Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER

Journal of Computational Physics 192 (2003) 355–364

JOURNAL OF COMPUTATIONAL PHYSICS

www.elsevier.com/locate/jcp

Short Note

A geometrical area-preserving Volume-of-Fluid advection method

Eugenio Aulisa [a], Sandro Manservisi [a], Ruben Scardovelli [a,*], Stephane Zaleski [b,1]

[a] *INFM-BO and DIENCA, Laboratory of Montecuccolino, University of Bologna, Via dei Colli 16, 40136 Bologna, Italy*
[b] *LMM, CNRS UMR 7607, UPMC, 8 rue du Capitaine Scott, 75015 Paris, France*

Received 25 February 2003; received in revised form 10 July 2003; accepted 15 July 2003

Abstract

A new class of algorithms that preserve mass exactly for incompressible flows on a Cartesian mesh are presented. They amount to piecewise-linear, area-preserving mappings of tessellations of the plane. They are equivalent to Volume-of-Fluid (VOF) advection methods which are decomposed into an Eulerian implicit scheme in one direction followed by a Lagrangian explicit step in the other one. It is demonstrated that mass conservation is exact for incompressible flows and that there are no undershoots or overshoots of the volume fraction which thus always remains constrained between 0 and 1.

AMS: 65M05; 76M20; 76T99

Keywords: Volume-of-Fluid method; Advection algorithms; Volume tracking; Area-preserving mapping; Interface tracking

1. Introduction

Transporting materials by an incompressible flow results in the conservation of volume and mass. Perhaps surprisingly, when dealing with two-phase and free-surface flows no numerical method exists that ensures this conservation to machine accuracy. The purpose of this paper is to introduce such a method, which is related to the Volume-of-Fluid method (VOF) but is in fact more general. We describe our method in two dimensions (2D), generalizations to the 3D space are briefly discussed. Consider a tessellation of a plane in elementary squares with area A (the generalization to rectangles is straightforward). The area of fluid in each cell at time step n is noted $C_{ij}^n A$, where C_{ij}^n is the fraction of the cell (i,j) occupied by the reference phase. A natural definition of mass conservation is a method which conserves the total area at each time step so that

[*] Corresponding author. Tel.: +39-051-644-1720; fax: +39-051-644-1747.
E-mail addresses: raus@mail.ing.unibo.it (R. Scardovelli), zaleski@lmm.jussieu.fr (S. Zaleski).
[1] Tel.: +33-1-44272558; fax: +33-1-44275259.

doi:10.1016/j.jcp.2003.07.003

$$\sum_{ij} C_{ij}^{n+1} = \sum_{ij} C_{ij}^{n}. \tag{1}$$

The Volume-of-Fluid (VOF) method is a convenient and popular technique to model interfaces in two-phase and free-surface flows, and seems naturally suited to realize condition (1). Cells with just one fluid will have a value of C equal to 0 or 1, while mixed cells will have an intermediate value of C. At any time step in the simulation the interface is not known: its shape has to be inferred from the knowledge of the scalar field C and then properly advected in the velocity field $\mathbf{v}$. This delineates a two-step procedure which is mainly geometric in nature [1]. If we consider a 2D Cartesian geometry, the PLIC (for Piecewise Linear Interface Calculation) technique reconstructs the interface in each computational cell with a segment perpendicular to the gradient of the scalar function C. We stress the fact that the reconstruction is not a unique process, since it relies on the algorithm adopted to calculate the gradient of C [1–5]. The actual position in the cell of this inclined segment is then uniquely determined from area conservation [6], but the reconstruction is in general not continuous across the cell boundary. The advection of the interface requires the calculation of cell boundary fluxes. This can be done independently along each coordinate direction, with multidimensionality obtained via an operator-split advection technique [1,2,4,5,7–9] or with multi-dimensional schemes [1,10–12]. The use of volume of fluid data and fluxes should lead directly to exact mass conservation but it is in fact not so. In the published methods, the advection operations results in various inconsistencies. In several cases, the new areas $C_{ij}^{n}A$ are simply not obtained from the old ones by adding fluxes because of difficulties with the splitting of the incompressibility condition [5]. Moreover, interface advection algorithms may produce some systematic errors, such as volume fractions that do not satisfy

