Journal of Computational Physics **166,** 302–335 (2001)
doi:10.1006/jcph.2000.6655, available online at http://www.idealibrary.com on IDEAL®

Two-Phase Flows on Interface Refined Grids Modeled with VOF, Staggered Finite Volumes, and Spline Interpolants

I. Ginzburg and G. Wittum

Interdisziplinäres Zentrum für Wissenschaftliches Rechnen (IWR), Im Neuenheimer Feld 368, 69120 Heidelberg, Germany
E-mail: ginzburg@itwm.uni-kl.de, Gabriel.Wittum@iwr.uni-heidelberg.de

Received August 26, 1999; revised April 6, 2000

A two-phase 2D model that combines the volume of fluid (VOF) method with implicit staggered finite volumes discretization of the Navier–Stokes equation is presented. Staggered finite volumes are developed on the basis of nonconforming Crouzeix–Raviart finite elements, where all components of the velocity lie in the middle of the element edges and the pressure degrees of freedom are found in the centers of mass of the elements. Staggered finite Volumes extend marker and cell (MAC) regular staggered grids to unstructured mesh. A linear saddle point problem, resulting from either the discretization or the Newton method, is solved for all unknown pressures and velocities. Interface is represented with spline interpolants which follow the VOF distribution. Adaptive mesh refinement is used to obtain a high level of uniform refining at the domain of dependence of the interface. The aligned grid is obtained by irregular refining of the cells which are intersected by a curve. The boundaries of its elements coincide with the slope segments going through the intersections of the curve with the underlying regular elements boundary. The deformable computational grids are used only to discretize the Navier–Stokes equation. The advection of volume fractions is done on the advection mesh, which corresponds to highest regular refining on the computational grid. Approximation of the surface tension on spline interpolants offers a straightforward way to describe correctly the pressure jumps on interface-fitted staggered grids. This allows deletion of the anomalous currents around a statical bubble and their effective reduction in real simulations. On the aligned grid, the continuity of the viscous stress is modeled exactly due to the finite volume approach. Using the proposed numerical techniques, single bubble rise is analyzed. © 2001 Academic Press

Key Words: two-phase flow; Navier–Stokes equation; volume of fluid method; finite volume method; rotated elements; saddle point problem; multigrid; linear interface interpolant; cubic splines; Laplace law; buoyant bubbles.

1. INTRODUCTION

In order to solve two-phase problems with immiscible fluids, incompressible Navier–Stokes equations are necessary. The difficulty arises from the presence of moving sharp fronts where pressure and velocity derivatives may have jump discontinuities. Numerous methods describing the interface and its propagation are often based on similar discretizations of the basic conservation laws. Here, we must deal with two principal directions, *surface tracking* and *surface capturing*, since our model can be seen as a hybrid of both. Excellent overviews of the interface methods are given by Hyman [31], Unverdi and Tryggvason [66], Sussman and Smereka [63], Rider and Kothe [54], Kothe [33], Rudman [60, 61], and Scardovelli and Zaleski [57].

Surface-tracking methods explicitly treat the interface as a discontinuity. Usually, it is specified by an ordered set of marker points, connected by an interpolation curve. The markers are advanced in the Lagrangian manner and then redistributed to obtain the best resolution of the interface (see Glimm *et al.* [25], Daly [19], and Popinet and Zaleski [45]). With the front-tracking method of Unverdi and Tryggvason [66], the interface is represented by an *interface grid*. Density and viscosity fields are computed from the interface grid position with the help of smooth indicator functions. Free Lagrangian methods do not treat the basic equations and the interface propagation separately. Instead, they advect the computational grid itself by the Lagrangian equations (see Fyfe *et al.* [21]). Galaktionov *et al.* [23] apply efficiently a front-tracking technique in strongly deformable geometries.

Surface-capturing methods are implicit with respect to the interface: by using special functions for the description of the multiphase distribution, they do not need the exact interface position to discretize the governing equations. The first volume-tracking methods are due to DeBar [20], Hirt and Nicholls [29], and Noh and Woodwards [43]. The volume of fluid (VOF) method [29] uses a volume fraction distribution (vof) to represent the two phases. The level set approach due to Osher and Sethian [44] uses a so-called level set function to implicitly capture the front. Both methods handle the complicated interfaces, including their merging and break up, more easily than the surface-tracking methods. The level set method advects the interface by solving a partial differential equation (PDE) for the level set function. Considerable efforts have been devoted to creating accurate advection schemes based upon interface reconstruction for the volume-tracking methods. An exhaustive review of this subject is Rider and Kothe [54].

To date, the first volume-tracking methods have used three different types of interface reconstruction: piecewise linear approximation across cell faces [20], piecewise constant ("stair stepped") approximation [29], and piecewise constant, the SLIC technique [43]. Following Youngs' second-order-accurate 2D method [71] and first-order 3D method [72], the most popular piecewise linear interface calculation (PLIC) methods reconstruct free surfaces of arbitrary geometry locally within each interfacial cell based on the estimation of the normal to the interface. The locality of the method results in the discontinuity of the interface on the cell edges. The SURFER code [26, 34, 57] computes the fluxes across the cell boundaries from the *split Lagrangian* propagation of the first-order PLIC interface. The volume-tracking algorithm of Rudman [60] develops the concept of Zalesak's flux-corrected transport without interface reconstruction. The method is intensively tested against the SLIC, Hirt–Nichols VOF, and first-order Youngs method. The FLAIR method of Ashgriz and Poo [3] is based on a linear reconstruction, continuous at the face between two adjacent interfacial cells. Second-order improvement of the FLAIR is discussed by Kim and No [51], where a parabolic curve holds both face slops and vof in interfacial cells. An approach similar to [3] is extended by Jeong and Yang [32] to 2D quadrilaterals elements. They also develop a geometrical smoothing algorithm which results in a piecewise linear continuous reconstruction specially adapted to filling processes. High-order linear interface reconstructions and advection methods have been developed by Pilliod [52], Puckett *et al.* [47], and Rider and Kothe [54]. Rider and Kothe also extend Youngs method [72] to calculate the normal to unstructured meshes, based on the linear square sense approach of Barth [11]. The accuracy of this method is compared with the second-order-least squares minimization [47, 52] and a second-order-accurate iterative procedure due to Swartz [65]. The Swartz method is applied in a new full remapping time integration scheme [41, 42], which represents a 2D illustration of recent enhancement to multidimensional volume-tracking advection schemes on both structured and unstructured grids. Very recently, Gao [22] has applied a volume-tracking technique to 3D unstructured tetrahedral collocated mesh.

A common problem of all methods is an accurate representation of the surface tension force which is concentrated on the interface. Often, the surface tension term is computed either with the continuum surface force (CSF) model of Brackbill *et al.* [13] or with the continuum surface stress (CSS) formulation of Lafaurie *et al.* [34]. The CSF represents the surface tension effects in a form of a smoothly varied volumetric force. Different methods for estimating the curvature, normals, and the surface delta function required for CSF model have been developed [2, 33, 61, 69]. The sensitivity of the results to the choice of smoothing kernels and the interface orientation, as well as to degree of smoothing, is not yet well understood. The CSS method [26, 34, 57] requires only the computation of the normals to the interface.

An inaccuracy caused by these methods manifests itself, in particular, in well-known *anomalous currents* around the stationary bubble (see [34, 57, 61, 69]). In simulations with the relatively strong surface tension effects, and especially in the presence of large density/viscosity jumps, the currents can progressively grow and destabilize the solution (see [2]). The currents can be reduced due to appropriate smoothing in the CSF and the CSS methods, but no remedy is found to delete them with these approaches. An alternative way is to model a correct pressure jump at the interface. In modeling the surface tension force on interface interpolants, cubic splines [19, 21, 45] have been found smooth enough to ensure an accurate discretization of the curvature. Popinet and Zaleski [45] annihilate the spurious currents due to taking into account, in addition to spline interpolation which goes through a set of marker points, the interface position while discretizing the pressure gradients.

An account of the interface position is rather delicate if the grid is fixed. Also, Coward *et al.* [17] show that commonly used simple viscosity averages significantly reduce the accuracy of VOF models at the interface. In order to introduce correctly the pressure jumps and the continuity of the viscous stresses, the remedy can be found in deformable grids, where element boundaries lie along the reconstructed interface. In combination with the VOF method, this approach was developed by Sato and Richardson [56] and Mashayek and Ashgriz [40]. In Sato and Richardson's *fringe element generation method*, the grid lies along a piecewise linear, continuously reconstructed interface which preserves the vof distribution. Lock *et al.* [38] and Barth and Sethian [12] use pseudo-concentration and level set techniques, accordingly, to track the front. Triangular mesh naturally provides a smooth front and an advection of the continuous and monotonic level set function is performed through the fixed finite element mesh.

Deformable grids induce the development of multiphase finite element schemes and finite volume schemes based on finite element approximations. In the works cited above, collocated discretization based on penalty function formulation is usually chosen [32, 38, 40, 56]. A special discontinuous piece-wise linear pressure approximation is used to model a pressure jump in [38]. Gao [22] employs a special mixed FEM formulation to obtain a stable discretization of Navier–Stokes equations. The outcome of front aligned grids is that the boundary conditions at the front are satisfied accurately without any Lagrangian moving of mesh system.

Summarizing the aforesaid, we emphasize two main factors in our work, which define the success of interface flow simulations: the correct approximation of surface tension force and the correct interface boundary conditions. In the first, we use the spline interpolants. In the second, we consider the grids aligned to the interface, including the use of unstructured (locally, near the interface) grids and finite volume discretization of Navier–Stokes equations. We keep in mind that all the algorithmic constructions have to be extended to complicated geometries and to 3D. Our choice of the algorithmic path is supported by the possibilities which offer the simulation tool UG [5, 50] to introduce any degrees of freedom while discretizing on adaptively refined and/or unstructured grids in 2D and 3D cases.

We propose to combine a VOF method with *staggered finite volumes* for Navier–Stokes equations. We develop this discretization for two-phase Navier–Stokes equations based on the nonconforming Crouzeix–Raviart finite elements [18]. The velocity degrees of freedom lie in the middle of the element edges. The pressure degrees of freedom are located in the centers of mass of the elements. Since pressure points do not lie on the element boundaries, approximation of the surface tension on spline aligned grids enables us to model correctly a pressure jump, to delete the anomalous currents around a statical bubble, and to reduce them effectively in real simulations. The benefits of using the VOF are that it tracks the interface topology and spline interpolation is “attached” to the obtained vof distribution. The construction of a cubic spline which preserves volume fraction distribution is presented. Three different types of computational grids are combined to track the pressure and velocity: uniform, adaptively regular refined, and interface aligned grids. Time integration of vof distribution is accomplished with a SURFER approach [26] on a so-called advection mesh, corresponding to a highest regular level of the computational grid.

The paper is structured in following way. In Section 2, the basic relations are given. In Section 3, interface reconstruction is presented. In Section 4, computational and advection meshes are constructed. Staggered finite volumes are introduced in Section 5. Different transgrid interpolations are discussed in Section 6. Laplace law is modeled in Section 7. Rise of single bubbles is analyzed in Section 8.

2. BASIC EQUATIONS

Assume that two immiscible phases, say liquid and gas, occupy a 2D rectangular domain Ω. The velocity field $\vec{u} = (u_x, u_y)$ and the pressure p of each phase obey the incompressible Navier–Stokes equation

$$\rho\frac{\partial\vec{u}}{\partial t} + \rho\nabla\cdot(\vec{u}\otimes\vec{u}) = -\nabla p + \rho\vec{g} + \vec{F}_s + \nabla\cdot(2\mu\mathbf{D}), \quad \mathbf{D} = \frac{1}{2}(\nabla\vec{u} + \nabla^t\vec{u}); \quad (1)$$

$$\nabla \cdot \vec{u} = 0. \tag{2}$$

In bulk, $\rho = \{\rho_l, \rho_g\}$ is the density and $\mu = \{\mu_l, \mu_g\}$ is the viscosity. For immiscible phases they are constant along a particle path; therefore the following relations hold:

$$\frac{\partial \rho}{\partial t} + \vec{u}\nabla\rho = 0\ ;\ \frac{\partial \mu}{\partial t} + \vec{u}\nabla\mu = 0. \tag{3}$$

The surface tension force acts on the interface S between the fluids. If the surface tension coefficient σ is constant, the surface tension force per unit area $\vec{F}_s$ is defined as

$$\vec{F}_s = \sigma\kappa\vec{n}\delta_S. \tag{4}$$

Here $\kappa(\vec{r}) = -(\nabla \cdot \vec{n})|_{\vec{r}}$ is a local curvature, taken positive if the center of curvature is in the gas phase; $\vec{n}(\vec{r})$ is the unit normal to S at $\vec{r}$, directed from the continuous phase into the gas phase. Distribution δ_S is concentrated on the interface S. A precise numerical computation of the curvature is difficult unless the interface is represented by a sufficiently smooth curve, e.g., twice continuously differentiable. Assuming that δ_S does not vary in a direction normal to interface, Lafaurie *et al.* [34] have shown that $\vec{F}_s$ can be included into Eq. (1) in the momentum-conserving form

$$\vec{F}_s \approx -\nabla \cdot \mathbf{T}\ , \quad \mathbf{T} = \sigma(\mathbf{I} - \vec{n} \otimes \vec{n})\delta_S. \tag{5}$$

In practice, δ_S is often approximated as $\delta_S = \|\nabla C\|$; $\|\cdot\|$ denotes the Euclidean norm of a vector. The finite difference "filters" to smooth C before its uses in (5) are discussed in [34, 57].

The velocity is assumed to be continuous across the interface: $[\vec{u}]_S = 0$. Here and below, $[\psi]_S = \psi_l - \psi_g$ denotes the jump across the interface S. Momentum conservation supplies the additional interface conjunction condition

$$[2\mu\mathbf{D} \cdot \vec{n} - p\vec{n}]_S = \vec{F}_s. \tag{6}$$

At the boundary of the physical domain Ω, we impose either no-slip or free-slip boundary conditions.

3. INTERFACE RECONSTRUCTION AND ADVECTION

3.1. *Linear Interface*

With the volume of fluid approach, we represent the distribution of phases by *volume fraction* C_k of the gas phase in cell k. The PLIC methods [37, 54, 71] uniquely define the interface in each cell with $0 < C < 1$ by a slope segment, which is perpendicular to a given normal $\vec{n}$:

$$\vec{n} \approx \nabla C. \tag{7}$$

If the domain of dependence of the given interfacial cell is refined regularly, ∇C can be computed with a nine-point finite difference stencil in 2D. Following [52, 54], the method for normal estimation which reproduces a line (a plane) exactly, regardless of its orientation with respect to fixed coordinate system, is referred to as second-order method. With this criterion, PLIC methods based on finite difference approximation of the normal done separately for each cell are only first-order accurate. When the accuracy and the convergence of the different algorithms are studied to calculate the normal [54], the first-order method (referred to as Youngs' second method in [54]) has better accuracy on coarse grids than the higher-order methods. The results presented in this paper are obtained with the first-order finite difference normal estimation.

