A General-Purpose Finite-Volume Advection Scheme for Continuous and Discontinuous Fields on Unstructured Grids

E. D. Dendy, N. T. Padial-Collins, and W. B. VanderHeyden

Los Alamos National Laboratory, Theoretical Division, Fluid Dynamics Group (T-3), and Los Alamos Computer Science Institute, Los Alamos, New Mexico 87545
E-mail: dendy@lanl.gov, nelylanl@lanl.gov, and wbv@lanl.gov

We present a new general-purpose advection scheme for unstructured meshes based on the use of a variation of the interface-tracking flux formulation recently put forward by O. Ubbink and R. I. Issa (*J. Comput. Phys.* **153**, 26 (1999)), in combination with an extended version of the flux-limited advection scheme of J. Thuburn (*J. Comput. Phys.* **123**, 74 (1996)), for continuous fields. Thus, along with a high-order mode for continuous fields, the new scheme presented here includes optional integrated interface-tracking modes for discontinuous fields. In all modes, the method is conservative, monotonic, and compatible. It is also highly shape preserving. The scheme works on unstructured meshes composed of any kind of connectivity element, including triangular and quadrilateral elements in two dimensions and tetrahedral and hexahedral elements in three dimensions. The scheme is finite-volume based and is applicable to control-volume finite-element and edge-based node-centered computations. An explicit–implicit extension to the continuous-field scheme is provided only to allow for computations in which the local Courant number exceeds unity. The transition from the explicit mode to the implicit mode is performed locally and in a continuous fashion, providing a smooth hybrid explicit–implicit calculation. Results for a variety of test problems utilizing the continuous and discontinuous advection schemes are presented.   ⓒ 2002 Elsevier Science (USA)

Key Words: advection; reconstruction; interface tracking; volume of fluid; unstructured meshes; unstructured grids; finite-volume method.

1. INTRODUCTION

The development of accurate, conservative, monotonic, compatible, shape-preserving numerical advection schemes has been the subject of intense study for several decades. Despite this effort, the quest for improved techniques continues. Leveque [5] provides an

extensive survey of the literature on this topic. Let us begin our discussion here with some definitions. In this paper *conservative* denotes an advection scheme that guarantees no loss of conserved quantities, such as mass, in the operation of moving material from one conservation volume to another under the action of advection. We use the modifier *monotonic* to denote an advection operator or scheme, which does not produce any unphysical extrema in the updated density field. This property is also sometimes referred to as total variation diminishing, or TVD. Similarly, we use the modifier *compatible* to denote an advection operator or scheme which does not produce any unphysical extrema in the updated mass-specific or "mixing ratio" quantities, such as specific internal energy or velocity. Finally, we use the modifier *shape preserving* to refer to a scheme that minimizes the distortion of the shape of features in the conserved field in multidimensional flows. An advection scheme that is shape preserving is sometimes also said to provide *corner coupling*. This typically means that material from a given donor cell is transported, in a single application of the advection scheme, to a neighboring cell, or "corner cell," which has only a vertex in common with the donating cell. Doing this corner-cell transport correctly leads to shape preservation. All of these characteristics are essential for accurate Eulerian flow physics simulation schemes.

Our goal here is to provide an advection scheme that is conservative, monotonic, compatible, and shape preserving for the cases of both continuous and discontinuous density fields in an integrated treatment. Furthermore, our goal is also to provide this scheme for unstructured grids. An example of the continuous-field case emerges in the simulation of a low Mach number compressible gas flow where the mass density fields, for example, are smooth continuous functions of space. An example of the discontinuous case would be the simulation of a free surface flow such as the sloshing of a liquid in a partially filled tank. Such an integrated scheme would be an attractive candidate in many of today's general-purpose flow simulation software, particularly those that address multimaterial and multifield flow problems.

Thus, we present in this paper a new general-purpose advection scheme for unstructured meshes based on the use of an interface-tracking flux formulation recently put forward by Ubbink and Issa [10], in combination with an extended version of the flux-limited advection scheme of Thuburn [8], for continuous fields. In what follows, we provide a brief review of the key features of the methods of Thuburn and Ubbink and Issa. Following this, we outline our scheme and then provide results on a variety of advection examples to show its accuracy and robustness. The examples include continuous-field advection, interface tracking, and explicit–implicit advection on quadrilateral and triangular element grids (all treated internally as unstructured grids). We conclude with a discussion of future work, including an outline of the use of our scheme in a continuous-remapping, Lagrangian–Eulerian scheme to accommodate right-hand-side physics.

2. OVERVIEW OF PRIOR WORK

Thuburn [8] developed a multidimensional flux-limiting scheme which ensures monotonic, compatible continuum advection with shape preservation. A key aspect of Thuburn's method is his criterion for the selection of limiting values for fluxed quantities based on information from both face and vertex neighbors. The inclusion of the vertex neighbor values provides the shape-preservation or corner-coupling properties of his scheme. Another

powerful aspect of Thuburn's method derives from the fact that as a flux-limiting scheme, it can accommodate a wide variety of fluxed field reconstruction techniques. Thuburn used a third-order UTOPIA scheme appropriate to structured meshes for many of his example calculations. However his limiting scheme accommodates the use of alternatives. We will exploit this flexibility in the scheme presented in this paper and provide a simplification to Thuburn's limiting procedure that makes it computationally more efficient on unstructured grids.

Thuburn's scheme also ensures the compatibility of mixing ratio quantities such as temperature and velocity, so it is applicable to a wide variety of applications. Finally, Thuburn describes how his scheme can be used on quantities such as mass density, which can exhibit new extrema as the result of nonsolenoidal transport velocity fields. Thus, the Thuburn scheme can be applied to virtually any advection task in a coupled simulation involving conservation of mass, momentum, energy, etc. Our second extension of the Thuburn's scheme will be to regularize his advection and limiting equations so that they maintain compatibility even in the limit of vanishing mass density. This extension is particularly important for the cases of multimaterial and multifield flow simulations.

Following up on the original volume-of-fluid (VOF) scheme of Hirt and Nichols [3], and on ideas for compressive reconstruction by Gaskell and Lau [2], Ubbink and Issa [10] have recently published a novel interface-tracking scheme. In their method they compute fluxed quantities from a weighted average of estimates based on continuous-field reconstruction and one based on simple compressive reconstruction. Ubbink and Issa's weighting factor is based on the angle between the interface surface normal and the conservation cell face surface normal and is astutely chosen to avoid the interface-wrinkling effects seen in previous versions of the VOF method. Ubbink and Issa use this reconstruction strategy for the computation of fluxed quantities inside a Crank–Nicholson time discretization of the advection operator to provide good shape preservation properties for interface-tracking simulations. To enforce monotonicity, they introduce a correction step to their scheme applied after each successive solution of the Crank–Nicolson system until they have a monotone future field.

We have developed two variations of the Ubbink–Issa method for use in our general–purpose scheme. Our first variation is completely explicit; the second utilizes Ubbink and Issa's Crank–Nicolson scheme. For the explicit scheme, we first modify the Ubbink–Issa flux quantity formulation to include our multidimensional continuous-field flux formulation in place of their use of the ULTIMATE-QUICKEST [4], high-order, one-dimensional formulation. This allows us to integrate our interface-tracking formulation neatly with our continuous-field formulation. We find that this formulation works well on both triangular and quadrilateral element meshes. In this mode, we apply our modified version of Thuburn's multidimensional limiter to ensure monotonicity. In our second variation, we use Ubbink and Issa's Crank–Nicolson approach to obtain the fluxed quantities and replace their iterative correction with an application of our modified version of Thuburn's limiter. This yields bounded future values without the need for an iterative correction scheme to enforce monotonicity. Thus, the properties of conservation, monotonicity, compatibility, and shape preservation are enforced simultaneously with the solution of the equations.

Finally, we also develop in our scheme an explicit/implicit capability by coupling our flux formulations for low Courant numbers to a fully implicit flux formulation, in a smooth fashion, to provide a continuously applicable method for virtually all Courant numbers.

3. ADVECTION SCHEME

In the scheme presented in this paper, we adopt and extend the flux-limiting scheme of Thuburn. We use both continuous-field reconstruction and our two modified versions of the reconstruction strategy of Ubbink and Issa to provide an integrated advection scheme applicable to both continuous-field advection and interface tracking for a wide variety of applications. For maximum flexibility, we provide some modest extensions to the Thuburn limiting scheme to accommodate fully unstructured grids and smooth compatible advection even in the limit of vanishing mass densities.

3.1. *Conservation Equations*

Without sacrificing generality, we start with the following prototypical pure-advection conservation system for mass with density ρ and an additional conserved quantity with density ρq. The equations are

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

and

$$\frac{\partial \rho q}{\partial t} + \nabla \cdot (\rho q \vec{u}) = 0. \tag{2}$$

Here $\vec{u}$ is the velocity field and q is a mass specific quantity sometimes called a mixing ratio. It is useful to consider the advective forms of these equations. Equation (1) can be written as

$$\frac{D\rho}{Dt} = -\rho \nabla \cdot \vec{u}, \tag{3}$$

and Eq. (2) as

$$\frac{Dq}{Dt} = 0, \tag{4}$$

where

$$\frac{D}{Dt} \equiv \frac{\partial}{\partial t} + \vec{u} \cdot \nabla$$

is the familiar material derivative which gives the time rate of change following the material motion. We can now more precisely discuss the concepts of monotonicity and compatibility. First, from Eq. (3) we can see that only velocity divergence can cause changes in mass density. So a monotonic advection scheme is one that does not introduce any artificial extrema into the mass density field due to numerical inaccuracies. In particular, for the case of incompressible or divergence-free velocity fields, a monotonic field should ensure that no new mass density extrema are created.