$$0 \leqslant C_{ij}^{n} \leqslant 1 \tag{2}$$

and thus are inconsistent with their definition (an example is given below in this note), or that are either $C_{ij}^{n} = \epsilon$ or $C_{ij}^{n} = 1 - \epsilon$, where ϵ is a small positive number, instead of being 0 or 1. This occurs randomly in the lattice and inside the bulk of each phase and may be seen on several published figures [5,13]. These errors introduce a non-local inconsistency by generating little holes in the bulk of the reference phase and some level of debris outside it. These integration inaccuracies have been called "wisp" [11] and should not be confused with the so-called "flotsam" or "jetsam", which are bigger in nature and mainly due to low-order reconstruction techniques such as SLIC (for Simple Line Interface Calculation) and algebraic methods [9,11,14]. The above-mentioned inconsistencies are difficult to correct: it is not obvious where the excess or missing mass should be disposed of, or retrieved. Code writers then routinely redistribute it in the surrounding cells with some diffusion algorithm or reset the volume fraction to 1 or 0 thus destroying mass conservation. See [1,11,14] for examples on how to remove overshoots and undershoots of the volume fraction C or to limit its values with a simple filtering $C = \mathrm{MIN}(1, \mathrm{MAX}(0, C))$.

From the more general point of view of the choice between various methods of interface tracking such as VOF, markers or level sets, mass conservation, if desirable, should be achieved by at least some sort of accounting of the volume in each cell. A few hybrid methods have appeared recently, some combining for instance level sets and particles [15] or markers and a level contour reconstruction method [16]. Although marker approaches may be very accurate, a natural way to obtain volume conservation is through some kind of VOF algorithms. This has led to several mixed approaches that combine a couple of these methods, such as VOF and level set [17], VOF and interface markers [18,19]. However, none of the proposed methods in their most recent version conserves mass exactly. In this respect, the status of VOF advection methods may be found in a previous paper where we describe a combined "Eulerian–Lagrangian Advection" [5] (hereafter we cite this paper as "ELA"). In ELA, we tabulated mass conservation for various VOF schemes and found that there is only one method that satisfies both mass conservation (1) and the consistency relation (2) without the need of any local redistribution algorithm. This technique was called the Eulerian Implicit–Lagrangian Explicit (EI–LE) method, however there was no proof that the method satisfied both

(1) and (2) in any given incompressible flow. In this note, we supply that proof, present the unsplit version of this technique, show that is comparable to other VOF unsplit advection algorithms in terms of performance, but much simpler to implement and strictly mass conserving.

2. The area-preserving linear-mapping method

Let Ω be a bounded domain with the reference phase 1 contained in the region $\Omega_1 \subset \Omega$ and $\mathbf{v} = (u, v)$ an arbitrary divergence-free flow field, i.e., $\nabla \cdot \mathbf{v} = 0$. We consider a staggered MAC mesh with square cells of size h, with the velocity horizontal and vertical components defined in the center of the cell sides, as shown in Fig. 1. To simplify even further the notation, we take a dimensionless velocity which is actually the CFL number, so that when we write u we mean $u\, \mathrm{d}t/h$. With this notation, the square cell has sides of unit length, the reference phase area and the volume fraction C have the same value and the thickness of the rectangular area crossing the right boundary in the time dt is just u_2. We have also a simplified expression for the discrete version of $\nabla \cdot \mathbf{v} = 0$

$$(u_2 - u_1) + (v_2 - v_1) \equiv D_x u + D_y v = 0. \tag{3}$$

The geometric nature of the VOF method has been clearly pointed out in [1], relatively to the reconstruction process and the calculation of boundary fluxes. In this note, we extend the geometrical interpretation to the whole advection as a linear mapping. For each cell, we define the two linear mappings Π_{xy} and Π_{yx}

$$\Pi_{xy} = \begin{cases} x'' = a(x + u_1), \\ y'' = by + v_1, \end{cases} \qquad \Pi_{yx} = \begin{cases} x'' = dx + u_1, \\ y'' = c(y + v_1), \end{cases} \tag{4}$$

where, with the same notation of Eq. (3), the four constants are $a = 1/(1 - D_x u)$, $b = (1 + D_y v) = 1/a$, $c = 1/(1 - D_y v)$ and $d = (1 + D_x u) = 1/c$. The Jacobian of the two transformations, by using (3), is equal to one: $J_{xy} = ab = J_{yx} = cd = 1$, thus they both are an area-preserving mapping. For example, with reference to Fig. 1, where the flow is compressed in the x-direction and expands along the y-coordinate, the first mapping Π_{xy} is understood by defining the two rectangles $A'B'C'D'$ and $A''B''C''D''$ related to the grid square cell $ABCD$. The first rectangle $A'B'C'D'$ is the pre-image of the grid cell $ABCD$ and the vertical sides $A'D'$ and $B'C'$ are the pre-images of the vertical sides AD and BC of the unit square. The pre-image is mapped onto the unit cell by the horizontal component of the mapping and represents the implicit part of the algorithm for the mapping Π_{xy}. Similarly, the vertical component of the flow maps the square cell $ABCD$ onto the rectangular image $A''B''C''D''$ and represents the explicit part of the method. In practice, we are

