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

Eugenio Aulisa[1], Sandro Manservisi[1], Ruben Scardovelli[1] and Stephane Zaleski[2]

February 4, 2003

1. INFM-BO and DIENCA-Lab. di Montecuccolino, Via dei Colli 16, 40136 Bologna, ITALY

2. LMM, CNRS UMR 7607, UPMC, 8 rue du Capitaine Scott, 75015 Paris, FRANCE

PROPOSED RUNNING HEAD: A geometrical area-preserving VOF advection method

Mailing addresses:

Ruben Scardovelli

DIENCA-Lab. di Montecuccolino, Via dei Colli 16, 40136 Bologna, ITALY

raus@mail.ing.unibo.it, phone: +39-051-6441711, fax: +39-051-6441747

Stephane Zaleski

LMM, CNRS UMR 7607, UPMC, 8 rue du Capitaine Scott, 75015 Paris, FRANCE

zaleski@lmm.jussieu.fr, phone: +33-1-44272558, fax: +33-1-44275259

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 tesselations 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 zero and one. The extension to the three-dimensional case requires the decomposition of the velocity field into three planar incompressible fields.

Keywords: Volume-of-Fluid method, advection algorithms, volume tracking, area-preserving mapping, interface tracking

Subject Classification: 65M05, 76M20, 76T99

1 Introduction

Transporting materials by an incompressible flow results in the conservation of volume and mass. Perhaps surprisingly, 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), generalisations to the three-dimensional space are discussed at the end of this paper. Consider a tesselation of a plane in elementary squares with volume V (the generalization to rectangles is straightforward). The volume of fluid in each cell at time step n is noted $C_{ij}^n V$, where C_{ij}^n is the volume fraction of the cell (i, j) occupied by the reference phase. A natural definition of mass conservation is a method which conserves the total volume at each time step so that

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

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 one or zero, 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 two-dimensional Cartesian geometry, the VOF/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, 2, 3, 4, 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 splitting technique [1, 2, 4, 5, 7, 8, 9] or with multidimensional schemes [1, 4, 10, 11]. 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 volumes C^n_{ij} are simply not obtained from the old ones by adding fluxes because of difficulties with the splitting of the incompressibility condition [5]. Moreover, in many cases the advection results in two types of error: i) the resulting volume fraction does not satisfy

$$0 \leq C^n_{ij} \leq 1 \tag{2}$$

and thus it is inconsistent with its definition. An example is given below in this note; ii) the resulting volume fractions are either $C^n_{ij} = \epsilon^2$ or $C^n_{ij} = 1 - \epsilon^2$, where ϵ is a small number, instead of being zero or one. This occurs randomly in the lattice and inside the bulk of each phase and may be seen on several published figures ([5, 12]). These errors introduce a nonlocal 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" [10] 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 VOF/SLIC (Simple Line Interface Calculation) and algebraic methods [9, 10, 13]. The inconsistencies (i) and (ii) above 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, 10, 13] for examples on how to remove overshoots and undershoots of the volume fraction C or to limit its values with a simple filtering $C = MIN(1, MAX(0, C))$.

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.

This has led to several mixed approaches that combine VOF with other methods, either VOF and level set [14] or VOF and markers [15]. 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 present the unsplit version of this technique, supply that proof and reinterpret the method as an instance of a class of geometrical methods, which can be generalized to approximations of the interface position and the velocity field using arbitrary discretization and accuracy, and thus truly independent of the choice of the formulation: marker, level set or VOF.

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, the velocity horizontal components u_2 and u_1 are defined on the center of the right and left sides, respectively, and the vertical components v_2 and v_1 on the top and bottom sides. 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 $u\,dt\,/\,h$. With this notation the cell has sides of length unity, the actual reference phase area and the volume fraction C are numerically the same 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 the divergence-free equation $\nabla \cdot \mathbf{v} = 0$ for the velocity field

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

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. Here we have extended the geometrical interpretation to the whole advection as a linear mapping.

We define the two mappings Π_{xy} and Π_{yx}

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

where 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 the discrete incompressibility constraint (3), is equal to one: $J_{xy} = a\,b = J_{yx} = c\,d = 1$, thus they both are an area-preserving mapping.