As for compatibility, we can see from Eq. (4) that mixing ratio quantities should remain constant following the fluid motion even for cases in which the velocity field has a divergence. Thus a compatible advection scheme is one in which no new extrema are artificially

created due to numerical inaccuracies even for the cases in which the velocity divergence is nonzero.

We now develop the discrete finite-volume versions of Eqs. (1) and (2). In preparation, let us define several quantities. Let V_k be the volume corresponding to the kth node of a given mesh. Let the average mass density and mixing ratio for that node be defined as

$$\rho_k \equiv \frac{1}{V_k} \int_{V_k} \rho \, dV,$$

$$q_k \equiv \frac{1}{\rho_k V_k} \int_{V_k} \rho q \, dV.$$

Then, following Thuburn, we can integrate Eqs. (1) and (2) over the control volume and, over a time interval, apply the divergence theorem and discretize the resulting surface integrals to obtain the finite-volume versions of Eqs. (1) and (2),

$$\rho_k^{m+1} = \rho_k^m + \sum_f C_{k-f}^{in} \hat{\rho}_f - \sum_f C_{k-f}^{out} \hat{\rho}_f \tag{5}$$

and

$$\rho_k^{m+1} q_k^{m+1} = \rho_k^m q_k^m + \sum_f C_{k-f}^{in} \hat{\rho}_f \hat{q}_f - \sum_f C_{k-f}^{out} \hat{\rho}_f \hat{q}_f, \tag{6}$$

respectively. The superscripts m and $m+1$ denote the known and unknown time levels, respectively, separated by a time interval, Δt. The summations in Eqs. (5) and (6) are carried out over all faces of the control volume. The quantities C_{k-f}^{in} and C_{k-f}^{out} are inflow and outflow Courant numbers for face f of the kth control volume and are defined as

$$C_{k-f}^{in} = H(-\vec{u}_f \cdot \hat{\mathbf{n}}_{k-f}) \frac{\hat{V}_f}{V_k} \tag{7}$$

and

$$C_{k-f}^{out} = H(\vec{u}_f \cdot \hat{n}_{k-f}) \frac{\hat{V}_f}{V_k}, \tag{8}$$

where H is the Heaviside function, $\vec{u}_f$ is the average velocity on face f, $\hat{n}_{k-f}$ is the outward surface unit normal of face f for the kth control volume, and $\hat{V}_f$ is the volume of material that passes through face f over the time interval Δt given by the formula

$$\hat{V}_f = \Delta t |\vec{u}_f \cdot \hat{n}_f| A_f, \tag{9}$$

where A_f is the area of face f and $\hat{n}_f$ is either one of the face normal vectors for the adjacent control volumes. The quantity $\hat{V}_f$ is often referred to as the *flux volume*.

The quantities $\hat{\rho}_f$ and $\hat{q}_f$ in Eqs. (5) and (6) are the average values of the ρ and q over the flux volumes for face f. We will refer to these quantities as *fluxed values* or *fluxed quantities*.

In order to compute the updated values of q we follow Thuburn [8] and divide the left- and right-hand side of Eq. (5) into the respective left and right sides of Eq. (6) and then divide the resulting right-hand side numerator and denominator by ρ_k^m to obtain

$$q_k^{m+1} = \frac{q_k^m + \sum_f C_{k-f}^{in}(\hat{\rho}_f/\rho_k^m)\hat{q}_f - \sum_f C_{k-f}^{out}(\hat{\rho}_f/\rho_k^m)\hat{q}_f}{1 + \sum_f C_{k-f}^{in}(\hat{\rho}_f/\rho_k^m) - \sum_f C_{k-f}^{out}(\hat{\rho}_f/\rho_k^m)}. \tag{10}$$

Equation (10) is the final update formulation presented by Thuburn [8] for the mixing ratio quantity. While this form was suitable to Thuburn's purposes, we must modify it in order to ensure smooth calculations in the limit as $\rho_k^m \to 0$. This limit is ubiquitous in multifield flows. Consider, for example, the simple case of a sedimenting two-phase flow. The density of the settling material is zero above the two-phase region. In computations of such a flow, we would like mixing ratio quantities to take on sensible values and not simply "blow up." Accordingly, we recast Eq. (10) by extracting q_k^m from the right-hand side as a separate term and then by multiplying the numerator and denominator of the remainder by ρ_k^m. This leaves Eq. (10) in the form

$$q_k^{m+1} = q_k^m + \frac{\sum_f C_{k-f}^{in}\hat{\rho}_f(\hat{q}_f - q_k^m) - \sum_f C_{k-f}^{out}\hat{\rho}_f(\hat{q}_f - q_k^m)}{\rho_k^{m+1} + \varepsilon}. \tag{11}$$

Here we have used Eq. (5) to further simplify the denominator of the second right-hand-side term. Note that we have also added the quantity ε to the denominator of the second term in Eq. (11). This is intended to be a small positive parameter that will be used to prevent division by zero in numerical computations. Now consider what happens in the event of ρ_k^{m+1} approaching zero. In this case, as long as the numerator of the second right-hand-side term approaches zero, the mixing ratio, q_k, will simply not change. This is automatic for the cases in which both inflow and outflow masses are zero. For other cases, this desirable behavior will emerge from the application of Thuburn's flux limiter with a few minor extensions. This is the subject of the next section.

3.2. *Modified Thuburn Flux Limiter*

In this section we discuss our modifications to Thuburn's flux limiter for the fluxed quantities in Eq. (11). Since Thuburn's limiting procedure is indifferent to the method used to compute fluxed quantities, we delay discussion of the formulations for the unlimited fluxed quantities to the next section. Note that, as Thuburn [8] teaches, his flux limiter can be used to limit both the mass density fluxed values, $\hat{\rho}_f$, as well as the mixing ratio fluxed values, $\hat{q}_f$. We continue to use this approach. We will elaborate more on this later. For now, we will discuss our modifications to Thuburn's limiting scheme to accommodate unstructured meshes and to allow for vanishing mass densities. We will confine our attention here only to our modifications and point the reader to Thuburn's paper [8] for the full explanation of his method.

3.2.1. Upstream-neighbor effects. The first modification of Thuburn's limiting proce- dure that we have introduced generalizes a particular aspect of his method to unstructured grids. One of the steps in Thuburn's limiting procedure involves widening the limiting bounds on fluxed quantities to reflect contributions from inflows from control volumes ad- jacent to the control volume upwind from a given face. This is the part of Thuburn's scheme

that provides the shape-preservation characteristics. In Thuburn's original prescription, only face neighbors from the upwind control volume that had a common vertex with the flux face in question were included. Thus, for a quadrilateral, the face neighbor opposite the flux face in question was excluded. For unstructured meshes, this exclusion, while not impossible, is quite onerous. It is much easier in, for example, the edge-based data structures that we used in many of our computations to simply include the effect of all face neighbors from the upstream control volume. This allowed us to accumulate the upstream-neighbor information at each node about upstream-face or, equivalently, edge-neighbor values simply using a sweep through all mesh edges. Then, when performing Thuburn's technique of widening bounds on fluxed quantities for a given face, we can simply look to this accumulated information on the upwind node for the face. We found that our results were not affected by including this additional information.

3.2.2. Protection against division by zero in the limiter. The second modification to Thuburn's limiting procedure is introduced to protect against division-by-zero problems when no outflow exists. Thuburn provides formulas for bounds on outflow fluxed quantities using the conservation equations and information from surrounding neighbor node data. Our alternative forms for these formulas are

$$
\left(\hat{q}_k^{(out)}\right)_{\min} = q_k^m + \frac{\sum_f \left\{ C_{k-f}^{in} \hat{\rho}_f \left[\left(q_f^{(in)}\right)_{\max} - q_k^m \right] \right\} - \left\{ \left(q_k^{m+1}\right)_{\max} - q_k^m \right\} \rho_k^{m+1}}{\sum_f C_{k-f}^{out} \hat{\rho}_f + \varepsilon}, \quad (12)
$$

$$
\left(\hat{q}_k^{(out)}\right)_{\max} = q_k^m + \frac{\sum_f \left\{ C_{k-f}^{in} \hat{\rho}_f \left[\left(q_f^{(in)}\right)_{\min} - q_k^m \right] \right\} - \left\{ \left(q_k^{m+1}\right)_{\min} - q_k^m \right\} \rho_k^{m+1}}{\sum_f C_{k-f}^{out} \hat{\rho}_f + \varepsilon}, \quad (13)
$$

where $(\hat{q}_k^{(out)})_{\min}$ and $(\hat{q}_k^{(out)})_{\max}$ are the minimum and maximum bounds for the outflow fluxed values for q for the kth control volume that ensure compatible advection. The quantities $(q_f^{(in)})_{\max}$ and $(q_f^{(in)})_{\min}$ are the maximum and minimum inflow values for the fluxed quantities for face f based on upstream data, including the effect of the nodes adjacent to upstream control volumes, as discussed in [8], and including our own modifications, discussed in Section 3.2.1 of this paper. Finally, $(q_k^{m+1})_{\max}$ and $(q_k^{m+1})_{\min}$ are the maximum and minimum future values for the quantity q based on the values from upstream neighbors [8].