3.2. *Cubic Spline Interpolant*

Piecewise parabolic interface reconstructions which preserve also the volume fraction distributions are usually drawn through only a local set of neighbors (see 5 × 5 block in [1], two neighboring cells in [51], 3 × 3 block in [53]). In the last work, the volume conservation is ensured in least-squares sense. Mosso *et al.* [42] draw a circle going through midpoints of three neighboring interfacial segments. At the same time, cubic splines [19, 21, 45], going through a *fixed position of marker points* (x_k, y_k), are found to be smooth enough to ensure the accurate numerical computations of the curvature in (4). The third-order polynomial parametric function $(x(s), y(s))$, $x(s) \in C^2$, $y(s) \in C^2$, has the following form in the interval between (x_{k-1}, y_{k-1}) and (x_k, y_k), $k \geq 1$:

$$\psi(s) = \frac{M_{k-1}^{\psi}}{6L_{k-1}}(s_k - s)^3 + \frac{M_k^{\psi}}{6L_{k-1}}(s - s_{k-1})^3 + \left(\frac{\psi_k}{L_{k-1}} - \frac{M_k^{\psi} L_{k-1}}{6}\right)(s - s_{k-1}) + \left(\frac{\psi_{k-1}}{L_{k-1}} - \frac{M_{k-1}^{\psi} L_{k-1}}{6}\right)(s_k - s). \tag{8}$$

Here, $\psi(s)$ denotes $x(s)$ or $y(s)$; s_k is a value of the parameter s at point (x_k, y_k): $s_k = \sum_{j=1}^{k}((x_j - x_{j-1})^2 + (y_j - y_{j-1})^2)^{\frac{1}{2}}$ and $L_{k-1} = s_k - s_{k-1}$. The continuity of the first derivatives at the points (x_k, y_k) supplies the necessary linear equations to determine the second derivatives M_k^x and M_k^y.

In VOF models, the points lying on the interface are unknown *a priori*. Our basic idea is to construct a cubic spline interpolant (8) which exactly cuts the phase distribution in each interface cell. Assume that the spline intersects the edges of a $h \times h$ interfacial cell k with the boundary $\partial\Sigma$ at points $\mathbf{C} = (x_{k-1}, y_{k-1})$ and $\mathbf{D} = (x_k, y_k)$ (see Fig. 5). Suppose further that we know *which* edges should be cut by the interface but we do not know *where* the interface cuts them. Then one of the coordinates of each cross point is fixed. Let ξ_k denote the unknown coordinate. Following [45], we use the Stokes theorem to express the volume V_k of the cell k cut by the cubic spline as

$$V_k = \oint_{\partial\Sigma} x\, dy. \tag{9}$$

This is schematically shown by the sequence of points **C**, **D**, **E**, **F**, **C** in Fig. 5. Our purpose is to satisfy the volume conservation relations

$$V_k = C_k h^2, \quad k = 1, \ldots, N. \tag{10}$$

If function $\int_{s_{k-1}}^{s_k} x(s)y'(s)ds$ is determined from (8), V_k is written in terms of the unknowns $\{M_{k-1}^x, M_k^x, M_{k-1}^y, M_k^y, \xi_{k-1}, \xi_k\}$. Together with the conditions of the continuity of the first derivatives at the points (x_k, y_k), relations (10) supply $3N$ nonlinear equations to find $3N$ unknowns, $\{(M_k^x, M_k^y, \xi_k), k = 0, \ldots, N-1\}$, in the case of the self-connected interfaces considered in this paper. Otherwise, additional boundary conditions should be imposed for the derivatives of the cubic spline at the boundary points (see [46]). When we solve this nonlinear system with the fully convergent Newton method [46], the Jacobian can be computed analytically. Two main problems concerning the solution should be emphasized. First, the solution can go outside the cell along the given edge. In this case, it does not keep the phase volume in the given cell but spreads it between two (or more) neighboring cells. If we encounter this situation, we accept the solution lying inside the interfacial cells that minimizes the residual of the nonlinear equations. This is in some sense similar to least-squares approximation to the solution in this situation.

Second, the solution of the nonlinear problem may or may not be unique and a special criterion should be used to fix one of the solutions. An example can be viewed in [56], where a continuous linear front reconstruction is uniquely defined due to a special fixing of the "interface contact" node at a no slip wall. We discuss in Section 7 how we fix the the solution in the case of the axisymmetric 2D bubbles where for any choice of the start point at the "nose" interfacial cell, the solution described just above can be found. If the second derivatives (M_k^x, M_k^y) are set equal to zero, the solution of the nonlinear system (10) corresponds to a *continuous linear interface* reconstruction which maintains the phase distribution. However, the same solvability problems still remain.

Below we describe how we define which edges of the interfacial cells should be cut by the continuous interface. Assume here that the interface should separate the domain into two subdomains (*inside* and *outside* the fluid) corresponding to $C = 1$ and $C = 0$. First, we construct the discontinuous linear interface with PLIC. Then, a preliminary parametric cubic spline, referred to as *spline A,* is drawn through the centers of the linear–slope segments. The cross points A_k of *spline A* with the cell edges determine the points $B_k(x_k^0, y_k^0)$ (see Fig. 1). The edges where B_k lie are those that are assumed above to be cut by the interface. A mass

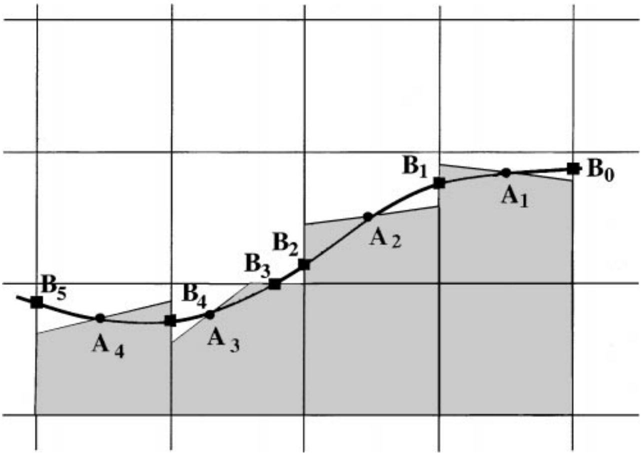


FIG. 1. Sketch for splines A and B. Spline A goes through the middle points of the PLIC interface (A_0, A_1, A_2, ...). Spline B goes through the intersections of Spline A with the grid's elements boundary (B_0, B_1, B_2, ...).

violating cubic spline, referred to as *spline B*, goes through the points B_k and provides an initial guess $(M_k^{x,0}, M_k^{y,0}, \xi_k^0)$ to cubic *spline C* which should preserve a phase distribution. We construct the *spline A* with the following algorithm:

- Choose a clockwise or counterclockwise order of the passing the interface. Assume here a counterclockwise direction where the gas phase ($C = 1$) is on the left.
- Reconstruct PLIC interface in each interfacial cell. With respect to chosen direction, denote its start point as p_s and end point as p_e. Mark all interfacial cells with "0."
- Choose arbitrarily an interfacial cell where the *spline A* should start.
- **I.** Denote the cell under consideration as $c.c$ (current cell). Mark it with a "1." Include the midpoint of the PLIC interface in the current cell in the subsequent points used to construct the *spline A*.
- **II.** To find the next point, analyze the neighbors of the current cell.

1. If the interfacial neighbor shares with the current cell an edge where $p_e(c.c)$ lies and if it is marked with "0," go to **I** (see Fig. 2a).

2. $Else$, examine the neighbors which share the edges where neither $p_s(c.c)$ nor $p_e(c.c)$ lies. If there is an interfacial neighbor marked with "0," go to **I** (see Fig. 2b).

3. $Else$, look for an interfacial cell $f.c$ (find cell) that satisfies the following four conditions:

(a) It is marked with "0."

(b) The gas phase is at the left (because of chosen counterclockwise direction) from the line connecting the $p_e(c.c)$ and $p_s(f.c)$.

(c) $\|p_e - p_s\|$ is smaller than any prescribed value eps.

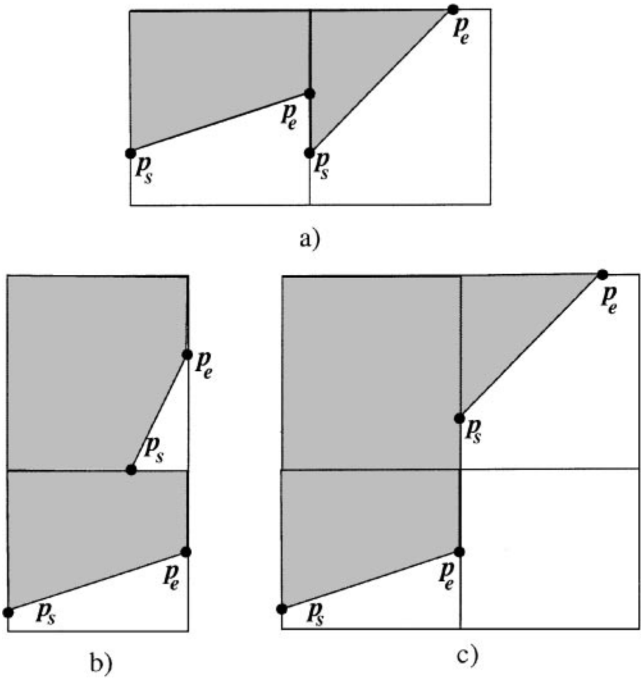


FIG. 2. Sketch for construction of spline A. (a) The interfacial neighbor shares with the current cell an edge where $p_e(c.c)$ lies. (b) The interfacial neighbor shares with the current cell an adge where neither $p_s(c.c)$ nor $p_e(c.c)$ lies. (c) Next cell is a neighbor.

(d) $p_s(f.c)$ is closest to $p_e(c.c)$ among the points which satisfy conditions (a)–(c). If such a cell is found, go to **I** (see Fig. 2c).

- **III.** The algorithm stops when no cell is found in **II.**3.

Let us comment on the algorithm above. If each interfacial cell has two or more interfacial neighbors, the algorithm always finds a subsequent cell to be visited by the spline in **II.**1–**II.**2. A case of only two interfacial neighbors is most frequent when the interface is adequately smooth (see, e.g., [3]). Some advection algorithms constrain themselves to a phase transfer from the partially filled cell to the empty cell (see, e.g., [51]). Interface smoothing is assumed only in **II.**3, which supposes that if the cell has no suitable interfacial neighbors or all of its neighbors are already visited, the next cell reasonably can be found in a given radius *eps*. In case of a self-connected *spline A*, the algorithm ends successfully if the first cell satisfies the conditions **II.**3.b-3.d. In cases when the gas phase consists of more than one disconnected part, start points lying at the interface of each part have to be chosen and the algorithm should be done for each part separately (see Fig. 3g).

4. COMPUTATIONAL AND ADVECTION GRIDS

We refer to a grid where basic equations are discretized and solved as a *computational* grid. The grid where time integration for vof is accomplished is referred to as an *advection* mesh. The regular virtual advection mesh, finer than the underlying regular computational grid, was also used by Rudman [61]. In our work, the computational grid is the finest grid of the hierarchy which can be composed from uniform, adaptively refined, and/or aligned grids. The advection mesh corresponds to the highest regular refining on the computational grid. Conceptually, the refinement levels of the advection mesh and of the computational

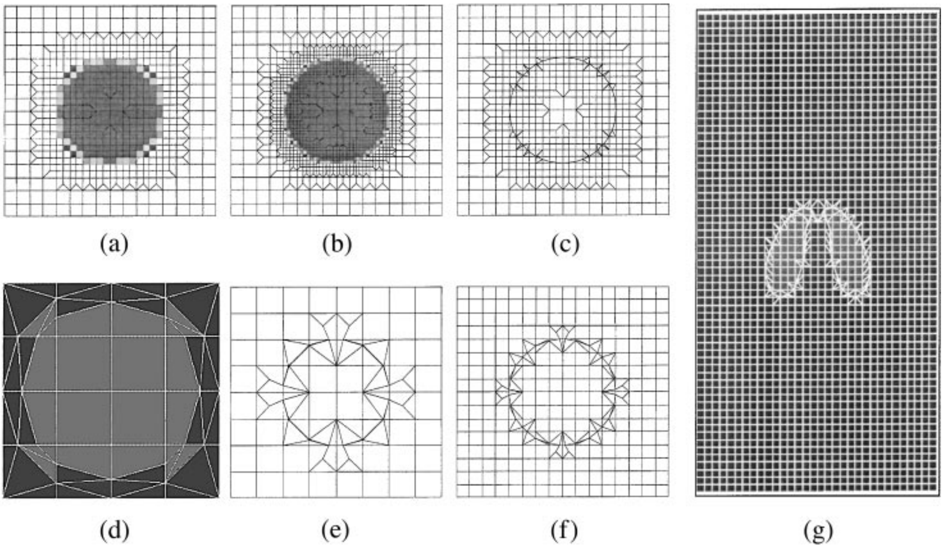


FIG. 3. Phase distribution C corresponds to circle $R = 0.25$ except in cases (d) and (g). One (a) and two (b) levels, adaptively refined grids. (c) *Spline B* aligned grid, constructed over one adaptive level. (d–f) PLIC aligned grids. (d) C corresponds to a square. Note that each "edge" along a circle in (e) hides the grid element. (g) Interface aligned grid fitted to *spline B* in case of bubble breaking up.

grid can be chosen independently. The transgrid operators discussed here and in Section 6 provide any necessary interpolations of the velocity from the computational to the advection mesh and C field interpolations, conversely.

4.1. *Computational Grids*

4.1.1. Adaptive refinement near the interface. Adaptive mesh refinement (AMR) becomes quickly popular in multiphase methods [12, 54]. Recently, Jeong and Yang [32] adopted adaptive refining through the refinement and mergence of elements on a quadrilateral mesh. Greaves and Borthwick [24] developed a panel division or removal of elements on a Cartesian mesh, dependent on the vorticity magnitude. M. Sussman *et al.* extended an adaptive projection method to hierarchy of adaptive nested regular grids. In their work, the cells where the curvature exceeds a prescribed threshold were refined. Our purpose was to minimize the computational expense while maintaining a high level of *uniform* refining at the domain of dependence of the interface. Our demand of the uniform refining comes from the use of the finite difference normal estimation in the element centers and corners. This is not required in the reconstruction algorithms designed on the unstructured grids. Moreover, we want to exclude at the present stage of a work with the aligned grids an additional error which could appear from the approximations between the differently refined adaptive subdomains. Also, the advantages of the adaptive refinement could be multiplied should the second-order interface reconstructions be used.

