An efficient bilinear interface reconstruction algorithm and consistent multidimensional unsplit advection scheme for accurate capturing of highly-curved interfacial shapes on structured grids

van der Eijk, Martin; Wellens, Peter

DOI
10.1016/j.jcp.2023.112656

Publication date
2024

Document Version
Final published version

Published in
Journal of Computational Physics

Citation (APA)
van der Eijk, M., & Wellens, P. (2024). An efficient bilinear interface reconstruction algorithm and consistent multidimensional unsplit advection scheme for accurate capturing of highly-curved interfacial shapes on structured grids. *Journal of Computational Physics*, *498*, Article 112656. https://doi.org/10.1016/j.jcp.2023.112656

Journal of Computational Physics 498 (2024) 112656

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

An efficient bilinear interface reconstruction algorithm and consistent multidimensional unsplit advection scheme for accurate capturing of highly-curved interfacial shapes on structured grids

Martin van der Eijk, Peter Wellens [*]

Delft University of Technology, Mekelweg 2, 2628CD, Delft, Netherlands

ARTICLE INFO

Keywords:
Volume of fluid
Interface capturing
Bilinear interface reconstruction
Multiphase flow
Unsplit advection scheme

ABSTRACT

A new bilinear interface reconstruction algorithm (BLIC) is presented to capture highly-curved interfaces more accurately on structured grids without a significant increase in computational costs compared to the standard piecewise linear interface calculation (PLIC) methods. The new reconstruction algorithm uses the initial PLIC segment and improves continuity of the interface using an averaging method. A curvature-weighted method improves the repositioning of the linear segments.

A new unsplit donating quadrant advection (DQA) scheme is introduced that is conservative and can create consistency with the momentum flux for two-phase flow models with a staggered MAC arrangement of variables within a grid cell. The consistent discretization of the fluxes prevents spurious interface velocities, negative densities, and instabilities. Standard 2D test cases and benchmarks demonstrate the performance of the BLIC and the DQA scheme, showing high accuracy and low costs compared to other available methods.

1. Introduction

1.1. Motivation and volume-of-fluid method

Modeling an interface between two separated fluid structures is a general application in industry. An application example for interface modeling is breaking water waves. Breaking wave impacts can enforce high peak pressures on marine structures. The forces are difficult to predict due to the complex free surface configuration of a breaking wave. Analytical expressions, assumptions, and empirical relations are currently used for the prediction of these forces which do not account for all their physical variability, hence the need for accurate free surface resolving numerical methods.

Sophisticated low-cost numerical two-phase flow models would decrease the uncertainty in determining the acting forces on marine structures. These models can also be used for a better understanding of the physics. This article proposes two new low-cost algorithms that improve the accuracy of modeling an interfacial flow compared to state-of-the-art approaches: an interface reconstruction algorithm and a stable interface advection scheme, both for use in numerical two-phase flow models.

There are multiple ways of interface modeling available for a two-phase flow model. One type of categorization is that between interface-capturing and interface-tracking techniques. The interface-tracking approach is a technique that explicitly transports Lagrangian markers surrounding an interface [39]. Disadvantages of such an approach are the difficulty of handling arbitrary changes of

---

[*] Corresponding author.
E-mail address: p.r.wellens@tudelft.nl (P. Wellens).

https://doi.org/10.1016/j.jcp.2023.112656
Received 27 November 2022; Received in revised form 22 September 2023; Accepted 19 November 2023

the topology and complex operations like the merging and breakup of interfaces [46]. In this article, we consider interface-capturing. Many interface-capturing techniques exist, e.g. Volume-of-Fluid (VOF) methods using a discrete volume fraction field C_f having a value between 0 and 1 to identify the position of the interface, and level-set methods using the signed distance to the interface.

The level-set method is easily differentiable, but often not fully mass conserving without special measures [50,33]. This article focuses on a VOF method that enforces strict volume conservation. A VOF method [17] uses a color function $f(\mathbf{x}, t)$ as an indicator of the material present at a defined position. The advection equation is

$$\frac{Df}{Dt} = \frac{\partial f}{\partial t} + (\mathbf{u} \cdot \nabla) f = 0, \tag{1}$$

where $\mathbf{u}$ is the interface velocity. The interface motion is approximated on a numerical grid of discrete cells covering the physical domain at hand. Cells are control volumes for governing equations like Eq. (1). In the discrete representation of Eq. (1), volume fraction field C_f is the average of the continuous color function over a given cell.

A VOF method for capturing the interface consists of two parts: the *geometrical interface reconstruction* and *fluid advection*. Geometrical interface reconstruction is required for finding the position of the interface from the color function. Fluid advection works by determining the donating regions and the fluxes. The donating region is the volume that is transported through the cell face while the flux is the quantity of the captured fluid going through the cell face. Our focus is on geometrical reconstruction using VOF although algebraic reconstruction [55,49] or no reconstruction with compressive terms would also have been options when our interest would have been on only the two phases of fluid. There are faster methods than geometric reconstruction, using explicit analytic formulas (a Moment of Fluid method [29]). Algebraic reconstruction requires less complex coding and computational costs than geometrical reconstruction, but may have lower accuracy for similar grid resolutions. With this article, however, we aim to resolve challenges with two-phase flow modeling especially when a (rigid) marine structure (a third phase) is to be represented in the domain, challenges that were encountered in our method [52] and in others. The implementation of the new schemes required to address the challenges is investigated in 2D on structured grids for two-phases. A reflection regarding their implementation in 3D is formulated.

1.2. Brief literature overview of geometrical interface reconstruction

Early algorithms to predict the interface orientation are the piecewise linear approximation [11], a stair case approximation [17], and simple line segments aligned with one of the grid axes (SLIC) [32]. These algorithms have a disadvantage that they can not keep fluid structures together due to the discontinuity in the interface from cell to cell; diffusion of the interface results in unphysical disconnecting droplets.

The Piecewise Line Interface Calculation (PLIC) method was an improvement over SLIC because it uses a linear function instead of a constant function of the spatial coordinates for determining the interface position in a grid cell. Even though the PLIC method still suffers from interface discontinuity at the faces of a grid cell, the PLIC method is often used to good effect. Many methods are available for determining the interface orientation, e.g. Parker and Youngs' method [36], (efficient) least square interface reconstruction (ELVIRA) [37], least-square gradient [43], height function scheme (Centered Columns) [37], Mixed Youngs-Centered (MYC) implementation [4], Centroid-Vertex Triangle-Normal Averaging (CVTNA) [24], piecewise continuous linear interface calculation (PCLIC) [53], and linear or quadratic fitting [48].

After the introduction of the PLIC method, other methods with higher-order functions in the spatial coordinates for the reconstruction were introduced. Price [40] proposed a parabolic reconstruction method (PPIC) based on a second-order equation for the interface segment. Similarly, Renardy and Renardy [42] presented a three-dimensional parabolic approach called PROST. Both methods showed an increase in accuracy but also in computational costs caused by iterative steps. The discussed parabolic reconstruction methods still display discontinuities in the interface. A reconstruction method that aims to reduce the discontinuity between the material interfaces is Patterned Interface Reconstruction (PIR) [30] using planar interfaces. This method is second-order accurate but does not fully satisfy continuity.

Further developed methods enforced the continuity of the interface. Sometimes even equality of the line segments' first derivative on either side of a cell faces is satisfied. The following references in this paragraph all found that a continuous representation of the interface reduces the diffusivity of the interface. Reconstruction methods based on cubic splines of Ginzburg and Wittum [14] and López et al. [25] (SIR) resulted in continuity and improved estimation of the curvature. However, the interfaces are wavy due to the non-locality of errors. Diwakar et al. [12] proposed the Quadratic Spline based Interface (QUASI) reconstruction algorithm satisfying the continuity and first derivative constraint. Although the QUASI method showed improved accuracy, the computational costs are an order higher than the standard low-order reconstruction methods. The noniterative PQLIC method [53] using quadratic lines has improved accuracy but, again, a significant increase in computational costs. Furthermore, the PQLIC method is not fully conservative. Another method worth mentioning is the piecewise circular arc interface calculation (PCIC) method [26] that makes use of a correction such that the interface is continuous. Other recently published methods provide higher-convergence rates but do not tackle topological changes of the interface like the methods discussed above [7,58].

The reconstruction methods using higher-order functions are accompanied by an increase in the difficulty of flux calculations, computational costs, and sometimes additional computational (iterative) steps. For increasing the accuracy of representing an arbitrary (highly-curved) fluid structure like a breaking wave with small flow features, the accuracy and sharpness improvement of the interface does not always outweigh the increase in computational costs. An example of such a highly-curved fluid structure is illustrated in Fig. 1 (wedge-entry experiments performed in dedicated setup at Delft University of Technology for validating multi-phase

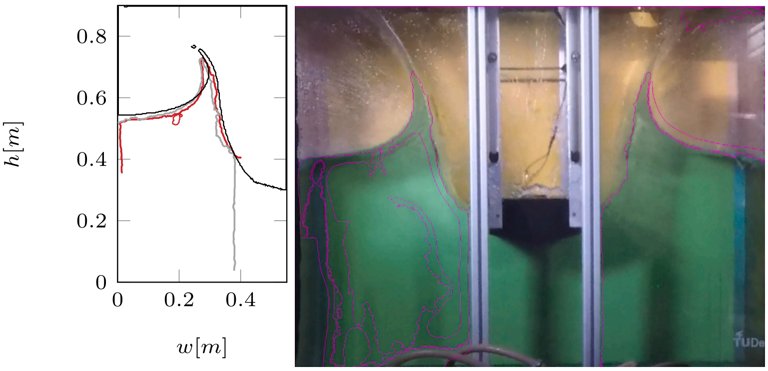


Fig. 1. Comparison free surface experimental & numerical data of a 2D wedge entry representing a section of a marine structure; — experiment left side, — experiment right side, — numerical result at 0.092[s]. The 'h' and 'w' indicate the height and width position, respectively. The impact speed is up to 7[m/s] for a wedge with a deadrise angle of 15[deg] and a width of 0.218[m]. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

numerical methods representing large deformations of the interface between phases; simulations performed with the method in this article).

Our new reconstruction algorithm aims to improve the accuracy of reconstructing highly-curved interfaces without a significant increase in computational costs compared to a PLIC algorithm using the Volume of Fluid method. The algorithm should be robust for an accurate interface reconstruction; the implementation needs to be straightforward. An iterative scheme is avoided such that no thresholds are needed. Compared to a PLIC algorithm, the presented scheme should reconstruct the interface with higher accuracy. We call the new algorithm the BiLinear Interface Calculation (BLIC) method. It is discussed in Sec. 2.

1.3. Brief literature overview of fluid advection schemes

The second part of a VOF method is fluid advection. Fluid advection geometrically estimates the fluid fluxes through the faces of a grid cell as a means of transporting the interface. The donating region and the corresponding reconstructed interface determine the size of the fluid flux. A distinction between two kinds of fluid advection schemes is made: an operator-splitting advection scheme and an unsplit advection scheme.

The operator-splitting schemes are characterized by the ease of determining the donating region and by how many calculation steps are required for the final fluid flux. These calculation steps can be split into two parts; determining the fluid flux in one direction and intermediate geometrical interface reconstructions. The advantage of a direction-split scheme is that it is straightforward to implement and robust. However, they show numerical diffusion and geometrical splitting errors that distort the interface [37]. The Conservative Operator Splitting for Multidimensions with Inherent Constancy (COSMIC) scheme [22] tries to minimize these errors and is used for comparison in this article with unsplit advection schemes. The choice for COSMIC as a reference is made because it was applied in our research field a number of times before [13,52]. The focus of this article is on unsplit advection schemes.

The second kind of fluid advection algorithm, unsplit advection schemes, prevents geometrical errors and the intermediate reconstruction step(s) by fluxing at once, sometimes in multiple directions, based on a polygon forming the donating region. The polygon is a plane figure described by a finite number of line segments resulting in a closed shape for every cell face. The polygon represents the donating region used for determining the flux. Polygon reconstruction is based on the velocity magnitude at the cell face and the surrounding velocity field.

The majority of unsplit advection schemes discussed here are for structured grids, but it is worth mentioning two recently proposed unsplit advection schemes for unstructured grids without going into more detail: a new triangulation algorithm with a modified Swartz method (UFVFC-Swartz) that shows high accuracy on unstructured grids [27], and a blended high-resolution scheme described as simple and efficient [19].

Many (multidimensional) unsplit advection schemes for structured grids are available, but not all advection schemes are without problems. The problems are characterized by Comminal et al. [9]: overlapping of donating regions (non-conservative), gaps between the donating regions (diffusion of interface), and non-conforming donating regions (undershoot or overshoot because the edges of the adjacent donating regions do not have the same length).

Early unsplit advection schemes were introduced by Rider and Kothe [43], Puckett et al. [41], Pilliod Jr and Puckett [37], and Harvie and Fletcher [16]. The Rider and Kothe scheme estimates the donating regions by a polygon based on face-centered velocities. Where the advection scheme is straightforward and has low costs, it is non-conservative because of overlapping donating regions and the diffusivity of the interface. The conservative Defined Donating Region (DDR) scheme of Harvie and Fletcher [16] prevents the overlap but increases the diffusivity of the interface and has a lower order of accuracy. Higher accuracy is obtained by Puckett's scheme, allowing, in contrast to DDR, fluid to enter and exit a cell in one time step while still being conservative. None of the discussed schemes results in conforming donating regions.

López et al. [25] proposed the Edge-Matched Flux Polygon Advection (EMFPA) scheme. Different than the advection schemes above, it is based on vertex velocities instead of face-centered velocities. The EMFPA scheme is accurate and conservative. However,

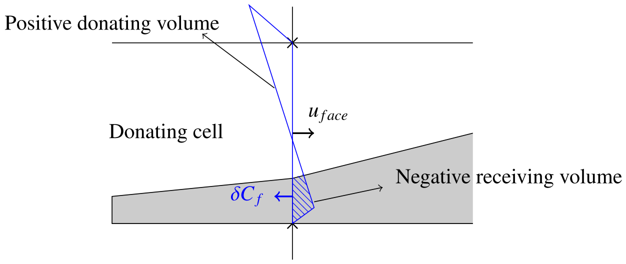


Fig. 2. Example with EMFPA scheme resulting in a negative donating region. The donating region is given by the blue lines ■. This region can generate a receiving flux larger than the donating one for a geometrically reconstructed interface, which is incorrect. The geometric fluid structure is given by ■. The flux δC_f (hatched with —) has a different direction (indicated by a blue arrow) than the face-centered velocity u_{face} (indicated by a black arrow). The faces of a grid cell are given by —. A cell vertex is given by ×.

the advection scheme did not initially result in conforming regions due to the volume corrections needed to satisfy the volume constraint, i.e. the volume of the polygon that satisfies the flux size based on the face-centered velocity.

More variations of the volume corrections of EMFPA are proposed [23,28], but they did not solve the non-conforming regions. Cervone et al. [8] proposed an additional volume correction that solved the problem of conforming regions. The FMFPA-3D scheme of Owkes and Desjardins [34] showed another solution for having conforming donating regions using EMFPA. They added a simplex to the polygon to create a solenoidal fluid flux.

The advection schemes discussed above are based on polygons constructed of linear edges. More sophisticated advection schemes, like Cellwise Conservative Unsplit advection (CCU) [9] using streaklines and the Stream scheme [15] using stream tubes, are not discussed in this article because of the computational costs. The CCU scheme showed to be around seven times more expensive than the EMFPA scheme.

Conclusions found in existing literature for a consistent two-phase flow solver [52] are the reason for introducing a new fluid-advection scheme. van der Eijk and Wellens [52] use a consistent approach to determine the mass and momentum fluxes on a staggered MAC arrangement of variables. A temporary continuity equation and densities discretized with the mass fluxes are used for consistency [59]. van der Eijk and Wellens [52] concluded that a direction-split advection scheme, like the COSMIC scheme, leads to inconsistency between the mass and momentum transfer and an increase in instability of the interface. The inconsistency is caused by a combination of the intermediate geometrical interface reconstruction step and the staggered MAC arrangement of variables. The densities resulting from the temporary continuity equation can even be negative, leading to instability of the interface.

EMFPA [34] prevents the intermediate geometrical reconstruction step while remaining conservative. However, as EMFPA is based on cell-vertex velocities, the created polygon can result in a so-called “negative” donating region. An example is illustrated in Fig. 2. The blue donating region self-intersects through the cell face resulting in a donating and a receiving volume at the same time. The receiving volume is of the opposite sign than the velocity centered at the cell face (u_{face}). There is the possibility that the final flux (δC_f) found with the receiving region is larger than the donating region or even the volume constraint. This is not correct, because the flux and the face-centered velocity have opposite signs.

From the authors’ point of view, the donating region in Fig. 2 is in contradiction with the momentum fluxes for two-phase flow solvers in which the velocity is assumed constant over the cell face; the momentum flux and mass flux based on the face-centered velocity can have a different sign from the VOF flux. The negative region can result in negative densities for a two-phase flow solvers such as van der Eijk and Wellens [52]. The problems of splitting schemes and the negative densities with EMFPA are discussed in more detail in Sec. 6.

The new advection scheme introduced below is a multidimensional unsplit VOF advection scheme that is conservative, conforming, and allows the material to enter and exit a grid cell in one time step. Face-centered velocities are used to prevent negative regions. We call the advection scheme DQA, which stands for donating quadrant advection. It is seen as a modification of EMFPA and the Rider-Kothe scheme that should obtain similar accuracy to EMFPA.