With the exception of the additional term, ε, in the denominators of Eqs. (12) and (13), these equations are algebraically identical to Eqs. (42) and (43) in [8]. (Note, the small quantity ε in Eqs. (12) and (13) is not necessarily the same as the ε in Eq. (11).) In our form, we have separated the right-hand side into two parts: the first being the time m value, q_k^m, which would correspond to explicit donor-cell advection, and the second containing the high-order correction. We have added the small positive quantity ε to the denominator simply to protect from division by zero in machine computations when no outflow exists. In this limit, $(\hat{q}_k^{(out)})_{\max}$ and $(\hat{q}_k^{(out)})_{\min}$ may have large values but will not be used since there is no ouflow.

3.2.3. Clipping to enforce future bounds. In principle, the use of Thuburn's limiter with the extensions outlined above should result in computed future values of q that are bounded by $(q_k^{m+1})_{\max}$ and $(q_k^{m+1})_{\min}$, that is, compatible. In practice, particularly for the case of vanishing mass densities, we have found that round-off errors can produce unbounded results. We therefore introduce an additional step to Thuburn's scheme after the update using Eq. (11), which clips any result that is out of bounds. We feel justified in doing

this because the results should be bounded. Any bounds violation is due only to round-off problems. Specifically, we perform the following operations at each node:

$$q_k^{m+1} \rightarrow \min\left[q_k^{m+1}, \left(q_k^{m+1}\right)_{\max}\right]$$

and

$$q_k^{m+1} \rightarrow \max\left[q_k^{m+1}, \left(q_k^{m+1}\right)_{\min}\right].$$

This then provides a compatible advection scheme. This only seems necessary in the limit of vanishing mass density and thus should not introduce any substantial conservation errors.

3.3. Limiting Mass Densities for Monotonic Advection

As Thuburn has pointed out [8], we can use his limiting scheme to place bounds on the fluxed mass densities themselves to avoid unphysical extrema. Thuburn's strategy is simply to replace q with ρ and the product $C\hat{\rho}$ with C in his limiting procedure. (Here C stands for either C_{k-f}^{in} or C_{k-f}^{out}.) By doing this, the limited mass densities are bounded by the surrounding mass density data after including the effects of compression and expansion by the divergent velocity field. The same strategy applies to our modified version of Thuburn's limiter presented above.

3.4. Fluxed Quantity Formulation

Now that we have outlined our modifications to the Thuburn limiting scheme, we are in a position to discuss our formulation for the computation of the flux quantities before limiting. Our approach here is first to provide an alternative to the third-order UTOPIA scheme used by Thuburn to one that is more appropriate to unstructured grids. Following this we then provide an overview of our modified version of the Issa–Ubbink flux formulation for interface-tracking computations.

3.4.1. Continuous field flux formulation. For the examples considered in this paper, we approximate the variation of a given field, ϕ, in the kth control volume as a linear function,

$$\phi_k(\vec{x}) = \phi_k + (\nabla\phi)_k \cdot (\vec{x} - \vec{x}_k) + \cdots, \tag{14}$$

where ϕ_k is the average value of ϕ in the control volume, $(\nabla\phi)_k$ is its gradient, and $\vec{x}_k$ is the cell centroid. The use of the linear function (14) provides more accuracy than a first-order donor cell scheme, as is well-known. We therefore term this our higher order formulation. Note also that we are not limited to a linear function; the formulation admits even higher order expressions. Finally, note that for some meshes, $\vec{x}_k$ is not always the same as the position of the node. In the examples that follow, for simplicity, we used the node position instead of the centroid and accept the error. We then obtain an estimate of the average value of ϕ in the flux volume by evaluating Eq. (14) at a point half way up the characteristic (back in time) following the fluid motion from the centroid of the face f, $\vec{x}_f$. That is,

$$\hat{\phi}_f = \phi_u\left(\vec{x}_f - \frac{1}{2}\vec{u}\Delta t\right), \tag{15}$$

where the subscript u denotes the upwind control volume.

The gradient in Eq. (14) must be supplied to the method by some suitable numerical differencing scheme. In our work we used a simple, so-called, Green–Gauss method

$$(\nabla\phi)_k \approx \frac{1}{V_k} \sum_f \phi_f A_f \vec{n}_{k-f}, \tag{16}$$

where the quantity ϕ_f is the average value of ϕ for face f. We employ two methods to compute this average value from surrounding nodal data. In the first method, we simply use the average of the two node values across the face f. In the second method, we compute average values of ϕ for each of the vertices of face f. These vertex values are computed as averages of local surrounding data. Since each face vertex corresponds to a unique mesh element and is approximately at the center of this element, we use the element node data as the surrounding data. These face vertex values are then averaged with the two values across the face to give the final face value. For example, the first method corresponds to a five-point stencil on structured quadrilaterals, while the second method corresponds to that of a nine-point stencil.

This approach is general and works for all element types. In our computations, the first method was used for continuous-field advection and the second method was used for interface tracking advection. This was done because we found that the second method gave improved shape preservation properties to the interface tracking calculations.

3.4.2. Discontinuous-field flux formulation. We now address the case of discontinuous fields. Specifically, we are interested in cases in which we are trying to track interfaces in our calculations. In these cases, the discontinuous-field is a volume fraction. In multifield flow simulations, the volume fraction, α, is related to the conserved macroscopic density, ρ, and the microscopic intrinsic material density, ρ^o as $\alpha = \rho/\rho^o$. We now discuss our modified formulation of the Ubbink–Issa formulation for volume fraction fluxes, $\hat{\alpha}$. Once these volume fraction fluxes are computed we may then reconstruct the mass density fluxes as $\hat{\rho} = \hat{\alpha}\hat{\rho}^o$.

3.4.3. Modified Ubbink–Issa formulation. Let us now consider directly the flux formulation for a volume fraction field. We assume that we are completely resolving material interfaces and the volume fraction is either 0 or 1 at any point in space in the continuous limit. Of course, in the numerical calculation, the average volume fraction in a control volume can be fractional if the volume is only partially filled with the material in question. This means that the nodal values for volume fraction can vary continuously between 0 and 1 over three contiguous control volumes. This is the case that Ubbink and Issa [10] have recently addressed.

The original VOF method by Hirt and Nichols [3] made use of the combination of a compressive reconstruction and a continuous-field reconstruction in order to maintain the sharp step profile of a discontinuous-field such as a fluid interface. Ubbink and Issa [10] recently provided a very effective means of smoothly varying between the compressive and continuous-field schemes by taking the weighted average between the two. They make use of a locally bounded, one-dimensional, differencing scheme first described by Gaskell and Lau [2] for the compressive reconstruction, and Leonard's one-dimensional, high-order ULTIMATE-QUICKEST [4] for their continuous-field reconstruction. The weighting between the compressive and continuous-field reconstruction is a function of the angle between interface normal vectors, as determined from upstream volume fraction gradients and cell

face normal vectors. Ubbink and Issa extend these hybrid one-dimensional treatments to multiple dimensions by time centering the fluxed values using a Crank–Nicolson scheme. That is, they apply their hybrid flux formulation to the time m and $m + 1$ data separately and then average to obtain the final flux values for volume fraction that are used in the conservation equation. Thus, their method has an implicit component, so finding the time $m + 1$ solution requires a system solver. To enforce monotonicity, Ubbink and Issa apply a correction between successive linear solutions to fix out-of-bounds values. For a more detailed description of their algorithm see [9, 10].

We employ the bulk of Ubbink and Issa's scheme here in our general-purpose framework. We make, however, two separate modifications to produce two alternative variations more suitable to our needs. In the first variation, we replace the use of the ULTIMATE-QUICKEST, one-dimensional high-order formulation for the continuous-field flux value with our multidimensional reconstruction scheme outlined above. We then use a weighting of this high-order flux value, $\hat{\phi}_{ho}$, with the one-dimensional compressive, $\hat{\phi}_c$, reconstruction as outlined in [10],

$$\hat{\phi}_f = \omega\hat{\phi}_c + (1 - \omega)\hat{\phi}_{ho}. \tag{17}$$

The weighting ω [10] varies smoothly between 1 and 0 as the angle between the face normal and the interface normal varies between 0 and $90°$. We then use this flux formulation in a fully explicit mode and employ our modified version of Thuburn's limiter on the resulting fluxed values, $\hat{\phi}_f$, to ensure monotonocity. In our second variation of the Ubbink–Issa scheme, we use a slightly modified version of their Crank–Nicolson formulation wherein we replace their iterative correction scheme with an *a posteriori* application of our modified version of Thuburn's limiting scheme [8], ensuring only one linear solution per time step. In the remainder of this paper the first method will be referred to as the "explicit" scheme while the latter will be referred to as the "implicit" or "Crank–Nicolson" scheme.

In Ubbink and Issa's scheme, the volume fraction flux is computed as a weighted average of the upwind and downwind node volume fractions

$$\hat{\alpha}_f = (1 - \beta_f)\alpha_D + \beta_f\alpha_A, \tag{18}$$

where α_D and α_A are the downwind and upwind node volume fractions. The weight factor β_f is a complicated function of the surrounding volume fractions and the material interface and control volume face surface unit normal vectors [10]. In order to avoid introducing a nonlinear system for the new time volume fraction, Ubbink and Issa advocate time lagging the weight factors in their Crank–Nicolson formulation for multidimensional computations. Thus, their final expression for the fluxed volume fraction is

$$\hat{\alpha}_f = (1 - \beta_f)\frac{\alpha_D^m + \alpha_D^{m+1}}{2} + \beta_f\frac{\alpha_A^m + \alpha_A^{m+1}}{2}, \tag{19}$$

where β_f is evaluated with time m data. Ubbink and Issa imply that the fully nonlinear fluxing value, with β_f values evaluated with both time m and $m + 1$ data, would actually do a better job of representing the time average of volume fluxes through a given time step. We initially implemented the full nonlinear version. Since we are now routinely solving nonlinear systems using the Jacobian–free Newton–Krylov solution methods [1], this

extension of their method did not place any additional requirements on our solution software. However, we did not see any appreciable additional benefit of the nonlinear fluxing values. Since the linear fluxing values are more computationally economical, we chose to retain these in our work.