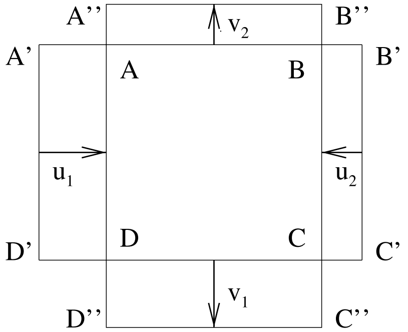

Fig. 1. The linear mapping Π_{xy} defined in the text maps the rectangle $A'B'C'D'$ onto $A''B''C''D''$. The vertical and horizontal sides are advected by the horizontal and vertical components u_i, v_i of the flow.

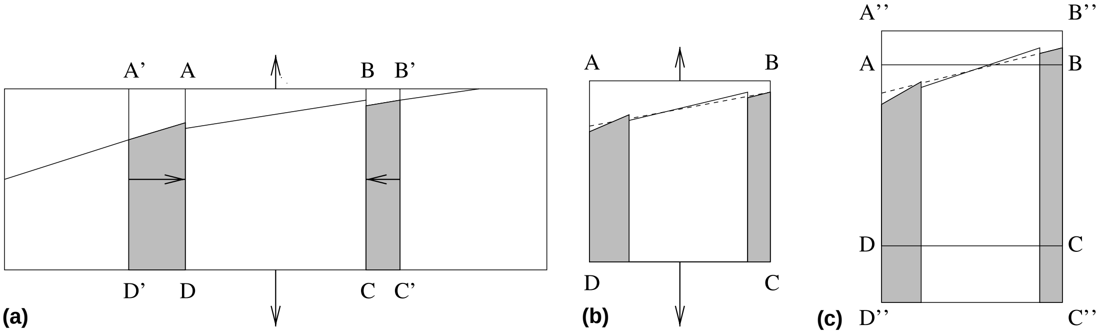

Fig. 2. A geometrical view of the whole advection procedure for the mapping Π_{xy}: (a) Eulerian fluxes along the x-coordinate; (b) mapping onto the unit square (implicit part of the method). In the split version of the algorithm, the intermediate reconstruction is shown as a dashed line; (c) Lagrangian advection in the y-direction (explicit component of the method).

transforming rectangles to rectangles, with two independent 1D mappings, as x'' is a function of x only and so y'' of y, which conserve the area when combined together.

We now restrict the application of this method to two-phase and free-surface flows and assume that the interface has been already reconstructed as a linear function in each computational cell with a PLIC algorithm based on the knowledge of the scalar function C at time step n. With reference to Figs. 1 and 2(a) and the linear mapping Π_{xy}, the reference phase inside the rectangle $A'B'C'D'$ comes from three different horizontal cells, in particular the contribution from the left cell is the reference phase inside the rectangle $A'ADD'$ of Fig. 1, shown as a gray area in Fig. 2(a). The whole procedure amounts to considering the piecewise-linear reconstructed interface in $A'B'C'D'$, computing its piecewise linear image by applying the linear mapping Π_{xy}, and calculating the area occupied by the reference phase in $A''B''C''D''$ which belongs to the central cell $ABCD$ and to the upper and lower cells. These areas can be computed in a very simple and efficient way with the routines described in [6]. The calculation is extended to the entire computational domain and the volume fraction C field is updated at the next time step $n+1$.

Mass is conserved exactly because of the area-preserving property of the linear mapping. The condition (2) is also preserved simply because the new fractions are calculated by adding areas whose sum by construction is always smaller than or equal to the area of a unit square. Moreover, arbitrarily shaped reconstructions (including spline approximations or parabolic arcs) could be mapped in this way if one can compute the images of the approximants. Short of an exact numerical mapping, approximate mappings would also conserve mass provided that the mass advected to $A''B''C''D''$ is constrained to be equal to the mass in $A'B'C'D'$. We remark that no holes or overlapping regions are produced because we start with an exact tessellation of the plane with rectangles $A'B'C'D'$ and end up with another tessellation with rectangles $A''B''C''D''$. This method is "unsplit" in the terminology of VOF methods since the interface advection is performed in a single step.