1.4. *Structure of the article*

The article starts with the introduction of the BiLinear Interface Calculation (BLIC) algorithm. In Sec. 3 the new BLIC method is compared with two standard PLIC methods for the static reconstruction of two shapes. In Sec. 4 the donating quadrant advection (DQA) scheme is introduced. The combination of the BLIC and DQA schemes are compared with other available methods in terms of accuracy and computational costs for traditional transport benchmarks in Sec. 5. In Sec. 6 the consistent application of the advection scheme for a two-phase solver [52] with a staggered arrangement of variables is discussed. The occurrence of negative densities when using a split advection scheme or the EMFPA scheme is highlighted. Instabilities caused by the reconstruction step and prevention of negative densities are illustrated in Sec. 6.4 for the example of a translating bubble. Conclusions are formulated in the final section.

2. BiLinear interface calculation

The new reconstruction method presented here will be called BiLinear Interface Construction (BLIC) method. It consists of the following steps:

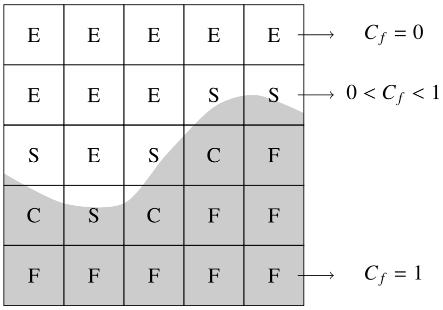


Fig. 3. Labeling of cells [52]; labels F, S, C, and E. Fluid is indicated by ■ ($C_f > 0$).

1. identification of cells that contain the interface (Sec. 2.1);
2. an initial Piecewise Linear Interface Construction (PLIC) per cell based on the gradient of the volume fractions C_f, with an inspection of the orientation of the interface in neighboring cells (Sec. 2.2);
3. determination of the wetted parts of the cell faces (Sec. 2.3);
4. weighted averaging of the wetted parts on either side of the cell face to improve continuity of the interface (Sec. 2.4);
5. addition of a control point to create a bilinear interface, while keeping the volume fraction C_f the same (Sec. 2.5).

The number of the step refers to the number of the subsection below in which the step is discussed, with an evaluation of the method in the last subsection. The method should remain low-in-costs, achieve higher accuracy for highly curved interfaces, and increase the continuity of the interface compared to PLIC methods. The scope of the implementation is limited to 2D. Considerations about extending the implementation to 3D are given at the end of this chapter. PLIC methods (e.g. Parker and Youngs, least-square gradient, ELVIRA) are portable to unstructured grids [18]. The BLIC method is a low-effort extension of PLIC which would be equally portable to unstructured grids.

2.1. Identification of the interface

Labeling of grid cells is used to account for the position of the interface between two fluids. The labels decide where the reconstruction of the interface takes place. The choice for the label of each cell is based on the volume fraction. The volume fraction C_f indicates the degree to which a grid cell is filled with fluid and takes a value between 0 and 1. We have adopted the labeling system of van der Eijk and Wellens [52], omitting solid structures (B label) for brevity. In Fig. 3, the labeling of the cells in a domain containing two fluids is illustrated, using label E(mpty) for cells completely filled with the lighter of the two fluids ($C_f = 0$), and the label S(urface) for cells with some of the heavier of the two fluids adjacent to E cells in the directions of the cartesian grid lines. Remaining cells are defined as F(luid) cells.

The label C(orner) is used to create an improved representation of the interface. C-labels are given to those F-cells that neighbor diagonally to a single E-cell, and make sure that those cells are recognized as part of the interface where in previous methods they were not. S-labeled cells and C-labeled cells need reconstruction of the interface between the two fluids. Why S-cells and C-cells are labeled differently is because of continuity of the interface and will become clear later in this section. S-cells and C-cells are also referred to as interface cells.

2.2. Initial PLIC reconstruction

A PLIC method uses a piecewise linear segment that approximates the real physical interface in a cell. An example of such a linear segment is illustrated in Fig. 4. It is described by the following 2D equation

$$m_x x + m_y y = \alpha, \tag{2}$$

where $\mathbf{m} = [m_x, m_y]^T$ is the interface orientation, $\mathbf{x} = [x, y]^T$ the position vector of a point on the interface, and α the distance to the origin so that the interface satisfies the volume constraint C_f for the given interface orientation. The distance to the origin α of the interface is found analytically, as in Scardovelli and Zaleski [47], either in 2D or in 3D.

The first step of a PLIC method is to identify the interface orientation, often based on the gradient of the volume fraction. Many methods to determine the interface orientation are published and discussed above. In the presented work, Parker and Youngs' method and Mixed Youngs-Centered (MYC) are discussed because of their ease of implementation and the relatively low costs [13]. Here, in any case, Parker and Youngs and MYC are compared to identify their effect on the final reconstruction. Diwakar et al. [12] mentioned that for their higher-order QUASI method, the interface orientation method for the initial PLIC is expected not to be relevant for the final reconstruction. We will find out if this is also the case for the BLIC method.

The Parker and Youngs' method, for a 2D grid, determines the gradients of the volume fraction C_f at four corners of the central grid cell using finite difference. The gradients are normalized and averaged, resulting in the interface orientation $\mathbf{m}$. The method is first-order accurate, but the errors in the reconstruction are comparable to second-order methods [37,48].

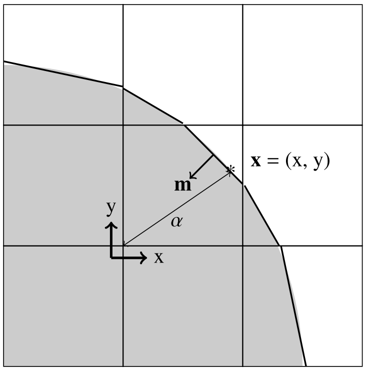


Fig. 4. A quarter circle reconstructed using piecewise linear segments (PLIC); the normal vector **m** and height value constant α for a random **x** are illustrated for a 3×3 stencil. Fluid is indicated by ▨ ($C_f > 0$).

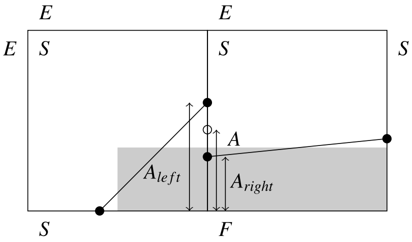


Fig. 5. Definition of face values A. Fluid is indicated by ▨. PLIC endpoints by ●. A more continuous endpoint ○ is found by averaging PLIC endpoints. The labeling of the cells and the surrounding cells is shown by means of letters F, S and E.

The Mixed Youngs-centered (MYC) method [4] is a mix between the Parker and Youngs' method and the standard height function approach with a stencil of 3×3 grid cells. It is described as a fast and accurate way to compute the interface normal **m**. Aulisa et al. [4] showed that the implementation outperforms the Parker and Youngs' method [36] in terms of accuracy and approaches second-order accuracy. Düz [13] reported the MYC implementation as a good compromise between accuracy and computational cost.

2.3. Face values

The next step of the BLIC method is determining the face values of the PLIC reconstruction. Every PLIC line contains two endpoints positioned at a cell face. An example is illustrated in Fig. 5. The face value is determined by a PLIC endpoint, illustrated by ●. A is a value between 0 and 1 that describes the part of a cell face that connects to the heavier fluid ($C_f = 1$). These endpoints are found analytically [47]. The 2D PLIC line forms a quadrangular or triangular shape, uniquely described by m_x, m_y, and α. For every cell with label S or C a PLIC line is computed. The endpoints of PLIC lines in the two cells on either side of a cell face in general do not coincide. The face apertures based on the PLIC lines left and right cell face, A_{left} and A_{right}, are combined so that face aperture A is the average of A_{left} and A_{right}.

The neighboring PLIC endpoints may be not positioned on the same cell face. This makes it harder to obtain continuity of the interface. Repositioning of the endpoints then is needed. Examples of how endpoints could be repositioned is illustrated in Fig. 6, which are inspired by the approach of Diwakar et al. [12]. The red arrows show the directions in which PLIC endpoints are moved. Every situation needs a different treatment to determine face aperture A, created by the more continuous interface that is represented by means of the dashed lines after repositioning the endpoints.

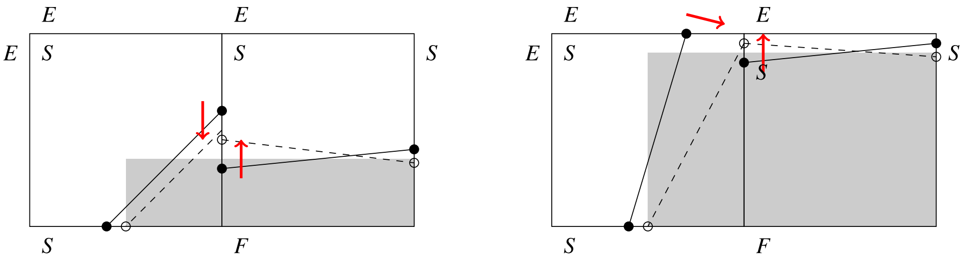
We want to avoid distinguishing between situations like in Fig. 6 to limit the involved computational effort. A such method should satisfy the following constraints

1. the face value A of an edge in an interface cell (S-or C-cell) neighboring a F-or E-cell is equal to 1 or 0, respectively;
2. a S-or C-cell has two face values A between 1 and 0;
3. the volume constraints are enforced without iteration or a significant increase in computational costs;

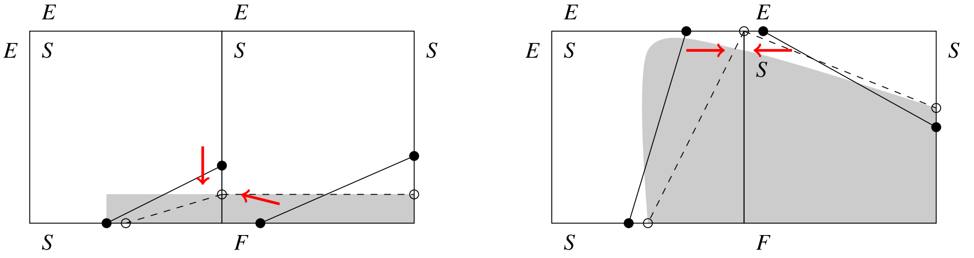
and is described in the next section.

2.4. Averaging method: curvature weighted

The proposed averaging method for determining the new face value A is based on the motto that *the curvature is more accurately predicted by a piecewise linear segment when the curvature is close to zero*. When averaging, the method gives more weight to the parts



(a) Both values A_{left} and A_{right} between 0 and 1 on common edge. (b) Full face value on common edge; $A_{left} = 1$ and $0 < A_{right} < 1$.



(c) A zero face value on common edge; $0 < A_{left} < 1$ and $A_{right} = 0$. (d) No common edge with face values A_{left} and A_{right} equal to 1.

Fig. 6. Four different situations of PLIC reconstructions to obtain continuity along the interface without satisfying the volume constraint yet [12]. Each situation ask for a different treatment to find the endpoints, which requires computational effort. A method that avoids distinguishing between situations is proposed in Sec. 2.4. Fluid is indicated by ▨. PLIC endpoint ●. Continuous endpoint ○. The labeling of the cells and the surrounding cells is shown by means of letters F, S and E.

of the interface that have low curvature. A piecewise linear representation for those parts is expected to be more accurate than for the parts of the interface with high curvature. Weighted averaging prevents the parts of the interface with high curvature from influencing the parts with low curvature, improving the overall accuracy of the interface reconstruction. A better curvature prediction gives a more accurate representation of the face values of the interface.

The curvature is calculated for interface cells (S- and C-cells). Curvature is not defined for the remaining labels. Many methods are available to predict curvature. In this article, we will only highlight two; the standard height function technique [10,38] and the classic technique using finite difference [1,36]. These methods are chosen for their relative ease of implementation and are used for illustrative purposes only.

The curvature (κ) is defined as

$$\kappa = -\nabla \cdot \left(\frac{\nabla \phi}{||\nabla \phi||} \right), \tag{3}$$

where ϕ can be a height value (a sum of aligned volume fractions) or just the volume fraction C_f. The standard height function technique is described by, for instance, Kleefsman et al. [20], using a 3×3 stencil. When the interface is more vertical than horizontal, the height function is defined parallel to the horizontal axis. The finite difference technique to calculate the curvature is based on the approach of Parker and Youngs [36]. The approach computes the normal vector to the interface at every corner of a cell. The difference in normal vectors at the corners results in the curvature of the free surface in that cell, defined at the cell center.

An example of a interface on a 3×3 stencil is shown in Fig. 7 (note: representation of the interface only this crude for illustration; examples with finite curvature are available in literature [1]). The curvature at the corner of the fluid body in this example should approach infinity; the values for the height function technique and the classic finite difference technique are given. Fig. 7 is an extreme example, with a strongly underresolved free surface configuration, to exaggerate the issue with the standard height function technique. The issue with the standard height function technique is that applying the method at sharp corners of the interface can lead to negative values of the curvature. To avoid negative values for the curvature, we continue with the classic finite difference technique. That approach does not guarantee convergence of curvature in general, but the results in the next chapter and in the remainder of the article show that it is well behaved in combination with BLIC.

For more information about accurate methods for determining curvature, one is referred to Popinet [38] and Abadie et al. [1], who resolved the issue with negative curvature by making the height function technique adaptive.

The new face value A is calculated as

$$A = \frac{\kappa_{right}^{n_\kappa} A_{left} + \kappa_{left}^{n_\kappa} A_{right}}{\kappa_{left}^{n_\kappa} + \kappa_{right}^{n_\kappa}}, \tag{4}$$

where n_κ is a weighting factor. This weighting factor is a free parameter for which a value is found by comparing simulation results in Sec. 3 for the static reconstruction of two different shapes. Referring to the constraints given in Sec. 2.3, the formulated method ensures that the face values of a F-cell or E-cell are 1 and 0, respectively. Repositioning of the endpoints (see Fig. 6) is automatic when the curvature-weighted approach is used. The curvature weighted approach should ensure that when an interface cell has more

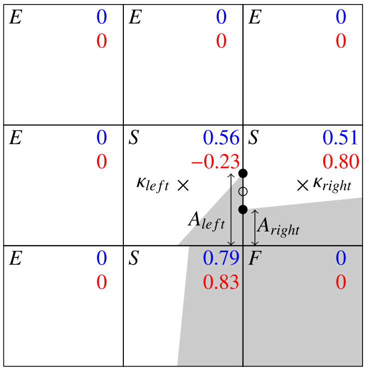
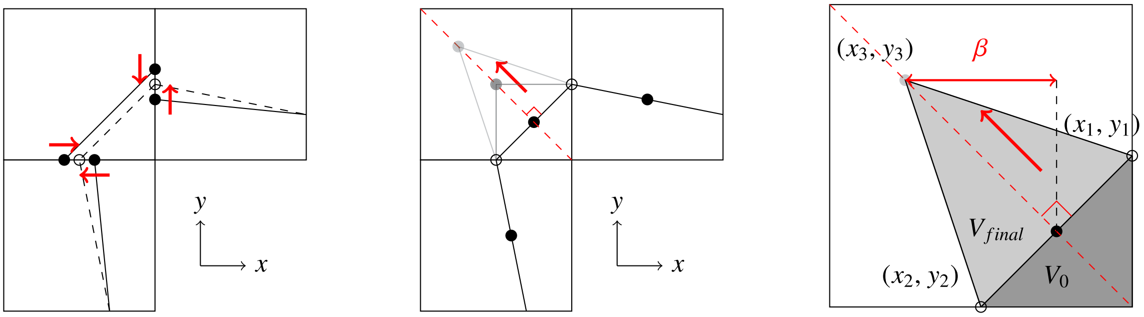


Fig. 7. Prediction of the curvature (κ) using the volume fractions C_f with two different techniques; the classic technique (🟦), and the height function technique (🟥). Fluid is reconstructed with original PLIC indicated by ■ ($C_f > 0$). PLIC endpoint ●. Continuous endpoint resulting in A ○. Center grid cell where κ is defined (×).



(a) Averaging the face values of PLIC.

(b) Adding the BLIC control point.

(c) Definitions for BLIC reconstruction.

Fig. 8. Procedure BLIC reconstruction to meet volume constraint. The volume V_0 after averaging the face values to A is ■. The final volume V_{final} satisfying the volume constraint is ▨. BLIC additional point ●. Continuous endpoint ○.

than two neighboring interface cells in vertical and horizontal direction, the number of face values remains two by applying it only for the lower two values of A.

2.5. Addition of BLIC point

When the continuity is improved by averaging as in Fig. 8a, the reconstruction will not satisfy the volume constraint. Therefore an additional point is introduced which will make the reconstruction bilinear. The additional point is initially positioned in the middle of the reconstructed line, see the ● in Fig. 8b. The point moves perpendicular to the continuous reconstructed line until it satisfies the volume constraint for C_f.

No iterations are needed to find the final position of the additional point. Using the definitions in Fig. 8c, the coordinates of the additional point (x_3, y_3) are found from

$$y_3 = \gamma(x_{3,0} + \beta) + b \text{ where } \gamma = -(x_1 - x_2)/(y_1 - y_2) \text{ and } b = y_1 - \gamma x_1. \tag{5}$$

The points are ordered counter clockwise. The third line can move over the line y(β) with initial point

$$x_{3,0} = \frac{x_1 + x_2}{2} \text{ and } y_{3,0} = \frac{y_1 + y_2}{2}. \tag{6}$$

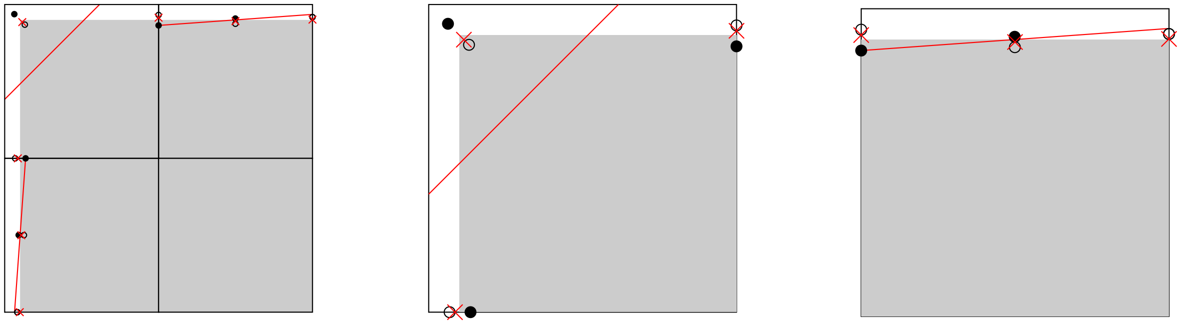
The value of β is found by

$$\beta = \frac{V_{final} - V_0}{\frac{1}{2}\gamma(x_2 - x_1) + \frac{1}{2}(y_1 - y_2)}, \tag{7}$$

where $V_{final} - V_0$ is the difference between the volume constraint and the volume before adding the extra control point.