We now deviate from [10], as mentioned above, by applying our modified version of Thuburn's limiter to the resulting converged time-centered, Crank–Nicolson flux values for density. These limited fluxing quantities are then used to obtain the future values of density from Eq. (5).

3.4.4. Explicit–implicit flux value formulation. Although we have not performed a Von Neumann stability analysis, our scheme appears to be stable as long as the sum of all outflow face Courant numbers is less than 1.0 for any node. We can, however, combine our explicit scheme with implicit schemes in order to compute at higher Courant numbers. We do this by combining flux values computed using our high-order explicit formulation with explicit upwind and a fully implicit flux value in a smooth continuous fashion. Let us denote the explicit first-order upwind flux value $\hat{\phi}_{f\text{-}donor}$, the implicit flux value $\hat{\phi}_{f\text{-}ss}$, and the limited explicit high-order continuous field flux $\hat{\phi}_{f\text{-}ho}$. In this paper, we used the upwind node time $m + 1$ value for $\hat{\phi}_{f\text{-}ss}$. Implicit donor-cell advection, while inaccurate, is known to be stable for all Courant numbers. This choice can, of course, be generalized to more sophisticated options in the future. Then our combined expression for the explicit–implicit fluxed value is

$$\hat{\phi}_f = \hat{\phi}_{f\text{-}implicit} + \hat{\phi}_{f\text{-}explicit}, \tag{20}$$

where

$$\hat{\phi}_{f\text{-}implicit} = \max\left[0, 1 - \frac{C_{\max}}{C_{D_f}}\right]\hat{\phi}_{f\text{-}ss} \tag{21}$$

and where the explicit fluxed value is a combination of the explicit first-order and high-order values

$$\hat{\phi}_{f\text{-}explicit} = \min\left[1, \frac{C_{\max}}{C_{D_f}}\right]\left\{\min\left[1, \left(\frac{C_{D_f}}{C_{\max}}\right)^{\beta}\right]\hat{\phi}_{f\text{-}donor}\right.$$
$$\left. + \left(1 - \min\left[1, \left(\frac{C_{D_f}}{C_{\max}}\right)^{\beta}\right]\right)\hat{\phi}_{f\text{-}ho}\right\}, \tag{22}$$

where C_{D_f} is the sum of all outflow Courant numbers for the upwind control volume for face f,

$$C_{D_f} = \sum_{f_u} C_{D\text{-}f_u}^{out}. \tag{23}$$

The quantity $C_{\max}$ is introduced as a maximum value for the sum of the outflow face Courant numbers, C_{D_f}, which can be controlled by the user. For stability $C_{\max}$ should be set to a value less than 1.0.

Equation (22) was obtained by blending the high-order and donor-cell fluxes so that the explicit flux tends toward the explicit donor-cell value as the Courant number sum, C_{D_f}, approaches $C_{\max}$. This is done to keep high-order flux values from introducing noise

as the evaluation point on the characteristic moves farther back and potentially outside the upstream control volume. The exponent β controls the rate of transition between the high-order flux values and the explicit donor value. As β approaches infinity the transition between the two schemes becomes a step function with the change occurring abruptly at C_{D_f} equals $C_{\max}$. In our calculations we chose a value of β equals 50 to ensure a smooth transition. This value for C_{D_f} delayed the onset of transition from high-order to first-order upwind such that it occurred between $C_{\max} - 0.05$ and $C_{\max}$. In our calculations, we set $C_{\max}$ to 0.85, which is safely away from the apparent stability limit of 1.0 for the explicit part of the scheme.

As an example of how this scheme works, consider the case C_{D_f} equals 2. For this case the final fluxed quantity is weighted as 0.425 from the explicit donor cell and 0.575 from the implicit value. Notice also that in the limit, as the Courant number tends to infinity, the explicit flux value tends to zero and the implicit flux value tends to the steady-state flux value. This property can be exploited so that the advection scheme can work in both a time-dependent and a steady-state service (see Section 5).

4. RESULTS AND DISCUSSION

In this section we present several test cases that exhibit the accuracy of the various components of our new integrated advection scheme. In each of the examples the density, ρ, is updated at each time step with Eq. (5). In several of the examples we also compute a temperature field, q (with unit specific heat), that is transported with the material according to Eq. (11). With this choice of variables, the product ρq can be usefully considered a mixture temperature. (For example, if we have the special case of a two-phase, nonconducting mixture, with each phase having unit microscopic material density, one phase with zero temperature, and the other with temperature q, then ρq would be the mixture temperature.) In all cases, we compute with flux values computed with the reconstruction methods discussed in this paper and limit them with our modified version of Thuburn's limiter [8]. We first show continuous-field advection on structured and unstructured meshes. Shape preservation on both quadrilateral and triangular element meshes is demonstrated. Our next two example calculations demonstrate the accuracy of our interface-tracking mode for the cases of simple translation and deformation in a vortical shear field. Both of these are common test problems used in validating interface-tracking methods. The simple translation test case includes the transport of a temperature field with the material as well. Here we demonstrate good shape preservation as well as compatible advection of a mass specific quantity, temperature. In the case of vortical shear flow, under time reversal, we show that we essentially recover the initial conditions. Error norms from these simulations on quadrilateral and triangular element meshes are tabulated along with those of [7, 10]. Finally, we show a computation of advection of concentration in a jet flow field, which shows the efficacy and utility of our explicit–implicit formulation.

Two domains, a unit square and a rectangle, have been used to compute the results presented in this section. Each domain is discretized, with structured quadrilateral elements and unstructured triangular elements having approximately the same number of nodes. Figure 1 shows a coarse representative example of the types of triangular meshes used for the example calculations in this paper. The quadrilateral meshes have a grid spacing of 0.01 for the unit square and 0.02 for the rectangular mesh in each coordinate direction.
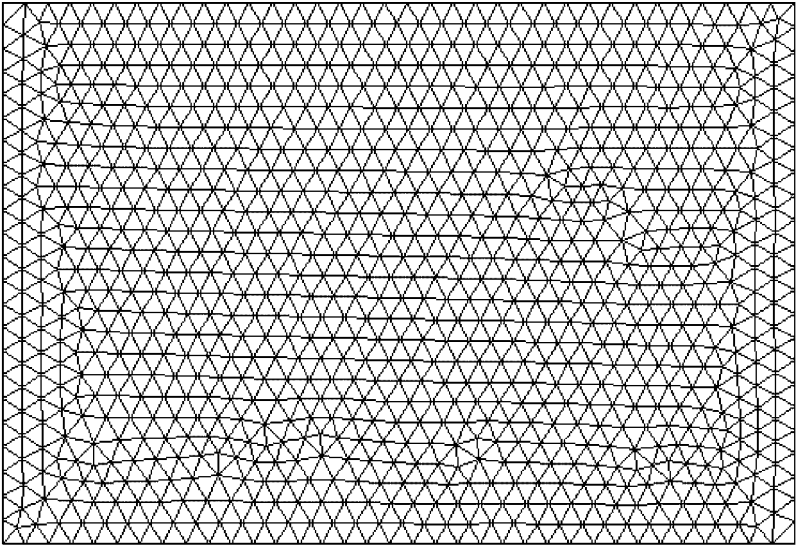
FIG. 1. Coarse representative example of a triangular element mesh.

4.1. *Shape-Preserving Continuous-Field Advection*

We first demonstrate the accuracy of our scheme for continuous-field advection on both structured quadrilateral and unstructured triangular grids. The domain for the computation is the unit square.

We start with an initial condition in which the density field is initialized with the distribution

$$\rho = \frac{1}{2} \begin{cases} 1 + \cos\left[\frac{\pi\sqrt{(x-0.2)^2+(y-0.2)^2}}{0.2}\right] & \text{for } \sqrt{(x-0.2)^2+(y-0.2)^2} < 0.2, \\ 0 & \text{otherwise.} \end{cases}$$

The temperature of this material is simply set to one where there is material and zero where there is not. That is,

$$T = \begin{cases} 1 & \text{for } \sqrt{(x-0.2)^2+(y-0.2)^2} < 0.2, \\ 0 & \text{otherwise.} \end{cases}$$

The existence of zero material density on the mesh provides us with a good test of our modifications to Thuburn's limiter. At time zero, the material is subjected to a velocity field along the diagonal of the unit square with unit velocities along the two coordinate directions. The Courant number for these computations, based on the node volumes, was 0.5. The computations were carried out over 240 time steps. The results of the computations are shown in Figs. 2–7.