3. Connection with previous methods

The linear mapping method can be related to the split EI–LE scheme described in ELA. We notice that each mapping Π can be divided in the sum of two 1D mappings. In the case of Π_{xy} we have the sequence

$$\Pi_x = \begin{cases} y' = y, \\ x' = a(x+u_1), \end{cases} \qquad \text{and} \qquad \Pi_y = \begin{cases} y'' = by' + v_1, \\ x'' = x'. \end{cases} \tag{5}$$

Transforming $A'B'C'D'$ into the unit square cell $ABCD$ by Π_x is equivalent to the EI step, while transforming $ABCD$ into $A''B''C''D''$ is equivalent to the LE step. This can be seen by translating the algebraic expressions of the two methods into the estimate of the areas in the rectangles. To this aim, we consider a 1D advection equation along the x-axis for the volume fraction C

$$\frac{\partial C}{\partial t} + u\frac{\partial C}{\partial x} = 0 \rightarrow \frac{\partial C}{\partial t} + \frac{\partial (uC)}{\partial x} = C\frac{\partial u}{\partial x}, \tag{6}$$

where the term on the r.h.s. of the second equation represents a 1D compression/expansion [1], and discretize this equation with a simple forward scheme in time, with u once again equal to the CFL number

$$C_{ij}^{n+1} = C_{ij}^{n} + F_{\text{left}} - F_{\text{right}} + \tilde{C}_{ij} D_x u, \tag{7}$$

where F_{left} and F_{right} denote the C fluxes across the left and right sides of the (i,j)th cell. If we consider the value C_{ij}^{n+1} for $\tilde{C}_{ij}$, we recover the implicit (EI) scheme

$$C_{ij}^{n+1} = \frac{C_{ij} + F_{\text{left}} - F_{\text{right}}}{1 - D_x u}. \tag{8}$$

The same result is obtained by applying the mapping Π_x, as shown in Figs. 2(a) and (b). On the other hand, if we substitute C_{ij}^{n} for $\tilde{C}_{ij}$ we get the explicit (LE) scheme depicted in Fig. 3

$$C_{ij}^{n+1} = C_{ij}^{n}(1 + D_x u) + F_{\text{left}} - F_{\text{right}}. \tag{9}$$

Both the LE and EI schemes obey the consistency condition (2).

The 2D split advection scheme by Li [2,20] consists of two LE steps, one along each coordinate direction. It does not conserve mass, but obeys the consistency condition. Rider and Kothe [1] developed a 2D split scheme based on an algebraic discretization of the advection equation as in (7), where an explicit step is followed by an implicit advection in the other direction. Puckett et al. [10] proposed the same scheme but with the explicit and implicit components in reversed order. A geometrical interpretation of the difference between the LE step and the explicit advection of [1,10] is shown in Fig. 3. Both schemes stretch the interface in the same way, but fluxes are calculated differently [5]: in the LE method the interface is first advected and then the fluxes are calculated, the opposite is true for the other method. With a simple example, we show that this can result in a volume fraction that does not satisfy the consistency condition (2). Consider a situation similar to Fig. 3, but with $u_1 = 2u_2 = 2u > 0$, the left cell with $C = 1$ and the central cell with the end points of the interface segment located at $(1 - u, 1)$ and $(1, 0.5)$, then $C_{ij}^{n} = 1 - u/4$. In both cases the volume fraction after time integration is given by (9) but with the Eulerian fluxes $F_{\text{left}} = 2u$, $F_{\text{right}} = 3u/4$ we find $C_{ij}^{n+1} = 1 + u^2/4 > 1$, while with the Lagrangian fluxes $C_{ij}^{n+1} = 1$ by construction. Similarly, it is possible to find a case in which $C_{ij}^{n+1} < 0$. The EI step is identical to the implicit part of the algorithm described in [1,10]. To our knowledge, there has been no published attempt to use an EI–EI method which would also preserve consistency but not mass. Finally, we notice that if the linear mapping

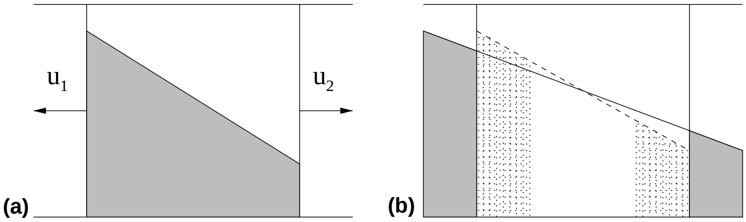

Fig. 3. Boundary fluxes along the x-direction for the 1D advection: (a) horizontal velocity and initial reconstruction; (b) Eulerian (dotted areas) and Lagrangian fluxes (grey areas).

scheme is run in its split version, after the first mapping Π_x another interface reconstruction is needed in the whole domain, shown as a dashed line in Figs. 2(b) and (c), before applying the second mapping Π_y. As a result, the flux calculation with the split algorithm is somewhat simpler, but a second reconstruction per time step is required.