For example, the use of the first mapping Π_{xy} in the advection of the interface is understood by defining two rectangles A'B'C'D' and A"B"C"D" related to each cell. The first one A'B'C'D' is the pre-image of the square cell ABCD and the vertical sides A'D' and B'C' are the pre-images of the vertical sides AD and BC of the square cell. The horizontal flow maps the pre-image onto the square cell and represents the implicit component of the algorithm for the mapping Π_{xy}. Similarly, the vertical flow maps the square cell ABCD onto the rectangular image A"B"C"D" and represents the explicit component of the method. The advection procedure is depicted in Fig. 1.

We need a reconstruction technique to obtain the location of the interface in the first rectangle A'B'C'D'. The simplest version of the method, already implemented in ELA, uses piecewise linear (PLIC) reconstructions. The reconstructions are then mapped by one of the two Π to obtain advected reconstructions. The PLIC reconstructions have a piecewise linear image by the linear mapping Π and thus the new mass in ABCD as well as the mass "given" to the cells below and above ABCD can be computed trivially [6]. Mass is conserved exactly because of the area-preserving property of Π. 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) could be mapped in this may provided that one is able to compute the images of the approximants. Ellipses with horizontal and vertical axes fall in this case. 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'.

Finally the method is "unsplit" in the terminology of VOF methods: no separate x and y advection are performed.

We remark that no holes or overlapping regions are produced because we start with an exact tesselation of the plane with rectangles A'B'C'D' and end with another tesselation with rectangles A"B"C"D". It is not easy to find other tesselations that can be linearly mapped on one another in a manner that approximates the exact flow when one considers the staggered velocity discretization in the MAC method. In particular, we found no such tesselations in 3D that would allow to perform advection in one step. However, if other discretizations are used, such as finite-element discretizations, on can find adequate tesselations obtained by deforming the triangular or finite elements in a volume-preserving way.

3 Connection with previous methods

The method above may be connected with the split $EI - LE$ method described in ELA, with a small change. We notice that the linear mapping Π can be divided in the sum of two one-dimensional mappings, in the case of Π_{xy} we have the sequence

$$\Pi_x = \begin{cases} y' = y \\ x' = a\,(x + u_1)\,, \end{cases} \tag{5}$$

and

$$\Pi_y = \begin{cases} y'' = b\,y' + v_1 \\ x'' = x'\,. \end{cases} \tag{6}$$

Transforming A'B'C'D' into the 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 *EI* method into the plain estimation of the areas in the rectangles. For the *LE* method the proof is straightforward as the Lagrangian method is naturally defined by the geometrical transformation of the interface segments.

Both operations obviously obey the consistency condition (2), but do not conserve mass individually because of compression or expansion in each direction. The CIAM method of Li [2, 16] is the sequence of two *LE* steps, one along the horizontal coordinate and one in the vertical direction. It does not conserve mass, but obeys the consistency condition. It is similar to the explicit part of the 2D split scheme by Rider and Kothe [1] obtained with an algebraic discretization of the advection equation for the volume fraction C. A geometrical interpretation is shown in Fig. 2, where the two 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. 2, but with $u_1 = 2\,u_2 = 2\,u > 0$, 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 $C^n_{ij} = 1 - u/4$. In both cases the volume fraction after time integration is given by the expression [5]

$$C^{n+1}_{ij} = C^n_{ij}\,(1 + D_x\,u) + F_{left} - F_{right}\,, \tag{7}$$

but with the Eulerian fluxes $F_{left} = 2\,u$, $F_{right} = 3u/4$ 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] and its geometrical representation is shown in Fig. 3a,b. There has been no published attempt to use an $EI - EI$ method which would also preserve consistency and not mass. The other major difference between the two split advection schemes is that we perform first the implicit part and then the explicit one, because this sequence corresponds to an exact tesselation of the plane. The whole procedure is depicted in Fig. 3 for the mapping Π_{xy}. If the scheme is run in its split version, the flux calculation is somewhat simpler but an extra reconstruction is needed, shown as a dashed line in Fig. 3b,c.