2.6. Evaluation

The curvature-weighted averaging is evaluated for the reconstruction of a sharp corner in Fig. 9, including the additional point satisfying the volume constraint. Two values for n_κ, 0 and 2, are compared with the end-positioning approach of Diwakar et al. [12].



(a) Complete corner reconstruction BLIC.

(b) Middle cell reconstruction difference BLIC.

(c) Outer cell reconstruction difference BLIC.

Fig. 9. Continuous face value calculation A for a 90 degrees corner. Red line is PLIC Youngs reconstruction (—). ● is the approach of Diwakar et al. [12] for determining the face values. ○ is the averaging method for $n_\kappa = 0$. x is the weighted averaging method for $n_\kappa = 2$. Fluid is indicated by ■ ($C_f > 0$).

Note that n_κ equal to 0 is simply averaging. The approach of Diwakar et al. [12] needs special treatments for repositioning endpoints as mentioned in Sec. 2.3. Compared to Fig. 6b their approach ignores the information of A_{left} and assumes A is equal to A_{right}.

The results in Fig. 9 show that for this case, the proposed curvature-weighted averaging method is an improvement compared to the approach of Diwakar et al. [12]. The value of n_κ affects the reconstruction and therefore needs extra evaluation in Sec. 3.

An additional remark concerns the BLIC point dividing the line segment. When this BLIC point is positioned outside of a grid cell, the initial PLIC line is used. The interface in that case is not continuous anymore. This is most likely to happen for S and C-cells which are almost empty or almost full. How often this happens is evaluated for the reconstruction of circle with 64 grid cells in the diameter. The chance is in the order of 0.1% and decreases to zero with grid refinement. For the reconstruction of a square with its sides resolved by 64 grid cells, the same results are obtained. The test case in Sec. 5.2 elaborates on this by also considering underresolved structures and shows how accurate BLIC remains compared with PLIC.

Another remark concerns the C-cells which is a reason why S-and-C-cells do not have the same label. There is the chance that a C-cell, connected to two S-cells, has an original PLIC interface with an opposing orientation than the S-cells before applying BLIC. This works against creating a smooth continuous BLIC interface when using the curvature-weighted approach. For this situation, the normal of the interface in the C-cell is reoriented such that it points in the direction of the average of the normal vectors of the interfaces in the neighboring S-cells. After reorienting the normal vector, the face values A in the BLIC algorithm are determined. The situation of opposing orientations in neighboring interface cells is even rarer than BLIC points being positioned outside of grid cells.

A final remarks concern the extension to 3D. The BLIC algorithm can be extended to 3D by following a similar procedure with planes and lines instead of lines and points. The BLIC algorithm presented in Sec. 2.5 can be formulated in 3D without increasing the complexity. The shape of the interface will be a pyramid after positioning the endpoints using the curvature-weighted approach. The cost increase of using BLIC instead of PLIC in 2D is evaluated in Sec. 5.3. The cost increase of using BLIC instead of PLIC in 3D is proportional to that in 2D.

3. Static reconstruction interface

The accuracy of the following reconstruction methods is compared for two static shapes: Parker and Youngs [36], Mixed Youngs Centered (MYC) [4], and BLIC. The transport of fluid is not involved, only reconstruction. The shapes that are evaluated are a circle and a square.

A domain of one by one is used. The width of the square is 0.512, and the circle radius is 0.368. The square is positioned in the center of the domain and the circle off-center at (0.525, 0.464) to prevent the reconstruction method from favoring a specific interface orientation [43]. The L_1 norm of the error between the exact interface and the reconstructed interface is calculated as

$$L_1 = \int\int |S_{exact}(x,y) - S_{reconstructed}(x,y)|\,dxdy, \tag{8}$$

in which S is the interface. The L_1 norm for both shapes and the different reconstruction methods is given in Table 1 as a function of the number of grid cells in the domain.

Circular shapes are common for fluid configurations, like droplets and bubbles. High accuracy in representing these shapes is important for contacts between bodies. The reconstruction using Parker and Youngs and MYC alone results in errors at least twice as high as when they are combined with BLIC.

Sharp corners in fluid bodies, like those in a square, are not present due to surface tension effects. However, when moving solid objects representing structures are present in a simulation, and when it is convenient to reconstruct the interface between solid objects and fluids in the same way as between fluids, then sharp corners do play a role. From that point of view, we want to know how the various reconstruction methods deal with sharp corners. The reconstruction error in the L_1 norm is reduced by an order of magnitude compared to Parker and Youngs and MYC when using the BLIC method.

Table 1
Reconstruction error static square / circle for different grid resolutions with $n_\kappa = 2$. The order of convergence is given in between parentheses.

Grid	Youngs		Youngs + BLIC		MYC		MYC + BLIC	
	Square	Circle	Square	Circle	Square	Circle	Square	Circle
10	4.81e-3	8.92e-4	3.51e-4	3.59e-4	5.64e-3	9.76e-4	8.82e-4	4.63e-4
20	6.81e-4 (2.82)	2.59e-4 (1.78)	7.61e-5 (2.21)	9.91e-5 (1.86)	7.72e-4 (2.87)	2.35e-4 (2.05)	7.61e-5 (3.53)	1.21e-4 (1.94)
40	2.48e-4 (1.46)	1.17e-4 (1.15)	2.57e-5 (1.57)	4.38e-5 (1.18)	3.40e-4 (1.18)	6.91e-5 (1.77)	1.96e-5 (1.96)	3.41e-5 (1.83)
80	7.20e-5 (1.79)	5.39e-5 (1.12)	5.62e-6 (2.19)	2.03e-5 (1.11)	8.82e-5 (1.95)	3.02e-5 (1.19)	1.93e-6 (3.34)	1.53e-5 (1.16)
160	4.06e-6 (4.15)	2.83e-5 (0.93)	1.90e-7 (4.89)	1.09e-5 (0.90)	4.44e-6 (4.32)	1.57e-5 (0.94)	4.26e-7 (2.18)	7.91e-6 (0.95)
320	1.79e-6 (1.18)	1.41e-5 (1.01)	6.24e-8 (1.60)	5.52e-6 (0.98)	1.97e-6 (1.17)	7.71e-6 (1.03)	1.72e-7 (1.32)	3.86e-6 (1.03)

Table 2
Reconstruction error static square / circle with BLIC (with Parker and Youngs initialization) for increasing value of n_κ, and therefore dependency of curvature κ.

n_κ	0		1		2		3		4	
	Square	Circle	Square	Circle	Square	Circle	Square	Circle	Square	Circle
10	1.26e-3	3.76e-4	4.68e-4	3.38e-4	3.51e-4	3.59e-4	2.34e-4	4.22e-4	1.19e-4	7.05e-4
20	8.00-4	1.03e-4	3.18e–5	9.91e-5	7.61e-5	9.91e-5	8.94e-5	9.85e-5	9.28e-5	9.79e-5
40	8.25e-5	4.06e-5	5.93e-6	3.96e-5	2.57e-5	4.38e-5	3.85e-5	4.58e-5	4.60e-5	4.67e-5
80	1.75e-5	1.76e-5	5.04e-6	1.85e-5	5.62e-6	2.03e-5	6.20e-6	2.13e-5	6.77e-6	2.19e-5
160	4.44e-6	9.39e-6	2.47e-7	9.67e-6	1.90e-7	1.09e-5	4.93e-7	1.15e-5	6.80e-7	1.18e-5
320	1.97e-6	4.69e-6	1.41e-7	4.94e-6	6.24e-8	5.52e-6	2.10e-7	5.84e-6	3.05e-7	6.02e-6

The results in Table 1 show that when an underresolved material interface like the corner of a square is present, BLIC decreases the reconstruction error significantly with a higher order of convergence. For the circle, which is not underresolved, the results show that the decrease in error with BLIC is less. For the circle, the reconstruction error with BLIC converges to the error obtained with PLIC on a grid that is twice as fine. This was expected because BLIC divides piecewise linear segments in two.

The effect of the magnitude of weighing factor n_κ on the reconstruction error in Eq. (4) is evaluated in Table 2 for BLIC (initialization with Parker and Youngs). The results show that the dependency on the curvature for the face apertures in the BLIC method improves the accuracy of reconstructing a square. For the square having sharp corners, the best performing value for n_κ is 2. A lower value for n_κ performs better than a high value for the reconstruction of a circle. BLIC performs, independent of the value of n_κ, better than Parker and Youngs when comparing the errors in Table 2 with Table 1. The reason for the improvement compared to PLIC when using higher values for n_κ is because the corners of a square in general are inherently underresolved, making PLIC so inaccurate. For circles, this inherent absence of resolution is not there, so that BLIC and PLIC have a comparable accuracy. Making n_κ adaptive is an interesting option for future research. In the remainder of this article, we will continue with n_κ is 2 with the underlying thought of representing fluid configurations with highly-curved interfaces.

The cost evaluation and comparison with other available reconstruction algorithms is done for the benchmarks in the next section.

4. Unsplit edge-matched upwind flux polygon advection scheme

A new unsplit multidimensional advection scheme has been developed. It is compared with a direction-split advection scheme, called COSMIC [22], for several benchmarks. Before introducing these advection schemes, a brief overview of the advection of the interface using the Volume-of-Fluid (VOF) approach is given.

4.1. VOF advection

The interface is captured using Eq. (1) under the assumption of incompressibility and the no-slip condition between the two fluids. In case of a Volume of Fluid (VOF) method, the color function f is replaced by a discrete volume fraction C_f. The discrete volume fraction indicates the filling ratio of one of the fluids in a grid cell and is the average of the color function.

The fluids are transported with the assumption of incompressibility

$$\int_{V_c} \frac{C_f^{n+1} - C_f^{n}}{\delta t} dV = -\int_{V} (\mathbf{u} \cdot \nabla) C_f dV = -\int_{V} \nabla \cdot \left(\mathbf{u} C_f\right) dV = -\oint_{S_c} \left(\mathbf{u}^n C_f^n\right) \cdot \mathbf{n} dS, \tag{9}$$

in which V and S are the volume and boundary of the grid cell, respectively. The superscript indicates the time level; $n+1$ is the new time level and n is the old-time level. Eq. (9) is the basis for the discretization of the advection equation. A straightforward discretization of the advection equation reads

$$C_f^{n+1} = C_f^{n} - \frac{1}{V} \sum_{face} \delta C_{f,face}, \tag{10}$$

in which subscript $face$ refers to a cell face, and the VOF flux δC_f is the amount of fluid transported from one cell to the next. Similar to Fig. 2, Fig. 10 shows an example of a flux at the cell face and the donating region for a one-directional flow field. The

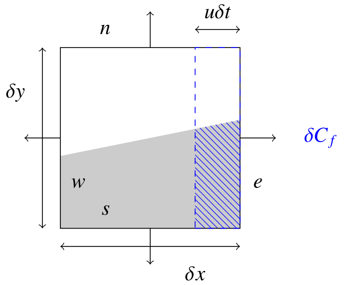


Fig. 10. Flux is calculated in a grid cell. Fluid is indicated by ■. The other fluid is white. The amount of fluid being transported (fluxed δC_f) is hatched with —. The cell faces are named. The donating region is indicated by the dashed line.

velocity field is defined at the cell faces of a grid cell. The VOF flux is based on the reconstructed interface segment of the donating cell. The flux is of the form

$$\delta C_f \sim C_f u \delta t \delta A, \tag{11}$$

where δA is the part of the area of a cell face in contact with the fluid being transported. The total area of the cell face in Fig. 10 is equal to δy. The naming of the cell faces, w(est), e(ast), n(orth), and s(outh), are illustrated.

4.2. COSMIC advection scheme

The COSMIC split scheme we compare the new method to consists of multiple transport and reconstruction steps [22]. The COSMIC scheme, in 2D, reads as follows

$$\begin{aligned} C_f^{*X} &= C_f^n - \frac{1}{V} \sum_{face=w,\ e} \delta C_{f,face}(u^n, C_f^n) + \frac{C_f^n}{V} \sum_{face=w,\ e} u_{face} \delta y, \\ C_f^{*Y} &= C_f^n - \frac{1}{V} \sum_{face=n,\ s} \delta C_{f,face}(v^n, C_f^n) + \frac{C_f^n}{V} \sum_{face=n,\ s} v_{face} \delta x. \end{aligned} \tag{12}$$

The interface of the intermediate volume fraction fields C_f^{*X} and C_f^{*Y} is again reconstructed, resulting in $C_f^{X,\ n+1/2}$ and $C_f^{Y,\ n+1/2}$. The superscripts $*$ and $n+1/2$ indicate intermediate time levels. A distinction is made between these two time levels ($*$ and $n+1/2$) for the discussion in Sec. 6 of why inconsistency can be found with COSMIC. Even though the values for C_f with time levels $*$ and $n+1/2$ in a cell are the same, the staggered arrangement of variables causes a difference for momentum control volumes after the reconstruction.

The new fraction field is found from

$$C_f^{n+1} = C_f^n - \frac{1}{V} \sum_{face=w,\ e} \delta C_{f,face}\left(u^n, \frac{C_f^n + C_f^{Y,\ n+1/2}}{2}\right) - \frac{1}{V} \sum_{face=n,\ s} \delta C_{f,face}\left(v^n, \frac{C_f^n + C_f^{X,\ n+1/2}}{2}\right). \tag{13}$$

4.3. Proposed Donating Quadrant Advection (DQA) scheme

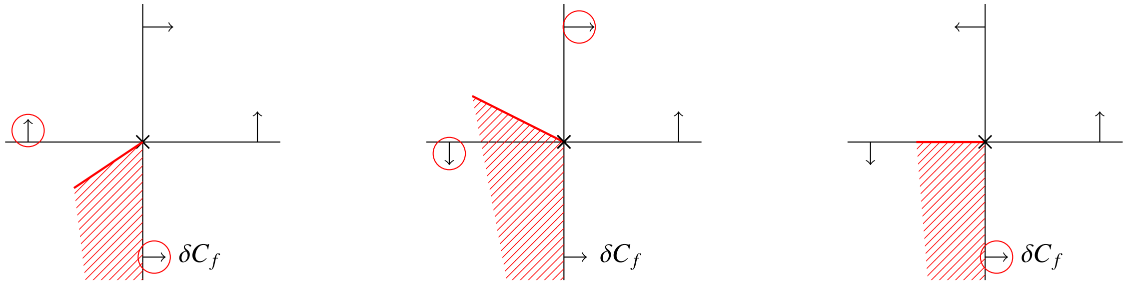
As mentioned in the introduction, we want the new advection scheme to satisfy the following conditions.

1. It will be an unsplit multidimensional scheme: to prevent distortion of the interface, and geometrical errors.
2. Be conservative and non-diffusive: no overlapping regions or gap between neighboring donating regions.
3. Feature conforming regions: the length of the edges of two neighboring donating regions should have the same length.
4. Allow fluid to enter and exit a grid cell in one time step.

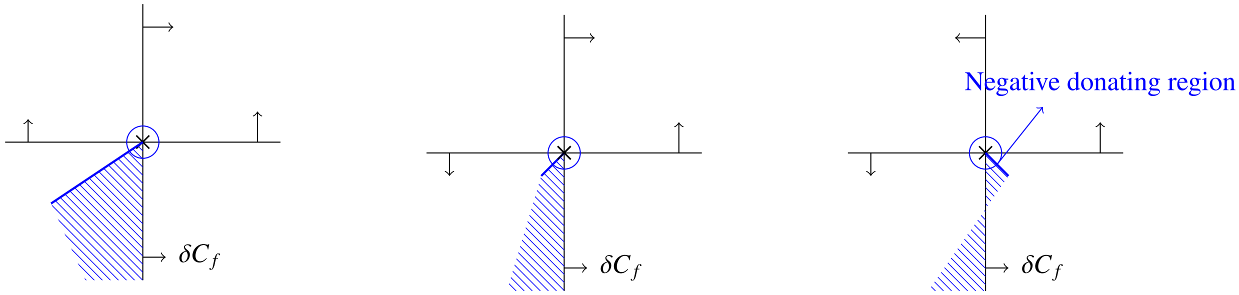
The EMFPA [34] scheme meets these requirements. The computational costs and accuracy of the new scheme should not be inferior to the EMFPA scheme. It also, in contrast to EMFPA, needs to be consistent with the mass and momentum transfer of two-phase flow solvers similar as presented in van der Eijk and Wellens [52]. Therefore, the subsection explaining the new scheme consists of four steps: positioning and determining the linear edges of the donating region, volume correction by adding an extra control point to satisfy the volume constraint, and checking if the final donating region complies with the requirements.