Assume that C is defined on a *regular base mesh* l_0 with step size h equal to 2^{-l_0}. Consider its element k. Our refinement criterion is if $\|\vec{n}(\vec{r})\|$ is different from zero for some corner $\vec{r}$ of the given element k, we mark the element k itself for regular refinement, as well as its neighbors which contain the given node $\vec{r}$. The regular marked element is subdivided into four regular *son* elements. When all regular refinements are done, some irregularly refined elements are placed in their neighborhood to guarantee the closure of the grid (see Figs. 3a and 3b).

The quantity of gas phase inside the finer element is estimated as the area of its intersection with the PLIC polygon constructed inside a father element. Since the vicinity of any interfacial cell is regularly refined, the adaptive refinement procedure can be applied anew. After repeating it l_{adap} times, we obtain a hierarchy consisting of l_{adap} adaptively refined grids. Here, l_{adap} takes any prescribed or dynamically estimated value. The procedure results in a smooth expansion of the refined layers from the maximal adaptive level to basic level.

4.2. *Interface Aligned Grid*

We are not making contrasts between aligned grids and AMR. On the contrary, we underline that these approaches supplement and amplify each other and we use them both. There are examples when AMR alone cannot help to overcome the difficulties, for example, in the case of Laplace law. The idea of an interface aligned grid is to completely separate the phases by the boundaries of the geometrical elements. This is fulfilled if the element face should lie along the reconstructed interface (see Fig. 3d). If the grid were aligned with the PLIC, very narrow elements with large angles or high aspect ratios would appear following closure of the grid because of the discontinuity of the reconstruction (see Figs. 3e, 3f). When the interface is represented by a smooth curve, its piecewise approximation by a slope segment going through the intersections of the curve with the grid's elements boundary gives rise to such an aligned grid (see [*CD*] in Fig. 5). The aligned grid is obtained with the irregular refinement of the cells which are intersected by a curve. We refer below to this simple grid adaptation to the front as "spline *rectifying* grid" (see Figs. 3a–3c and 3g).

The spline aligned grid may happen to contain "bad" elements which deteriorate the performance of the solution procedure. Similarly to [56], we have applied a simple remedy: two nodes merge together if the distance between them is smaller than εh. Typically, we have experimented with $\varepsilon < 0.1$. More effective methods are used in [40], where the internal nodes have to be moved in order to improve the grid properties. The criteria on the lengths of the cells proposed very recently in [23] present the possibility of covering a region with a strong curvature by small elements.

4.3. *Time Integration on Advection Mesh*

Let initial conditions $\vec{u}^0$ and p^0 be given at $t = t_0$. By using the initial distribution C^0, the first approximation to the solution $\vec{u}^{\Delta t}$ is computed. Then, with the mean velocity $\frac{1}{2}(\vec{u}^0 + \vec{u}^{\Delta t})$, C^0 is advected to its new position $C^{\Delta t/2}$. The solution $\vec{u}^{\Delta t}$ is recalculated then with $C^{\Delta t/2}$. Then $C^{t+\Delta t/2}$ is used to discretize the governing equations in the time interval $(t, t + \Delta t)$. In particular, deformable computational grids correspond to $C^{t+\Delta t/2}$. In numerical experiments with the rising bubbles we advance C on the regular advection mesh, using a 2D geometrical split approach similar to one implemented in the SURFER code [26]. The advection mesh has a space step $h = 2^{-(l_0+l_{adap})}$, corresponding to the maximal regular refining of the neighborhood of the interface on the computational grid. In the case of the aligned grids, velocity is projected first on the underlying mesh (uniform or adaptively refined), with help of the same transgrid operators as for solution procedure. Near the interface where the advection schemes are really active, no further velocity interpolation is needed. When the advection scheme needs a divergence-free velocity field outside the interface region, the interpolation to the insufficient point is done as follows.

Irregular refined elements are formally replaced by four regular son cells. Requiring the velocity components lying on the edges of the new cells to be divergence-free in the sense of (11), they are locally computed from the velocity solution inside this element. This reconstruction from the irregular to regular sons is based on the relation (21). Final interpolation inside regular *coarse* cells lying on the different levels l, $l_0 \leq l < l_0 + l_{adap}$, is done with the approximations [37], in x direction for u_x and in y direction for u_y. This approximation is piecewise constant along each edge and preserves the continuity property (11) for son elements. They are also used in [61] to interpolate the solution from the staggered computational grid to finer advection mesh.

When new phase distribution is computed on the advection mesh, the mesh may be coarsened to obtain a new hierarchy of the adaptively refined grids. In practice we proceed as follows. First, we project C on the base level l_0 and consequently construct the adaptively refined grids as described above. Using again the regular refining near the interface, the volume fractions on the highest regular grid l, $l = l_0 + l_{adap}$, are redefined as a copy from the advection mesh. In this way, they correspond to the computed solution without any approximations. The projection on the base level enables us to reconstruct the adaptive grid without coarsening of its previous elements. When $l_{adap} > 1$, C is redefined on the underlying adaptively refined grids as a subsequent projection from the finer level. While using the flux based time integration schemes on unstructured grids, extension of CFL criterion [22, 32] can lead to very small time steps because of the presence of the elements with small characteristic lengths.

5. DISCRETIZATION OF BASIC EQUATIONS

5.1. *Staggered Finite Volumes*

To approximate Navier–Stokes equations with surface tension, we use finite volume discretization based on the nonconforming Crouzeix–Raviart finite element [18]. The finite volume approach helps us to separate the phases. Moreover, it deals with the fluxes directly and in this way brings more control over them than the finite element approach. Similar to the well-known staggered marker and cell distribution [28] of the discretization points, the unknown pressure is located at the center of the element and the velocity variables are found in the middle of its edges. However, in contrast to the MAC model, each edge contains *both* components of the velocity. This approach enables us to treat the basis equations in the same manner on structured (uniform) or unstructured grids. The velocity is considered to be piecewise linear (or bilinear) with respect to the so-called "rotated element" [48], which connects the centers of the edges. The pressure is considered to be piecewise constant over the element. The theoretical estimates [9, 18, 48] show that such approximations of the Stokes equation are stable; i.e., they satisfy the so-called discrete Babuska–Brezzi–Ladyzhenskya condition (BBS) and have the second-order convergence rates for the velocity components and first-order for pressure components. When the approximation of the convection term in the Navier–Stokes equation does not spoil the properties of the stiffness matrix, needed for stability, the above results are also extended to Navier–Stokes equations. This discretization can be generalized also to 3D case following [9]. Unlike with a collocated approach [59] or a penalized formulation, since no additional coupling between the pressure and the velocity is needed [see (11), (20)], the continuity equation is satisfied exactly. This enforces mass conservation in advection schemes, especially if they are flux-based. When the element faces lie along a front, the correct description of pressure jumps follows since pressure discretization points are not found on the element boundary. Below we briefly view the main steps of the discretization procedure.

- With the finite volume method, mass conservation Eq. (2) is integrated over a control volume $CV_p(\vec{r})$ which coincides with the geometrical element with the center $\vec{r}$. The Gauss theorem reduces the integration to a boundary integral over ∂CV_p. Its approximation with quadratures involves the velocity variables $\vec{u}(ip)$ lying on the boundary of the element (see points ip_0-ip_3 in Fig. 4) as

$$\sum_i (\vec{u} \cdot \vec{s}_p)|_{ip_i} = 0, \quad ip_i \in \partial CV_p. \tag{11}$$

Below, n_α denotes the outer unit normal to the boundary segment ∂CV_α of a control volume; s_α is equal to n_α scaled with the length of the segment.

In order to discretize the Navier–Stokes equation (1), we integrate it over control volumes $CV_u(\vec{r})$ now constructed for each velocity discretization point $\vec{r}$. The control volume is subdivided into the subcontrol volumes *SCV*; each *SCV* belongs only to one geometrical

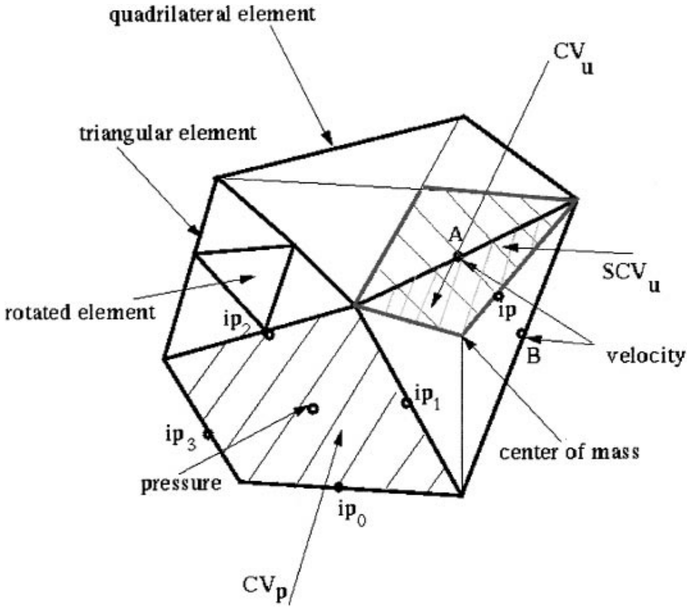


FIG. 4. Sketch for staggered non-uniform grid.

element (see Fig. 4). With the help of the Gauss theorem we represent Eq. (1) as

$$\rho_u \int_{CV_u} \frac{\partial \vec{u}}{\partial t} + \rho_u \oint_{\partial CV_u} \mathbf{V}\vec{n}_u\, ds + \oint_{\partial CV_u} \{p - 2\mu\mathbf{D}\}\vec{n}_u\, ds$$

$$= \rho_u \int_{CV_u} \vec{g} + \int_{CV_u} \vec{F}_s; \;\; \mathbf{V} = \vec{u} \otimes \vec{u}. \tag{12}$$

- The pressure is assumed to be an elementwise constant function. Density ρ and viscosity μ are computed as a volume averaging inside each geometrical element:

$$\rho = C\rho_g + (1 - C)\rho_l; \quad \mu = C\mu_g + (1 - C)\mu_l. \tag{13}$$

The density $\rho_u(\vec{r})$ is derived from (13) by averaging over two subcontrol volumes sharing the given velocity discretization point $\vec{r}$. This averaging rises from a nonconservative form chosen here. The conservative formulation requires a special effort to compute correctly the mass fluxes in case of large-density ratios on the boundary between the control volumes (see [61]).

Spatial velocity approximations are fulfilled on so-called *rotated* elements [48]: the corners of the rotated element coincide with the velocity discretization points (see Fig. 4). Let the finite element (f.e.) approximation of arbitrary scalar or vector field ϕ inside or outside a given triangular/quadrilateral element be written as

$$\phi^{f.e.}(\vec{r}) = \sum_{j=1}^{m} N_j(\vec{r})\phi_j. \tag{14}$$

Here, N_j are standard linear/bilinear basis functions; m is the number of the element corners. While using (14), we will distinguish between f.e. interpolation over the geometrical element and f.e. *rotated* interpolation when basis functions correspond to the rotated element. In both cases, the gradients of ϕ are represented as

$$\nabla\phi^{f.e.}(\vec{r}) = \sum_{j=1}^{m} \nabla N_j(\vec{r})\phi_j. \tag{15}$$

The rotated f.e. interpolations (14)–(15) are used to compute the velocity and its gradients at integration points lying on the ∂CV_u. Note that in the case of the interface aligned grid, when no phase mixture is presented by the relations (13), Eq. (12) with the relations (14)–(15) enables us to satisfy the tangential conjunction condition (6) exactly on the reconstructed interface. In particular, stationary linear two-phase Couette flow is exactly described for any viscosity ratio and any inclination of the interface, provided that the interface is exactly reconstructed.

- If neither the equation of the interface nor its curvature is known, $\vec{F}_s$ can be applied in the form (5) on any computational grid:

$$\int_{CV_u} \vec{F}_s = -\oint_{\partial CV_u} \mathbf{T}\cdot\vec{n}_u\, ds. \tag{16}$$

Following [34], we approximate δ_S as $|\nabla C|$. The components of the tensor $\mathbf{T}$ at the integration points are then computed with f.e. relations (14) from their nodal values, obtained in the corners of the finest regular element to which the point belongs. If the interface is represented by a parametric spline $(x(s), y(s))$, the surface tension force over the interface segment $(\mathbf{A}, \mathbf{B})$, lying inside the control volume (see sketch for Frenet formulas in Fig. 5), can be computed as

$$\int_{CV_u} \vec{F}_s = \oint_{A}^{B} \sigma\kappa\vec{n}\, ds = \sigma(\vec{t}_B - \vec{t}_A); \quad \vec{t} = \frac{(x'(s), y'(s))}{(x'^2(s) + y'^2(s))^{\frac{1}{2}}}. \tag{17}$$

In the case of the staggered aligned grid, the intersection points of the CV_u with a spline coincide with the corners of the grid elements by the construction. No extra effort is needed to correctly discretize the pressure jump, compared to the situation that occurs in this case on the regular grids (see [45]).

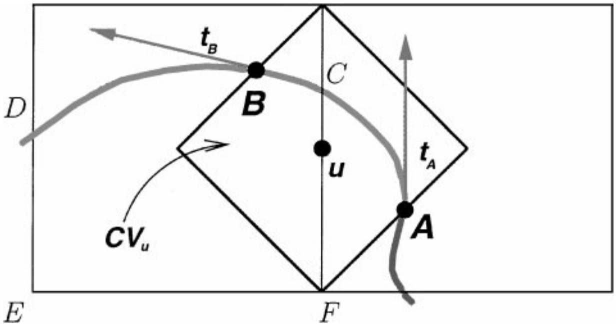


FIG. 5. Sketch of bisection of spline with control volume $CV_u(\vec{r})$.

5.2. *Time Discretization and Discussion*

The velocity in the continuity equation (11) is usually treated with an implicit Euler time approximation. Let ψ stand for p, **D**, or **V** in (12). Then, our time discretization can be written in a general form as

$$\psi = \alpha^{+}\psi^{t+\Delta t} + \alpha^{-}\psi^{t}. \tag{18}$$

Here, $\alpha^{+} = 1$, $\alpha^{-} = 0$ for the implicit Euler scheme and vice versa for the explicit Euler method; $\alpha^{+} = \alpha^{-} = \frac{1}{2}$ for the Crank–Nicolson technique. When $\mathbf{V}(\vec{u})$ is approximated with the implicit schemes, a nonlinear system of equations arises. We solve it with the Newton method using a linear search technique. The Jacobian of the system is analytically computed without any linearization of the convective term. The initial guess is the solution obtained at the previous time step or its approximation on a new computational grid. As an alternative treatment of the convection term, we linearize it with respect to time:

$$\mathbf{V}^{t+\Delta t} = \mathbf{V}^{t} + \frac{\partial \mathbf{V}^{t}}{\partial \vec{u}}\frac{\partial \vec{u}}{\partial t}\Delta t; \qquad \frac{\partial \vec{u}}{\partial t} \approx \frac{\vec{u}^{t+\Delta t} - \vec{u}^{t}}{\Delta t}. \tag{19}$$

The second-order time linearization scheme (19) preserves the stability properties of the implicit approach and avoids the necessity of solving the nonlinear system of equations.