Figures 2 and 3 show the initial and final density resulting from advection using the present scheme and the first-order donor cell scheme. Figure 2 shows peak heights while Fig. 3 shows shape preservation. In both figures, a is the initial configuration, b is the result using the current scheme on the quadrilateral element mesh, c is the result using the current scheme on the triangular element mesh, and d is the result for first-order donor cell as a
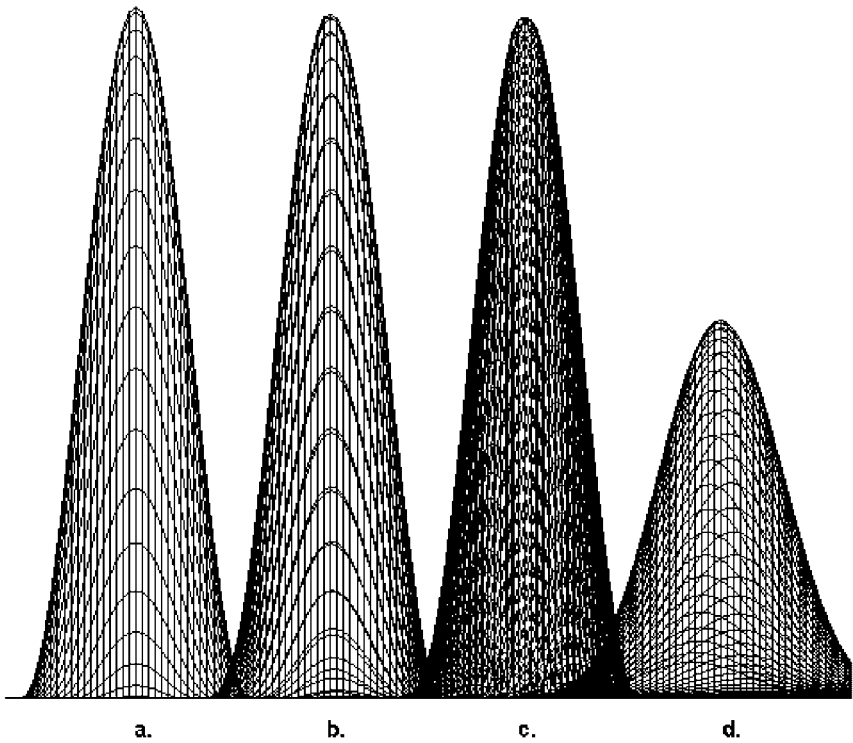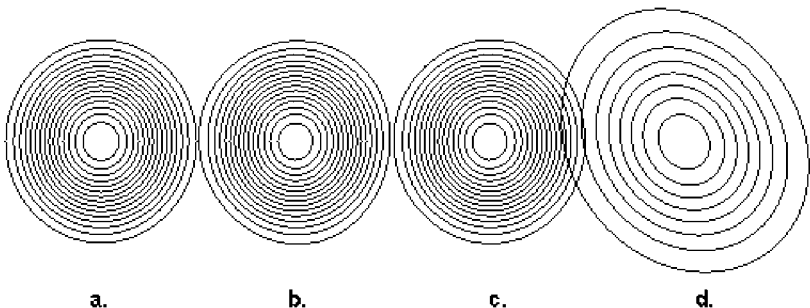
FIG. 2. Continuous-field advection results demonstrating peak preservation for density. (a) Initial condition; (b) quadrilateral element mesh results using current scheme; (c) triangular element mesh results using current scheme; (d) quadrilateral element mesh results using first-order donor cell.

comparison. Figures 3a–3d show the same results in contour form. Here, for Figs. 3a–3c, the inner and outer contour values are 0.0625 and 0.9775 respectively. In Fig. 3d, we have inner and outer contour values of 0.0625 and 0.5. The results in these figures show the monotonicity, the good peak preservation, and the shape-preservation capabilities of the scheme.

As a point of reference for these results, we show in Fig. 3d the corresponding results for a first-order upwind method on the quadrilateral mesh. These results show considerable peak

FIG. 3. Continuous-field advection results demonstrating shape preservation for density. (a) Initial condition; (b) quadrilateral element mesh results using current scheme; (c) triangular element mesh results using current scheme; (d) quadrilateral element mesh results using first-order donor cell.
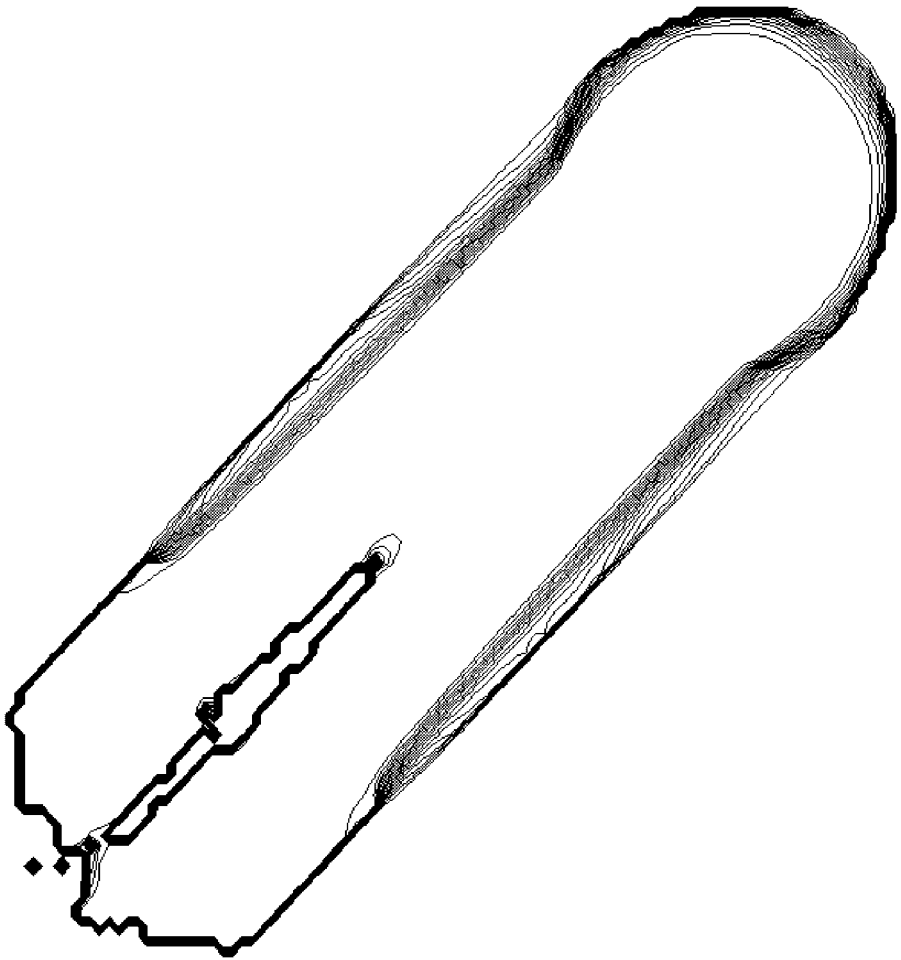
FIG. 4. Line contour plot of the final temperature field for quadrilateral mesh computation. The temperature inside tightly packed contour lines is one; the temperature on the outside is zero.

height loss due to numerical diffusion and loss of shape. The percentage difference between the initial and final peak values are 1.2, 1.5, and 45.4% for Figs. 2b–2d, respectively.

Figures 4 and 5 show line contour plots of the temperature fields, q, resulting from the advection computations. In each case, the temperature is one over the entire path of the material. This is expected by the design of our update, Eq. (11), which leaves the temperature of a node unchanged if it empties or is empty. This is an artifact of having a zero-density material in the background. In a real multiphase flow calculation, for example, this sort of thing would not happen because the temperature of the material outside the plume would be dictated by the physics of the other material. We saw no compatibility problems with the temperatures remaining bounded between zero and one.

A more useful way to look at the temperature in this case is to look at what we called the mixture temperature, ρq, as shown in Figs. 6 and 7. The layout of these figures is the same is in Fig. 2, with the exclusion of the donor cell results. In these figures, we see a more physically intuitive temperature pattern with the mixture temperature nonzero only where the material is.
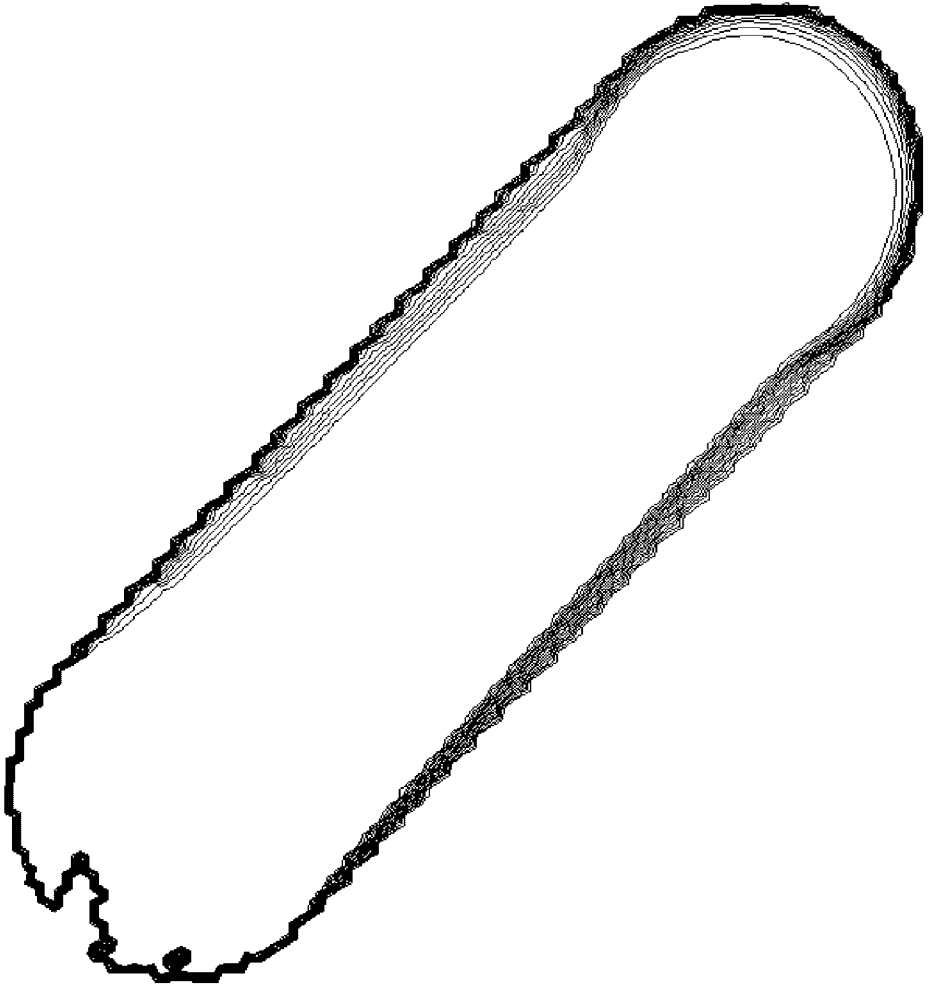
FIG. 5. Line contour plot of the final temperature field for triangular mesh computation. The temperature inside tightly packed contour lines is one; the temperature on the outside is zero.

