Experimental Validation of 3-D Lagrangian VOF Model: Bubble Shape and Rise Velocity

B. G. M. van Wachem and J. C. Schouten
Laboratory of Chemical Reactor Engineering, Dept. of Chemical Engineering and Chemistry,
Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands

A novel 3-D computational fluid dynamics model using an advanced Lagrangian interface tracking scheme was studied to find the time-dependent behavior of gas bubbles rising in an initially quiescent liquid. A novel least-square approach is used to determine the normal behavior at the interface for an accurate reconstruction and advection of the interface based on mollification of the color function by convolution. The incompressible Navier−Stokes equations are solved using an accurate discretization scheme to obtain the flow field of the gas and liquid phase. Detailed experiments of single rising bubbles of different sizes were performed to compare the shape, rise velocity and pressure signal of the bubble with the performed simulations. The developed Lagrangian volume-of-fluid model could accurately track the motion and shape of the gas−liquid interface embedded in a flow field with significant vorticity.

Introduction

Gas−liquid bubble columns are frequently employed in the chemical, petrochemical, and wastewater industries. Most applications involve gas−liquid mass transfer with chemical reactions. The simple construction, favorable mass- and heat-transfer properties, and low operating cost are distinct advantages of bubble columns over other gas−liquid reactors. However, despite its industrial relevance, many important hydrodynamic phenomena associated with the gas-liquid flow occurring in bubble columns, such as bubble formation, bubble coalescence, and bubble breakup are still poorly understood. These hydrodynamic phenomena strongly influence the complete behavior of the bubble column, that is, mass transfer, heat transfer, and chemical reactions depend upon them. The fundamental understanding of the behavior of individual bubbles in liquid under various conditions will strongly increase the insight in the physical phenomena mentioned.

To study the time-dependent behavior of individual gas bubbles in a fundamental framework, we have developed a three-dimensional computational fluid-dynamics (CFD) model based on the volume-of-fluid (VOF) concept. Our CFD model provides an abundance of data without the necessity for empirical constant or tweaking parameters. Hence, the model can be used, for instance, to calculate the interfacial area, the mass transfer between the two phases, the influences of the geometry, or study the mixing behavior in the flow. The VOF method is a powerful tool for modeling incompressible fluid flows that contain a free surface (e.g., Hirt and Nicholls, 1981). In the VOF method, the fluid location is recorded by employing a VOF function, or color function, which is defined as unity within the fluid regions and zero elsewhere. In practical numerical simulations employing a VOF algorithm, this function is unity in computational cells occupied completely by fluid of phase 1, zero in regions occupied completely by phase 2, and a value between these limits in cells that contain a free surface. In the VOF algorithm, the color function is semidiscontinuous, while in the closely related level-set algorithm, this function is completely continuous.

An advantage of the level-set algorithm is its simplicity in computing derivatives of the color function, required, for instance, to calculate the curvature of the surface. A disadvantage of this approach is that the numerical representation of the transport equation to determine the values of the color function is prone to numerical error, leading to a loss or gain of mass of either of the phases. Successful implementations

Correspondence concerning this article should be addressed to B. G. M. van Wachem.
Current address of B. G. M. van Wachem: Dept. of Thermo and Fluid Dynamics, Chalmers University of Technology, SE-412 96 Gothenburg, Sweden.

of the level-set methods have been demonstrated, for instance, by Koren and Venis (1999) and Sussman et al. (1994). However, simulations of gas−liquid systems using this method have not been validated experimentally yet.

In VOF methods, the color function is a semidiscontinuous function, facilitating the calculation of the properties of each of the phases and making it possible to present an accurate numerical scheme for solving the color transport equation. However, an accurate calculation of the curvature of the interface, by determining the derivative of the color function, is very difficult from a numerical point of view.

Recently, a great deal of progress has been made in the numerics and application of the VOF and level-set algorithms (for example, Kothe and Rider, 1995; Sussman and Puckett, 2000). However, there is very little experimental validation to verify VOF or level-set simulation results. Moreover, when employing the VOF algorithm, it is still difficult to obtain a good estimate of the curvature of the free surface. This article presents two robust ways to determine the curvature of the free surface, and validates the outcome of air−water VOF simulations with detailed experiments.