The capabilities of staggered finite volume discretization have to be tested, first, in one-phase simulations. Because of space restrictions, numerical rate of convergence is considered here only for one of examples from [6]. The velocity field $\vec{u}^{0} = (-\partial\psi^{0}/\partial y, \partial\psi^{0}/\partial x)$, $\psi^{0} = \pi^{-1}\sin^{2}\pi y \sin^{2}\pi x$ is initialized inside the unit square domain. The velocity is equal to zero at the boundary. Two cases are considered: Stokes flow ($\mu/\rho = 0.1$, convection term is dropped) and Reynolds number 100 ($\mu/\rho = 0.01$). Since no upwind approximation is included in this paper, inviscid flow is not considered. In each case, we compute solutions on the regular grids ($h = 1/2^{n}$, $n = 4, \ldots, 8$) with $\Delta t = h/2$ (CFL $= 1/2$). Time discretization of the pressure and the diffusion terms is done with the Crank–Nicolson scheme; the convection term is described with the relations (19) in the case of Re $= 100$. Table I displays the order of the velocity approximation, $\sqrt{U_{2h,4h}/U_{h,2h}}$, measured at time $T = 0.5$. Here, $U_{h,2h}$ is the L_2 norm of the pointwise difference between the velocity solution obtained on the grid with step $2h$ and the velocity projected from the finer grid. Velocity projection is done with the relations (22). We observe asymptotical second-order convergence, which shows that we are in desired regions of stability and asymptotic approximations, at least in moderate Reynolds numbers.

TABLE I
Convergence Results for Single Vortex

Case	16–32	Rate	32–64	Rate	64–128	Rate	128–256
Stokes	6.41e-4	1.99	1.62e-4	1.99	4.07e-5	1.99	1.03e-5
Re = 100	2.78e-2	1.72	9.39e-3	1.91	2.59e-3	1.98	6.60e-4

5.3. *Solution Procedure*

At each time step, the discretization described above leads to a saddle-point problem (SPP) for velocity and pressure unknowns:

$$\begin{pmatrix} A & B \\ B^t & 0 \end{pmatrix} \begin{pmatrix} v \\ p \end{pmatrix} = \begin{pmatrix} f \\ 0 \end{pmatrix}. \tag{20}$$

Matrix A describes $\vec{u}$-$\vec{u}$ coupling of the discretized momentum equation, B addresses its $\vec{u}$-p counterpart; matrix B^t denotes the discretization of the continuity equation. Although the SPP brings more difficulties to the solution procedure compared with the solution of symmetric positive definite systems (SPD), they are well developed and the effectiveness of the modern state-of-the art algorithms is compared with the ones for SPD systems [7]. In particular, the multigrid (MG) approach [8, 27, 68, 70] has been proven to be very effective for SPP; i.e., its convergence rate does not depend on the discretization parameter (number of unknowns) and the jumps of the coefficients, at least for “good” grids. The “bad” grids, i.e., where the elements with big angles or high aspect ratios present, can result in bad convergence rates of the iterative solvers. This is true for the SPD as well as the SPP problem. There are approaches to overcoming this difficulty (algebraic multigrid, block preconditioner depending on the anisotropy). In our computations we do not use such sophisticated procedures but, rather increase the number of smoothing steps.

In our work we use the V-cycle of the MG as a preconditioner for BiCGstab method [4, 58]. As the MG smoother, we use either SIMPLE transforming operator of Wittum [70] or a *collective* Gauss–Seidel technique following Vanka [67]. The prolongation of the velocity component is done with the damped rotated f.e. relation (14). The prolongation of the pressure component is constant elementwise. The restriction is computed as the transposition to the prolongation in both cases. In order to reduce the residual about 10^6–10^7 times, the typical convergence of BiCGstab observed in our computations was 5–10 iteration steps on uniform and adaptively refined grids and 15–30 steps on the spline aligned grids, by using 1–4 pre- and post-smoothing iterations on each multigrid level. A comparison of two solution techniques ([70] and [67]) in different real problems and in functions of element properties is under study and will be reported in a separate paper.

6. TRANSGRID SOLUTION INTERPOLATIONS

Once the velocity field $\vec{u}^t$ is computed, the phase field $C^{t-(1/2\Delta t)}$ is advanced to $C^{t+(1/2\Delta t)}$. The new phase distribution requires the construction of a new computational grid unless it is uniform. The discretization of the basic equations in the next time interval $(t, t+\Delta t)$ requires the old solution $\vec{u}^t$ and, eventually, p^t at the new discretization points. The old solution can be interpolated from the previous computational grid or prolongated from a coarse grid which is not altered between two time steps. We discuss now some transgrid operators used in computations presented below.

6.1. *Velocity Approximation*

Assume that the velocity solution is given at the discretization points of some computational grid. The solution $\vec{u}(\vec{r})$ at an arbitrary point $\vec{r}$ can be computed by using the f.e. relations (14) on the rotated element of the geometrical element where $\vec{r}$ lies. Then, the averaging is done over all geometrical elements to which the point belongs. Below, we use this approximation scheme to interpolate the velocity from the old computational grid to new discretization points. The continuity equation (11) will not be generally satisfied by the interpolated solution on a new computational grid.

6.2. *Divergence-Free Velocity Restriction*

Consider some father element and the set of its son elements lying on the finer grid. Assume that the velocity in discretization points $\vec{r}_k$ of the son elements obeys (11). Since any son velocity discretization point, not lying on the father boundary, belongs to two son elements, the sum of the relations (11) over all son elements is reduced to the summation over the son discretization points lying on the boundary ∂CV_p of the father element:

$$\sum_k (\vec{u} \cdot \vec{s}_p)|_{\vec{r}_k} = 0, \quad \vec{r}_k \in \partial CV_p. \tag{21}$$

We put the velocity $\vec{u}$, defined at the edge of the father element, as equal to the weighted average of the velocity at son discretization points $\vec{r}_k$ *lying on the same edge*:

$$\vec{u} = \frac{1}{\sum_k \|\vec{s}_p(\vec{r}_k)\|} \sum_k (\|\vec{s}_p\| \vec{u})|_{\vec{r}_k}. \tag{22}$$

Due to relation (21), the velocity (22) is divergence-free in the sense of (11) w.r.t. the father element. Starting from the finest level and proceeding in this way, one can construct divergence-free fields on all underlying levels. We use this projection to compute the stiffness matrix on coarse grids required for the multigrid technique and to advance the phase field on the underlying base mesh in case of the aligned grid.

6.3. *Pressure Interpolation*

We need the restriction of the pressure solution to coarser levels both for the multigrid method and for its expansion to new discretization points. Projection of the pressure is simple: the arithmetical mean pressure value over all son elements is appropriated to their father. In order to approximate the pressure in centers of new geometrical elements, we proceed as follows. First, we approximate the pressure solution at the nodes of the old computational grid as the arithmetical mean of the pressure values over all elements which share the given node. Then, we project this nodal solution to all coarser levels. Once a hierarchy of new grids is reconstructed, the nodal solution is interpolated from the finest unalterable level into the son nodes with f.e. relations (14). The solution at the new discretization points, i.e. the centers of new elements, is computed as an arithmetical mean value of the nodal solution. The procedure is then successively repeated for subsequent finer levels.

7. LAPLACE LAW

The validation of the Laplace law for a stationary bubble represents a well-known test of surface tension methods. At equilibrium, the pressure jump across the interface is related to its radius R according to (6)

$$p_{\text{in}} - p_{\text{out}} = \frac{\sigma}{R}. \tag{23}$$

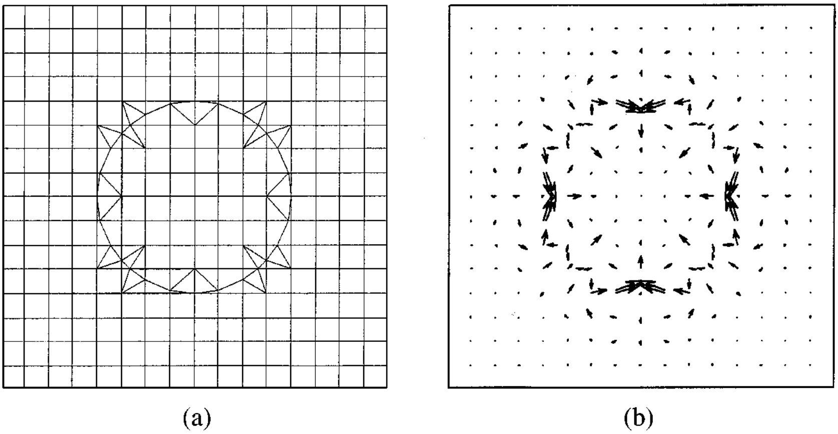


FIG. 6. Anomalous currents (b) on the grid (a) aligned with the exact circle. The surface tension force is computed in momentum conserving form (16), (5).

Here, p_{in} and p_{out} are pressure values inside and outside the bubble. We set $\overline{p}_{\text{in}}$ and $\overline{p}_{\text{out}}$ to be equal to a weighted mean value computed over all elements lying completely inside or outside the bubble; the weight coefficient is equal to the element area. Well-known numerical artifacts, called in the literature "anomalous" or "spurious" currents around a macroscopically static bubble, are typical for VOF models. In order to demonstrate that the anomalous currents originate from a bad approximation of the surface tension term, we consider the two following experiments. The physical data are given in lattice units in this section.

First, we rectify a circle of the radius $R = 0.2$, centered in the unit square, by a *continuous linear* interface. The aligned grid follows this reconstruction, as displayed in Fig. 6a. Both fluids have the same densities and viscosities equal to one. The surface tension force $\vec{F}_s$ with $\sigma = 1$ is applied in both forms: conservative [(5) and (16)] and exact [(17)]. The amplitude of the anomalous currents, $\max(\|\vec{u}\|)$, and measurements of the Laplace law are given in Table II. The behavior of the currents on the aligned, adaptively refined or regular grids is similar when the relations (5), (16) are used: $\max(\|\vec{u}\|)$ is proportional to σ/μ_l and it varies weakly if the space resolution increases. This follows the same lines as the results

TABLE II
Results for a Static Bubble

	Amplitude of the anomalous currents			
	$\vec{F}_s$: rel. (16), (5)	$\vec{F}_s$: rel. (17)		
h	Exact circle	Exact circle	*Spline B*	*Spline C*
1/16	0.0413	8.67×10^{-9}	1.00×10^{-2}	1.64×10^{-3}
1/32	0.0293	9.22×10^{-9}	6.00×10^{-3}	1.89×10^{-4}
1/64	0.0289	1.23×10^{-8}	1.76×10^{-3}	4.89×10^{-5}
	Laplace law: $(\overline{p}_{\text{in}} - \overline{p}_{\text{out}})\frac{R}{\sigma} = 1$			
1/16	0.936	1.0	1.00282	0.999905
1/32	0.969	1.0	1.00087	1.000010
1/64	0.988	1.0	1.00019	1.000000

in [34, 35] where the anomalous currents obtained with finite difference approximations of relation (16) on uniform grids stay constant when space resolution increases. In this way, one cannot benefit from the adaptively refined grids.

On the other hand, since the pressure gradient across the interface is correctly described on the aligned grid where phase separation inside of a control volume is done by the boundary between two elements, the anomalous currents vanish when relation (17) is used with the exact curvature and exact values of normal vector $\vec{n}$. Consequently, an exact pressure distribution (23) is obtained in this case (see second column in Table II).

Second, we construct the cubic splines B and C, corresponding to the volume fraction distribution for a circle $R = 0.2$. In order to fix *spline C*, we iterate the solution of the nonlinear problem until the curvature at the nose becomes equal to one at the neighboring point. $\vec{F}_s$ is implemented then in the form (17) on the spline aligned grids.

Rudman [61] and Aleinov and Puckett [2] do not comment on how much the amplitude of the currents is reduced with their approaches. Williams *et al.* [69] reduce them by factors of 3 to 5, using an approach similar to [2, 6]. Table II shows that the amplitude of the spurious currents is reduced by a factor of 20–500 on *spline C* and by a factor of 4–10 on *spline B* , even on relatively coarse grids. The currents are better reduced in case of the mass conserving *spline C*. In both cases, the current strength decreases with the spatial resolution. These results confirm the ability of cubic spline aligned grids to represent the surface tension accurately.

8. RISING BUBBLES

8.1. *Dimensionless Parameters*

The rising bubbles are often classified in terms of the following group of dimensionless parameters: the Reynolds number Re, the Froude number Fr, and the Weber number We:

$$\mathrm{Re} = \rho_l LU/\mu_l, \quad \mathrm{Fr} = U^2/gL, \quad \mathrm{We} = \rho_l LU^2/\sigma, \tag{24}$$

where U is the steady state bubble velocity. On the other hand, the descriptions in terms of the Reynolds, the Eötvös number E_0, and the Morton number M are also widespread:

$$E_0 = g\rho_l d^2/\sigma, \quad M = g\mu_l^4/\rho_l\sigma^3. \tag{25}$$

Here, $d = 2R$, R being the bubble radius. In order to introduce the first group, let us rewrite Eq. (1) in the following dimensionless variables:

$$x' = x/L, t' = tU/L,\ \vec{u}' = \vec{u}/U,\ p' = p/(\rho_l U^2),\ \rho' = \rho/\rho_l \text{and}\ \mu' = \mu/\mu_l. \tag{26}$$

If we substitute (26) into (1) and drop the primes, it becomes

$$\frac{\partial\vec{u}}{\partial t} + \nabla\cdot(\vec{u}\otimes\vec{u}) = \frac{\vec{g}_u}{\mathrm{Fr}} + \frac{1}{\rho}\left[-\nabla p + \frac{\vec{F}_s}{\mathrm{We}} + \frac{1}{\mathrm{Re}}\nabla\cdot(2\mu\mathbf{D})\right]. \tag{27}$$

The dimensionless density and viscosity outside the bubble are equal to one; $\vec{g}_u$ is a unit gravitation vector. When it is not specially indicated, we compute the parameters (24) with

$L = 2R$. If U is put equal to $(Lg)^{1/2}$, then We turns out to be the Eötvös number E_0 and Re becomes equal to Re_0,

$$\mathrm{Re}_0 = (2R)^{\frac{3}{2}}\sqrt{g}\rho_l/\mu_l. \tag{28}$$

A dimensionless combination of these parameters, E_0^3/Re_0^4, yields the Morton number M. On the other hand, M is $\mathrm{We}^3/(\mathrm{Re}^4\mathrm{Fr})$ and E_0 is We/Fr when $L = d$. In fact, the Eötvös number is a dimensionless size of a bubble; it belongs to the interval $(10^{-2}, 10^3)$. The Morton number depends only on the properties of the liquid; it varies strongly depending on the viscosity of the surrounding phase, from 10^{-14} for liquid metals up to 10^8 in viscous oils. The parameters of an air bubble in water ($\mu_l = 1.137 \times 10^{-2}$ g/(cm s), $\rho_l = 1.0$ g/cm^3, $\sigma = 72.8$ dynes/cm) give rise to $M = 4.25 \times 10^{-11}$. In [63, 66], the reader can find an exhaustive list of the literature concerning numerical simulations of a single bubble.