4.3.1. Positioning of the linear edges

The new scheme is based on face-centered velocities, like Rider-Kothe [43], to prevent the so-called negative donating flux-region, illustrated in Fig. 2. The negative donating region can result in a VOF flux of opposite sign than the fluxing face velocity. We consider this inconsistent with the direction of the face-centered velocity and the effect thereof is elaborated upon in Sec. 6. The donating



(a) DQA 1: Edge of polygon in the corresponding cell. (b) DQA 2: Edge of polygon in another cell. (c) DQA 3: Velocity change of sign.



(d) EMFPA 1: Edge of polygon in the corresponding cell. (e) EMFPA 2: Edge of polygon in another cell. (f) EMFPA 3: Velocity change of sign.

Fig. 11. Three different situations of flux polygon construction using velocities defined on the cell faces. The thick linear edge is determined by the encircled velocities. Grid cells are illustrated with −. The amount of fluid being transported with DQA (fluxed δC_f) is virtual hatched with — and with EMFPA is hatched with —. A cell vertex is given by × where the velocities are averaged for EMFPA. A velocity position is given by an arrow.

region needs to be on one side of the cell face, given by the sign of the face velocity. In contrast to Rider-Kothe, the new scheme is conservative and has no overlap with neighboring donating regions. We will name the scheme DQA, short for Donating Quadrant Advection.

Three different situations for constructing a linear edge of the donating region are illustrated in Fig. 11. The difference between the vertex velocity-based EMFPA scheme in Figs. 11d to 11f and the new scheme in Figs. 11a to 11c is given. The encircled velocities in Fig. 11 are used for determining the magnitude and direction of the thick-lined linear edge. The vertex velocities used for EMFPA are found by averaging the surrounding face-centered velocities.

The DQA scheme depends on the direction of the fluxing face-centered velocity. The quadrant where the linear edge is positioned decides which single face-centered velocity is used. The linear edge is always positioned in a quadrant where, in 2D, two (vertical and horizontal) face-centered velocities are donating. When no quadrant has two donating face-centered velocities, in Fig. 11c, a non-sloped linear edge is made to prevent a negative donating region. The use of a non-sloped linear edge has the disadvantage that it can lead to a gap between neighboring donating regions because the non-sloped edge does not match with the edge of the donating region of the upper velocity in Fig. 11c. The improved robustness and loss of accuracy as a result of this procedure is evaluated by means of benchmarks in Secs. 5.1 and 5.2 and by comparing with EMFPA. The results show that DQA has similar accuracy compared to EMFPA, without the risk of negative densities.

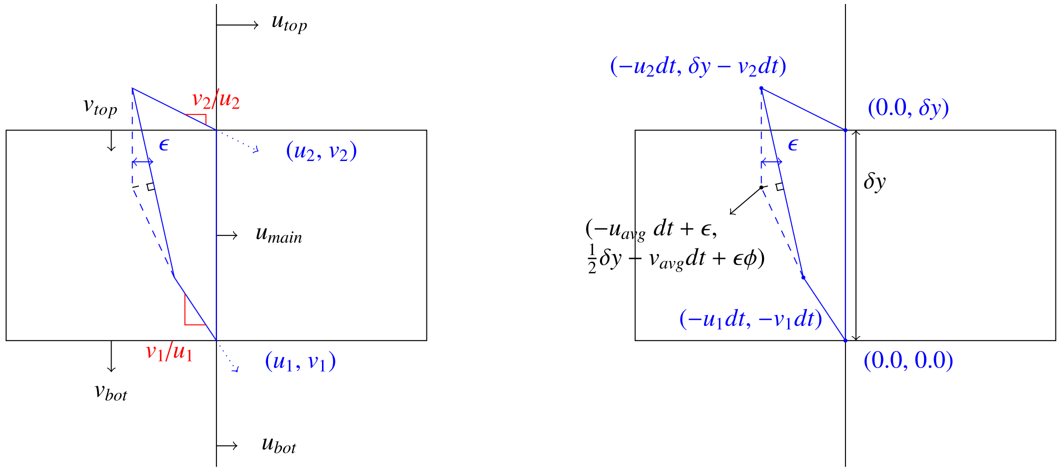
4.3.2. Determination of the linear edges

Determining the complete donating region with DQA, including the coordinates of the vertices, is based on Fig. 12. The flux velocity u_{main} determines the volume of the donating region. The vertical velocities v_{bot} and v_{top} are used to determine the slope of the linear edges. The distance value ϵ and slope value ϕ are related to the volume correction discussed in the next subsection.

The velocities are defined such that the positioning of the linear edges complies with the three situations illustrated in Fig. 11. The velocities in Fig. 12a for the linear edges of the donating region are defined as follows

$$
\begin{aligned}
[u_1,\ v_1] &= \begin{cases} [u_{main},\ v_{bot}], & \text{if } \operatorname{sign}(v_{bot}/u_{main}) \neq \operatorname{sign}(u_{main}) \\ \boxed{[u_{bot},\ v_{bot}]}, & \text{else if } \operatorname{sign}(u_{bot}) = \operatorname{sign}(u_{main}) \\ [u_{main},\ 0.0], & \text{otherwise} \end{cases} \\
[u_2,\ v_2] &= \begin{cases} [u_{main},\ v_{top}], & \text{if } \operatorname{sign}(v_{top}/u_{main}) = \operatorname{sign}(u_{main}) \\ \boxed{[u_{top},\ v_{top}]}, & \text{else if } \operatorname{sign}(u_{top}) = \operatorname{sign}(u_{main}) \\ [u_{main},\ 0.0], & \text{otherwise} \end{cases}
\end{aligned}
\qquad (14)
$$

Donating region construction in other directions is approached similarly. The scheme, like EMFPA [34], can be straightforwardly extended to 3D.



(a) Example of face-centered velocities needed.

(b) Coordinates defining the polygon; additional point is black.

Fig. 12. Definition of DQA scheme; in case $u_1 = u_{main}$, $u_2 = u_{top}$, $v_1 = v_{bot}$, and $v_2 = v_{top}$. Donating region is given by —. Slopes are given by —. The volume correction with slope ϕ is illustrated for the approach of Owkes and Desjardins [34]. The cell face area δA is equal to δy.

4.3.3. Volume correction

The donating regions in Fig. 12 do not initially satisfy the volume constraint. Many volume correction methods are available [23,28,34,8]. The correction method of Owkes and Desjardins [34] is adopted by adding an extra control point. This extra point ensures that the volume of the donating region satisfies the volume constraint. Conforming donating regions are found when applying this correction method.

The distance value ϵ, illustrated in Fig. 12b, should result in a donating region with a volume of $u_{main}dt\delta A$ (the volume constraint). The additional point that satisfies the volume constraint is found with u_{avg} and v_{avg} being equal to $\frac{1}{2}(u_1 + u_2)$ and $\frac{1}{2}(v_1 + v_2)$, respectively. The distance is found from

$$\epsilon = dt\frac{-u_2v_1dt + u_1v_2dt + 2u_{main}\delta A - (u_1 + u_2)\delta A}{(v_2 - v_1)dt + (u_1 - u_2)\phi dt - \delta A}, \tag{15}$$

in which δA is equal to δy for Fig. 12.

The slope value ϕ can be determined in many ways. Evaluation of the effect of ϕ on the accuracy is needed. Owkes and Desjardins [34] uses the normal vector of the original linear edge as the direction for creating the additional point. This method is illustrated as an example in Fig. 12. Other ways to determine the slope value ϕ are

$$\begin{aligned}
&\phi = \frac{v_2 - v_1}{u_2 - u_1} && \text{(Owkes and Desjardins [34]),}\\
&\phi = 0 && \text{(this work, referred to as ‘Zero slope’ in Table 7),}\\
&\phi = \frac{v_2 + v_1}{u_2 + u_1} && \text{(this work, referred to as ‘Slope in mean flux direction’ in Table 7).}
\end{aligned} \tag{16}$$

The evaluation of the slope value is done in the next section. The slope value of Owkes and Desjardins [34] is used for comparison with other computational methods.

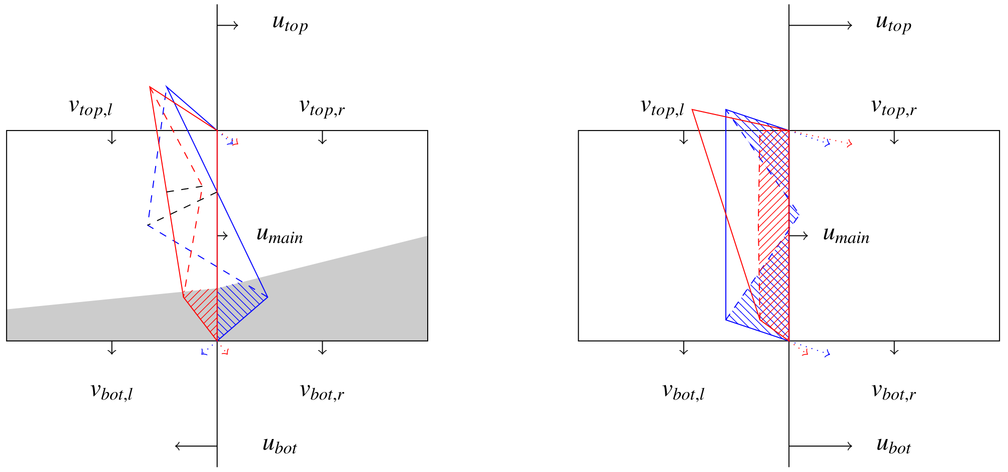
The extra correcting point can result in a non-convex donating flux region. When the shape is non-convex, the donating region is split into multiple convex triangles. This increases the computational costs as the triangles are dealt with separately, but it solves the problem of determining the VOF flux of a non-convex region. The intersection points of the donating region with the BLIC interface are expensive to determine for non-convex shapes. The computational costs are analyzed in one of the following sections.

4.3.4. Reducing order with DDR scheme

Owkes and Desjardins [34] showed that their advection scheme is unconditionally stable. We want to discuss what constraints are needed to keep the two-phase solver described in Sec. 6 stable. When the volume correction is large, the additional point can lead to an overlap of the donating region with neighboring regions. Or it can result in a negative donating region. The DQA method can easily switch to a lower-order DDR scheme when this happens. The difference between DDR and DQA, illustrated in Fig. 13b, is that the donating polygon never intersects a cell face; material cannot enter and exit a cell in one time step. The DDR scheme is found when the boxed step in Eq. (14) is omitted. DDR is robust and conservative, but increases the diffusivity of the interface.

The criteria to switch from DQA back to DDR are

1. The correcting point should not lead to overlap with neighboring polygons or the polygon itself to prevent mass changes.



(a) Shear flow: presented scheme prevents flux of opposite sign than velocity.

(b) In a wake: presented scheme uses DDR to prevent negative region.

Fig. 13. Difference between EMFPA (🟦) and the DQA scheme (🟥) for two different cases. The dashed line is the donating region after volume correction. The solid line is the original donating region. The hatched area is the flux δC_f. Liquid is indicated by ⬜. The air is white. Note that the hatched area for the EMFPA scheme can be larger than the volume constraint $u_{main}\delta t\delta A$, but DQA is not.

2. The correcting point should not be at the opposite side of the cell face compared to the other points of the donating region (later referred to as 'negative donating region') because it can induce instability of the two-phase solver.

When these criteria are not met, one of the provided constraints is violated, and the DDR method is used for that particular cell face. Note that the correction method we use can also cause mass issues due to overlap when combined with EMFPA. Alternative correction methods will be a research topic of ours for the near future.

The chance that switching back to DDR occurs is negligible for cases with imposed velocity fields. However, for the case in Fig. 1, with large interface (free-surface) deformations, the switch back provides robustness, in the sense that negative donating regions that can induce instabilities of the two-phase solver are prevented. For a case similar to the one shown in Fig. 1, using a relatively coarse grid resolution (roughly 100 cells in the width and height of the domain) and a Courant number of 0.50, the number of times the algorithm switches back to DDR is smaller than 1% of the total number of time steps. Switching back especially occurs when water and air near the interface have large velocities in opposite direction. At those moments, the correction to meet the volume constraint can lead to overlap or the situation discussed in the Sec. 4.3.5, so that switching back to DDR is necessary to prevent instability. The percentage of time steps, in which the algorithm switches back to DDR, decreases when lowering the Courant number. Grid refinement with a factor of two reduces the number of times the algorithm switches back with a power of two. We consider the chance of the algorithm switching back small. The effect of switching back to DDR on the accuracy is evaluated in Sec. 5.2.

4.3.5. Difference EMFPA and DQA for two examples

In Fig. 13 the full donating region with EMFPA and DQA is compared for two cases. These two cases are considered to be the most different. The length of the velocity vectors is to scale. The flux size is illustrated by the hatched area. The volume correction method of Owkes and Desjardins [34] is used with both EMFPA and DQA.

In Fig. 13a, a shear flow is considered, an example of EMFPA having a flux of opposite sign than the fluxing face-centered velocity. The DQA scheme prevents this by using the face-centered velocity based on the position of the linear edges of the donating region. In Fig. 13b, an example of DQA switching to DDR is given. A negative donating region is created due to the volume correction, which is allowed by EMFPA but not by the DQA scheme.

The authors want to point out that, even though the chance is small, EMFPA can result in a negative donating region that gives a flux larger than the volume constraint. In other words, the size of the hatched area can be larger than the sum of the receiving and the donating region. As a requirement, DQA strictly enforces volume conservation.

5. Fundamental transport cases

5.1. Zalesak slotted disk rotation

The first benchmark case with advection of a fluid structure to be discussed is the Zalesak slotted disk [57]. The Zalesak slotted disk is a well-known benchmark for volume-tracking methods. The disk is rotated for one complete anti-clockwise revolution in a velocity field with uniform vorticity

$$u = -\Omega(y - y_0), v = \Omega(x - x_0), \tag{17}$$

Table 3
Error E^n for slotted disk rotation ($n = 1$). Other schemes are compared with the presented ones. Youngs is an abbreviation for the Parker and Youngs method. A Courant number of 0.25 is used.

Algorithm	Error (E^1)
SLIC [32]	8.38e-2
Hirt-Nichols [17]	9.62e-2
CICSAM [51]	2.02e-2
High-resolution VOF - unstructured grid [19]	1.01e-2
Puckett - stream [15]	1.00e-2
Puckett - DDR [16]	9.73e-3
Puckett - EMFPA [25]	9.73e-3
Youngs - stream [15]	1.07e-2
Youngs - DDR [16]	1.56e-2
Youngs - EMFPA [25]	1.06e-2
QQ - THINC [56]	1.42e-2
SIR - EMFPA [25]	8.74e-3
ACLSVOF uniform triangular grid [3]	7.19e-3
ACLSVOF adaptive triangular grid [3]	1.25e-2
Linear least square fit - EI-LE [48]	9.42e-3
Quadratic fit - EI-LE [48]	5.47e-3
Quadratic fit + continuity - EI-LE [48]	4.16e-3
QUASI - EMFPA [12]	2.69e-3
Youngs - DQA (this work)	1.23e-2
Youngs - COSMIC (this work)	1.25e-2
MYC - DQA (this work)	1.06e-2
MYC - COSMIC (this work)	1.09e-2
Youngs BLIC - DQA (this work)	7.76e-3
MYC BLIC - COSMIC (this work)	6.16e-3
MYC BLIC - DQA (this work)	5.85e-3

Table 4
Error E^1 for slotted disk rotation ($n = 1$) for different grid resolutions using MYC. A Courant number of 0.25 is used. The order of convergence is in between parentheses.

Algorithm	50×50	100×100	200×200
MYC - DQA (this work)	1.39e-1	3.45e-2 (2.01)	1.06e-2 (1.70)
MYC - COSMIC (this work)	1.41e-1	3.47e-2 (2.02)	1.09e-2 (1.67)
MYC BLIC - COSMIC (this work)	1.42e-1	2.08e-2 (2.77)	6.16e-3 (1.76)
MYC BLIC - DQA (this work)	1.38e-1	2.30e-2 (2.58)	5.85e-3 (1.98)

where $\mathbf{u} = [\mathrm{u}, \mathrm{v}]^T$ is the 2D velocity field, $\mathbf{x} = [\mathrm{x}, \mathrm{y}]^T$ the spatial position, and the position with subscript '0' the axis of rotation. The domain is four by four with the center of rotation at coordinate (2.00, 2.00). The constant angular velocity Ω is 0.50. The disk has a radius of 0.50 and is initially positioned at coordinate (2.00, 2.75). The slot has a width of 0.12 and stops in the center of the disk. The grid resolution used for the benchmark is 200×200. These parameters are adopted from the work of Rudman [44] and Diwakar et al. [12]. The error is defined as

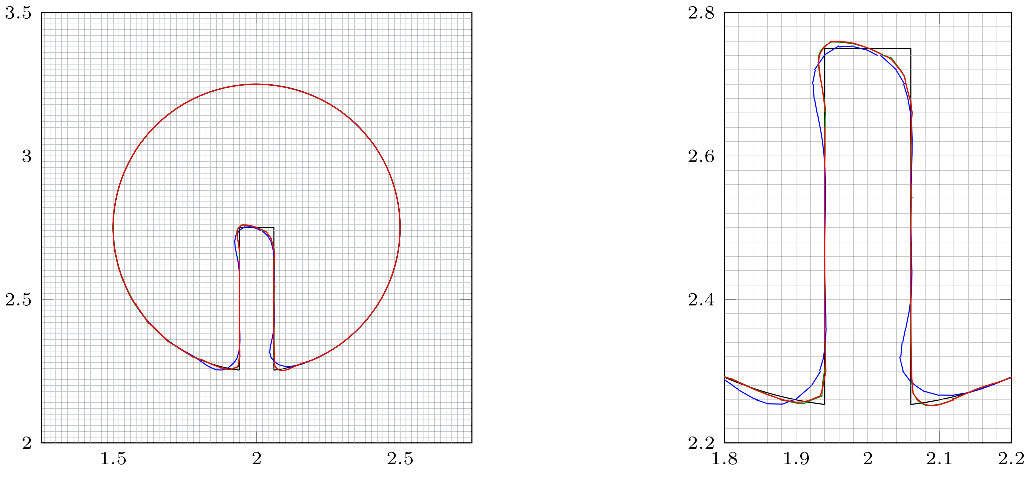
$$E^n = \frac{\sum_{i,j} |C^n_{f,i,j} - C^0_{f,i,j}|}{\sum_{i,j} C^0_{f,i,j}}, \tag{18}$$

in which n represents the number of revolutions of the slotted disk.