Multiphase Model

In this work we consider two separate, incompressible fluid phases. The two phases are separated by a piecewise-linearly reconstructed interface (that is, Gueyffier et al., 1999; Rider and Kothe, 1998). The Navier−Stokes equation for the incompressible fluid phases reads

$$\rho\frac{\partial \boldsymbol{u}}{\partial t} + \rho\nabla\cdot(\boldsymbol{uu}) = -\nabla p + \nabla\cdot\bar{\bar{\tau}} + \rho \boldsymbol{g} + \boldsymbol{S}, \qquad (1)$$

with the continuity equation for an incompressible fluid,

$$\nabla\cdot\boldsymbol{u} = 0, \qquad (2)$$

where ρ is the density of the local fluid, p is the local pressure, $\boldsymbol{S}$ is the surface tension, $\bar{\bar{\tau}}$ is the viscous stress tensor, and $\boldsymbol{u}$ is the velocity field. The velocity field holds both the liquid and the gas velocity. Viscosity and density are assumed constant in each of the phases, but may vary from phase to phase, with values ρ_i and μ_i for phase i.

The majority of researchers employing a VOF or level-set type of method solve a transport equation to determine the location of phase 1 or 2 (Hirt and Nicholls, 1981)

$$\frac{\partial\xi}{\partial t} + \boldsymbol{u}\cdot\nabla\xi = 0, \qquad (3)$$

in which ξ is represented as a color function, denoting either phase 1 or phase 2. In the level-set method, the scalar ξ is a smooth function.

The first VOF-type approach was suggested by (Hirt and Nicholls, 1981). Although this scheme is still often applied, it performs badly on the Rudman translation test (Rudman, 1997) and the Rider−Kothe reversed single-vortex test (Rider and Kothe, 1998). Also other piecewise constant schemes [for example, Lafaurie et al. (1993)] show a large amount of smearing of the interface and a violation of the conservation of each of the phases (Kothe and Rider, 1995). The application of so-called surface-sharpening models, as present in some commercial CFD codes, can somewhat prevent the smearing of the interface, but lacks any physical background.

Piecewise-linear schemes are able to track a linear surface, including its orientation. This piecewise-linear profile more closely represents the actual fluid geometry. Youngs' VOF method (Youngs, 1982) and the stream scheme (Harvie and Fletcher, 2000) are examples of these. These schemes are much more complex than piecewise-constant schemes, especially in three dimensions, but they have been shown to be significantly more accurate (Rudman, 1997; Rider and Kothe, 1998; Harvie and Fletcher, 2000).

In this work, we have applied a Lagrangian advection method of a piecewise-linear reconstruction of the interface. Lagrangian advection schemes have proven to be robust and accurate in solving the values for color functions (Gueyffier et al., 1999; Kothe et al., 1999).

Discretization and numerical treatment

The momentum equations are discretized on a staggered grid (Harlow and Amsden, 1965; Patankar, 1980), which is more natural than using an orthogonal grid, because no artificial boundary conditions are required and spurious modes are prevented. On a staggered grid, the cell faces contain the normal velocity components. The remaining variables are stored at the cell centers.

The discretization of Eq. 1 with the continuity equation leads to a differential algebraic system (Wesseling, 2000)

$$\frac{d\boldsymbol{u}}{dt} + N(\boldsymbol{u}) + Gp = \boldsymbol{f}(t) \qquad (4)$$

$$D\boldsymbol{u} = g(t), \qquad (5)$$

where N is a nonlinear algebraic operator arising from the discretization of the inertia and viscous terms, G is a linear operator representing the discretization of the pressure gradient, D is a linear operator representing the discretization of the continuity equation, and $\boldsymbol{f}$ and g are known source terms, including contributions due to boundary conditions.

For the time dependence, we have used the Crank−Nicolson scheme of Eq. 4. Hence, the time discretization of Eq. 4 leads to

$$\boldsymbol{u}^n - \boldsymbol{u}^{n-1} + \frac{1}{2}\Delta t\{N(\boldsymbol{u}^n) + N(\boldsymbol{u}^{n-1})\} + \Delta t G p^{n-1/2} = \frac{1}{2}\Delta t\{\boldsymbol{f}(t^n) + \boldsymbol{f}(t^{n-1})\}. \qquad (6)$$

Equation 6 is not solved as is, but first a prediction, $\boldsymbol{u}^*$, is made, which satisfies the momentum equation, but which does not satisfy the continuity equation. For the pressure, we use the best available guess, namely $p^{n-3/2}$. The pressure at a new time level can be obtained by applying operator D to Eq. 6 and subtracting the discretized continuity equation, Eq. 5

$$DG\delta p = \frac{1}{\Delta t}\{D\boldsymbol{u}^* - g(t^n)\}, \quad p^{n-1/2} = p^{n-3/2} + \delta p, \qquad (7)$$

where the application of the two operators DG leads to a Poisson-type of equation. After the computation of $p^{n-1/2}$,

$\boldsymbol{u}^n$ follows from Eq. 6. Although the discretized equations (Eqs. 6 and 7) are second-order accurate in Δt, to make the time-dependent solution formally second-order accurate, it is necessary to compute the initial pressure at $t = 0$

$$\frac{dg(0)}{dt} + DN[\boldsymbol{u}(0)] + DGp(0) = Df(0). \tag{8}$$

The operators D and G are discretized centrally.