8.2. *Numerical Experiments*

If not specially indicated, the conditions of the calculations are as follows. The dimension of the box is 1×2 cm; the initial bubble radius is $R = 1/6$ cm; the start position of its center is at (0.5 cm, 0.75 cm); $g = 980$ cm/s^2; the density of the liquid $\rho_l = 1.0$ g/cm^3. Bubble velocity U is computed from the change in the position of the bubble nose using the central differential in time. After the advection step, if necessary, we adjust C to be in the interval $(0, 1)$. This may result in total volume (or *mass*) violation, otherwise preserved by the algorithm. The relative loss $M^{\mathrm{rel.}}(t)$ of the total mass Ma of the gas phase is defined as

$$M^{\mathrm{rel.}}(t) = \frac{Ma(t=0) - Ma(t)}{Ma(t=0)}. \tag{29}$$

The time step satisfies the CFL condition on the advection mesh ($\max(\|\vec{u}\|)\frac{\Delta t}{h} < \frac{1}{2}$) and the restriction given by Brackbill *et al.* [13], coming from the stability condition at capillary level.

An implicit Euler time discretization is used in both conservation relations (11) and (12). No upwind approximation is used. The convection term is discretized with relation (19). We have not found any noticeable difference between these solutions and implicit Euler discretization, at least for the medium range Reynolds numbers under consideration. Surface tension force is implemented in the exact form (17) on the spline aligned grids and on the adaptively refined grids used in Section 8.2.4. Important here is while computing the surface tension in a form (16), (5), the solution was destroyed by the anomalous currents in this last experminent. Implementation of the surface tension on the cubic splines helped us to obtain the stable solutions.

8.2.1. Study of the interface interpolants. We consider here a relatively large bubble with $E_0 = 40.1$ and $M = 125$. Other dimensionless parameters of the experiment are $\mathrm{Re}_0 = 4.77$, $\mathrm{Re} = 0.88$, $\mathrm{We} = 1.4$, $\rho_l/\rho_g = 40$, $\mu_l/\mu_g = 88$. The streamlines at the steady state are shown in Fig. 8 in reference frame moving with the bubble. Mass violating *spline B* and mass conserving *spline C* are used to represent the interface, whereas the solution is obtained on a regular 32×64 grid. *Spline C* is fixed in the same manner as in Section 7 for the case of a stationary bubble. We show in Fig. 7 that even in the case of the smooth bubble shape variation, *spline C* happens to oscillate more than *spline B*. We suggest that

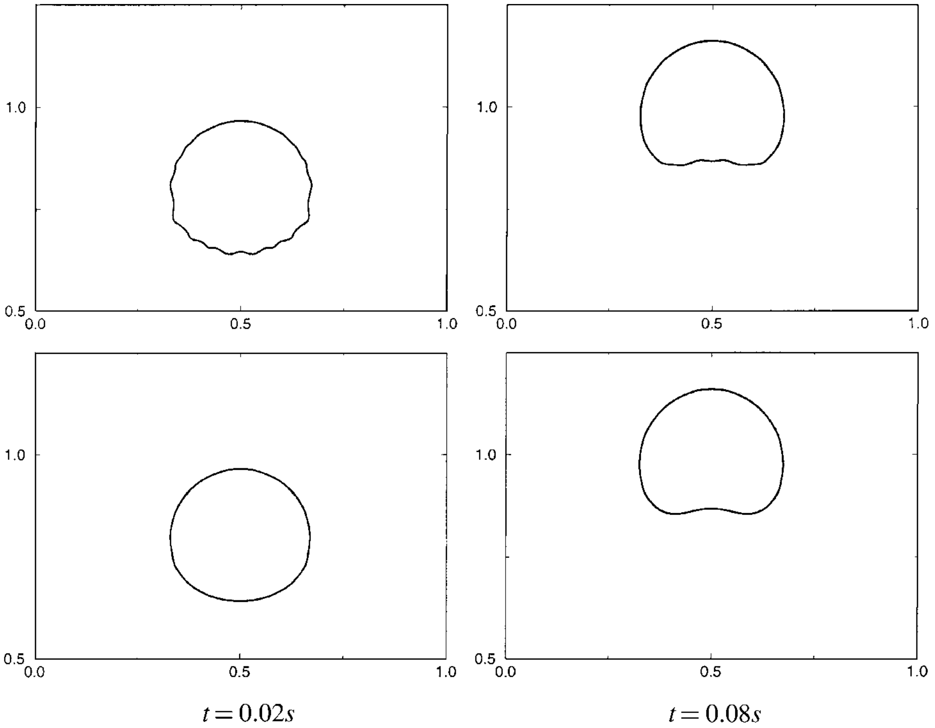


FIG. 7. Mass violating *splines B* (bottom row) and mass conserving *splines C* (top row) in case $M = 125$, $E_0 = 40.1$, $\mu_l/\mu_g = 88$.

the best mass conserving spline should be one with the minimal curvature. Computing the surfaces of minimal curvature [12, 15] is based on the idea that while evolving with the speed equal to $-\kappa$ (κ being a local curvature), the surface will come to a "minimal" surface. In [15], the surface is attached to a fixed "frame." In our case, we conjecture that a frame

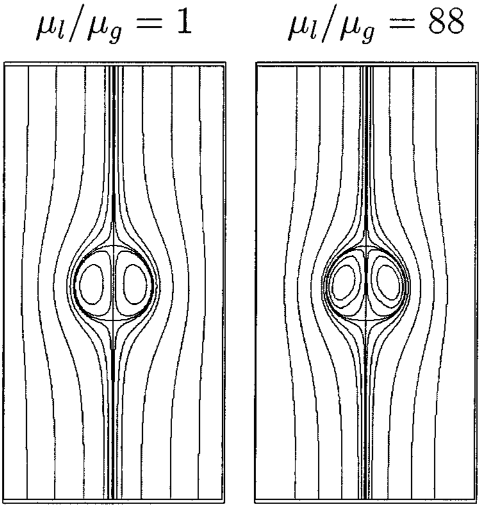


FIG. 8. Streamlines in case of rising bubble with $M = 125$, $E_0 = 40.1$.

could correspond to a moving set of points which satisfy the nonlinear system of equations for mass preserving spline. Should our conjecture succeed, that the best mass conserving spline is one with the minimal curvature, no additional criterion will be needed.

In all simulations below, we align the grid to the interface approximated by *spline B*. We confirm that even in the case of strong deformations, the relative mass violation caused by *spline B* on rather coarse grids ($h = 2^{-5}$) is less than 1%. On the other hand, the mass violation induced by the "spline rectifying" on the aligned grid has usually the same order of magnitude whether it is fitted to a *spline B* or to *spline C* interface interpolant. Thus, if no finer approximation to the spline interface is done by the subsequent refinement, the reconstruction of the interface with *spline B* is justified from a practical point of view. In the simulations presented below, cubic *splines B* are found smooth enough to ensure an accurate evaluation of the curvature. The convergence rates of the normal and curvature estimations obtained from *spline A*, *spline B*, and *spline C* interpolants have to be compared with the results presented in [2, 42, 53] and further studied for distorted interfaces as well.

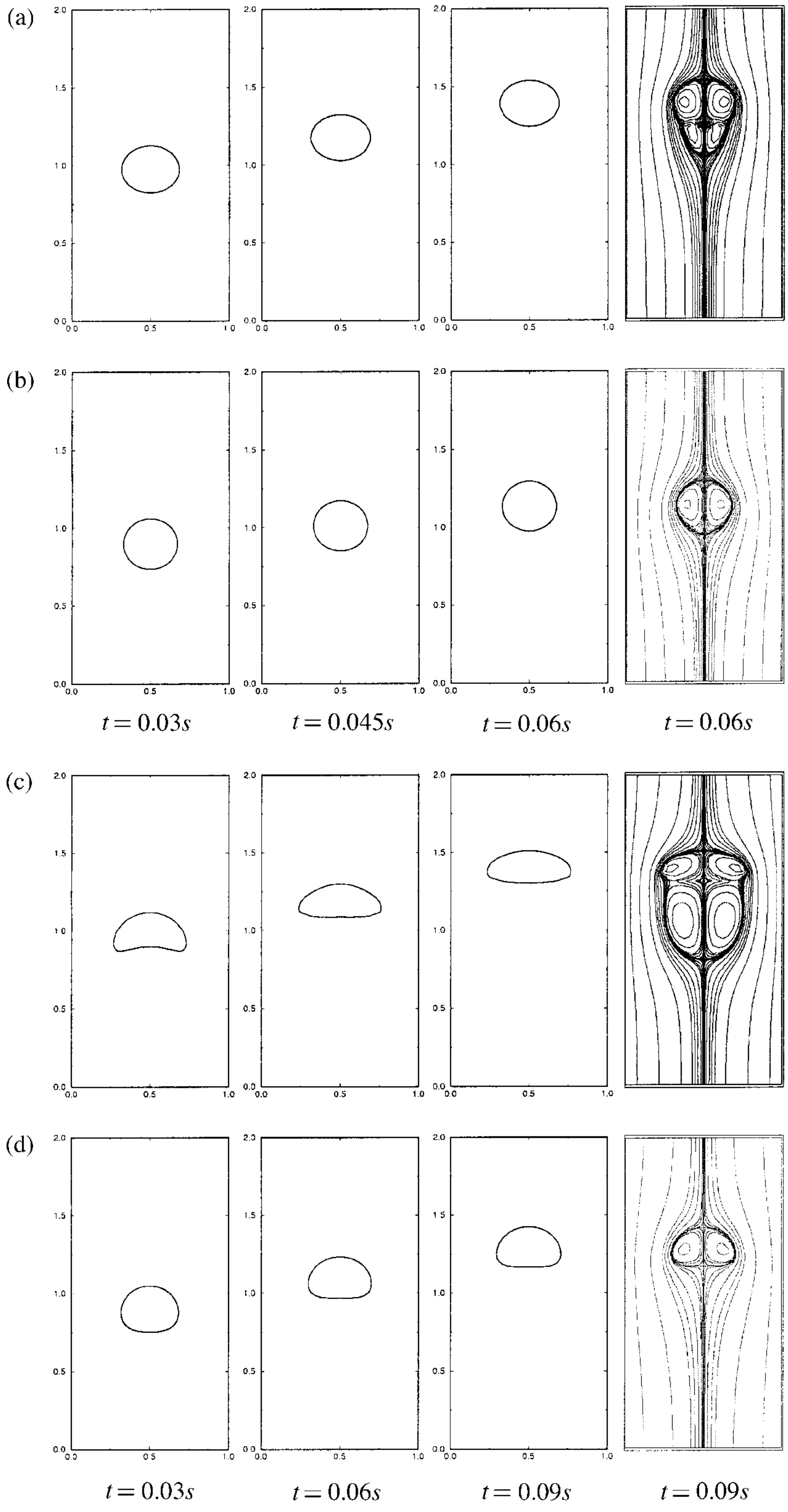
8.2.2. First experiment. Buoyant two-dimensional bubbles with three different Eötvös numbers: $E_0 = 1$, $E_0 = 10$, and $E_0 = 104$ are modeled by Unverdi and Tryggvason [66] with a front-tracking method. They use a finite difference MAC-type discretization on a 65 × 129 regular grid. Four experiments with different M are considered for each E_0. The Morton number M decreases with the fluid viscosity μ_l, whereas the bubble viscosity μ_g and the density ratio are kept constant; so μ_l/μ_g decreases from the calculations with higher to smaller M.

Here, we calculate the bubbles with the same E_0, M, μ_l/μ_g, and $\rho_l/\rho_g = 40$ as in [66]; free-slip boundary conditions are used. The results are shown in Figs. 9 and 10. All calculations, except those of the skirted bubbles, are done on *spline B* aligned computational grids, constructed over 32 × 64 regular mesh. Skirted bubbles displayed in Fig. 10e are obtained on a regular 64 × 128 regular grid while the spline technique has not yet been implemented to deal with the reconnection. The time step $\Delta t = 5 \times 10^{-5}$ s except for Fig. 9b where it is doubled. Relative mass loss (29) is given at $t = 0.06$ s. When the moving is done on the same mesh and with the same time step, $|M^{\text{rel.}}|$ ncreases with Re (cf. $M^{\text{rel.}}$ in cases (c) and (d)).