In Table 3, the errors of Eq. (18) for various reconstruction and advection schemes after one rotation of the disk are given with a reference to the article from which it was taken. The results of the newly presented reconstruction and advection scheme are compared with the state-of-the-art schemes in Table 3. For an equal comparison, a Courant number of 0.25 is used based on the maximum velocity [25]. The number of times DQA switches back to DDR, as discussed in Sec. 4.3.4, is 0%. López et al. [25] commented that the largest error for this case is typically found near the slot and Scardovelli and Zaleski [48] found that PLIC methods normally smooth out parts of the interface with high curvature, like the slot.

The error E^1 after one rotation of the disk and the order of convergence obtained when using the reconstruction and advection methods in this article, is shown in Table 4. For higher grid resolutions, the error can potentially be governed by the number of time steps [48]. Decreasing the Courant number results in higher errors E^1, because it increases the number of reconstruction and advection steps. Where a Courant number of 1.00 with a grid of 200×200 (MYC BLIC - DQA) results in an error of 4.78e-3, a Courant number of 0.50 gives 5.30e-3, and 0.25 gives 5.85e-3. This effect is also mentioned by Diwakar et al. [12], showing that it does not happen with their QUASI scheme.

In Fig. 14, the application of BLIC showed that the corners of the slot are better represented than with PLIC. A more accurate PLIC method like MYC reduces the error compared to Paker and Youngs, also in combination with BLIC. The unsplit DQA advection scheme shows a slight reduction of the error compared to COSMIC and a similar error as EMFPA. The combination of DQA and MYC



(a) Complete slotted disk.

(b) Zoom in on slot.

Fig. 14. The interfaces of the Zalesak' slotted disk for one turn with 200×200 (interface based on reconstruction line). MYC BLIC - DQA is indicated by 🟥. MYC BLIC - COSMIC is indicated by 🟩. MYC - COSMIC is indicated by 🟦. Reconstruction of interface is given for a number of rotations n = 1.

BLIC obtains similar accuracy for the slotted disk as higher-order methods like the Quadratic fit methods [48]. The computational costs are compared in Section 5.3.

5.2. Rider–Kothe single vortex

A simple translation or solid body rotation of a fluid structure like the slotted disk by itself is not enough for testing interface modeling methods. No final judgment can be made because this kind of case is not realistic and more suitable for benchmarking and debugging [44]. A more physical case for interface modeling is the reversed single vortex field case by Bell et al. [5], modified by Rider and Kothe [43].

A solenoidal velocity distribution is imposed for the reversed single vortex field in a domain of one by one. The velocity field is determined by the stream function field

$$\Psi(x,y,t) = \frac{1}{\pi}\sin^2(\pi x)\sin^2(\pi y)\cos\left(\frac{\pi t}{T}\right), \tag{19}$$

with T the period of the reversed motion of the vortex. An initial circle with a radius of 0.15 is positioned at (0.50, 0.75) and deformed until $t = T/2$. Hereafter, the deformed structure morphs back to its initial shape until $t = T$. The error E^T is determined by the difference between the initial and final volume fraction field multiplied by the grid spacing [12]

$$E^T = \sum_{i,j} \delta x_i \delta y_j |C^T_{f,i,j} - C^0_{f,i,j}|. \tag{20}$$

According to Scardovelli and Zaleski [48] the main error for the single vortex, as well as for the slotted disk, is caused by the number of reconstructions rather than the advection algorithm.

Period T is chosen to be 2.0. Table 5 shows the errors E^T for single vortex simulations with the advection and reconstruction methods presented in this article. The methods described here are compared to methods from the existing literature. Three different grid resolutions 32×32, 64×64, and 128×128 are considered. A Courant number of 0.95 is used, the Courant number at each time level being based on the maximum velocity component in the domain at that time level. In the existing methods, a Courant number of 1.00 was used, but for that number using DQA can lead to mass losses due to overlap in donating polygons caused by the correction point (see Sec. 4.3.4). This is elaborated upon below.

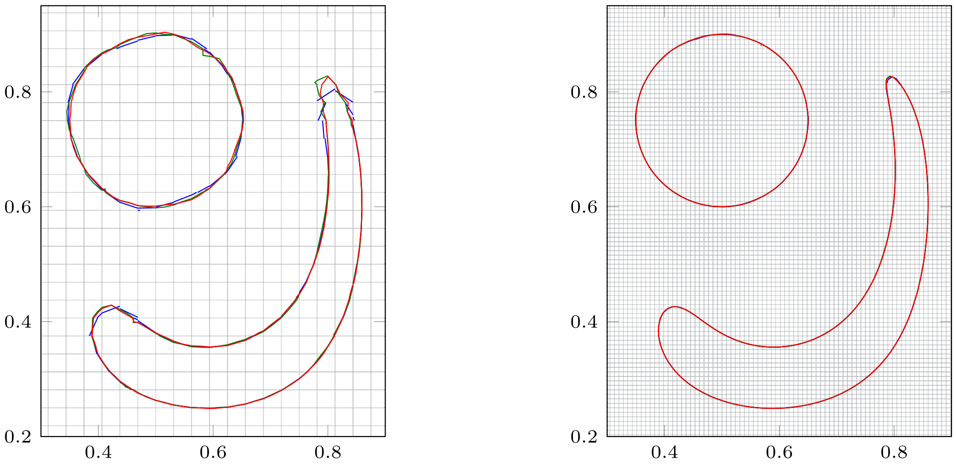
MYC BLIC - DQA has a lower error compared to other available methods. The method competes in terms of accuracy with higher-order methods like PCIC (piecewise circular arcs) and QUASI (quadratic splines). The choice for a more-accurate PLIC method, like MYC compared to Parker and Youngs, improves the result for reversed single vortex. For a period T of 2.0 and grid 128×128, MYC BLIC - DQA has a similar error as PCIC - EMFPA. Referring back to Sec. 4.3.1, the accuracy of DQA is similar to EMFPA when comparing Youngs - DQA with Youngs - EMFPA. These results show that the use of a non-sloped linear edge has negligible influence on the accuracy for this case.

Fig. 15 shows the difference in interface at $t = T/2$ and $t = T$ for six of the single vortex simulations in Table 5, comparing methods on a grid of 32×32 and on a grid of 128×128. The interface obtained with MYC BLIC - DQA on the coarser grid comes closest to the interface obtained on the finer grid. On the finer grid the difference between methods is not visible any more. Switching back from BLiC to PLIC was not necessary for the simulations used to make Fig. 15; they all met the constraints defined in Sec. 2.6.

The effect of the Courant restriction on the accuracy of the simulations is evaluated for the methods discussed above. The error E^T, obtained using grid 128×128, is shown in Table 6 for values of the Courant number 0.1, 0.5, 0.95 and 1.00. Similar to the

Table 5
Error E^T at different grid resolutions for the reversed single vortex field with $T = 2.00$ and Courant number 0.95. Methods presented in this article are compared with methods from the existing literature, for which a Courant number of 1.00 was used. Youngs is an abbreviation of the Parker and Youngs method. The order of convergence is in between parentheses.

Algorithm	32x32	64x64	128x128
Hirt & Nichols + LHF [13]	1.01e-2	5.25e-3 (0.94)	2.47e-3 (1.09)
QQ - THINC [56]	6.70e-2	1.52e-2 (2.16)	3.06e-3 (2.29)
Puckett - stream [15]	2.37e-3	5.65e-4 (2.07)	1.32e-4 (2.10)
Puckett - Rider & Kothe [43]	2.36e-3	5.85e-4 (2.01)	1.31e-4 (2.16)
Puckett - EMFPA [25]	2.14e-3	5.39e-4 (1.99)	1.29e-4 (2.06)
Youngs - stream [15]	2.49e-3	7.06e-4 (1.82)	2.23e-4 (1.66)
Youngs - EMFPA [25]	2.31e-3	6.89e-4 (1.75)	2.25e-4 (1.61)
LSG - COSMIC [13]	2.74e-3	7.01e-4 (1.97)	1.96e-4 (1.84)
LSG - EI-LE [13]	2.70e-3	6.93e-4 (1.96)	1.89e-4 (1.87)
ELVIRA - COSMIC [13]	2.55e-3	6.50e-4 (1.97)	1.51e-4 (2.11)
ELVIRA - EI-LE [13]	2.54e-3	6.47e-4 (1.97)	1.45e-4 (2.16)
CVTNA - PCFSC [24]	2.34e-3	5.38e-4 (2.12)	1.31e-4 (2.04)
Linear least square fit - EI-LE [48]	1.75e-3	4.66e-4 (1.91)	1.02e-4 (2.19)
Quadratic least square fit - EI-LE [48]	1.88e-3	4.42e-4 (2.09)	9.36e-5 (2.24)
Quadratic least square fit + continuity - EI-LE [48]	1.09e-3	2.80e-4 (1.96)	5.72e-5 (2.29)
Mixed markers and VOF method [3]	1.00e-3	2.69e-4 (1.89)	5.47e-5 (2.30)
SIR - EMFPA [25]	8.62e-4	2.37e-4 (1.86)	5.62e-5 (2.08)
QUASI - EMFPA [12]	6.65e-4	1.57e-4 (2.08)	4.33e-5 (1.86)
PCIC - EMFPA [26]	5.61e-4	1.46e-4 (1.94)	4.17e-5 (1.81)
Youngs - DQA (this work)	2.14e-3	5.15e-4 (2.05)	1.66e-4 (1.63)
MYC - DQA (this work)	2.09e-3	4.32e-4 (2.27)	1.08e-4 (2.00)
MYC - COSMIC (this work)	2.74e-3	6.96e-4 (1.98)	1.80e-4 (1.95)
Youngs BLIC - DQA (this work)	1.23e-3	1.88e-4 (2.71)	6.25e-5 (1.60)
MYC BLIC - COSMIC (this work)	1.68e-3	4.04e-4 (2.06)	1.40e-4 (1.53)
MYC BLIC - DQA (this work)	1.21e-3	2.04e-4 (2.57)	4.42e-5 (2.20)



(a) 32x32.

(b) 128x128.

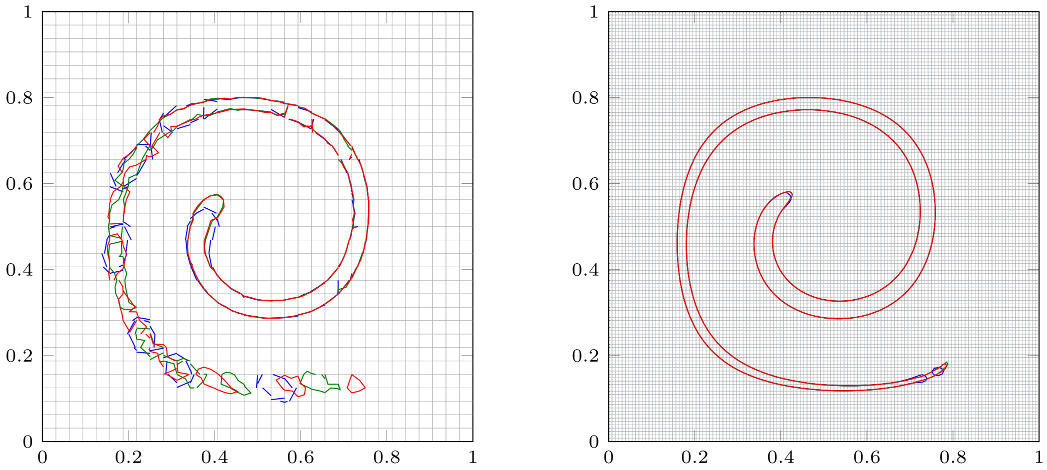
Fig. 15. The interfaces for the reversed single vortex from Rider and Kothe [43] with $T = 2.00$ and Courant number 0.95 (interface based on reconstruction line). MYC BLIC - DQA is indicated by 🟥. MYC BLIC - COSMIC is indicated by 🟩. MYC - COSMIC is indicated by 🟦.

slotted disk, a decrease in Courant number does not lower the error when using DQA. The larger error for lower Courant numbers is due to the increased number of reconstructions. When using COSMIC the error does decrease when going from Courant number 1 to Courant number 0.95, and from 0.95 to 0.5, but that due to the fact that COSMIC, with the reconstruction step, is not mass conserving for Courant numbers above 0.5 [54]. Note again that DQA is not mass conserving either at a Courant number of 1.0 because of the overlap in donating regions. The mass losses for DQA depend on the grid size. The mass losses for MYC BLIC - DQA with Courant number 1.00 are 0.18%, 0.62% and 1.50% respectively, for grids 128×128, 64×64 and 32×32. For grids 128×128 and 64×64 DQA switched back to DDR in 0.00% of all time steps, for grid 32×32 DQA switched back in 0.50% of all time steps.

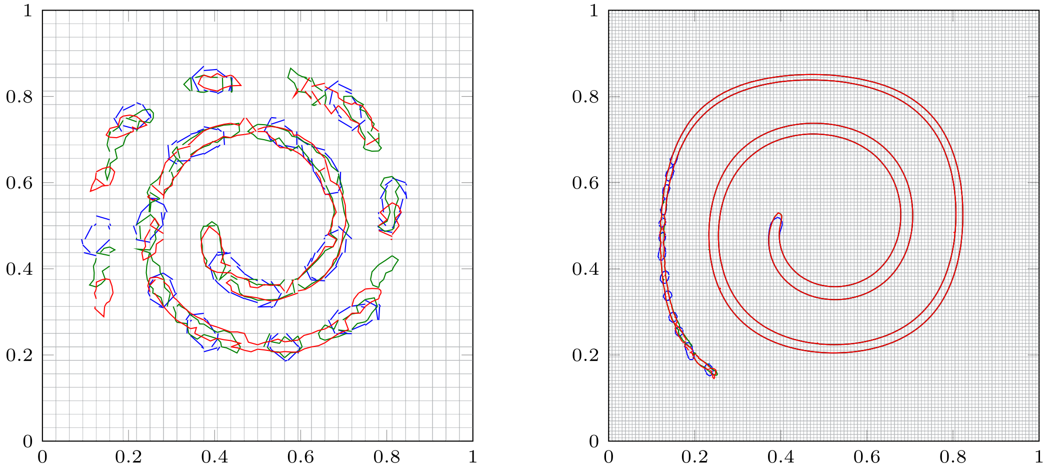
In Fig. 16, two different grid resolutions and time instances of the interface for a non-reversed single vortex are illustrated. The BLIC method keeps the fluid body better together than PLIC. The number of detached droplets is reduced when using BLIC. Sharper parts of the vortex, especially the front, and the tail are represented better with BLIC than with PLIC. With BLIC the interface could not always be reconstructed for grid resolution 32×32 and switched back to PLIC due to the correction point of BLIC being outside

Table 6
Error E^T at grid resolution 128×128 for different Courant numbers. The mass changes are given as a percentage of the initial mass in between parentheses.

Courant number	0.1	0.5	0.95	1.0
Youngs - DQA (this work)	3.06e-4 (0.00%)	2.66e-4 (0.00%)	1.66e-4 (0.00%)	3.30 (0.04%)
MYC - DQA (this work)	1.35e-4 (0.00%)	1.21e-4 (0.00%)	1.08e-4 (0.00%)	2.49e-4 (0.18%)
MYC - COSMIC (this work)	1.39-4 (0.00%)	1.30e-4 (0.00%)	1.80e-4 (0.04%)	2.47e-4 (0.07%)
Youngs BLIC - DQA (this work)	2.23e-4 (0.00%)	8.22e-05 (0.00%)	6.25e-5 (0.00%)	2.22e-4 (0.18%)
MYC BLIC - COSMIC (this work)	1.53e-4 (0.00%)	6.36e-05 (0.00%)	1.40e-4 (0.04%)	1.98e-4 (0.08%)
MYC BLIC - DQA (this work)	1.34e-4 (0.00%)	4.99e-5 (0.00%)	4.42e-5 (0.00%)	2.15e-4 (0.18%)



(a) 32x32 at t = 2.00. (b) 128x128 at t = 2.00.



(c) 32x32 at t = 3.00. (d) 128x128 at t = 3.00.

Fig. 16. The interfaces for a non-reversed single vortex with different reconstruction and transport schemes at two time instances (interface based on reconstruction line). MYC BLIC - DQA is indicated by 🟥. MYC BLIC - COSMIC is indicated by 🟩. MYC - COSMIC is indicated by 🟦.

of a cell. The number of time steps BLIC switched to PLIC for that grid until t = 3.0 is 8.9% with MYC BLIC - DQA and 13.7% with MYC BLIC - COSMIC. Even when having to switch back to PLIC this many time, there is still an advantage of using BLIC because its error E^T is smaller than for plain PLIC. For grid 128×128, BLIC switched back to PLIC in fewer than 0.3% of all time steps.

The effect of determining the slope value ϕ for the DQA scheme is evaluated for the three approaches provided in Eq. (16). The results are given in Table 7 for the combination MYC BLIC - DQA. The volume correction of Owkes and Desjardins [34] has the best results for this case, even though they mentioned that the effect of the volume corrections is probably small.