The extension to the 3D space of this method is not direct. In [21], we suggest the following procedure to circumvent this difficulty. We consider the three 2D velocity fields $\mathbf{v}^1 = (u^1, v^1, 0)$, $\mathbf{v}^2 = (0, v^2, w^2)$ and $\mathbf{v}^3 = (u^3, 0, w^3)$ and require each 2D field to be incompressible, $\nabla \cdot \mathbf{v}^i = 0$, and their sum to be the 3D field $\mathbf{v}$, $\sum_i \mathbf{v}^i = \mathbf{v}$. Therefore, we have six linear scalar equations for each grid cell and if N_{tot} is the total number of computational cells then the inverse of a $(6N_{\text{tot}}) \times (6N_{\text{tot}})$ sparse matrix is needed. For fixed grids the inverse matrix is not sparse, but it is the same at each time step and it can be economically stored for small-medium size problems. Otherwise, it has to be inverted at each time step, requiring a consistent computational effort. This may somewhat slow down the algorithm, but the cost of this inversion is smaller than that required to invert for the pressure in a multiphase flow application.

4. Advection accuracy tests

In this section, we consider two advection problems which are widely used to test the accuracy of reconstruction and advection algorithms: the Zalesak slotted disk rotation [22] and the reversed single vortex test [1,23]. We consider only a limited number of tests and show that the linear mapping method is comparable to other advection algorithms in terms of performance, while its major strengths are the simplicity and the mass conservation property. Three reconstruction techniques are mentioned in the tests. The ELVIRA and Puckett's algorithms [4] are both PLIC methods that approximate linearly the interface by minimizing a discrete error function between the actual volume fraction values in a 3×3 block of cells around the cell (i,j) and those obtained with a linear reconstruction. The first method considers only six possible values of the angular coefficient of the straight line, the second one minimizes the same error function with an iterative search of the optimal angular coefficient. They both reconstruct a linear interface exactly, but the second approach is slightly more performing when the local interface curvature is high [11]. The third technique (SZ) approximates the interface in each cell with two consecutive segments, it relies on an initial PLIC reconstruction and is based on a least square fit [5]. We compare our advection method with three other unsplit schemes: the defined donating region (DDR) [12] and Stream [11] by Harvey and Fletcher and one (RK) by Rider and Kothe [1]. The donating region of each of these schemes is conceptually depicted in Fig. 4. The DDR scheme deforms the rectangular fluxing area through the right boundary to a trapezoid, to avoid an overlap with the flux through the top side, but it conserves the area. The RK scheme removes and adds triangles near the cell corners along the characteristic lines, in particular the dashed height of the top triangle removed in Fig. 4(c) is equal to v_2. Finally, the Stream scheme divides the cell side in n_{st} equal segments that represent the exit of elementary flux tubes which are reconstructed backwards along the characteristic lines. The sum of these areas defines the donating region.

4.1. Zalesak slotted disk rotation

In this test, the unit square is divided in 100^2 cells and a circle with radius equal to 15 cells is centered at $(0.5, 0.75)$. A vertical rectangular cut is produced with a width equal to one-third of the radius and the upper bridge has the same maximum thickness. A full rotation with constant angular velocity is performed in 628 steps. This advection study tests mainly the accuracy of the reconstruction algorithm near regions of high curvature, since both terms in Eq. (3) are zero, thus there is no 1D compression/expansion. As a matter of fact, we notice in Fig. 5 that the two lines corresponding to the ELVIRA reconstruction algorithm, combined with the split and unsplit versions of the linear mapping method, cannot be distinguished at this