At the steady state, bubble shapes and streamlines are found in good qualitative agreement with the results displayed in Figs.3 and 4 in [66]. Consider first small bubbles or the bubbles with high surface tension when $E_0 < 40$. Following the classification in [14], they are found in a spherical or ellipsoidal regime (see Fig. 9 and Table III). For $E_0 = 1$, the top of the bubble becomes slightly flatter than the back and the wake appears due to stronger deformation when M decreases. This corresponds to the steady state axisymmetric shapes

TABLE III
Parameters for First Experiment

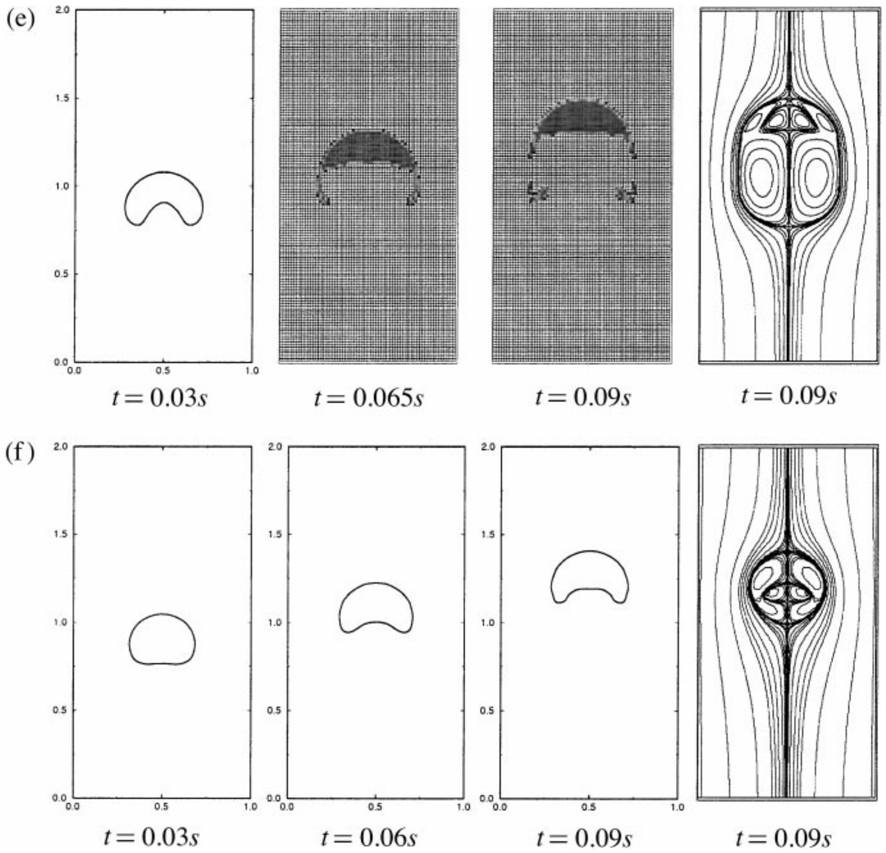
	μ_l/μ_g	E_0	M	Re_0	Re	We	$M^{\text{rel.}}$
(a)	88	1	10^{-7}	56.23	42	0.58	2.0×10^{-4}
(b)	493	1	10^{-4}	10	4.46	0.2	5.2×10^{-5}
(c)	88	10	10^{-4}	56.23	21.55	1.47	5.3×10^{-5}
(d)	493	10	10^{-1}	10	3.5	1.18	2.8×10^{-5}
(e)	85	104	10^{-1}	57.9	22.5	15.6	-1.3×10^{-5}
(f)	479	104	10^{2}	10.3	3.58	12.53	2.2×10^{-5}



$t = 0.03s$ $t = 0.045s$ $t = 0.06s$ $t = 0.06s$

$t = 0.03s$ $t = 0.06s$ $t = 0.09s$ $t = 0.09s$

FIG. 9. Rising bubbles with $E_0 = 1$ (a, b) and $E_0 = 10$ (c, d). Further details are given in Table III.



$t = 0.03s$ $t = 0.065s$ $t = 0.09s$ $t = 0.09s$

$t = 0.03s$ $t = 0.06s$ $t = 0.09s$ $t = 0.09s$

FIG. 10. Rising bubbles with $E_0 = 104$. Further details are given in Table III.

drawn by Ryskin and Leal [55, Fig. 2]. Oblate ellipsoids obtained for $E = 10$ are in close agreement with [66]; when M decreases, the separation occurs at the rim of the ellipsoid and "egg-shaped" closed wake appears behind it.

If $E_0 > 40$, the bubbles are found in spherical cap regime. When Re increases, the bubbles change likewise from oblate ellipsoidal to the oblate-ellipsoidal cap shapes, and then to the spherical-cap shapes (see photos presented by Bhaga and Weber [10, Figs. 2 and 3]. Large bubbles in high M systems at Re of order 10 to 50 can develop thin annular films of dispersed fluid, usually referred to as "skirt." Skirted bubbles are studied by Hnat and Buckmaster [30]; in their experiments, the values We/Re > 2.32 imply the appearance of a skirt for Re > 9. Although this criterion is not satisfied by the 2D bubble displayed in Fig. 10e, the skirt partially envelopes the wake similar to the idealized skirted spherical cap bubble drawn in Fig. 2 in [30]. The wake structures agree with the schematic wake diagram for skirted bubbles (see [14, Fig. 8.5]): internal circulation consists of two vortexes while toroidal vortex develops behind the bubble.

8.2.3. Second experiment. Large gas bubbles are modeled with the axisymmetric level set method in the recent work of Sussman and Smereka [63]. Their dimensionless parameters, calculated with $L = R$, coincide with the experimental parameters of *bubble A* and *bubble C* in Table I of Hnat and Buckmaster [30]: Re $= 9.8$, Fr $= 0.76$, We $= 7.6$ and Re $= 24.4$, Fr $= 0.88$ and We $= 27.2$, accordingly. In both cases, $\mu_g/\mu_l = 0.0085$ and

$\rho_g/\rho_l = 0.0011$. Both bubbles have the same Morton number ($M = 0.065$) and equal surface tension coefficient σ. Their volumes are different and correspond to $E_0 = 39.3$ and $E_0 = 123.1$, respectively.

We consider here two bubbles of the same radius $R = 1/6$ cm but with different σ. Morton and Eötvös numbers coincide with the data given above; $\Delta t = 5 \times 10^{-5}$ s. Since our experiment is two-dimensional, the steady state nose velocity U is less than the expected value $U^{3\mathrm{d}}$ in the 3D case: $U^{3\mathrm{d}} = (\mathrm{Fr} gR)^{1/2}$. Consequently, the Reynolds numbers of the current experiment differ from those in [30, 63]. The evolution of the bubbles is displayed in Figs. 11 and 12. The form of the wakes outside the bubbles agrees with the experimental data (see [30, Fig. 1]). In agreement with Bhaga and Weber [10], in the intermediate time the lower edge becomes sharper if Re increases. The bubble shapes agree quite well with the evolutions presented in Fig. 4 and Fig. 6 in [63] (here, the time values are different because of the difference in rise velocity). Similar to the results of Sussman and Smereka, *bubble C* develops the skirts and then continues to rise with nearly the constant speed at the nose as expected from the experimental measurements (see [10, 14]). Rise velocity of *bubble C* is shown in Fig. 13. The skirts break off and travel behind the bubble. This behavior is similar to the motion of the large bubble displayed in Fig. 10e which has close E_0 and M values.

Collins [16] derives the velocity of a two-dimensional spherical capped bubble rising along the axis of a channel of finite width $2b$ as

$$U^{2d} = \left[\frac{gb}{6\pi}(3 - \tanh^2\alpha)\tanh\alpha\right]^{\frac{1}{2}}, \quad \alpha = \frac{\pi\overline{c}}{2b}. \tag{30}$$

Here $2\overline{c}$ is the length of the body. Collins shows that until $\overline{c}/b$ exceeds 0.4, $\overline{c}$ is equal to the radius of curvature at the front stagnation point $\overline{a}$. In order to compare U with the predictions (30), no-slip boundary conditions are used. The radius of curvature $\overline{a}$ is estimated by fitting a circle through interface points closest to the nose. We obtain $U \approx 6.37$ cm/s, $\overline{a} = 0.298$ cm, $U^{2\mathrm{d}} \approx 6.849\,\mathrm{cm/s}$ for *bubble A* and $U \approx 6.39$ cm/s, $\overline{a} = 0.33$ cm, $U^{2\mathrm{d}} \approx 6.958\,\mathrm{cm/s}$ for *bubble C*. We believe that the discrepancy with the prediction (30) may be related to the inaccuracy in curvature estimation. Moreover, the no-slip boundary condition on the top of the box can slow the flow. Besides, the steady shapes of the obtained bubbles still resemble more the oblate–ellipsoidal cap than the spherical cap.

If the moving is done on the same advection mesh, bubble shapes obtained on the regular and adaptively refined grids practically coincide (see Figs. 11 and 12). They are in close agreement with the shapes obtained on the aligned grid in the case of *bubble A* (cf. middle and bottom rows in Fig. 11). In the case of *bubble C*, some difference appears between the shapes when the skirts develop (see middle row in Fig. 12). Indeed, we usually see that the indentation at the rear is a bit less developed on the aligned grids. This may be related to the fact that the interfacial cells are treated differently during the discretization on regular and aligned grids. Moreover, the skirts look thicker and mass is preserved better on the aligned grid since its corresponding advection mesh (underlying regular mesh) is coarser than in the case of nonaligned grids, but equal time steps are used in all computations.

8.2.4. Third experiment. Bubbles that rise in viscous liquids have been studied by Bhaga and Weber [10]. They have found that for $M > 4 \times 10^{-3}$ and $\mathrm{Re} < 110$, a closed toroidal wake develops behind the bubble. For $\mathrm{Re} > 110$ the wake appears to be open and unsteady. In their work, Bhaga and Weber show the steady shapes and the streamlines around rising

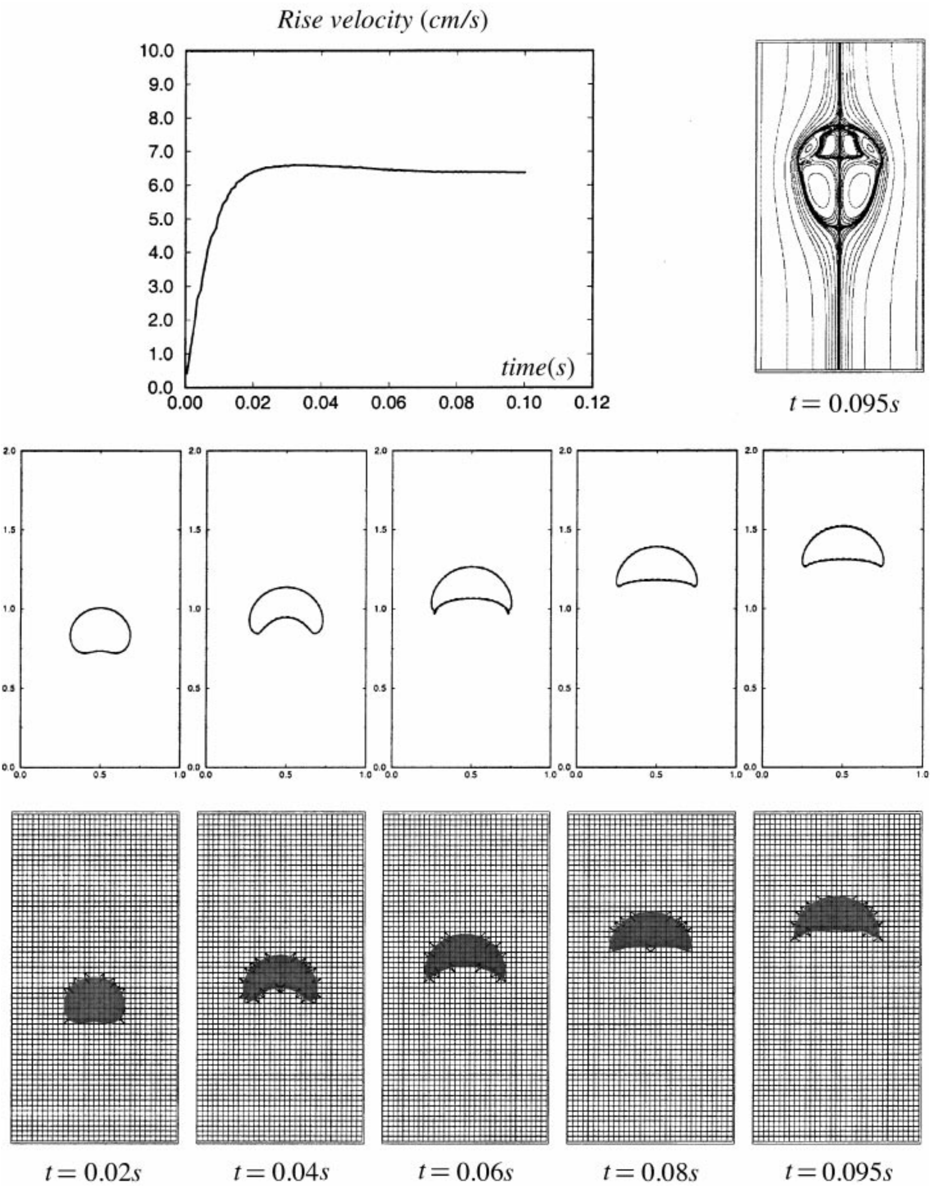


FIG. 11. Evolution of *bubble A*: $\mu_g/\mu_l = 0.0085$, $\rho_g/\rho_l = 0.0011$, $E_0 = 39.3$, $M = 0.065$, $\mathrm{Re}_0 = 31.1$, $\mathrm{Re} = 11$, $\mathrm{We} = 4.9$. Bottom row: Aligned computational grids in course of motion. Middle row: Cubic splines correspond to computations on regular 64×128 mesh (dashed line) and adaptively refined grid (solid line), which is constructed over 32×64 regular mesh. At $t = 0.06$ s $M^{\text{rel.}}$ on the aligned, adaptively refined, and uniform grids is -9.45×10^{-6}, 2.32×10^{-5}, and 2.73×10^{-5}, respectively. Top row: Rise velocity is plotted in time. Streamlines are shown at steady state.

bubbles in four situations with decreasing M (see Fig. 19 in [10]: it corresponds to cases (a), (d), (f), (g) in their Table 3). The Morton number decreases with the viscosity of the liquid, from $M = 848$ to the critical value $M_{\text{cr}} = 4.63 \times 10^{-3}$, which is relatively high since very viscous liquids are used. In this section, we take $\rho_l = 1.314\,\text{g/cm}^3$ and $\rho_l/\rho_g = 1090.5$, corresponding to an *air* bubble ($\mu_g = 1.78 \times 10^{-4}$) in liquids [10]. We show in Figs. 14a