Table 7
Error E^T for different ϕ in Sec. 4.3.3 with MYC BLIC - DQA for the reversed single vortex field with $T = 2$ and Courant number is 0.95.

ϕ	32x32	64x64	128x128
Owkes and Desjardins [34]	1.03e-3	2.18e-4	4.42e-5
Zero slope	1.11e-3	2.22e-4	4.69e-5
Slope in mean flux direction	1.27e-3	2.52e-4	6.03e-5

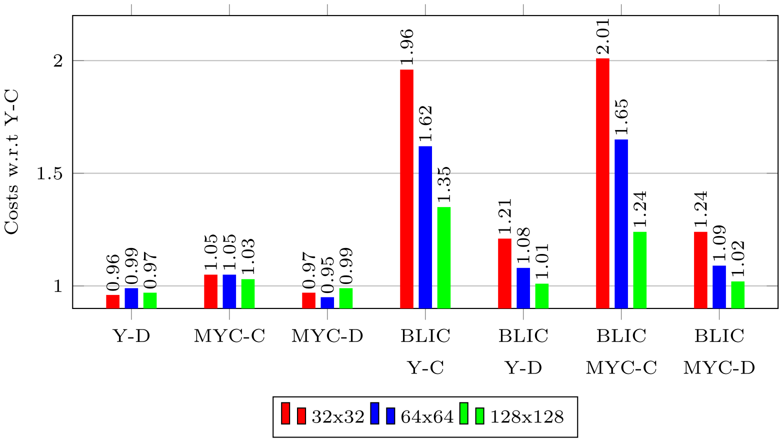


Fig. 17. Comparison of computational costs normalized with the costs of Parker and Youngs - COSMIC (Y-C) for the single vortex case based on 384 timesteps for three different grid resolutions. (D) is DQA. (C) is COSMIC. (Y) is Parker and Youngs' method. (MYC) is Mixed Youngs and Centered method.

5.3. Computational costs

The code is written in Python and compiled using the Numba package [21]. The compiled code is run on a 12-core stand-alone machine. In Fig. 17, the computational costs of the reconstruction and advection methods are compared for the reversed single vortex, normalized with the costs of the combination Parker and Youngs - COSMIC (Y-C). The costs at three different grid resolutions are compared. The average computational time is based on 384 time steps. The sensitivity of the costs is tested by repeating the same simulation on the same machine. After repeating five times, the cost variation is between 1% and 3%.

The application of the MYC reconstruction method over the Parker and Youngs' method is neglectable in computational costs. BLIC has between 20% and 1% more computational costs than Parker and Youngs. Similarly, as the results of Diwakar et al. [12], the costs relatively decrease by increasing the grid resolution.

The costs of the advection scheme become dominant for a higher grid resolution. The presented unsplit advection scheme (DQA) results in an average of around 5% less computational costs than the splitting scheme (COSMIC) when using a PLIC method. Similarly, the extra costs of using BLIC with the COSMIC scheme compared to the DQA scheme are even more dominant. The relative increase in computational costs is because of the additional intermediate reconstruction steps for COSMIC. The DQA scheme does not make use of intermediate reconstruction steps.

Compared to other reconstruction methods in the existing literature of which the costs are documented, BLIC appears to have fairly low costs. The quadratic least-square reconstruction algorithm with continuity equation seems twice as costly as Parker and Youngs [48]. The QUASI method, showing high accuracy in representing the interface, is around 30 times higher in costs and ELVIRA 10 times compared to Parker and Youngs [12]. The computational costs of the SIR scheme are comparable with BLIC, around 1.1 times higher in costs than Parker and Youngs [25]. PCIC with continuity correction is around 15 times higher in costs compared to Parker and Youngs [26]. Note that it is difficult to compare with accuracy to the reported computational costs of methods in the existing literature. Not for every method cost indications are given. And for ELVIRA, for instance, reports [12,25,26] appeared inconsistent. Costs, moreover, depend on the efficiency of coding and the hardware that was used. For these reasons, the comparison in Fig. 17 can potentially be valuable, because all methods were implemented in the same code and run on the same machine.

6. Consistent mass-momentum transport

'Consistent' is the name used for two-phase methods with staggered variables that make sure that mass and momentum fluxes are determined in the same way, carefully considering that the control volumes for mass and momentum are also staggered with respect to each other [45,59,52,2,35]. Consistent modeling prevents momentum losses and prevents spurious velocities from being generated near the interface between fluids, that – in some cases – lead to instability.

Consistency is analyzed for the transport and reconstruction methods described in this article, in combination with the two-phase method of van der Eijk and Wellens [52]. The results of van der Eijk and Wellens [52] show temporal plots of momentum losses

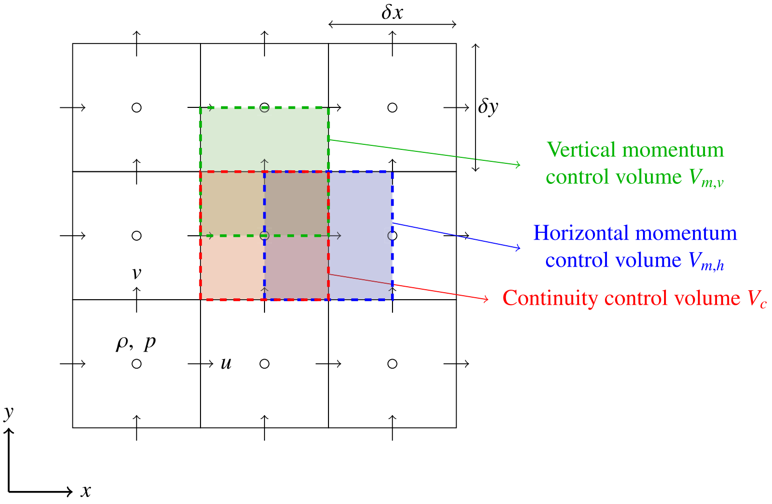


Fig. 18. Standard MAC configuration of variables (staggered); pressure p is defined in the cell center (○), the horizontal velocity u field is sampled on the vertical faces (→), the vertical velocity v is sampled on the horizontal faces (↑). The overlap of the continuity control volume (—), with a vertical momentum control volume (—), and with a horizontal momentum control volume (—) is shown.

which are nearly zero. When spurious velocities do occur and when they can be explained in terms of the discretization of mass and momentum, we call them 'inconsistencies'. The discretization of the density is key for explaining the inconsistencies.

6.1. Brief overview of two-phase solver

A brief overview of the two-phase solver is needed to explain from where inconsistencies originate. The governing equations for two-phase flow are formulated as if it is a single homogeneous mixture [31]. The motion of the mixture is described in terms of a single velocity and a single pressure field, solving the continuity equation and the momentum equation, together with the advection equation for the interface. Viscous, capillary, and gravitational effects are neglected for the purpose of this article.

A 2D Cartesian grid of cells is adopted. According to the staggered MAC configuration of variables, velocities ($\mathbf{u} = [u,\ v]^T$) are defined in grid cell faces, and scalar variables, the pressures (p), volume fractions (C_f), and densities (ρ), are defined in grid cell centers. How continuity control volumes (V_c) and momentum control volumes (V_m), different for each velocity direction, are defined with respect to the grid cells is shown in Fig. 18.

Considering the time discrete version of the continuity, momentum and advection equation, using Forward Euler in time for brevity of notation, the equations are integrated over control volumes to obtain the weak form. The weak form is the basis of the discretization in space. First, the fluids are advected using Eq. (9), repeated here as Eq. (21) to be complete

$$\int_{V_c} \frac{C_f^{n+1} - C_f^n}{\delta t}\, dV = -\oint_{S_c} \underbrace{\left(\mathbf{u}^n C_f^n\right) \cdot \mathbf{n}}_{consistent}\, dS. \tag{21}$$

Here, S_c refers to boundary of the control volume V_c shown in Fig. 18. Transport and reconstruction of the interface are solved for continuity control volumes with labels S and C (refer to the labeling system in Sec. 2.1). Because the fluids are considered incompressible, the continuity equation reduces to

$$\oint_{S_c} \mathbf{u}^{n+1} \cdot \mathbf{n}\, dS = 0. \tag{22}$$

The time discrete momentum equation is solved in steps, the main step being

$$\int_{V_m} \frac{\rho^{n+1}\mathbf{u}^{n+1} - \rho^{n+1}\bar{\mathbf{u}}}{\delta t}\, dV = -\oint_{S_m} p^{n+1}\mathbf{n}\, dS, \tag{23}$$

in which auxiliary vector field $\bar{\mathbf{u}}$ in Eq. (23), without viscous, capillary and gravitational effects, is solved from

$$\int_{V_m} \frac{\bar{\rho}\bar{\mathbf{u}} - \rho^n\mathbf{u}^n}{\delta t}\, dV = -\oint_{S_m} \underbrace{\rho^*\mathbf{u}^n \left(\mathbf{u}^n \cdot \mathbf{n}\right)}_{consistent}\, dS, \tag{24}$$

whereas auxiliary density $\bar{\rho}$ is solved with a temporary continuity equation that is integrated over momentum control volumes

$$\int_{V_m} \frac{\bar{\rho}-\rho^n}{\delta t} dV = - \oint_{S_m} \underbrace{\left(\mathbf{u}^n \rho^*\right) \cdot \mathbf{n}}_{consistent} dS. \tag{25}$$

Here, ρ^* is a density for which a consistent discretization in space is used. A *consistent* space discretization for momentum and mass is obtained when the advective terms indicated by the word 'consistent' are treated the same with continuity control volumes and with momentum control volumes, and when the spatial discretization of ρ^n and ρ^{n+1} is the same on continuity control volumes and on momentum control volumes. The density ρ^{n+1} is solved differently from $\bar{\rho}$.

The new density ρ^{n+1} is associated with the fluid volume in a momentum control volume. The fluid volume $\mathcal{A}$ in a momentum control volume is calculated with volume fraction field C_f^{n+1} and the fluid-interface orientation: $\rho = \mathcal{A}\rho_f + (1-\mathcal{A})\rho_{1-f}$, with ρ_f the constant density of the fluid that is advected (typically the heavier of the two fluids), and ρ_{1-f} the constant density of the other fluid.

The auxiliary density $\bar{\rho}$ is calculated using ρ^*. The value of ρ^* depends on the size of the VOF flux δC_f at the boundary of the momentum control volume, so that consistency is obtained between the mass flux and the momentum flux. The momentum flux $\rho^*\mathbf{u}$ is discretized using

$$\rho^*\mathbf{u} = \left(C\rho_f + (1-C)\rho_{1-f}\right)\mathbf{u}, \tag{26}$$

in which C is a scalar fraction indicating the VOF flux normalized by the size of the donating region [59]. This discretization ensures consistency between mass and momentum flux.

$$C = \frac{|\delta C_f|}{|\mathbf{u}|\delta t \delta A}. \tag{27}$$

The density ρ^* is above zero by definition and, therefore, absolute values of flux and velocity are used.

6.2. Application of an advection scheme to the solver

Besides calculating the VOF fluxes around the faces of the continuity control volume for Eq. (21), additional VOF fluxes need to be calculated around the faces of the momentum control volumes for Eqs. (25) and (24) to find ρ^*. Owkes and Desjardins [35] dealt with this in a similar way. The VOF flux positions for all control volumes are illustrated in Fig. 19, including examples of donating regions for a flow that is directed to the right and to below. Fig. 19 shows one of the nine continuity control volumes in Fig. 18. The continuity control volume is subdivided in four quadrants so that fluxes of mass and momentum can be matched in continuity control volumes, horizontal and vertical control volumes. An arrow is drawn at every face of a quadrant, representing a VOF flux that requires computation. Per 2D grid cell, not four, but twelve VOF fluxes are calculated.

The thick-lined arrows in Fig. 19 are the cell-face-centered velocities solved from the system in Eq. (23). These velocities are assumed constant along a cell face. Therefore, the unknown sub-velocities for the faces of the quadrants are equal to the solved face-centered velocity, for instance for the velocities on the right-hand side of the cell: $\mathrm{u}_{right} = u_{right,1} = u_{right,2}$. The same applies to u_{left}, v_{bot}, and v_{top}. The velocities in the middle of a grid cell (u_{mid} and v_{mid}) are not the same as the velocities in the cell faces. An averaging procedure is used to find these velocities. They are found from $v_{mid} = \frac{1}{2}\left(v_{top} + v_{bot}\right)$ and $u_{mid} = \frac{1}{2}\left(u_{left} + u_{right}\right)$. The sum of the two fluxes at the position of $v_{mid,1}$ and $v_{mid,2}$ in Fig. 19 results in the total VOF flux through the bottom face of the vertical momentum control volume needed for ρ^*. The same is true for the horizontal control volume fluxes at the position of u_{mid}.

The sub-velocities in Fig. 19 are used to reconstruct the donating region. The linear edges of the donating regions (black dashed lines in Fig. 19 representing the (sub-)donating regions) are calculated as in Sec. 4 using the sub-velocities of the cell quadrants. The sum of two donating regions along a cell face should be equal to the volume constraint, where 'volume constraint' means the total amount of fluid to be fluxed. For example, the volume constraint for the donating regions at the face indicated with $right$ is $u_{right}\delta y = \frac{1}{2}u_{right,1}\delta y + \frac{1}{2}u_{right,2}\delta y$.

Similarities with Owkes and Desjardins [35] are found in dividing the cell into sub-cells. Even though they make use of vertex-centered velocities and adopt the method of Rudman [45], reconstructing the interface in sub-cells to create consistency and a conservative scheme, a difference is found in the way of correcting the donating region such that they comply with the volume constraint. In our case, for every sub-velocity a donating region with a correction is defined, resulting in a less complex system but more correction calculations.

6.3. Inconsistency

A method is considered to be fully consistent when the auxiliary density $\bar{\rho}$ is the same as ρ^{n+1}, see Eqs. (23) and (24). The method in Rudman [45] is fully consistent with both transport and reconstruction of the interface taking place in cell quadrants, but also computationally expensive. Reconstruction handled in complete cells, instead of in cell quadrants, can lead to small inconsistencies. The overlap of the interface reconstruction in the continuity and momentum control volume is the reason for this. The problem is that $\bar{\rho}$ does not account for the reconstruction of the fraction field in a complete cell (or continuity control volume), while ρ^{n+1} does. Collocated approaches, like in Bussmann et al. [6] and in Rudman [45], do not have this issue.

Zuzio et al. [59] use grid cells divided up in quadrants to determine mass fluxes, but reconstruct the interface in complete grid cells. In theory, reconstructing the interface in this way can lead to inconsistency because during reconstruction fluid can be moved

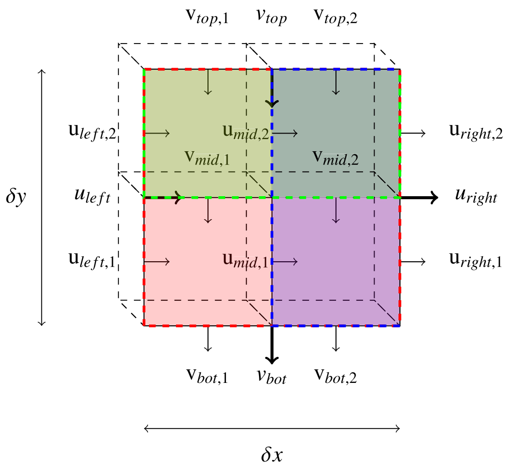
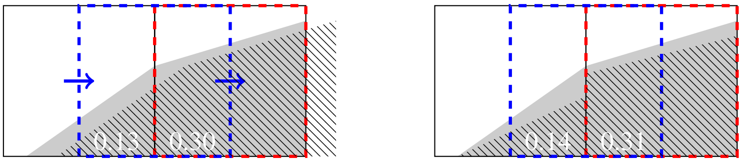


Fig. 19. Subdivision of a grid cell in quadrants. DQA donating regions needed for a consistent discretization with a staggered arrangement are also indicated; solid black is the border of the grid cell and dashed black borders are the donating regions based on the sub-velocities. The u_{main} velocity for every donating region is given. Cell-face-centered velocities are thick-lined. The other arrows are sub-velocities. The overlap of the continuity control volume (—) with a vertical momentum control volume (—) is indicated, as well as the overlap with a horizontal momentum control volume (—).



(a) Fluid configuration before (grey) and after (hatched) transport. Volume fraction in momentum control volume (number) matches the fluxes.

(b) Fluid configuration after transport *and* reconstruction (hatched). Volume fraction in momentum control volume does *not* match fluxes anymore.

Fig. 20. Interface transport and reconstruction resulting in inconsistency between $\bar{\rho}$ and ρ^{n+1}, because $\bar{\rho}$ is based on the fluxes (leading to fluid configuration in (a)) and ρ^{n+1} is based on the volume fraction after reconstruction (fluid configuration in (b)). Volume fraction before transport is indicated by ■. Volume fraction after transport is indicated by hatched area. The cell boundary (—), the horizontal momentum control volume (—), and continuity control volume (—) are also shown. The arrows represent the VOF fluxes through the faces of the momentum control volume. The values give the volume fraction of half the momentum control volume, normalized by the size of the total momentum control volume.