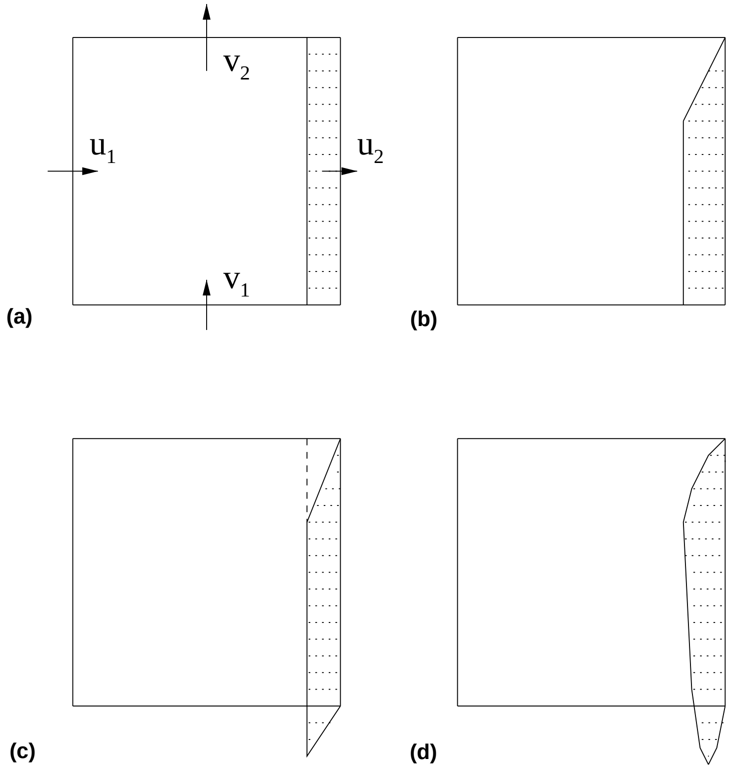

Fig. 4. Different donating regions through the right boundary: (a) 1D split, (b) DDR, (c) KR, (d) Stream schemes.

resolution. On the other hand, with the SZ reconstruction algorithm and the split EI–LE advection we obtain definitively a more precise and symmetric result. For comparison with published data we consider the Rudman–Zalesak problem [9] where the square has side 4 and is divided in 200^2 cells. The circle center is at $(2, 2.75)$, the widths of the slot and of the upper bridge are 6 and 20 cells, respectively. A full rotation is performed in 2524 steps and the non-dimensional error E is defined as

$$E = \frac{\sum_{ij} |C_{ij} - \tilde{C}_{ij}|}{\sum_{ij} C_{ij}},$$

where C is the exact volume fraction field and $\tilde{C}$ is the numerical value obtained by a different combination of reconstruction and advection algorithms. Results for different methods are presented in Table 1, where the n_{st} variable in the Stream scheme is set to 1000. The DDR scheme has the worst performance and does not describe with sufficient accuracy the transverse fluxes near the cell corners. The SZ reconstruction algorithm produces the best result because of the better approximation of the interface where the curvature is high.

4.2. Reversed single vortex test

A more precise assessment of the reconstruction and advection algorithms is made with a flow containing a non-uniform vorticity field [1,5,9,11]. A circle of radius 0.15 is centered at $(0.5, 0.75)$ in a unit square domain. All boundaries are periodic and the velocity field $\mathbf{v}$ is specified by the stream function

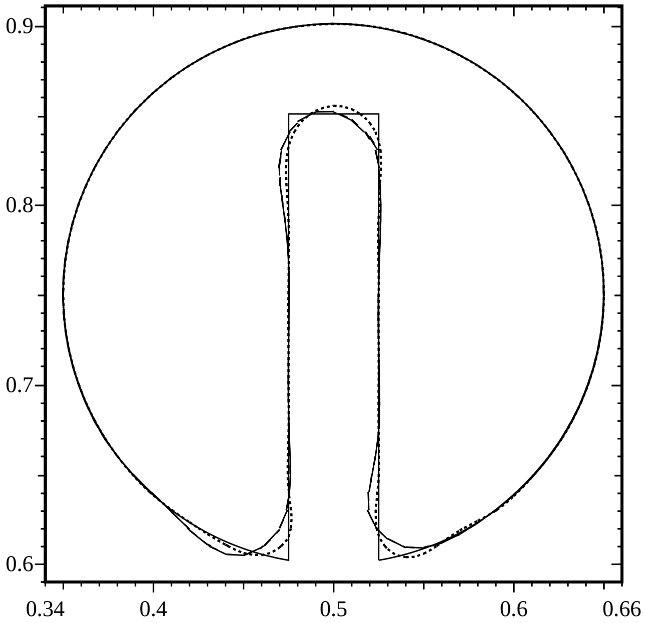

Fig. 5. Zalesak slotted disk test. The slotted disk (continuous solid line) is shown together with three reconstructed interfaces after one full rotation: ELVIRA/EI–LE (solid segments), ELVIRA/Π (dashed segments) and SZ/EI–LE (dotted segments). The first two reconstructed interfaces cannot be distinguished at this resolution.