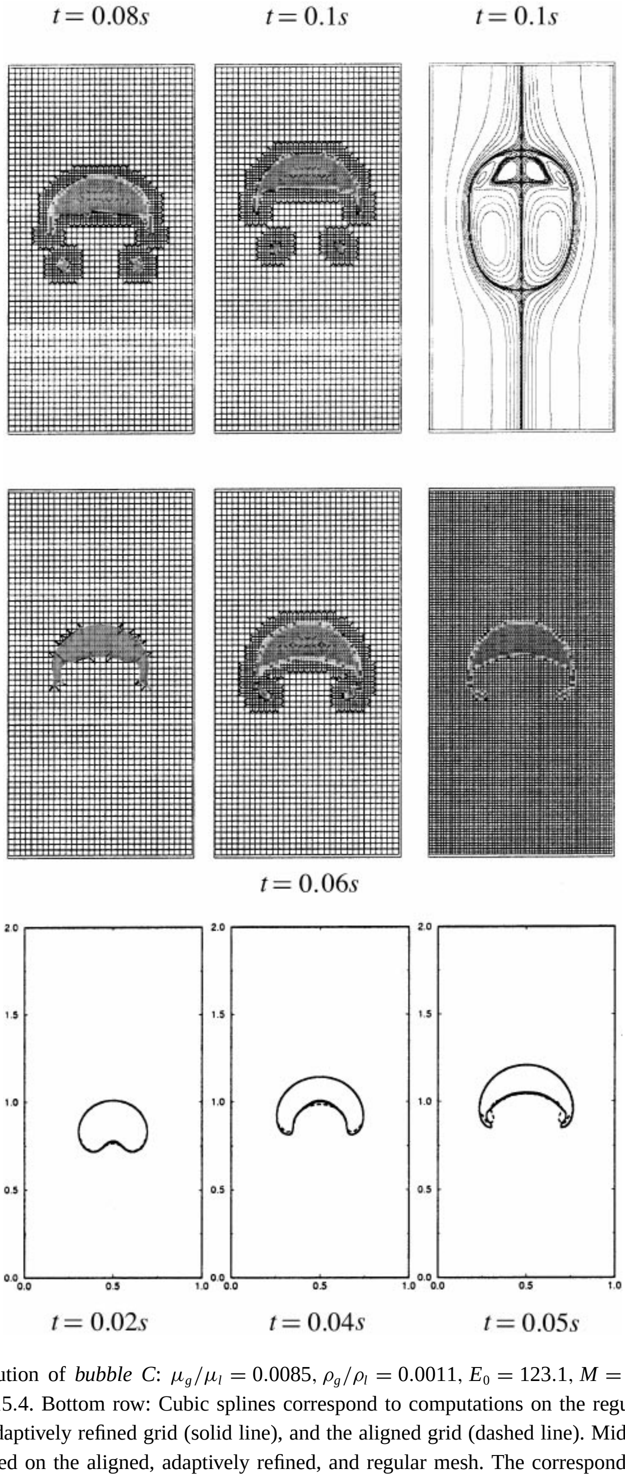


FIG. 12. Evolution of *bubble C*: $\mu_g/\mu_l = 0.0085$, $\rho_g/\rho_l = 0.0011$, $E_0 = 123.1$, $M = 0.065$, $\mathrm{Re}_0 = 73.2$, $\mathrm{Re} = 25.9$, $\mathrm{We} = 15.4$. Bottom row: Cubic splines correspond to computations on the regular 64 × 128 mesh (dotted line), the adaptively refined grid (solid line), and the aligned grid (dashed line). Middle row: Phase distribution is displayed on the aligned, adaptively refined, and regular mesh. The corresponding values M^{rel} are -1.82×10^{-5}, -6.73×10^{-5}, and -6.56×10^{-5}. Top row: Further motion of the skirted bubble is done on the adaptive grid.

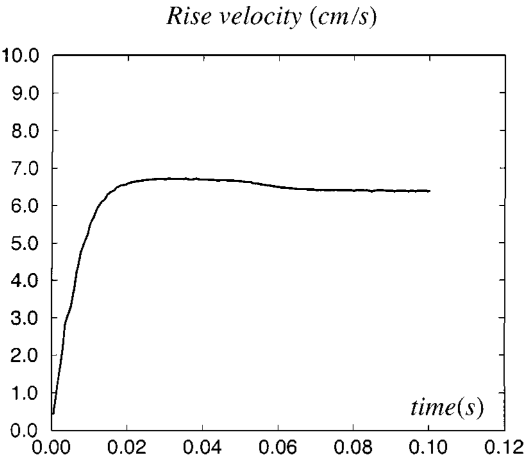


FIG. 13. Rise velocity of *bubble C*.

and 14b and Table IV the evolution of an air bubble in liquids used by Bhaga and Weber for two Morton numbers, $M = 848\,(\mu_l/\mu_g = 1.5 \times 10^5)$ and $M = 5.51\,(\mu_l/\mu_g = 4.2 \times 10^4)$. The solutions for an air bubble at $M = 0.103$ $(\mu_l/\mu_g = 1.5 \times 10^4)$ and $M = 4.63 \times 10^{-3}$ $(\mu_l/\mu_g = 7184)$ are obtained on the aligned grids up to the moment when the skirts break off. The computations on the aligned grids are performed with the $\vec{F}_s$ in the form (17). When $\vec{F}_s$ is computed in the conservative form (5), (16), the parasite currents quickly grow because of the small bubble viscosity and destroy the interface, whether the grid is aligned to the interface or not. If the bubble viscosity increases, we reach the stable solutions in these cases (see Figs. 14c, 14d and the corresponding viscosity jumps in the legend).

As the alternative approach, we discretize $\vec{F}_s$ on the cubic spline interpolants also in the case of *nonaligned grids*. For this purpose, we use the relations (17) for each pair of points defining the intersections of the cubic *spline B* with the boundary of the control volume. Then, stable solutions for air bubbles are obtained for all considered M numbers as in the case of the aligned grids. This is demonstrated in Figs. 14b and 15. Up to now, no special account of the interface position is included during the discretization of the pressure gradients on nonaligned grids. Nevertheless, computing with the spline interpolants considerably diminishes the anomalous currents on them.

The bubble shapes and the behavior of the surrounding liquid qualitatively agree with the experimental results. The bubbles take oblate ellipsoidal cap shapes. The spherical cap form is not yet reached in case (d), unlike in the experiment of Bhaga and Weber, since the attained Reynolds numbers are smaller than in real 3D experiments. A strong indentation at the rear

TABLE IV
Parameters for Third Experiment

	μ_l/μ_g	E_0	M	Re_0	Re	We	$\Delta t(s)$
(a)	1.5×10^5	116	848	6.55	1.69	7.75	1×10^{-4}
(b)	4.2×10^4	116	5.51	23.1	7.93	13.72	5×10^{-5}
(c)	250	116	0.103	62.4	22.09	14.54	5×10^{-5}
(d)	100	115	4.63×10^{-3}	134.6	47.75	14.46	2.5×10^{-5}

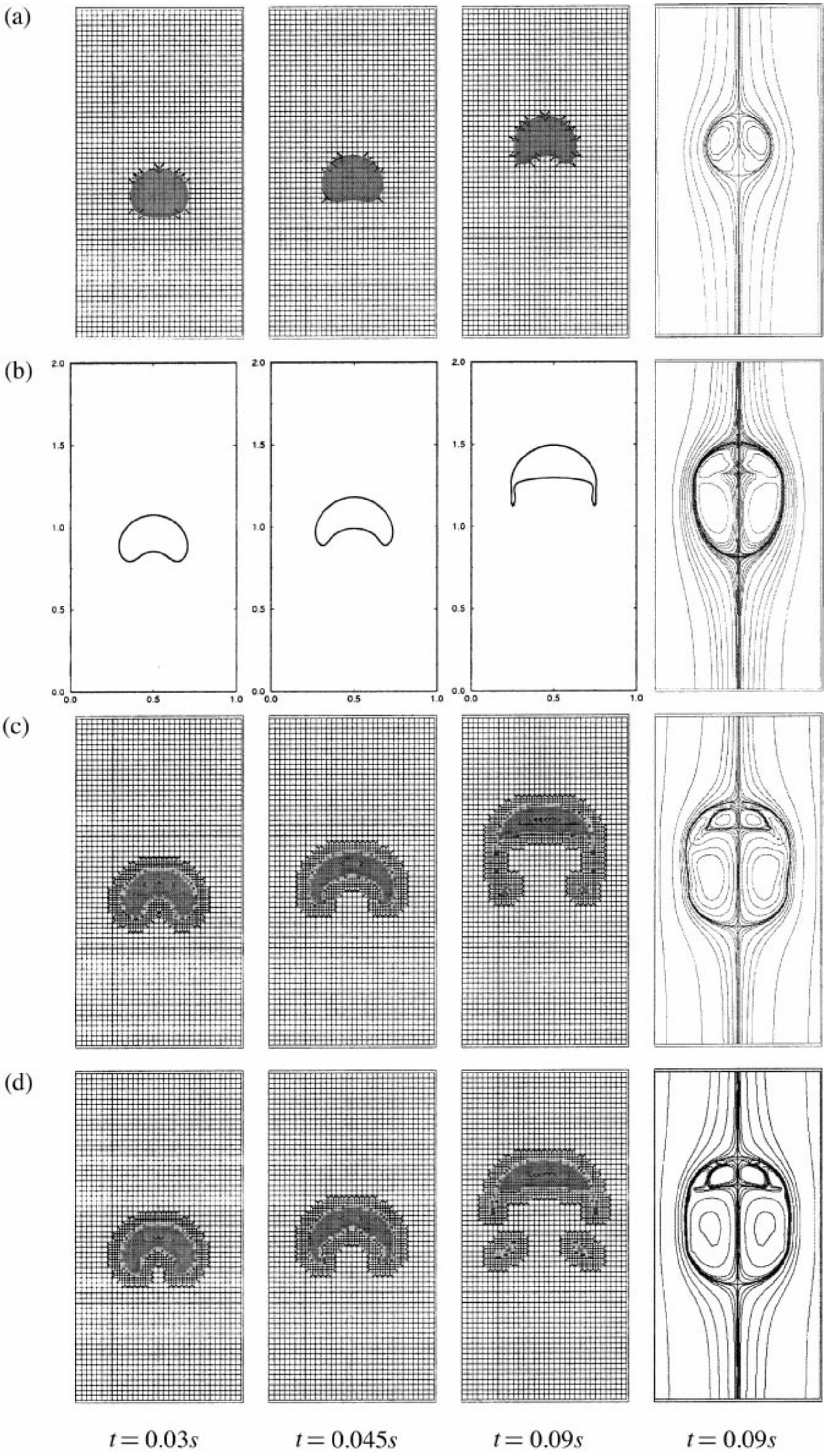


FIG. 14. Eötvös and Morton numbers correspond to the experiment of Bhaga and Weber. The evolution of the bubbles is modeled with the rotated discretization. (a) Aligned grid; (b–d) Adaptively refined grids.

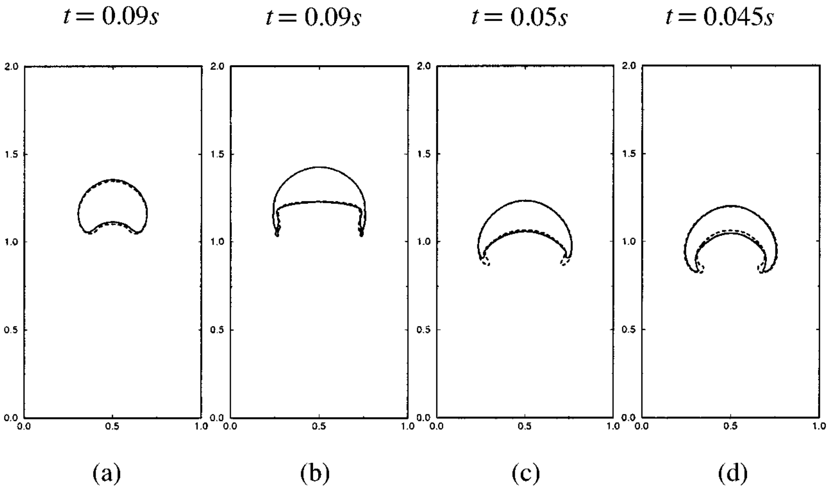


FIG. 15. Bubble shapes are obtained with the rotated (solid line) and MAC (dashed line) discretizations. (a–d): E and M correspond to Fig. 14; μ_l/μ_g corresponds to air bubble (1.5×10^5, 4.2×10^4, 1.5×10^4, and 7184, accordingly). Rotated discretization: Spline B aligned grid fitted over 32 × 64 regular mesh in case (a), adaptively refined grids otherwise. MAC discretization is done on regular grids: 32 × 64 in case (a), 64 × 128 mesh in other cases.

during the motion results in the appearance of closed toroidal wakes. Bhaga and Weber fit the frontal surface of the bubble and compare the boundary of the wake with the boundary of an ellipsoid in the case of an oblate ellipsoidal cap. We find that the wake grows continuously when going from smaller to higher Re. The wake is greater in vertical direction than the frontal ellipsoid, except for the smallest Re. In this way, the study of the wakes provides results similar to the experimental measurements. The bubbles with $M = 5.51$, $M = 0.103$, and $M = 4.63 \times 10^{-3}$ are skirted. This is not the case in the physical experiment, but it corresponds to the previous numerical computations of large bubbles (see Figs. 10e and 12) as well as to the results obtained with our MAC-type description, discussed below.

For a comparison, the MAC-type central finite difference (f.d.) approximation of Eqs. (1) and (2) on a regular grid was implemented. Explicit as well as implicit time discretizations can be used to treat both convection and diffusion terms. A linear saddle point problem is solved for all unknown pressures and velocities, similar to the current discretization. Surface tension is implemented with a f.d. approximation of (5), following [26, 34]. It should be underlined that this discretization remains stable also for a small bubble viscosity in all experiments discussed in this section. The interface shapes, plotted with the cubic spline interpolant, are compared in Fig. 15. In case (a) of the smallest Re and the greatest viscosity ratio, the small difference in bubble shapes is mainly related to the difference in the discretizations on the regular and aligned grids. The skirts agree quite well in the next three experiments where we mainly attribute the difference to the different treating of the surface tension force. This is confirmed by the computations in the cases shown in Figs. 14c and 14d, when the relations (5) are used in both discretization schemes. Finally, we do not think that the advection schemes are responsible for the skirt formation in our model since our preliminary computations with the unsplit advection scheme [54] provide very close results. Remarkably, the 2D level set method [62] and axisymmetric level set method [63] also exhibit a tendency to form skirts.

9. CONCLUSION

We have introduced a two-dimensional two-phase model based on an implicit finite volume discretization of the Navier–Stokes equations on unstructured computational grids, which either adaptively refine the interface or produce a grid that is aligned with it. In computations with the interface aligned grids, the jumps of the pressure and the continuity of the viscous stresses are kept on the front. Another important advantage that we see is the ability to accurately compute the surface tension force on cubic spline aligned grids. We show that the anomalous currents disappear when the interface is represented by a true circle. They are significantly reduced by using spline aligned grids. Staggered finite volumes and the construction of the adaptive/aligned grids can be extended to 3D and to fully unstructured mesh in a straightforward manner based on 3D elements [9] and the development [22, 41, 54] of VOF algorithms designed on the unstructured mesh, respectively. The reconstruction of the interface as spline interpolant in a 3D case can be done with parametric non-uniform rational B-splines (NURBS) surfaces [35], due to their flexibility in local choice of the rational polynomials (i.e., quotients of two polynomials). In particular, parametric surfaces can represent spheres, ellipsoids, and many other surfaces. An algorithm which computes a control point configuration closest to the given one, such that the free form solids contain volumes of the given sizes, is presented in [49]. The method [49] can use NURBS basis functions as well.