4 Rider-Kothe Reversed Single Vortex Test

A precise assessment of reconstruction and advection algorithms is made with a flow containing a nonuniform vorticity field [1, 5, 9, 10]. A circle of radius 0.15 is centered at point $(0.5, 0.75)$ in a unit square domain. All boundaries are periodic and the velocity field $\mathbf{v}$ is specified by the stream function

$$\Psi(x, y, t) = \frac{1}{\pi} \sin^2(\pi x) \, \sin^2(\pi y) \, \cos\left(\frac{\pi t}{T}\right). \tag{8}$$

The fluid stretches and spirals about the center of the domain, reaching a maximum deformation at time $t = T/2$ and when $t = T$ is back to the initial configuration. We consider a period $T = 2$ and calculate the error as

$$E = \sum_{i,j} h^2 \, |C_{i,j}^f - C_{i,j}^i| \,, \tag{9}$$

where h is the constant grid size and $C_{i,j}^f$ and $C_{i,j}^i$ are the final and initial volume fraction values at cell (i, j). The results are given in Table I. When we use the same ELVIRA reconstruction technique developed by Pilliod and Puckett [4] in conjunction with the Π and $EI - LE$ advection algorithms, we notice a bigger error for the split $EI - LE$ than for the unsplit Π. This difference is entirely due the doubling of the number of reconstructions. In both cases we alternate in time the coordinate direction of the implicit part of the algorithm, in particular Π_{xy} with Π_{yx}. Moreover, we believe that the difference in the results obtained with our new scheme and the unsplit one by Rider and Kothe [1] is mainly due to a slightly different reconstruction technique than in the advection method. We adopt the ELVIRA technique, where an error functional is minimized by using a finite number of tentative values of the normal to the interface in a given cell based on backward, central and forward finite differences of the volume fractions of the surrounding cells. In [1] a minimum of the same functional is found with an iterative procedure by varying the value of the normal vector. The second approach is reported to give better results at low grid resolutions [10]. As a matter of fact, we see that the difference between the two methods is vanishing with grid refinement. We also point out the importance of the reconstruction of the interface by presenting in Table I lower overall errors obtained with two different combined algorithms [5]: first a linear fit reconstruction with the Π advection and then a more involved method where the interface is approximated in each cut cell with two consecutive segments, which give also an estimate of the local radius of curvature, together with the split $EI - LE$ advection.

5 Extension to three dimensions

The extension to the three-dimensional space of this method is not direct. In [17] we suggest the following procedure to circumvent this difficulty. We consider the sequence of three two-dimensional velocity fields $\mathbf{v} = (u, v, w) = \mathbf{v}^1 + \mathbf{v}^2 + \mathbf{v}^3 = (u^1, v^1, 0) + (0, v^2, w^2) + (u^3, 0, w^3)$, and we require that each 2D field be incompressible and that their sum be the original 3D field. Therefore in each grid cell (i, j, k)

the following system of 6 linear equations has to be satisfied

$$\begin{cases} u^1 + u^3 = u \\ v^1 + v^2 = v \\ w^2 + w^3 = w \\ D_x\, u^1 + D_y\, v^1 = 0 \\ D_y\, v^2 + D_z\, w^2 = 0 \\ D_x\, u^3 + D_z\, w^3 = 0\,. \end{cases} \qquad (10)$$

Let N_{tot} be the total number of computational cells, then we need to invert a $(6N_{tot})\times(6N_{tot})$ sparse matrix. If the grid remains fixed the matrix has to be inverted only once, since it is constant in time.

6 Conclusion

We have presented a new advection algorithm in two-dimensional 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 acting on an element of a tessellation of the plane. Thus no holes or overlapping regions are produced in the advection, and the numerical values of the volume fraction C are always constrained between zero and one. It provides a simple multidimensional or unsplit method.

The method could be applied with higher-order reconstruction of the interface without fundamental changes. Higher order approximation will require some numerical quadrature to determine the mass falling in each square after the advection. However one basic requirement is to map the initial tesselation on the final one by an area preserving mapping.

In the three-dimensional space the procedure requires the decomposition of the velocity field in three planar incompressible fields, which involves a one-time sparse matrix inversion. 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.

References

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

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

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

[4] J. E. Pilliod Jr and 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 and S. Zaleski. Interface reconstruction with least-square fit and split eulerian-lagrangian advection. *Int. J. Numer. Meth. Fluids*, 41:251–274, 2003.