Finally, we also performed these computations on two unit square meshes with discretizations twice and four times as coarse as the base meshes used in the above examples. From these computations we obtained error norms which showed that the method displays *second-order* convergence for both the quadrilateral and the triangular element meshes.

4.2. *Compatible, Shape-Preserving Interface Tracking Advection*

We now demonstrate the accuracy of the interface-tracking component of the new scheme. For the translation of material in a constant velocity field we start with a unit step in the shape of a hollow square for both density and temperature in the lower left corner of the rectangular domain, centered at $(x, y) = (0.48, 0.48)$. The internal width of the square is 0.4 and the external width is 0.8. This is the same prescription used by Ubbink and Issa [10]. We compute on both the quadrilateral and triangular element meshes of the rectangular domain
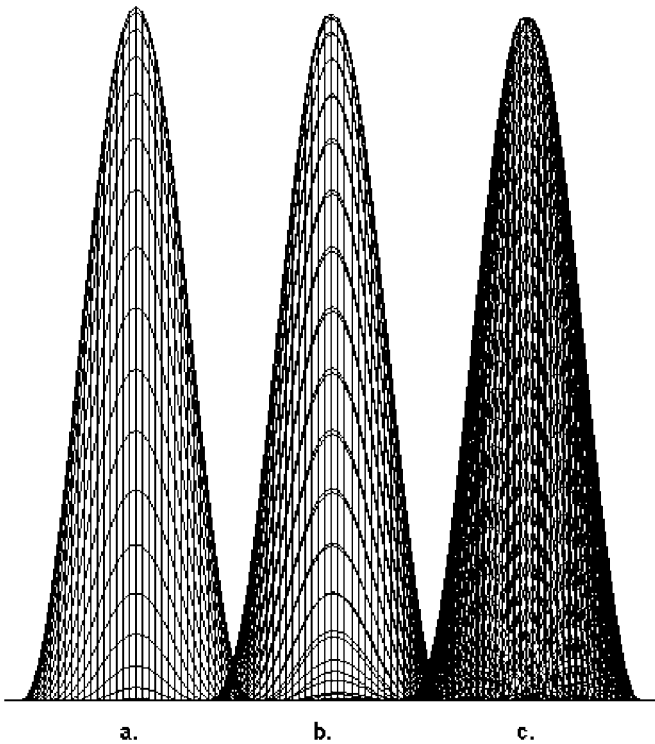
FIG. 6.   Continuous-field advection results demonstrating peak preservation for the mixture temperature (ρq). (a) Initial condition; (b) quadrilateral element mesh results using current scheme; (c) triangular element mesh results using current scheme.

discussed in Section 4. In this case we advect with a velocity field of $(u, v) = (2.0, 1.0)$ and compute at a Courant number of 0.25 over 750 time steps.

The results from these computations for both quadrilateral and triangular meshes are shown in Figs. 8 and 9. The "explicit" and "Crank–Nicolson" methods are as described in Section 3.4.3.

As with the continuous-field advection, we saw no noncompatible temperatures, nor any nonmonotonic density results for all cases. The shape-preservation characteristics of the our schemes look very good as well. To quantify the accuracy of the calculation we computed
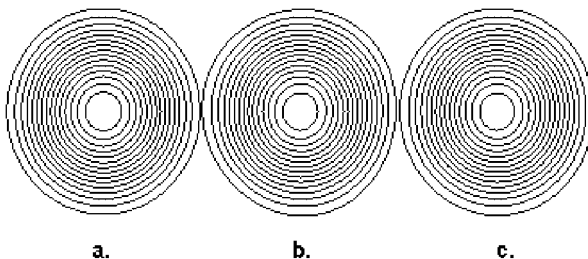

FIG. 7.   Continuous-field advection results demonstrating shape presevation for the mixture temperature (ρq) (a) Initial condition; (b) quadrilateral element mesh results using current scheme; (c) triangular element mesh results using current scheme.
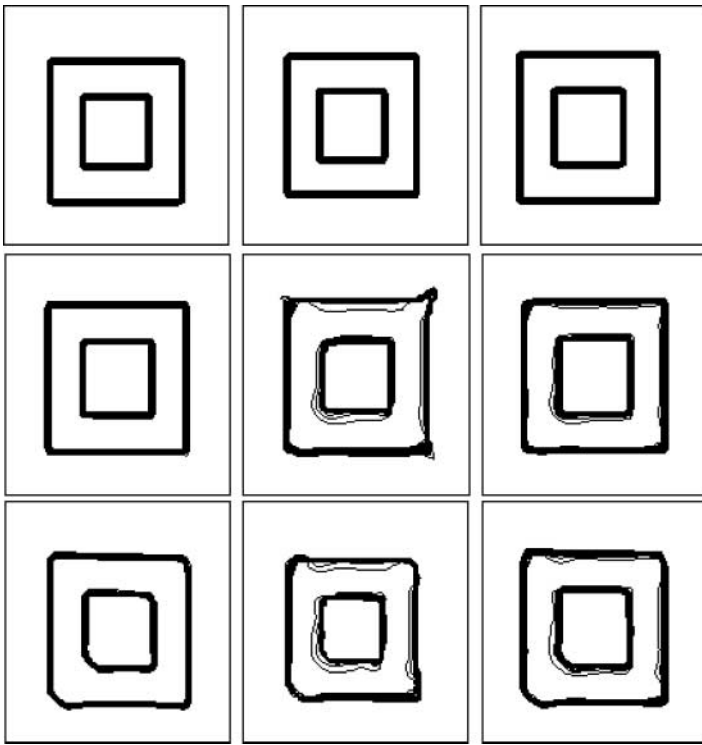
FIG. 8. Density, temperature, and mixture temperature for interface-tracking computations on the quadrilateral element mesh. (Top) Initial conditions; (middle) Crank–Nicolson; (bottom) explicit.
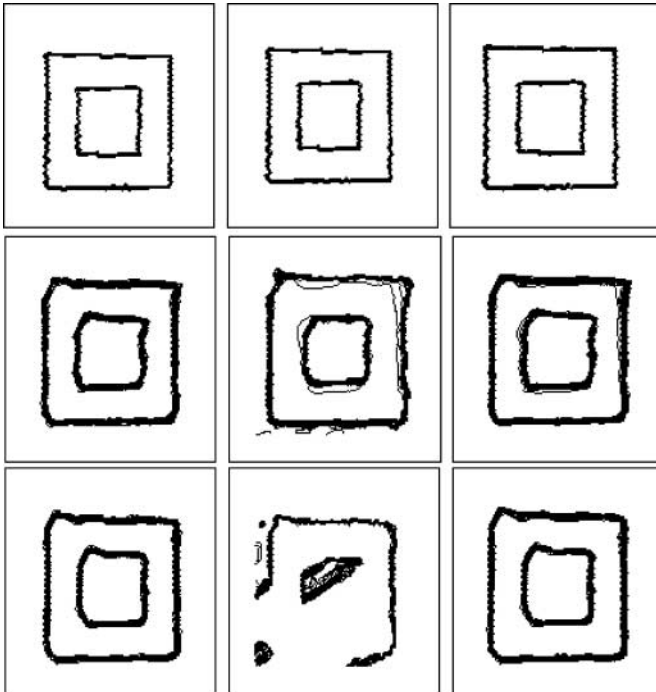

FIG. 9. Density, temperature, and mixture temperature for interface-tracking computations on the triangular element mesh. (Top) Initial conditions; (middle) Crank–Nicolson; (bottom) explicit.

TABLE I
L_1-Error Norms for the Box Translation Problem

Method	L_1-error norm
SLIC [6]	0.1320
Hirt–Nichols [3]	0.0069
FCT-VOF [7]	1.63e-8
Youngs [11]	0.0258
Ubbink and Issa, structured [10]	0.0250
Ubbink and Issa, unstructured [10]	0.0397
Present Crank–Nicolson scheme, quadrilaterals	0.0182
Present Crank–Nicolson, triangles	0.0512
Present explicit scheme, quadrilaterals	0.0311
Present explicit scheme, triangles	0.0800

the L_1-error norm of the final state as defined in [10]. Table I shows the L_1-error norms as presented in [7, 10], as well as those for the current scheme for this example problem. We see that both our implicit and our explicit schemes on the quadrilateral element mesh are comparable to those of Ubbink and Issa's. Note that Rudman's FCT–VOF and Hirt and Nichol's method do exceedingly well for quadrilaterals. This is an artifact due to the alignment of the interface with cell faces and the one-dimensional, direction-split nature of these two schemes. As is seen in the next section the order of these errors is not typical of these schemes for general problems. Shortcomings of our two new methods show up in terms of the L_1-error norms for the triangular element mesh. However, the contour plots show acceptable shape preservation for this problem. As in Section 4.1 it is important not to be confused by the temperature plots in Figs. 8 and 9. This pattern is expected by the design of our update, Eq. (11), as explained in Section 4.1.