Spline interface reconstruction also enables us to reduce considerably the anomalous currents in VOF models, even if they are based on regular grids. Taking into account the interface position and the jumps on it while discretizing the pressure gradients should result in further improving the accuracy on such a grid. The combination of high-order upwinding methods with staggered finite volumes is necessary for real air/water computations with small Morton and high Reynolds numbers. We are currently working on solution techniques for saddle-point problems more tuned to interfacial problems. Our current construction of the aligned grids is done locally and the symmetry is preserved. We benefit at the advection step from the fact that the grid underlying the aligned grid or the adaptively refined grid itself is regularly refined near the interface. When an approximation of boundary conditions at the boundaries of the domain needs to use the unstructured grids already at the low levels, or when a strong distortion of the interface needs to use a nonregular adaptive strategy, grid generators can be employed in two as well as in three dimensions to construct the sequence of grids approximating the front. Also, advection schemes based on spline interpolants rather than on the PLIC method, could be developed in the future.

Although the current representation of the interface is based on the VOF method on regular grids, the model can work with other front descriptions. In particular, one can reconstruct the zero level set from the level set function provided by the level set approach. Then the model will combine such advantages of the last method as front advection via the solution of a PDE and the handling of interface merging and reconnections, with the versatility of finite volumes and finite elements on unstructured grids. Finally, our solution procedure does not use projection methods, but solves the entire linear system for the unknown pressures and velocities. In this way, a change in the time discretization scheme does not affect the solution method. Since the necessity to use the implicit schemes for the convection term in two-phase computations with strong surface tension and/or flux-based advection algorithms is not obvious, the model could benefit from the projection methodology when combined with the finite element discretizations on the interface aligned grids.

ACKNOWLEDGMENTS

We acknowledge support from the SFB 412 and the CNRS-DFG French–German research program in computational fluid mechanics. The support of the **UG** numerical group has been very helpful in the implementation. I.G. is grateful to D. Gueyffier, H. Rentz-Reichert, W. J. Rider, M. Rieber, R. Scardovelli, and S. Zaleski for useful discussions. The authors thank G. Mazurkevich, W. J. Rider, and A. Schatz for their critical readings of the manuscript.

REFERENCES

1. N. Ashgriz and J. Y. Poo, A computational method for determining curvatures, *J. Comput. Phys.* **84**, 483 (1989).
2. I. Aleinov and E. G. Puckett, Computing surface tension with high order kernels, in *Proc. 6th International Symposium on Computational Dynamics*, edited by K. Oshima, Lake Tahoe. CA, September 4–8, 1995.
3. N. Ashgriz and J. Y. Poo, FLAIR: Flux line-segment model for advection and interface reconstruction, *J. Comput. Phys.* **93**, 449 (1991).
4. R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. Van der Vorst, Templates for the solution of linear systems: Building blocks for iterative methods, *SIAM* (1994).
5. P. Bastian, K. Birken, K. Johannsen, S. Lang, N. Neuss, H. Rentz-Reichert, and C. Wieners, UG—A flexible software toolbox for solving partial differential equations, *Computing and Visualization in Science* **1**, 27 (1997).
6. J. B. Bell, P. Colella, and H. M. Glaz, A second-order projection method of the incompressible Navier–Stokes equation. *J. Comput. Phys.* **85**, 257 (1989).
7. J. H. Bramble and J. E. Pasciak, A preconditioning technique for indefinite systems resulting from mixed approximations of elliptic problems, *Math. Comput.* **50**, 1 (1988).
8. J. H. Bramble, *Multigrid Methods,* Pitman Res. Notes in Math. Sci., ISSN 0296-3674, 1993.
9. F. Brezzi and M. Fortin, Mixed and hybrid finite element methods, *Springer Series in Comp. Math.* **15** (Springer-Verlag, 1991).
10. D. Bhaga and M. E. Weber, Bubbles in viscous liquids: Shapes, wakes, and velocities, *J. Fluid Mech.* **105**, 61 (1992).
11. T. J. Barth, *Aspects of Unstructured Grids and Finite-Volume Solvers for Euler and Navier–Stokes Equations,*** VKI/NASA/AGARD Special Course on Unstructured Grid Methods for Advection Dominated Flows, AGARD Publication R-787, 1995.
12. T. J. Barth and J. A. Sethian, Numerical schemes for the Hamilton–Jacobi and level set equations on triangulate domains, *J. Comput. Phys.* **145**, 1 (1998).
13. J. U. Brackbill, D. B. Kothe, and C. Zemach, A continuum method for modeling surface tension, *J. Comput. Phys.* **100**, 335 (1992).
14. R. Clift, J. R. Grace, and M. E. Weber, Bubbles, Drops, and Particles (Academic Press, San Diego, 1978).
15. D. L. Chopp, Computing minimal surfaces via level set curvature flow, *J. Comput. Phys.* **106**, 77 (1993).
16. R. Collins, A simple model of the plane gas bubble in a finite liquid, *J. Fluid Mech.* **22**, 763 (1965).
17. A. V. Coward, Y. Y. Renardy, M. Renardy, and J. R. Richards, Temporal evolution of periodic disturbances in two-layer Couette flow, *J. Comput. Phys.* **132**, 346 (1997).
18. M. Crouzeix and P.-A. Raviart, Conforming and nonconforming finite element methods for solving the stationary stokes equations. *Revue Francaise d'Automatique, Informatique et Recherche Opérationnele* **R-3**, 33 (1973).
19. B. J. Daly, A technique for including surface tension effects in hydrodynamic calculations, *J. Comput. Phys.* **4**, 97 (1969).
20. R. DeBar, Fundamentals of the Kraken Code, Technical Report UCIR-760, LLNL, 1974.
21. D. E. Fyfe, E. S. Oran, and M. J. Fritts, Surface tension and viscosity with Lagrangian hydrodynamics on a triangular mesh. *J. Comput. Phys.* **76**, 394 (1988).

22. D. M. Gao, A three-dimensional hybrid finite element–volume tracking model for mould filing in casting processes, *Int. J. Numer. Methods Fluids* **29**, 877 (1999).

23. O. S. Galaktionov, P. D. Anderson, G. W. M. Peters, and F. N. Van de Vosse, An adaptive front tracking technique for three-dimensional transient flows, *Int. J. Numer. Methods Fluids* **32**, 201 (2000).

24. D. M. Greaves and A. G. L. Borthwick, On the use of adaptive hierarchical meshes for numerical simulation of separated flows, *Int. J. Numer. Methods Fluids* **26**, 303 (1998).

25. J. Glimm, J. Grove, B. Lindquist, O. McBryan, and G. Tryggvason, The bifurcation of tracked scalar waves, *SIAM J. Sci. Stat. Comput.* **9**, 61 (1988).

26. D. Gueyffier, J. Lie, R. Scardovelli, and S. Zaleski, Volume of fluid interface tracking with smoothed surface stress methods for three-dimensional flows, unpublished (1998).

27. W. Hackbusch, Multi-Grid Methods and Applications (Springer-Verlag, New York, 1985).

28. F. H. Harlow and J. E. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface, *Phys. Fluids* **8**, 2182 (1965).

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

30. J. G. Hnat and J. D. Buckmaster, Spherical cap bubbles and skirt formation, *Phys. Fluids* **19**, 182 (1976).

31. J. M. Hyman, Numerical methods for tracking interfaces, *Physica D* **12**, 396 (1984).

32. J. H. Jeong and D. Y. Yang, Finite element analysis of transient fluid flow with free surface using VOF (VOLUME-OF-FLUID) method and adaptive grid. *Int. J. Numer. Methods Fluids* **26**, 1127(1998).

33. D. B. Kothe, *Perspective on Eulerian finite volume methods for incompressible interfacial flows*, lecture notes presented at Free Surface Flow Workshop, International Centre for Mechanical Sciences, Udine, Italy (September 1–5, 1997). [In *Free Surface Flows*, edited by H. C. Kuhlmann and H.-J. Rath (Springer-Verlag, New York), p. 267].

34. B. Lafaurie, C. Nardone, R. Scardovelli, and S. Zaleski, Modeling merging and fragmentation in multiphase flows with SURFER, *J. Comput. Phys.* **113**, 134 (1994).

35. Les Piegl and Wayane Tiller, The NURBS Book (Springer-Verlag, New York, 1997).

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

37. J. Li, Calcul d'Interface Affine par Morceaux, *C. R. Acad. Sci. Paris* **320**, série IIb, 391 (1995).

38. N. Lock, M. Jaeger, M. Medale, and R. Occelli, Local mesh adaptation technique for front tracking problems, *Int. J. Numer. Methods Fluids* **28**, 719 (1998).

39. T. S. Ludgren and Mansour, Vortex ring bubbles, *J. Fluid Mech.* **224**, 177 (1991).

40. F. Mashayek and N. Ashgriz, A hybrid finite-element-volume-of-fluid method for simulating free surface flows and interfaces, *Int. J. Numer. Methods Fluids* **20**, 1363 (1995).

41. S. J. Mosso, B. K. Swartz, D. B. Kothe, and S. P. Clancy, Recent enhancements of volume tracking algorithm for irregular grids (Los Alamos National Laboratory, Los Alamos, NM, LA_UR_96_277, 1996), presented at the Parallel CFD Conference, Capri, Italy (March 20–23, 1996).

42. S. J. Mosso, B. K. Swartz, D. B. Kothe, and R. C. Ferrell, A Parallel, Volume-Tracking Algorithm for Unstructured Meshes (Los Alamos National Laboratory, Los Alamos, NM, LA_UR_96_2420, 1996).

43. W. F. Noh and P. R. Woodward, SLIC (simple line interface method), in *Lecture Notes in Phys.* **59**, edited by A. I. van de Vooren and P. J. Zandbergen (Springer-Verlag, Berlin/New York, 1976), p. 330.

44. S. Osher and J. A. Sethian, Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton–Jacobi formulations, *J. Comput. Phys.* **130**, 269 (1997).

45. S. Popinet and S. Zaleski, A front-tracking algorithm for the accurate representation of surface tension, submitted for publication.

46. W. H. Press, S. A. Teukolsky, W. T. Wetterling, and B. P. Flannnery, Numerical Recipes in C. (Cambridge Univ. Press, Cambridge, UK, 1992).

47. E. G. Puckett, A. S. Almgren, J. B. Bell, D. L. Marcus, and W. J. Rider, A high-order projection method for tracking fluid interfaces in variable density incompressible flows, *J. Comput. Phys.* **79**, 12 (1988).

48. R. Rannacher and S.Turek, Simple nonconforming quadrilateral stokes element, *Numer. Methods Partial Diff. Equations* **8**, 97 (1992).

49. A. Rappoport, A. Sheffer, and M. Bercovier, *Volume preserving free form solids, IEEE Trans. Visualiz. Comput. Graphics* **2**(1), 19 (1996).

50. H. Rentz-Reichert, Robuste Mehrgitterverfahren zur Lösung der inkompressiblen Navier–Stokes Gleichung: Ein Vergleich, PH.D. thesis (Stuttgart University, 1996).

51. S.-O. Kim and H. C. No, Second order model for free surface convection and interface reconstruction, *Int. J. Numer. Methods Fluids* **26**, 79 (1998).

52. J. E. Jr. Pilliod and E. G. Puckett, *Second Order Accurate Volume-of-Fluid Algorithms for Tracking Material Interfaces*, Technical Report No. LBNL-40744 (Lawrence Berkley National Laboratory).

53. G. R. Price, G. T. Reader, R. D. Rowe, and J. D. Bugg, A piecewise parabolic interface calculation for volume tracking, *Proceedings of the Sixth Annual Conference of the Computational Fluid/Dynamics Society of Canada* (University of Victoria, Victoria, British Columbia, 1998).

54. W. J. Rider and D. B. Kothe, Reconstructing volume tracking, *J. Comput. Phys.* **141**, 112 (1998).

55. G. Ryskin and L. G. Leal, Numerical solution of free-boundary problems in fluid mechanics, Part 2, Buoyancy-driven motion of a gas bubble through a quiescent liquid, *J. Fluid Mech.* **148**, 19 (1984).

56. T. Sato and S. M. Richardson, Numerical simulation method for viscoelastic flow with free surfaces-fringe element generation method. *Int. J. Numer. Methods Fluids* **26**, 555 (1994).

57. R. Scardovelli and S. Zaleski, Direct numerical simulation of free-surface and interfacial flow, *Annu. Rev. Fluid Mech* **31**, 567 (1999).

58. G. L. Sleijpen, H. A. van der Vorst, and D. R. Fokkema, BiCGstab(l) and other hybrid BI-CG methods, *Numer. Algorithms* **7**, 75 (1994).

59. G. E. Schneider and M. J. Raw, Control volume finite element method for heat transfer and fluid flow using collocated variables, *Numer. Heat Transfer* **11**, 363 (1987).

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

61. M. Rudman, Volume tracking methods for incompressible multifluid flows with large density variations, *Int. J. Numer. Methods Fluids* **28**, 357–378 (1998).

62. M. Sussmann, P. Smereka, and S. Osher, A level set approach for computing solutions to incompressible two-phase flow. *J. Comput. Phys.* **114**, 146 (1994).

63. M. Sussmann and P. Smereka, Axisymmetric free boundary problems, *J. Fluid Mech.* **341**, 269 (1997).

64. M. Sussmann, A. S. Almgren, J. B. Bell, P. Collela, L. Howell, and M. Welcome, An adaptive level set approach for incompressible two-phase flows, *J. Comput. Phys.* **148**, 81 (1994).

65. B. Swartz, The second order sharpening of blurred smooth borders, *Math. Computat.* **52**, 675 (1989).

66. S. H. Unverdi and G. Tryggvason, A front-tracking method for viscous, incompressible, multi-fluid flows, *J. Comput. Phys.* **100**, 25 (1992).

67. S. P. Vanka, Block-implicit multigrid solution of Navier–Stokes equations in primitive variables, *J. Comput. Phys.* **65**, 138 (1986).

68. P. Wesseling, *An Introduction to Multi-Grid Methods* (Wiley, 1992).

69. M. W. Williams, D. B. Kothe, and E. G. Puckett, Convergence and accuracy of kernel-based continuum surface tension models, LANL Report LA-UR-98-2268, invited paper of the Chia-Shuh Yih Memorial Symposium, in *Proceedings of the Thirteenth U.S. National Congress of Applied Mechanics, Gainesville, FL 1998.*

70. G. Wittum, Multi-grid methods for Stokes and Navier–Stokes Equations, Transforming smoothers: Algorithms and numerical results, *Numer. Math.* **54**, 543 (1989).

71. D. L. Youngs, Time-dependent multi-material flow with large fluid distortion, in *Numerical Methods for Fluid Dynamics*, edited by K. W. Morton and M. L. Norman, 1986.

72. D. L. Youngs, An Interface Tracking Method for a 3D Eulerian Hydrodynamics Code, Technical Report 44/92/35 (AWRE, 1984).