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

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

[8] D.L. Youngs. Time-dependent multi-material flow with large fluid distortion. In K.W. Morton and M.J. Baines, editors, *Numerical Methods for Fluid Dynamics*, page 273. Academic Press, New York, 1982.

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

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

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

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

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

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

[15] E. Aulisa, S. Manservisi, and R. Scardovelli. A mixed markers and volume-of-fluid method for the reconstruction and advection of interfaces in two-phase and free-boundary flows. Submitted to J. Comput. Phys.

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

[17] 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.

grid/error	(a)	(b)	(c)	(d)	(e)	(f)
32^2	1.73e-3	2.46e-3	2.52e-3	1.09e-3	2.36e-3	2.37e-3
64^2	4.32e-4	6.10e-4	6.46e-4	2.80e-4	5.85e-4	5.65e-4
128^2	9.44e-5	1.34e-4	1.45e-4	5.72e-5	1.31e-4	1.32e-4

Table I: Errors for the single vortex test, for the following combined reconstruction and advection algorithms: (a) linear fit/Π, (b) ELVIRA/Π, (c) ELVIRA/$EI - LE$, (d) two consecutive segments/$EI - LE$, (e) Puckett/Rider and Kothe, (f) Puckett/Stream. Results (c) and (d) are taken from [5], (e) from [1] and (f) from [10], respectively.

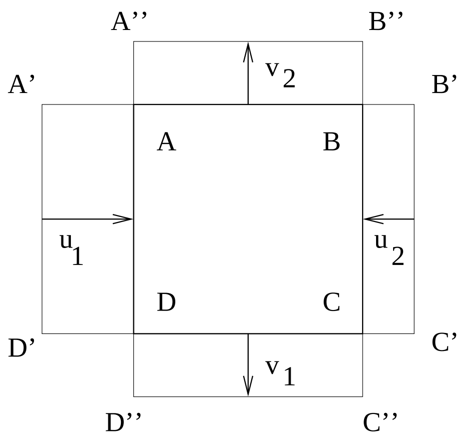

Figure 1: The linear mapping Π_{xy} defined in the text maps the rectangle A'B'C'D' into A"B"C"D". The horizontal and vertical sides are advected by the horizontal and vertical components u_i , v_i of the flow.

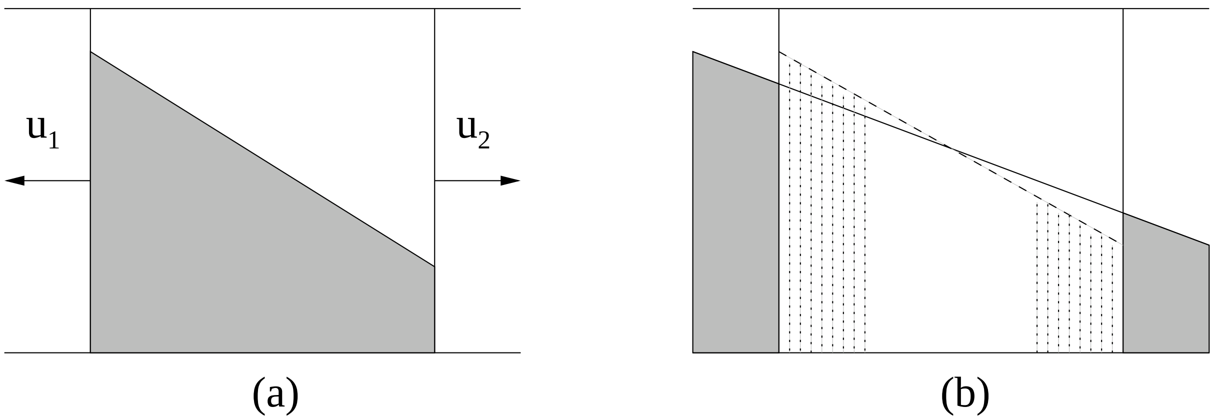

Figure 2: Boundary fluxes along the x-direction for the one-dimensional advection: a) horizontal velocity and initial reconstruction; b) Eulerian (dotted areas) and Lagrangian fluxes (grey areas)