Table 1
Errors for the Rudman–Zalesak slotted disk test

Reconstruction/Advection algorithms	Error
Puckett/DDR	1.50e – 2
Puckett/Stream	1.00e – 2
ELVIRA/EI–LE	1.00e – 2
SZ/EI–LE	4.16e – 3
ELVIRA/Π	1.00e – 2

The first two results are from [11,12], the second two from [5].

$$\Psi(x,y,t) = \frac{1}{\pi}\sin^2(\pi x)\sin^2(\pi y)\cos\left(\frac{\pi t}{T}\right). \tag{10}$$

The fluid stretches and spirals about the center of the domain, reaching a maximum deformation at time $t = T/2$, while at $t = T$ is back to the initial configuration [24]. The maximum CFL is equal to 1. We consider a period $T = 2$ and for consistency with previous work we now calculate the error as

$$E = \sum_{i,j} h^2 |C_{i,j} - \tilde{C}_{i,j}|,$$

which has the dimension of an area. We alternate in time the coordinate direction of the implicit part of the algorithm, both in the split and unsplit advections, which are coupled to the ELVIRA reconstruction technique. The results are given in Table 2. For completeness, we mention that when we run this test on the 32^2 grid with the ELVIRA reconstruction and with two consecutive LE or EI steps we observe an increasing in the reference phase area of about 6.4% with the explicit scheme and a decreasing of about 7.4%

Table 2
Errors and convergence rates for the single vortex test

Grid	Error/(Order)				
	(a)	(b)	(c)	(d)	(e)
32^2	2.37e − 3/(2.07)	2.36e − 3/(2.01)	2.52e − 3/(1.95)	1.09e − 3/(1.96)	2.46e − 3/(2.01)
64^2	5.65e − 4/(2.10)	5.85e − 4/(2.16)	6.46e − 4/(2.15)	2.80e − 4/(2.19)	6.10e − 4/(2.19)
128^2	1.32e − 4	1.31e − 4	1.45e − 4	5.72e − 5	1.34e − 4

(a) Puckett/Stream, (b) Puckett/RK, (c) ELVIRA/EI–LE, (d) SZ/EI–LE, (e) ELVIRA/Π. The data in the first column are from [11], those in the second one from [1], the following two columns are from [5].

with the implicit one. The split version of the linear mapping algorithm is for this test slightly less performing than unsplit one, because of the double number of reconstructions combined with the effects of the 1D compression or expansion which deforms appreciably the interface in the intermediate reconstruction. However, a better reconstruction algorithm, such as SZ, has a stronger impact on the performance than the change from split to unsplit advection. We believe that the difference in the results obtained by the linear mapping and the Stream (here with $n_{\mathrm{st}} = 100$) and RK unsplit schemes is mainly due to the reconstruction algorithm, since the ELVIRA method is slightly less performing than the Puckett's one when the radius of curvature is not much bigger than the grid spacing. As a matter of fact, we notice that the Π algorithm is converging at a slightly faster rate.

5. Conclusion

We have presented a new advection algorithm in 2D Cartesian geometry. At each time step a linear mapping is used to advect the reconstructed interface into neighboring cells. For a divergence-free velocity field the Jacobian of the linear mapping is equal to one, thus it is an area-preserving mapping. The mapping of the whole grid is piecewise linear, each piece transforming a rectangular element of the starting tessellation of the plane onto the final one, while preserving the area. Thus, no holes or overlapping regions are produced in the advection and the numerical values of the volume fraction C are always constrained between 0 and 1. It provides a simple multidimensional or unsplit method. The method could be applied with higher-order reconstructions of the interface without fundamental changes, but a more complicated numerical quadrature would be required to determine the mass falling in each cell after the advection.

It is also interesting to consider possible extension to other interface tracking methods. For instance, if the interface is tracked by a level set, the level set function could be updated by $\phi(x, t_{n+1}) = \phi(\Pi^{-1}(x), t_n)$, where Π is an operator transforming a tessellation of the plane in an area conserving way.

One of the drawbacks of the current method is the fact that the global mapping composed by all the individual linear mappings of the rectangles is discontinuous. Therefore, particles close to each other but in different rectangles of the tessellation follow different paths. This makes the extension to markers methods more difficult. It would be interesting to have continuous linear mappings. This could be achieved with triangular tessellations of the plane, in which the linear mappings interpolate the velocities at the vertices of the triangulations.