4.3. *Interface Tracking in a Vortical Shear Field*

Here we perform a calculation of the deformation of a circular unit step of material in a vortical shear field on the quadrilateral and triangular element meshes with both the C–N and explicit methods. This is a common test problem for interface tracking. In the computation, we advect the material with a prescribed shear velocity field for one rotation. We then reverse the velocity field and advect the material back to the original state. The exact solution to the problem has the material returning to its original position and circular shape. The *x* component of velocity for the forward rotation is

$$u(x, y) = +\sin(\pi x)\cos(\pi y). \tag{24}$$

The corresponding *y* component of velocity is

$$v(x, y) = -\cos(\pi x)\sin(\pi y). \tag{25}$$

The signs of the expressions are flipped for the reverse-rotation part of the computation. The initial and final position of the material is $(x, y) = (0.5, 0.2\frac{1+\pi}{\pi})$ with a radius of 0.2. We performed our computation with a time step such that the maximum Courant number on the quadrilateral element mesh was no greater than 0.25. This corresponds to 2000
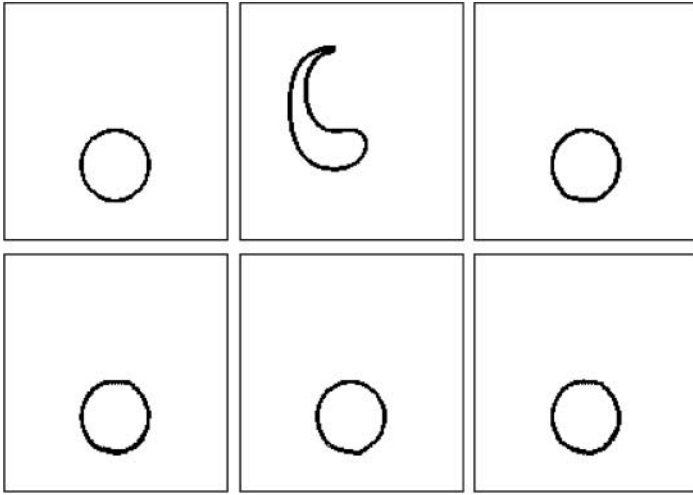
FIG. 10. Contour plot of density for interface-tracking in a vortical shear. The top row shows plots at times of 0.0, 0.048, and 0.08 for the Crank–Nicolson scheme on the quadrilateral mesh. The bottom shows the final result at time 0.08 from left to right for the Crank–Nicolson interface-tracking scheme on the triangular mesh, the explicit interface-tracking scheme on the quadrilateral mesh, and the explicit interface-tracking scheme on the triangular mesh.

total time steps. The number of time steps was retained for the triangular element mesh computation; the maximum Courant number was 0.3. The results of the computations are shown in Fig. 10. The L_1-error norms from Refs. [7, 10], along with those of our methods, are given in Table II.

As seen in the contour plots, the original circular shape is very nearly recovered in all cases. For quadrilateral element meshes the error norms reveal that our explicit method is more accurate than all other methods, except for Young's method. This is not wholly unexpected since the Young scheme involves considerable geometric reconstruction. The results on the triangular element meshes are very promising. The error norms are acceptable and no "flotsam" [7] was generated.

TABLE II
L_1-Error Norms for the Vortical Shear Problem

Method	L_1-error norm
SLIC [6]	0.0459
Hirt–Nichols [3]	0.0660
FCT-VOF [7]	0.0314
Youngs [11]	0.0086
Ubbink and Issa, structured [10]	0.0290
Ubbink and Issa, unstructured [10]	0.0182
Present implicit scheme, quadrilaterals	0.0181
Present implicit scheme, triangles	0.0388
Present explicit scheme, quadrilaterals	0.0155
Present explicit scheme, triangles	0.0346

4.4. *Explicit–Implicit Advection in a Jet Flow Field*

To test our explicit–implicit advection scheme a velocity field with varying Courant numbers was necessary. In this final example, we perform a simulation of the transport of a concentration pulse in a two-dimensional jet in which the velocity decreases by 90% over the computational domain. We take our velocity field from the velocity potential for a two-dimensional, inviscid-flow source doublet where the potential is given by

$$\varphi = -\frac{\mu}{2\pi} \frac{\cos \vartheta}{r},$$

where r, ϑ are polar coordinates and μ is a source strength. On the unit square, the velocity components, (u, v), are

$$u = \frac{\mu}{2\pi} \frac{\xi^2 - \eta^2}{r^4}$$

and

$$v = \frac{\mu}{\pi} \frac{\xi \eta}{r^4},$$

where $r^2 = \xi^2 + \eta^2$ and ξ and η are scaled and shifted spatial coordinates given by $\xi = s_x(x + \delta_x)$ and $\eta = s_y(y + \delta_y)$. For our example problem, we took, $\delta_x = 0.46$, $\delta_y = -0.5$, $s_x = 1.0$, $s_y = 2\pi(\delta_x)^2$, and $\mu = 1$, so that $u = 1$ at $(x, y) = (0, 0.5)$. The velocity field and some corresponding streamlines are shown in Fig. 11. The coordinate $(x, y) = (0, 0.5)$ is
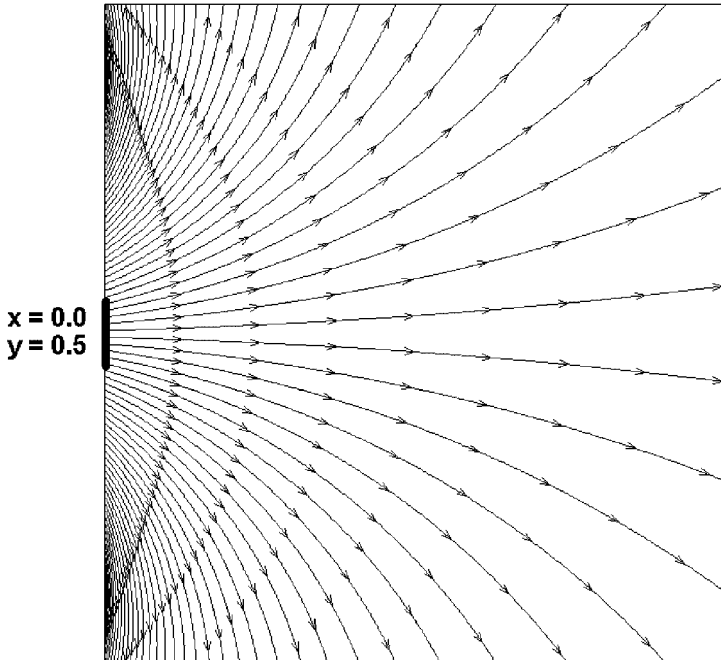

FIG. 11. Velocity field and stream lines for jet flow advection problem.

noted as well. The analytical solution for the location of a particle after time $(t - t_0)$ has been found to be in cylindrical coordinates

$$t - t_0 = \left(\frac{r_0}{\sin \theta_0}\right)^3 \frac{\pi}{\mu} \left\{ \theta - \theta_0 + \frac{1}{2}(\sin 2\theta_0 - \sin 2\theta) \right\}$$

for the θ component and

$$r = r_0 \frac{\sin \theta}{\sin \theta_0}$$

for the radial component, where r_0 and θ_0 are the radial and angular position of the particle at time t_0. The relations

$$\cos \theta = \frac{\xi^2}{\sqrt{\xi^2 + \eta^2}}$$

and

$$r = \frac{\xi}{\cos \theta},$$

where ξ and η are defined as above, are used to transform between the Cartesian and cylindrical locations of the particle.

In the example calculations, the density field was constant, and initially, the temperature field was zero. At time zero, the temperature of the inflowing material between $(x, y) = (0, 0.45)$ and $(x, y) = (0, 0.55)$ was set to one for a period of 0.5 time units, after which the inflow temperature was set back to zero. Thus, a finite pulse of temperature was introduced into the flow. The results of the computations are shown in Fig. 12 for the quadrilateral element mesh and Fig. 13 for the triangle element mesh. The analytical solution is shown by the solid lines, where the value inside the enclosed curve is one and zero elsewhere.

