., HALL, M., LEE, P., MADALA, R., SARMA, A., TURNER, M. and WAIT, T. (2002), *An operational multiscale hurricane forecasting system*, Mon. Wea. Rev. *130*, 1830–1847.

HARTEN, A. (1983), *High resolution schemes for hyperbolic conservation laws*, J. Comp. Phys. *49*, 357–393.

HÓLM, E.V. (1995), *A fully two-dimensional, nonoscillatory advection scheme for momentum and scalar transport equations*, Mon. Wea. Rev. *123*, 536–552.

HOURDIN, F. and ARMENGAUD, A. (1999), *The use of finite-volume methods for atmospheric advection of trace species. Part I: Test of various formulations in a general circulation model*, Mon. Wea. Rev. *127*, 822–837.

HUBBARD, M.E. (1999), *Multidimensional slope limiters for MUSCL-type finite volume schemes on unstructured grids*, J. Comp. Phys. *155*, 54–74.

HUBBARD, M.E. and NIKIFORAKIS, N. (2003), *A three-dimensional, adaptive, Godunov-type model for global atmospheric flows*, Mon. Wea. Rev. *131*, 1848–1864.

ISELIN, J., PRUSA J.M. and GUTOWSKI, W.J. (2002), *Dynamic grid adaptation using MPDATA scheme,* Mon. Wea. Rev. *130*, 1026–1039.

JABLONOWSKI, C., HERZOG, M., PENNER, J.E., OEHMKE, R.C., STOUT, Q.F. and VAN LEER, B. (2004), *Adaptive grids for weather and climate models*, *ECMWF Seminar on Recent Developments in Mumerical Methods for Atmospheric and Ocean Modelling*, Reading, England.

JAMESON, A., SCHMIDT, W. and TURKEL, E. (1981), *Numerical solution of the euler equations by finite volume method using Runge-Kutta time stepping schemes*, AIAA Paper 1981–1259.

KURIHARA, Y. and BENDER, M. A. (1980), *Use of a movable nested-mesh model for tracking small vortex*, Mon. Wea. Rev. *108*, 1792–1809.

LEVEQUE, R.J. *Finite volume methods for hyperbolic problems*, (Cambridge University Press 2002). 558 pp.

LIN, S.J., CHAO, W.C., SUD, Y.C. and WALKER, G.K. (1994), *A class of van Leer-type transport schemes and its application to the moisture transport in a general circulation model,* Mon. Wea. Rev. *122*, 1575–1593.

LIU, M. and CARROLL, J.J. (1996), *A high-resolution air pollution model suitable for dispersion studies in complex terrain*, Mon. Wea. Rev. *124*, 2396–2409.

LÖHNER, R. (2001), *Applied CFD techniques: an introduction based on finite element methods*, John Wiley and Sons Ltd. 366 pp.

LOTTATI, I. and EIDELMAN, S. (1994), *A second-order Godunov scheme on a spatial adapted triangular grid*, Applied Numerical Mathematics, *14*, 353–365.

LUO, H., BAUM, J.D. and LÖHNER, R. (2003), *Extension of HLLC scheme for flows at all speeds*, AIAA Paper 2003–3840.

MAVRIPLIS, D.J. (2003), *Revisiting the least-squares procedure for gradient reconstruction on unstructured meshes*, AIAA Paper 2003–3986.

MITCHELL, C.R. (1994), *Improved reconstruction schemes for the Navier-Stokes equations on unstructured meshes*, AIAA Paper 1994–0642.

MÜLLER, R. (1992), *The performance of classical versus modern finite volume advection schemes for atmospheric modeling in a one-dimensional test-bed*, Mon. Wea. Rev. *120*, 1407–1415.

NOYE, B.J. and TAN, H.H. (1989), *Finite difference methods for solving the two-dimensional convection-diffusion equation*, Int. J. Num. Methods Fluids *9*, 75–98.

PIETRZAK, J. (1998) *The use of TVD limiters for forward-in-time upstream-biased advection schemes in ocean modeling*, Mon. Wea. Rev. *126*, 812–830.

PRUSA, J.M. and SMOLARKIEWICZ, P.K. (2003), *An all-scale anelastic model for geophysical flows: dynamic grid deformation*, J. Comp. Phys. *190*, 601–622.

SARMA, A. AHMAD, N., BACON, D. BOYBEYI, Z., DUNN, T., HALL, M. and LEE, P. *Application of adaptive grid refinement to plume modeling*. In Air Pollution *VII*. (WIT Press 1999). pp. 59–68.

SKAMAROCK, W. and KLEMP, J.B. (1993) *Adaptive grid refinement for two-dimensional and three-dimensional nonhydrostatic atmospheric flows*, Mon. Wea. Rev. *121*, 788–804.

SMOLARKIEWICZ, P.K. (1982), *The multi-dimensional Crowley advection scheme*, Mon. Wea. Rev. *113*, 1968–1983.

SMOLARKIEWICZ, P.K. (1984), *A fully multidimensional positive definite advection transport algorithm with small implicit diffusion*, J. Comp. Phys. *54*, 325–362.

SPEKREIJSE, S. (1987), *Multigrid solution of monotone second-order discretizations of hyperbolic conservation laws*, Math. Comp. *49*, 135–155.

STANIFORTH, A.N. and MITCHELL, H.L. (1978), *A variable-reolution finite element technique for regional forecasting with the primitive equations*, Mon. Wea. Rev., *106*, 439–447.

STANIFORTH, A., COTE, J. and PUDYKIEWICZ, J. (1987), Comments on "*Smolarkiewicz's Deformational Flow*", Mon. Wea. Rev. *115*, 894–900.

SYKES, R.I. and HENN, D.S. (1995), *Representation of velocity gradient effects in a Gaussian Puff model*, J. App. Met. *34*, 2715–2723.

TORO, E. F., *Riemann Solvers and Numerical Methods for Fluid Dynamics*. (Springer Verlag 1999) 624 pp.

VARVAYANNI, M., DAVAKIS, E. DELIGIANNIS, P. and CATSAROS, N. (1999), *Pollutant dispersion over complex surfaces of variable landuse using unstructured prismatic grid*, J. Geophys. Res. *104*, 275–293.

VAN LEER, B. (1979), *Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov's method*, J. Comp. Phys. *32*, 101–136.

VENKATAKRISHNAN, V. (1993), *On the accuracy of limiters and convergence to steady state solutions*, AIAA Paper 1993–0880.

WEISS, J.M., MARUSZEWSKI, J.P. and SMITH, W. A. (1999), *Implicit solution of preconditioned Navier-Stokes equations using algebraic multigrid*, *AIAA J. 37*, 29–36.

ZALESAK, S.T. (1979), *Fully multidimensional flux-corrected transport algorithms for fluids*, J. Comp. Phys. *31*, 335–362.

(Received January 21, 2005, accepted June 24, 2005)