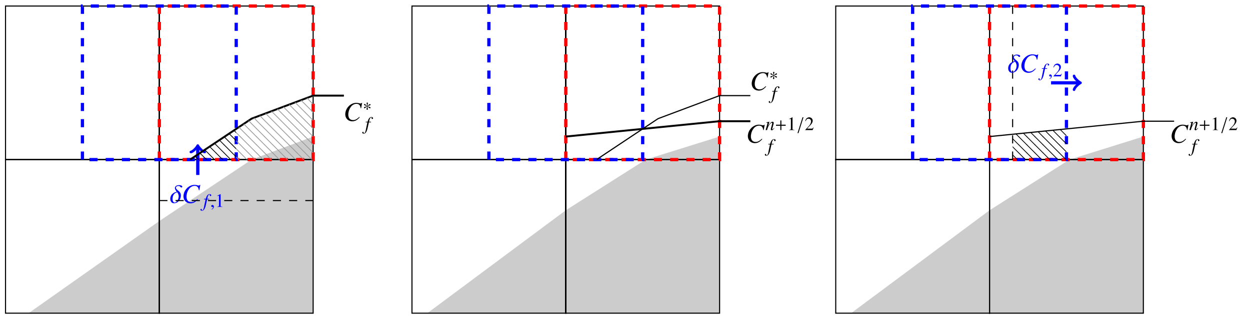
from one quadrant to another, without having mass fluxes associated with that movement. Zuzio et al. [59] have demonstrated, however, without explicitly stating it in their article, that the effect of this type of inconsistency on the overall accuracy of the method is marginal, which was confirmed later by van der Eijk and Wellens [52] when using a similar approach in combination with the COSMIC transport method that reduces direction-split errors through an additional step including intermediate reconstruction.

An example of the non-problematic inconsistency described above with single-step advection methods is given in Fig. 20. The interface is transported in 1D with a constant velocity. The grey area shows the fluid configuration before transport. The hatched area shows the fluid configuration after transport. The numbers represent the volume fraction in half of a momentum control volume (in blue), normalized by the size of the momentum control volume. Fig. 20a gives the configuration of the fluid after transport (net value of all fluxes), but before reconstruction. The volume fraction in the halves of the momentum control volume then has a value that correspond to the fluxes. After reconstruction, in Fig. 20b, the volume fraction in the halves of the momentum control volume has changed. And because density field $\bar{\rho}$ is based on the VOF fluxes, and density field ρ^{n+1} is based on the volume fraction, an inconsistency between $\bar{\rho}$ and ρ^{n+1} arises.

For incompressible flow modeling using the mixture formulation, the conditions $\rho_{1-f} < \bar{\rho} < \rho_f$ and $\rho_{1-f} < \rho^{n+1} < \rho_f$ for $\rho_{1_f} < \rho_f$ need to hold. Density ρ^{n+1} is associated with the volume fractions in a momentum control volume. As $\mathscr{A}$, as well as C_f, is bounded between 0 and 1, ρ^{n+1} always satisfies these criteria. Density $\bar{\rho}$ is based on the VOF fluxes. It therefore depends on the advection scheme whether $\bar{\rho}$ satisfies the conditions.

Upon more careful consideration, it was found that the intermediate reconstruction in COSMIC lead to a more substantial inconsistency, resulting in spurious velocities, negative densities $\bar{\rho}$ and even instability. Compared with the non-problematic reconstruction step discussed above, this intermediate reconstruction does violate the criteria given. The mechanism is explained below and is true for all direction-split schemes that include intermediate reconstruction (another example is the MACHO scheme [22]).

A sequence of steps of how a negative value for $\bar{\rho}$ can come about with COSMIC is given in Fig. 21 using the notations in Sec. 4.2. The hatched area indicates the size of the flux. The advection of the interface is split in fluxing the vertical direction first (Fig. 21a) to obtain the intermediate volume fraction field C_f^*, then reconstructing the interface to obtain $C_f^{n+1/2}$ (Fig. 21b), followed by fluxing in horizontal direction to obtain C_f^{n+1} (Fig. 21c).



(a) Vertical flux results in C_f^*.

(b) Reconstruction step for continuity CV to $C_f^{n+1/2}$.

(c) Horizontal flux to obtain C_f^{n+1}.

Fig. 21. Problematic inconsistency due to intermediate reconstruction in COSMIC. Fluid C_f^n is indicated by ■. The cell boundary (—), the horizontal momentum control volume (—), and continuity control volume (—) are shown. The black hatched area indicates the flux for the momentum control volume. The black dashed lines is the donating region for the given face-centered velocity.

The initial density ρ^n in Fig. 21 is equal to zero inside the momentum control volume. Using the variables from the situation of Fig. 21 in Eq. (25) results in an expression like

$$\bar{\rho} = -\frac{1}{\delta V_m}(\delta C_{f,2} - \delta C_{f,1})\rho_f, \tag{28}$$

in which it is assumed that the only non-zero VOF fluxes are $\delta C_{f,1}$ and $\delta C_{f,2}$, that δV_m is the size of the momentum control volume, and that $\rho_f >> \rho_{1-f}$. When, due to intermediate reconstruction $\delta C_{f,2}$ is larger than $\delta C_{f,1}$, so when the hatched area in Fig. 21c is larger than in Fig. 21a, then a negative value for $\bar{\rho}$ is obtained. Our experience is that negative densities lead to spurious velocities, which affect the time step through the Courant criterion. When the time step is kept the same, the spurious velocity may lead to instability.

Solutions to prevent negative densities could be to apply reconstruction in cell quadrants as in Rudman [45] with much increased computational effort. Another solution altogether could be to adopt a discretization based on a collocated arrangement of variables and control volumes. The solution we chose to obtain a robust, accurate and consistent method for two-phase flow simulations was to develop DQA, an unsplit advection method.

DQA is inspired by EMFPA. When researching the application of EMFPA for our applications, another problematic inconsistency was encountered that can lead to negative values of $\bar{\rho}$, and even instability, when the VOF flux is of opposite sign from the velocity in the same control volume face. Where the donating region for direction-split methods is always solely on one side of the control volume face, a multidimensional unsplit advection scheme like EMFPA can generate negative donating regions, as shown in Figs. 2, 11f, and 13b. Negative donating regions can result in a VOF flux δC_f directed inwards of a momentum control volume, while the velocity $\mathbf{u}$ points outwards.

An example of a negative donating region is illustrated in Fig. 22, in which the VOF flux δC_f is larger than the available fluid volume. Using Eq. (26) results in momentum $\rho^* u$ of opposite sign from the VOF flux. Using Eq. (25)

$$\bar{\rho} = \left(\mathcal{A}^n - \frac{\delta C_f}{\delta V_m}\right)\rho_f \tag{29}$$

for the situation in Fig. 22, in which the value of $\mathcal{A}^n$ is smaller than the VOF flux $\frac{\delta C_f}{\delta V_m}$, while assuming that $\rho_f >> \rho_{1-f}$, leads to a negative value for $\bar{\rho}$, spurious velocities and, potentially, instability. There is even the chance with EMFPA, although small, that the VOF flux δC_f in the negative donating region is larger than the volume constraint $u\delta t\delta A$. Then, fraction C in front of fluid ρ_{1-f} in Eq. (26) becomes negative, at which time mass conservation is not longer enforced. DQA, which does not allow for negative donating regions by design, prevents the problematic inconsistency.

6.4. The effect of the advection scheme on stability

Simulations with the two-phase solver of van der Eijk and Wellens [52] are performed to demonstrate the effect of inconsistency on the results. Two cases are considered. The first is designed to show the negative densities associated with the intermediate reconstruction step when using COSMIC for fluid transport. The second case is designed to show the negative densities resulting from negative donation regions when using EMFPA for fluid transport. The results of both cases are compared with the simulation results in which DQA is used for fluid transport. In all simulations, BLIC-MYC is used to reconstruct the interface.

The setup of the first case is shown in Fig. 23. A 2D cylinder (circle) composed of high-density fluid is translated diagonally through the domain at a constant velocity of 14.14[m/s]. The cylinder is surrounded by a low-density fluid with a uniform velocity of the same magnitude and in the same direction as the velocity of the high-density cylinder. The relative velocity between fluids, therefore, is zero and further velocity changes should not occur. The density ratio between fluids is 10^6. The effect of the grid

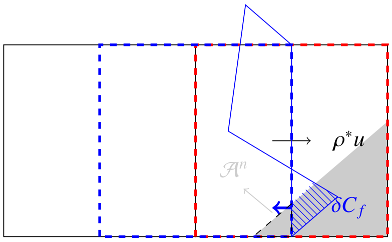


Fig. 22. Negative donating region when using EMFPA. Volume fraction C_f is indicated by ■. The donation region is shown in (—). The hatched area is the VOF flux δC_f. Momentum flux $\rho^* u$ is of opposite direction from the VOF flux δC_f. In case the fluid volume ($\mathcal{A}^n$) is smaller than the VOF flux δC_f, a negative density $\bar{\rho}$ is found, potentially leading to spurious velocities and instability.

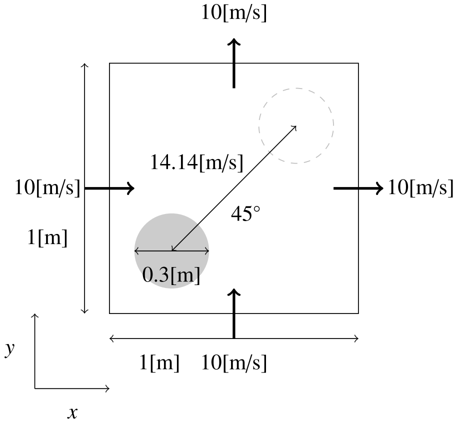
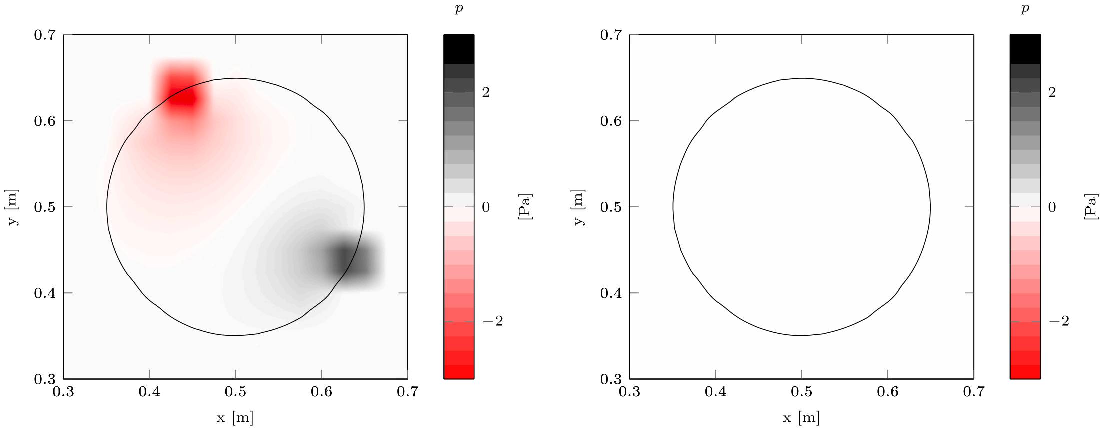


Fig. 23. Simulation setup with cylinder of high-density fluid translating diagonally through low-density fluid at zero relative velocity.

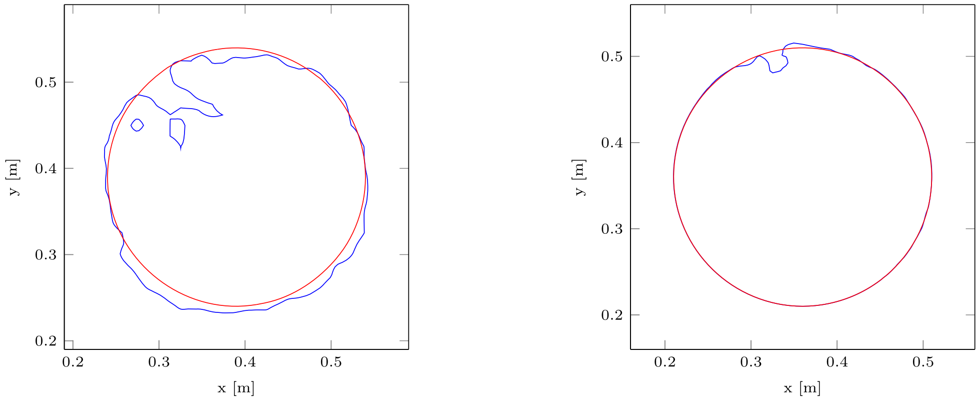


(a) COSMIC. (b) DQA.

Fig. 24. Contour plot of pressure in 40×40 simulation after 50 time steps. The interface is shown as a continuous black line. Results with COSMIC, to the left, feature regions of low and high pressure caused by spurious velocities. The simulation with DQA gives the expected results for this case without pressure gradients.

resolution on the results is investigated by considering three grid resolutions with 40×40, 80×80, and 120×120 grid cells. The time step is kept restrained at a value corresponding to an initial Courant number of 0.2. The Courant number is based on the size of the momentum control volume. The simulations finish after 100 time steps.

The results of the 40×40 simulation with the diagonal translation of the fluid cylinder are shown in Fig. 24. Contour plots of the pressure after 50 time steps are given. The pressure distribution in the simulation with COSMIC is showing a region with lower pressure and a region with higher pressure which is unexpected for a simulation with zero relative velocity between the high-density fluid cylinder and the surrounding fluid. These regions are caused by spurious velocities that themselves are caused by the negative density that has developed according to the mechanism described above. As shown on the right of Fig. 25, a simulation with DQA instead of COSMIC, but otherwise the same, yields the expected results and does not show any pressure gradients.



(a) 80×80, after 56 time steps.

(b) 120×120, after 66 time steps.

Fig. 25. The interface between fluids in 80×80 and 120×120 simulations at two time levels. COSMIC is —. DQA is —. The interface in the simulations with COSMIC is highly distorted, whereas the simulations with DQA give the expected results.

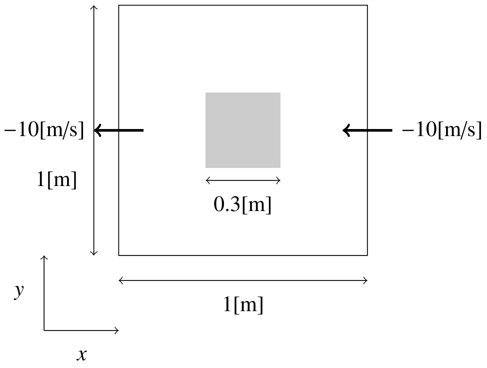


Fig. 26. Simulation setup with square of high-density fluid at rest surrounded by low-density fluid moving with an undisturbed velocity magnitude of 10[m/s].

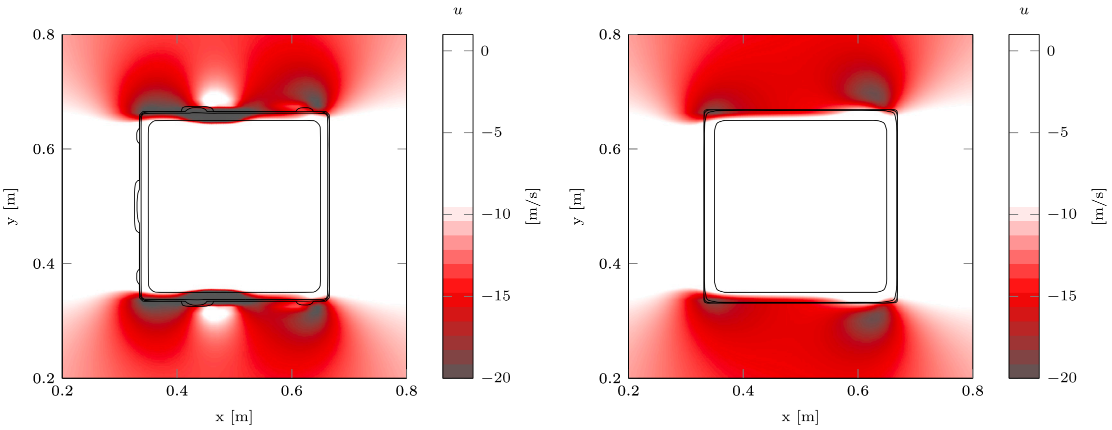
Fig. 25 compares the interfaces between COSMIC and DQA for two grid resolutions, 80×80 and 120×120. For both grid resolutions, the interface with COSMIC is highly distorted as a result of the spurious velocities, whereas the interface with DQA is as expected. The interface distortion does decrease with increasing grid resolution. Both simulations with COSMIC became unstable, but the simulation with a grid resolution of 120×120 became unstable at a later time step than the simulation with a grid resolution of 80×80.

The density $\bar{\rho}$ should be in between ρ_f and ρ_{1-f}. However, the simulation with COSMIC resulted for every time step in over- and undershoots of the density $\bar{\rho}$ in the order of 50. The over- and undershoots increase when the simulation becomes unstable until it crashes. An overshoot of 50[kg/m^3] compared to a density ratio of 10^6 seems small but can result in instability. The spurious velocities are smaller for lower density ratios. It must be noted that the simulations became unstable because we did not allow the time step to decrease according to the Courant criterion.

It is our experience, from van der Eijk and Wellens [52] and the present article, that exaggerating the density ratio with respect to more common ratios such as for water and air, helps trigger issues when they are there. The undershoots and overshoots of the intermediate density $r\bar{h}o$ seemed inconspicuous, but set a chain of events into motion that can even lead to unstable simulations.