References

[1] W.J. Rider, D.B. Kothe, Reconstructing volume tracking, J. Comput. Phys. 141 (1998) 112–152.

[2] Jie Li, Calcul d'interface affine par morceaux (piecewise linear interface calculation), C.R. Acad. Sci. Paris, série IIb (Paris) 320 (1995) 391–396.

[3] B.J. Parker, D.L. Youngs, Two and three dimensional Eulerian simulations of fluid flow with material interfaces, Technical Report 01/92, 1992, UK AWE.

[4] J.E. Pilliod Jr., E.G. Puckett, Second-order accurate volume-of-fluid algorithms for tracking material interfaces. Technical Report, Lawrence Berkeley National Laboratory, 1997, No. LBNL-40744.

[5] R. Scardovelli, S. Zaleski, Interface reconstruction with least-square fit and split Eulerian–Lagrangian advection, Int. J. Numer. Meth. Fluids 41 (2003) 251–274.

[6] R. Scardovelli, S. Zaleski, Analytical relations connecting linear interfaces and volume fractions in rectangular grids, J. Comput. Phys. 164 (2000) 228–237.

[7] G. Strang, On the construction and comparison of difference schemes, SIAM J. Numer. Anal. 5 (1968) 506.

[8] D.L. Youngs, Time-dependent multi-material flow with large fluid distortion, in: K.W. Morton, M.J. Baines (Eds.), Numerical Methods for Fluid Dynamics, Academic Press, New York, 1982, p. 273.

[9] M. Rudman, Volume-tracking methods for interfacial flows calculations, Int. J. Numer. Meth. Fluids 24 (1997) 671–691.

[10] A.S. Almgren, E.G. Puckett, J.B. Bell, D.L. Marcus, W.J. Rider, A high-order projection method for tracking fluid interfaces in variable density incompressible flows, J. Comput. Phys. 130 (1997) 269–282.

[11] D.J.E. Harvie, D.F. Fletcher, A new volume of fluid advection algorithm: The stream scheme, J. Comput. Phys. 162 (2000) 1–32.

[12] D.J.E. Harvie, D.F. Fletcher, A new volume of fluid advection algorithm: The defined donating region scheme, Int. J. Numer. Meth. Fluids 35 (2001) 151–172.

[13] M. Renardy, Y. Renardy, J. Li, Numerical simulation of moving contact line problems using a volume-of-fluid method, J. Comput. Phys. 171 (2001) 243–263.

[14] B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, G. Zanetti, Modelling merging and fragmentation in multiphase flows with surfer, J. Comput. Phys. 113 (1994) 134–147.

[15] D. Enright, R. Fedkiw, J. Ferziger, I. Mitchell, A hybrid particle level set method for improved interface capturing, J. Comput. Phys. 183 (2002) 83.

[16] S. Shin, D. Juric, Modeling three-dimensional multiphase flow using a level contour reconstruction method for front tracking without connectivity, J. Comp. Phys. 180 (2002) 427–470.

[17] M. Sussman, E.G. Puckett, A coupled level set and volume-of-fluid method for computing 3D and axisymmetric incompressible two-phase flows, J. Comput. Phys. 162 (2000) 301–337.

[18] E. Aulisa, S. Manservisi, R. Scardovelli, A mixed markers and volume-of-fluid method for the reconstruction and advection of interfaces in two-phase and free-boundary flows, J. Comput. Phys. 188/2 (2003) 611–639.

[19] E. Aulisa, S. Manservisi, R. Scardovelli, A coupled marker and local area conservation method for three-dimensional interface tracking, J. Comput. Phys. (submitted).

[20] D. Gueyffier, A. Nadim, J. Li, R. Scardovelli, S. Zaleski, Volume of fluid interface tracking with smoothed surface stress methods for three-dimensional flows, J. Comput. Phys. 152 (1999) 423–456.

[21] E. Aulisa, Tecniche di ricostruzione e di convezione nello spazio tridimensionale dell'interfaccia di separazione tra sistemi bifase composti da fluidi immiscibili e incomprimibili. Master's thesis, University of Bologna, 2001.

[22] S.T. Zalesak, Fully multidimensional flux-corrected transport algorithms for fluids, J. Comput. Phys. 31 (1979) 335.

[23] J.B. Bell, P. Colella, H.M. Glaz, A second-order projection method for the incompressible Navier–Stokes equations, J. Comput. Phys. 85 (1989) 257.

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