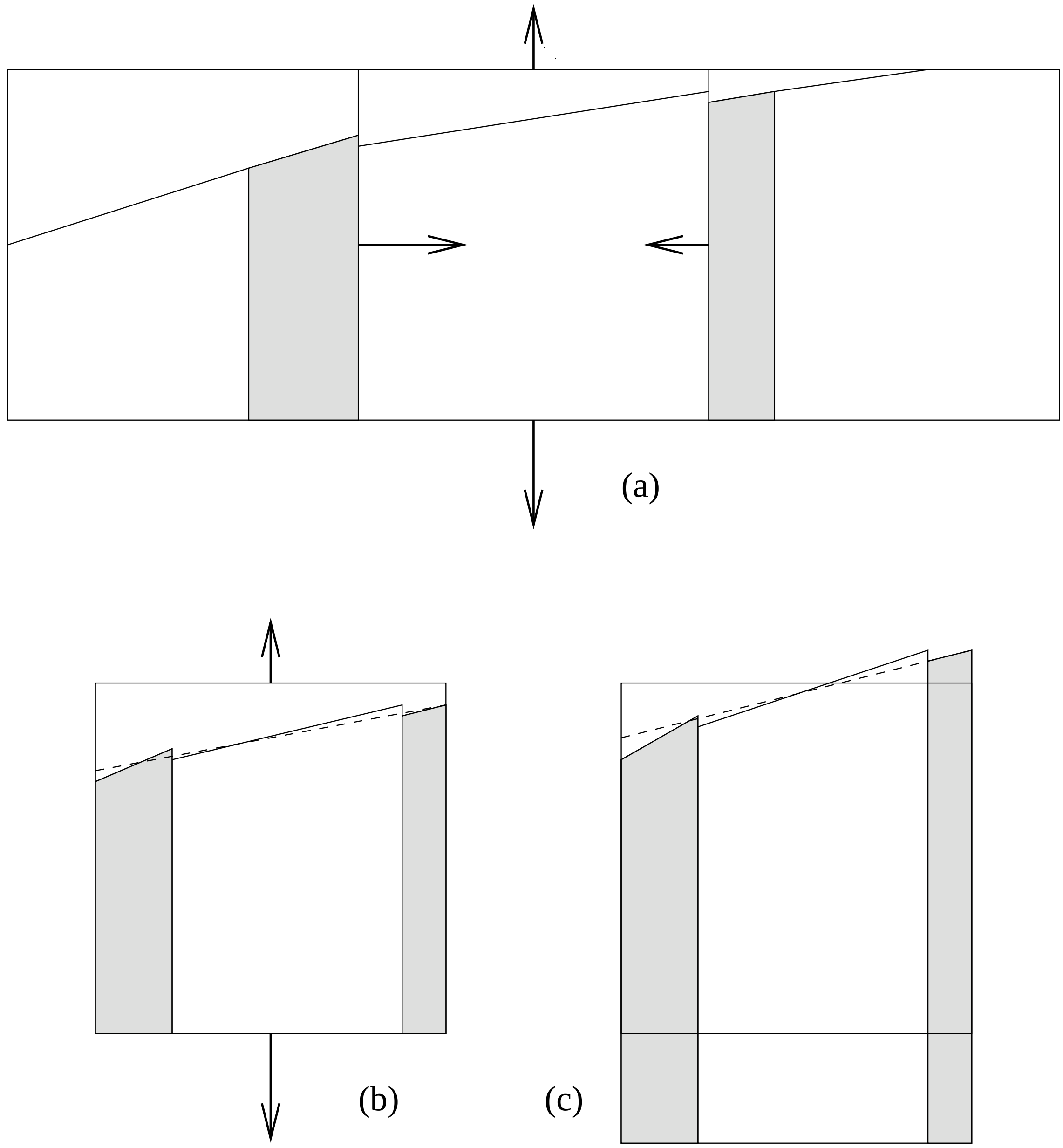

Figure 3: A geometrical view of the whole advection procedure for the mapping Π_{xy}: (a) Eulerian fluxes along the x coordinate; (b) mapping onto the unit square (implicit part of the method). In the split version of the algorithm the intermediate reconstruction is shown as a dashed line; (c) Lagrangian advection in the y direction (explicit component of the method).