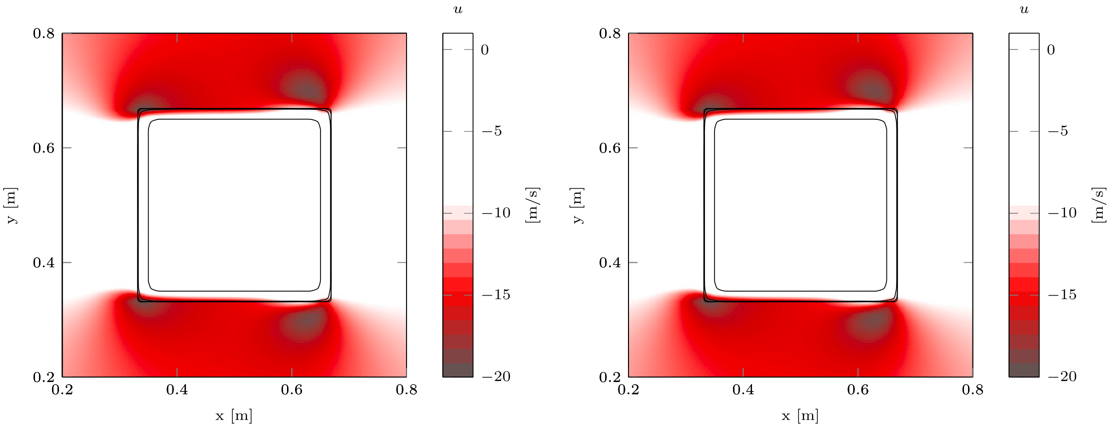
The second case is designed to demonstrate the effect of having negative donating regions when using the EMFPA unsplit advection scheme in a two-phase flow solver with a high-density ratio between fluids. Negative donating regions can also lead to negative densities and spurious velocities. The mechanism is different from that with COSMIC, but the effect on the robustness of the solver is the same. The previous case discussed does not show any issues for EMFPA because no negative donating regions are created with the imposed constant velocity field. The chance of negative donating regions increases for shear flows as in Fig. 13a.

The simulation setup is shown in Fig. 26. A 2D square of high-density fluid is at rest while being surrounded by low-density fluid moving with an undisturbed velocity magnitude of 10[m/s]. The density ratio between fluids is 10^6. Two grid resolutions, 86×86 and 87×87, are considered to show how sensitive EMFPA can be for the specific configuration of fluids. The time step is allowed to vary so that at any time the Courant number is lower than 0.4. The simulations end at $t = 0.05$[s]. The thought behind the setup is to represent a shear flow, in which negative donating regions are constructed with EMFPA as in Fig. 13a. DQA does not allow negative regions to be formed. In absence of gravity and viscous stresses, the setup should not result in deformation or movement of the square of fluid. The differences between the simulation results with EMFPA and with DQA will be discussed.



(a) EMFPA 86×86.

(b) DQA 86×86.



(c) EMFPA 87×87.

(d) DQA 87×87.

Fig. 27. Contour plot of horizontal velocity and interface at $t = 0.005$[s] in simulations with EMFPA and DQA at slightly different grid resolutions. Contour levels of the interface represent volume fraction values of 0.5, 0.01, 0.001, and 0.0001, becoming smaller from the high-density square towards the low-density surrounding fluid. EMFPA generates spurious velocities near -70[m/s] (not shown, minimum of scale is -20[m/s]).

The differences between the simulation results with EMFPA and with DQA are most apparent when the horizontal velocity field is shown together with the interface between fluids. Fig. 27 shows the horizontal velocity and interface of four simulations at time $t = 0.005$[s]. The top row in Fig. 27 is for grid resolution 86×86 and has the EMFPA results on the left and the DQA results on the right. The EMFPA results for grid 86×86 feature a distorted interface caused by negative densities, resulting in spurious velocity oscillations that attain a magnitude as high as 70[m/s] (the scale in the plot goes until 20 to be able to compare with the other plots in the figure). The bottom row of Fig. 27 shows the results of the simulation with a grid resolution of 87×87, again with EMFPA on the left and DQA on the right. For grid 87×87, the results of EMFPA and DQA are the same and equal to what could be expected. The difference between the results of the simulations with EMFPA, with a marginal difference in grid resolution, indicates how sensitive a two-phase flow solver can be to negative densities.

The interface deformation with EMFPA is related to the specific position of the interface in a grid cell with a grid resolution of 86×86 and to EMFPA using vertex velocities. When a grid cell containing the interface is nearly filled, then, due to the high velocity of the low-density fluid, a negative donating region can be formed with a VOF flux of the opposite sign from the main velocity in that cell (see Fig. 13a). That VOF flux deforms the interface, which induces a negative density. With grid resolution 87×87, the grid cells near the interface have a lower volume fraction which prevents the negative donating region from being formed. DQA is not sensitive to this mechanism because it does not allow the creation of negative donating regions.

The spurious horizontal velocities in the 86×86 EMFPA simulation are a consequence of negative densities which are caused by the VOF fluxes in interface cells of opposite sign from the velocity in those cells. Negative densities $\bar{\rho}$ were computed with values up to -350 kg/m^3 in more than 50% of the time steps according to the mechanism explained above: solving the velocity field $\bar{\mathbf{u}}$ in Eq. (24), using Eqs. (26) and (25) yields a negative density $\bar{\rho}$, resulting in a change in sign of the velocity. The pressure field needs to ensure a divergence-free field and, therefore, induces a larger correction of the velocity field, giving even higher velocities.

7. Conclusion

A new bilinear interface calculation (BLIC) method is introduced for reconstructing the interface in a 2D Volume-of-Fluid (VOF) method. A new multidimensional consistent unsplit VOF advection method, DQA, which is short for Donating Quadrant Advection, is presented using face-centered velocities that is conservative, low-cost, and accurate. It allows the fluid to enter and exit a cell in a time step.

Both BLIC and DQA are tested with standard 2D benchmarks like static reconstruction of two different shapes, Zalesak's slotted disk, and a reversed single vortex field. The BLIC method uses a robust curvature-weighted averaging method with a piecewise linear segment as starting point to create continuity of the interface along the cell faces. Weighting using the curvature increases the accuracy of the BLIC method, especially for underresolved material interfaces. The accuracy of BLIC for higher grid resolutions converges to the accuracy of PLIC on a grid twice as fine. The additional computational costs compared to Parker and Youngs' PLIC method is up to 20%, decreasing to 1% for higher grid resolutions. However, the increase in accuracy of representing the interface is up to an order of magnitude. The BLIC method needs a less dense grid and less computational time to reach similar accuracies as the PLIC method.

The DQA scheme we present shows similar accuracy as the state-of-the-art vertex velocities-based EMFPA scheme for traditional benchmarks. The computational costs of the unsplit DQA scheme are lower than for a direction-split scheme like COSMIC, mostly because of the intermediate reconstruction steps a direction-split scheme uses. The combination of the BLIC and DQA methods is competitive with higher-order methods in terms of accuracy but outperforms most of them in computational cost based on provided data from literature.

An important additional benefit of the DQA scheme is introduced to ensure consistency for an existing consistent two-phase flow solver with a staggered arrangement of variables. Consistency means using the same discretization techniques for determining the momentum, mass, and VOF fluxes and carefully matching them in the different control volumes that are staggered with respect to each other. Consistency is achieved by incorporating the VOF fluxes in the discretization of the density. Negative densities due to inconsistency are found for an intermediate reconstruction step in COSMIC, and for a VOF flux of the opposite sign than the velocity in EMFPA. The negative densities result in spurious velocities and instability near the fluid interface. The DQA scheme prevents these negative densities and remains stable and accurate in a simulation of a translating high-density cylinder in a low-density fluid, for which COSMIC fails, and in a simulation with a square of high-density fluid in a shear flow with low-density fluid, for which EMFPA fails.

CRediT authorship contribution statement

Martin van der Eijk: Conceptualization, Methodology, Software, Validation, Formal analysis, Writing, Visualization.
Peter Wellens: Conceptualization, Methodology, Formal analysis, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] T. Abadie, J. Aubin, D. Legendre, On the combined effects of surface tension force calculation and interface advection on spurious currents within volume of fluid and level set frameworks, J. Comput. Phys. 297 (2015) 611–636.

[2] T. Arrufat, M. Crialesi-Esposito, D. Fuster, Y. Ling, L. Malan, S. Pal, R. Scardovelli, G. Tryggvason, S. Zaleski, A mass-momentum consistent, volume-of-fluid method for incompressible flow on staggered grids, Comput. Fluids 215 (2021) 104785.

[3] 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 (2003) 611–639.

[4] E. Aulisa, S. Manservisi, R. Scardovelli, S. Zaleski, Interface reconstruction with least-squares fit and split advection in three-dimensional Cartesian geometry, J. Comput. Phys. 225 (2007) 2301–2319.

[5] J.B. Bell, P. Colella, H.M. Glaz, A second-order projection method for the incompressible Navier-Stokes equations, J. Comput. Phys. 85 (1989) 257–283.

[6] M. Bussmann, D.B. Kothe, J.M. Sicilian, Modeling high density ratio incompressible interfacial flows, in: Fluids Engineering Division Summer Meeting, 2002, pp. 707–713.

[7] B.K. Campbell, An arbitrarily high-order three-dimensional Cartesian-grid method for reconstructing interfaces from volume fraction fields, J. Comput. Phys. 426 (2021) 109727.

[8] A. Cervone, S. Manservisi, R. Scardovelli, S. Zaleski, A geometrical predictor–corrector advection scheme and its application to the volume fraction function, J. Comput. Phys. 228 (2009) 406–419.

[9] R. Comminal, J. Spangenberg, J.H. Hattel, Cellwise conservative unsplit advection for the volume of fluid method, J. Comput. Phys. 283 (2015) 582–608.

[10] S.J. Cummins, M.M. Francois, D.B. Kothe, Estimating curvature from volume fractions, Comput. Struct. 83 (2005) 425–434.

[11] R. DeBar, Fundamentals of the kraken code, Technical Report UCIR-760, 1974.

[12] S. Diwakar, S.K. Das, T. Sundararajan, A quadratic spline based interface (quasi) reconstruction algorithm for accurate tracking of two-phase flows, J. Comput. Phys. 228 (2009) 9107–9130.

[13] B. Düz, Wave generation, propagation and absorption in CFD simulations of free surface flows, Ph.D. thesis, University of Technology Delft, 2015.
[14] I. Ginzburg, G. Wittum, Two-phase flows on interface refined grids modeled with vof, staggered finite volumes, and spline interpolants, J. Comput. Phys. 166 (2001) 302–335.
[15] D.J. Harvie, D.F. Fletcher, A new volume of fluid advection algorithm: the stream scheme, J. Comput. Phys. 162 (2000) 1–32.
[16] D.J. Harvie, D.F. Fletcher, A new volume of fluid advection algorithm: the defined donating region scheme, Int. J. Numer. Methods Fluids 35 (2001) 151–172.
[17] C.W. Hirt, B.D. Nichols, Volume of fluid (vof) method for the dynamics of free boundaries, J. Comput. Phys. 39 (1981) 201–225.
[18] K. Ito, T. Kunugi, H. Ohshima, T. Kawamura, A volume-conservative plic algorithm on three-dimensional fully unstructured meshes, Comput. Fluids 88 (2013) 250–261.
[19] D. Kim, C.B. Ivey, F.E. Ham, L.G. Bravo, An efficient high-resolution volume-of-fluid method with low numerical diffusion on unstructured grids, J. Comput. Phys. 446 (2021) 110606.
[20] K. Kleefsman, G. Fekken, A. Veldman, B. Iwanowski, B. Buchner, A volume-of-fluid based simulation method for wave impact problems, J. Comput. Phys. 206 (2005) 363–393.
[21] S.K. Lam, A. Pitrou, S. Seibert, Numba: a llvm-based python jit compiler, in: Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC, 2015, pp. 1–6.
[22] B. Leonard, A. Lock, M. MacVean, Conservative explicit unrestricted-time-step multidimensional constancy-preserving advection schemes, Mon. Weather Rev. 124 (1996) 2588–2606.
[23] P. Liovic, D. Lakehal, A Newton–Krylov solver for remapping-based volume-of-fluid methods, SIAM J. Sci. Comput. 31 (2009) 865–889.
[24] P. Liovic, M. Rudman, J.L. Liow, D. Lakehal, D. Kothe, A 3d unsplit-advection volume tracking algorithm with planarity-preserving interface reconstruction, Comput. Fluids 35 (2006) 1011–1032.
[25] J. López, J. Hernández, P. Gómez, F. Faura, A volume of fluid method based on multidimensional advection and spline interface reconstruction, J. Comput. Phys. 195 (2004) 718–742.
[26] R.K. Maity, T. Sundararajan, K. Velusamy, An accurate interface reconstruction method using piecewise circular arcs, Int. J. Numer. Methods Fluids 93 (2021) 93–126.
[27] T. Marić, H. Marschall, D. Bothe, An enhanced un-split face-vertex flux-based vof method, J. Comput. Phys. 371 (2018) 967–993.
[28] J. Mencinger, I. Žun, A plic–vof method suited for adaptive moving grids, J. Comput. Phys. 230 (2011) 644–663.
[29] T. Milcent, A. Lemoine, Moment-of-fluid analytic reconstruction on 3d rectangular hexahedrons, J. Comput. Phys. 409 (2020) 109346.
[30] S. Mosso, C. Garasi, R. Drake, A smoothed two-and three-dimensional interface reconstruction method, Comput. Vis. Sci. 12 (2009) 365–381.
[31] A. Murrone, H. Guillard, A five equation reduced model for compressible two phase flow problems, J. Comput. Phys. 202 (2005) 664–698.
[32] W.F. Noh, P. Woodward, Slic (simple line interface calculation), in: Proceedings of the Fifth International Conference on Numerical Methods in Fluid Dynamics June 28–July 2, 1976 Twente University, Springer, Enschede, 1976, pp. 330–340.
[33] E. Olsson, G. Kreiss, A conservative level set method for two phase flow, J. Comput. Phys. 210 (2005) 225–246.
[34] M. Owkes, O. Desjardins, A computational framework for conservative, three-dimensional, unsplit, geometric transport with application to the volume-of-fluid (vof) method, J. Comput. Phys. 270 (2014) 587–612.
[35] M. Owkes, O. Desjardins, A mass and momentum conserving unsplit semi-Lagrangian framework for simulating multiphase flows, J. Comput. Phys. 332 (2017) 21–46.
[36] B.J. Parker, D.L. Youngs, Two and three dimensional Eulerian simulation of fluid flow with material interfaces, in: Atomic Weapons Establishment, 1992.
[37] J.E. Pilliod Jr, E.G. Puckett, Second-order accurate volume-of-fluid algorithms for tracking material interfaces, J. Comput. Phys. 199 (2004) 465–502.
[38] S. Popinet, An accurate adaptive solver for surface-tension-driven interfacial flows, J. Comput. Phys. 228 (2009) 5838–5866.
[39] S. Popinet, S. Zaleski, A front-tracking algorithm for accurate representation of surface tension, Int. J. Numer. Methods Fluids 30 (1999) 775–793.
[40] G.R. Price, A Piecewise Parabolic Volume Tracking Method for the Numerical Simulation of Interfacial Flows, University of Calgary, 2000.
[41] E.G. Puckett, A.S. Almgren, 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.
[42] Y. Renardy, M. Renardy, Prost: a parabolic reconstruction of surface tension for the volume-of-fluid method, J. Comput. Phys. 183 (2002) 400–421.
[43] W.J. Rider, D.B. Kothe, Reconstructing volume tracking, J. Comput. Phys. 141 (1998) 112–152.
[44] M. Rudman, Volume-tracking methods for interfacial flow calculations, Int. J. Numer. Methods Fluids 24 (1997) 671–691.
[45] M. Rudman, A volume-tracking method for incompressible multifluid flows with large density variations, Int. J. Numer. Methods Fluids 28 (1998) 357–378.
[46] R. Scardovelli, S. Zaleski, Direct numerical simulation of free-surface and interfacial flow, Annu. Rev. Fluid Mech. 31 (1999) 567–603.
[47] R. Scardovelli, S. Zaleski, Analytical relations connecting linear interfaces and volume fractions in rectangular grids, J. Comput. Phys. 164 (2000) 228–237.
[48] R. Scardovelli, S. Zaleski, Interface reconstruction with least-square fit and split Eulerian–Lagrangian advection, Int. J. Numer. Methods Fluids 41 (2003) 251–274.
[49] K.M. Shyue, F. Xiao, An Eulerian interface sharpening algorithm for compressible two-phase flow: the algebraic thinc approach, J. Comput. Phys. 268 (2014) 326–354.
[50] M. Sussman, P. Smereka, S. Osher, A level set approach for computing solutions to incompressible two-phase flow, J. Comput. Phys. 114 (1994) 146–159.
[51] O. Ubbink, R.I. Issa, A method for capturing sharp fluid interfaces on arbitrary meshes, J. Comput. Phys. 153 (1999) 26–50.
[52] M. van der Eijk, P.R. Wellens, Two-phase free-surface flow interaction with moving bodies using a consistent, momentum preserving method, J. Comput. Phys. 474 (2023) 111796.
[53] T. Vignesh, S. Bakshi, Noniterative interface reconstruction algorithms for volume of fluid method, Int. J. Numer. Methods Fluids 73 (2013) 1–18.
[54] G.D. Weymouth, D.K.P. Yue, Conservative volume-of-fluid method for free-surface simulations on Cartesian-grids, J. Comput. Phys. 229 (2010) 2853–2865.
[55] F. Xiao, Y. Honma, T. Kono, A simple algebraic interface capturing scheme using hyperbolic tangent function, Int. J. Numer. Methods Fluids 48 (2005) 1023–1040.
[56] B. Xie, F. Xiao, Toward efficient and accurate interface capturing on arbitrary hybrid unstructured grids: the thinc method with quadratic surface representation and Gaussian quadrature, J. Comput. Phys. 349 (2017) 415–440.
[57] S.T. Zalesak, Fully multidimensional flux-corrected transport algorithms for fluids, J. Comput. Phys. 31 (1979) 335–362.
[58] Q. Zhang, Fourth-and higher-order interface tracking via mapping and adjusting regular semianalytic sets represented by cubic splines, SIAM J. Sci. Comput. 40 (2018) A3755–A3788.
[59] D. Zuzio, A. Orazzo, J.L. Estivalèzes, I. Lagrange, A new efficient momentum preserving level-set/vof method for high density and momentum ratio incompressible two-phase flows, J. Comput. Phys. 410 (2020) 109342.