In the each of the figures, the top two graphics are from a simulation in which the time step was such that the Courant number was below one for all nodes. The bottom two graphics, on the other hand, were from computations in which the Courant number for the node at $(x, y) = (0.0, 0.5)$ was 7.5 and decreased across the mesh to 0.075. The two graphics on the left side of each figure show the temperature at a time of 0.5 after the initial injection. The two graphics on the right side of each figure show the temperature at a time of 5 after the initial injection. From the figures, we can see that the high Courant number results, which were computed using the explicit–implicit extension of our scheme, discussed in Section 3.4.4, are qualitatively similar to the fully explicit, low-Courant-number results. The most obvious discrepancy is in the early results in the left graphics, where numerical spreading is apparent. Although there are still differences at later time, the results are quite similar and there is no evidence of instabilities in our calculations. This similarity and stability suggests that our explicit–implicit scheme would be a useful technique to use in problems in which localized high-velocity regions exist and computer resource constraints prohibit a fully explicit computation. While there are inaccuracies associated with the explicit–implicit technique, qualitatively realistic results can be obtained.
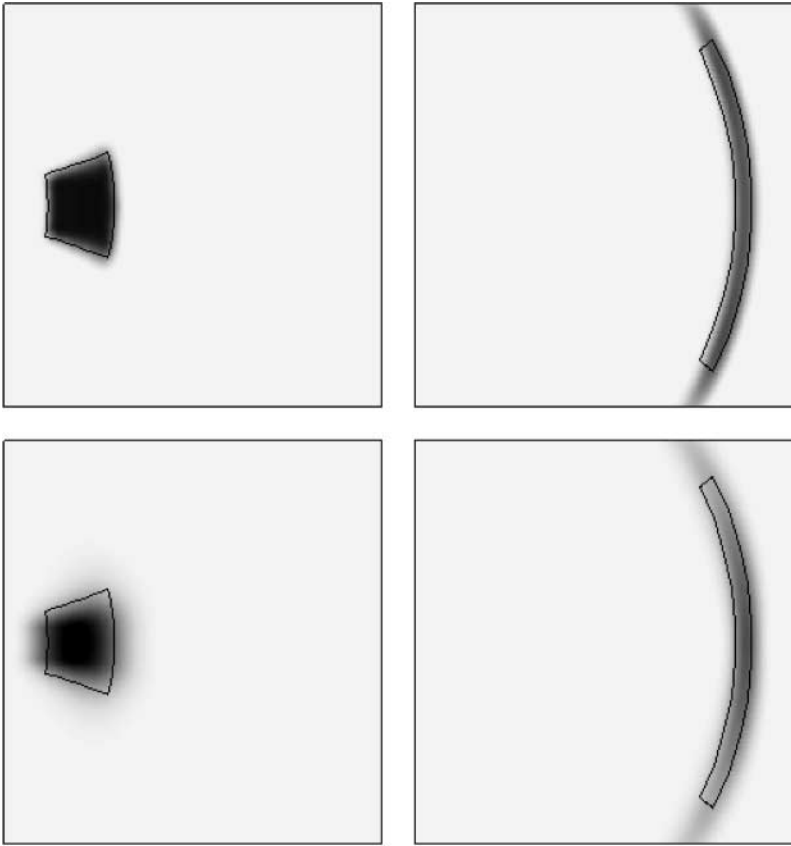
FIG. 12. Temperature contours for jet flow advection problem on quadrilateral element mesh. The solid lines are from the analytical solution.

4.5. *Discussion*

The motivation for the work presented in this paper was to develop an accurate general-purpose advection scheme for unstructured, as well as structured, meshes for both continuous and discontinuous fields. For the case of continuous-field advection, fluxed quantities are computed from an upstream, time-centered evaluation of the reconstructed field based on node data and node-centered gradients. The use of our extended version of Thuburn's limiter ensures monotonic and compatible future values of mass and mass-specific quantities. This scheme has been determined to have second-order convergence on quadrilateral and triangular element meshes. For discontinuous fields, two variations of the Ubbink–Issa [10] method have been developed. The first is a fully explicit scheme that combines our continuous-field face quantities with those obtained by the compressive difference scheme. These are then limited with our extended Thuburn limiter. In the second variation, we solve for the face flux quantities using Ubbink and Issa's Crank–Nicolson formulation but follow this with an *a posteriori* application of our extended version of Thuburn's limiter instead of their iterative correction scheme to enforce monotonicity. This ensures only one linear solution per time step. Both methods have been shown to do quite well on both quadrilateral and triangular element meshes. Of particular note was the ability of our explicit scheme to out-perform all other methods except for Young's method on the vortical shear flow
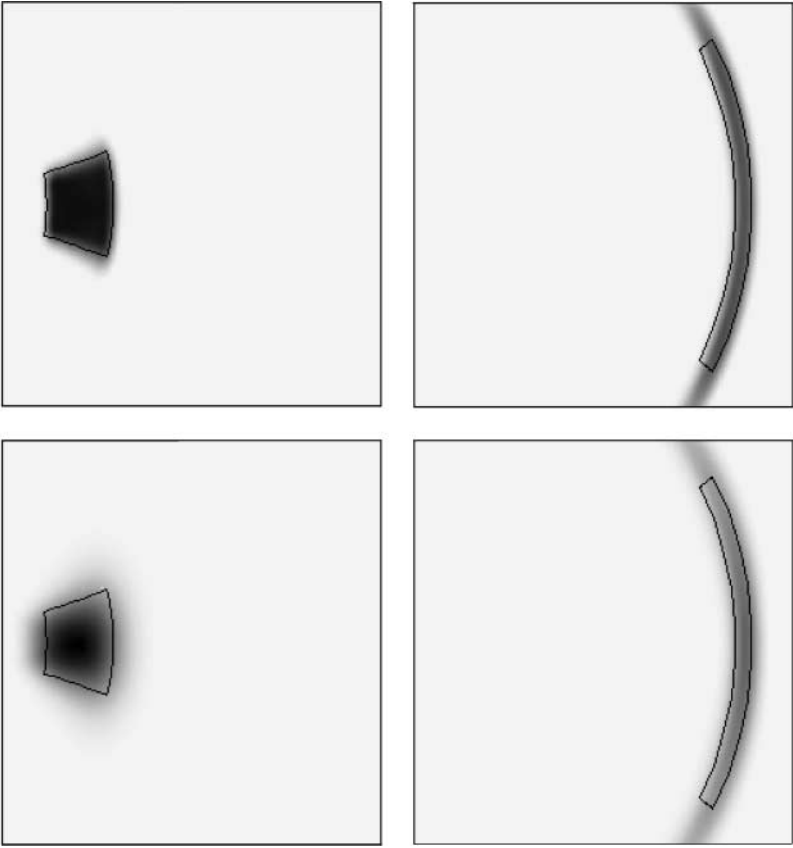
FIG. 13. Temperature contours for jet flow advection problem on triangular element mesh. The solid lines are from the analytical solution.

problem on quadrilateral element meshes. This is extremely promising, particularly for a method that does not use directionally split advection nor interface reconstruction.

5. FUTURE WORK—LAGRANGIAN–EULERIAN FORMULATION

We conclude our discussion here with a look toward the use of our advection scheme in a general-purpose flow simulation code. For such a role, we must be able to use the scheme in equations more complicated than Eqs. (1) and (2), specifically for the case in which the right-hand sides are not zero. For these cases, our plan is to use a Lagrangian–Eulerian integration approach to solving the equations. Let us briefly illustrate what we mean on the generic equation

$$\frac{\partial \phi}{\partial t} + \nabla \cdot \phi \vec{u} = f(\phi), \tag{26}$$

where $f(\phi)$ is any Lagrangian physics, such as diffusion or pressure gradient acceleration, for example. We can apply the Lagrangian–Eulerian scheme in two steps by discretizing in time as follows. We first perform the Lagrangian step where we include the implicit flux

part of the advection operator as A_{impl},

$$\phi^L = \phi^m + \Delta t\, A_{impl}(\phi^L) + \Delta t f(\phi^L). \tag{27}$$

We then follow this with the Eulerian advection step

$$\phi^{m+1} = \phi^L + \Delta t\, A_{expl}(\phi^L), \tag{28}$$

where $A_{impl}(\phi^L)$ is the advection operator using only implicit flux values as defined by Eq. (21) and $A_{expl}(\phi^L)$ is the advection operator with fluxed quantities computed using Eq. (22) and using the Lagrangian data as the known nodal data. This formulation has the desirable property that it interpolates from a fully explicit Langrangian–Eulerian scheme to a fully implicit scheme. Thus the formulation will handle high-accuracy time-dependent problems and steady-state problems. Moreover, the algorithm will handle problems with both high- and low-Courant-number nodes simultaneously.

ACKNOWLEDGMENT

REFERENCES

1. P. N. Brown and Y. Saad, Hybrid Krylov methods for non-linear systems of equations, *SIAM J Sci. Stat. Comput.* **11**, 450 (1990).

2. H. Gaskell and A. K. C. Lau, Curvature-compensated convective transport: SMART, a new boundedness-preserving transport algorithm., *Int. J. Numer. Methods Fluids* **8**, 617 (1988).

3. C. W. Hirt and B. D. Nichols, Volume of fluid (VOF) method for the dynamics of free boundaries, *J. Comput. Phys.* **39**, 201 (1981).

4. B. P. Leonard, Stable and accurate convective modeling procedure based on quadratic upstream interpolation, *Comput. Methods Appl. Mech. Eng.* **19**, 59 (1979).

5. R. J. Leveque, High-resolution conservative algorithms for advection in incompressible flow, *SIAM J. Numer. Anal.* **33**, 627 (1996).

6. W. F. Noh and P. Woodward, *SLIC (Simple Line Interface Calculation)*, Lecture Notes in Physics (Springer-Verlag, New York, 1976), Vol. 59.

7. M. Rudman, Volume-tracking methods for interfacial flow calculations, *Int. J. Numer. Methods Fluids* **24**, 671 (1997).

8. J. Thuburn, Multidimensional flux-limited advection schemes, *J. Comput. Phys.* **123**, 74 (1996).

9. O. Ubbink, *Numerical Prediction of Two Fluid Systems with Sharp Interfaces*, Ph.D. thesis (University of London, 1997).

10. O. Ubbink and R. I. Issa, A method for capturing sharp interfaces on arbitrary meshes, *J. Comput. Phys.* **153**, 26 (1999).

11. D. L. Youngs, Time-dependent multi-material flow with large fluid distortion, in *Numerical Methods for Fluid Dynamics*, edited by K. W. Morton and M. J. Baines (Academic, New York, 1982), p. 273.