Convective Flux. For the contribution of the convective flux to operator N in Eq. 5 we have employed the flux-limited total variation diminishing scheme ISNAS (Zijlema, 1996). The convective term is assumed to be written as $(\boldsymbol{u}\phi)$ for each direction. The face values of each velocity component, ϕ, are approximated by the first-order upwind scheme, corrected by adding an appropriate antidiffusive flux controlled by a limiter. Hence, $\phi_{i+1/2}$ in the direction of U is approximated by

$$\phi_{i+1/2} = \begin{cases} \phi_i + \dfrac{1}{2}\Psi\left(r^{+}_{i+1/2}\right)(\phi_i - \phi_{i-1}) & U_{i+1/2} > 0 \\ \phi_i + 1 - \dfrac{1}{2}\Psi\left(r^{-}_{i+1/2}\right)(\phi_{i+2} - \phi_{i+1}) & U_{i+1/2} < 0, \end{cases} \tag{9}$$

with the flux limiter

$$\Psi_{\kappa}(r) = \frac{(r + |r|)((1+\kappa)r + 1 - \kappa)}{(1+r)^2} \qquad 0 \leq \kappa \leq 1, \tag{10}$$

where

$$r^{+}_{i+1/2} = \frac{\phi_{i+1} - \phi_i}{\phi_i - \phi_{i-1}} \text{ and } r^{-}_{i+1/2} = \frac{\phi_{i+1} - \phi_i}{\phi_{i+2} - \phi_{i+1}}. \tag{11}$$

The value $\kappa = 1/3$ leads to a formal third-order accurate scheme (Zijlema, 1996).

Boundary conditions

At the outlet of the domain the pressure is set and the stress is set to zero,

$$p = p_{\infty} + \tau_{zz} \quad \text{and} \quad \tau_{yz} = 0, \quad \tau_{xz} = 0 \tag{12}$$

At the inlet of the domain, the z velocity is specified, and the x and y velocity components are set to zero. There is no necessity for a boundary condition for the pressure at the inlet, as this has been taken care of by applying the operator D. At the walls of the domain, all velocity components are set to zero. All of the preceding boundary conditions are taken into account in the applied operators and in the variables $\boldsymbol{f}$ and g.

Interface normal and surface tension

The interface normal, $\boldsymbol{n}$, in a continuous framework is mostly given as the gradient of the color function

$$\boldsymbol{n}(\boldsymbol{x}) = \frac{\nabla\xi(\boldsymbol{x})}{|\nabla\xi(\boldsymbol{x})|}. \tag{13}$$

To include surface tension in the framework of the VOF or level-set method, a continuum surface tension is usually applied in the form of

$$\boldsymbol{S} = \sigma\kappa(\boldsymbol{x})\boldsymbol{n}(\boldsymbol{x})\Gamma(\boldsymbol{x}), \tag{14}$$

where σ is the surface tension and is assumed constant, Γ is an interface indicator function, and κ is the curvature of the interface. In the most popular continuum surface-tension model, by Brackbill et al. (1992), the interface indicator, the curvature of the interface, and the interface normal are directly related to the color function,

$$\Gamma(\boldsymbol{x}) = |\nabla\xi(\boldsymbol{x})| \tag{15}$$

$$\kappa(\boldsymbol{x}) = -\nabla\cdot\boldsymbol{n}(\boldsymbol{x}). \tag{16}$$

As explained earlier, in the level-set method, the calculation of the preceding equations to determine the normal and curvature of the interface is fairly straightforward, as implementing a finite difference representation of the smooth color function ξ leads to no numerical problems. However, in the VOF method, the color function is discontinuous, and the calculation of the curvature and the normal is not straightforward from a numerical point of view. Still, many researchers employ a finite difference approximation of these equations, leading to large so-called parasitric currents, a nonphysical velocity field close to the interface (Lafaurie et al., 1993; Williams et al., 1998).

The effects of wall adhesion at the interface can be estimated in terms of the equilibrium contact angle between the fluid and the wall, θ_{eq}. This angle is experimentally determined under static conditions, and depends upon the smoothness and geometry of the wall. The normal to the interface at points on the wall is

$$\boldsymbol{n} = \boldsymbol{n}_{\text{wall}} \cos\theta_{eq} + \boldsymbol{n}_t \sin\theta_{eq}, \tag{17}$$

where $\boldsymbol{n}_t$ lies in the wall and is normal to the contact line between the interface and the wall, and $\boldsymbol{n}_{\text{wall}}$ is the unit wall normal directed into the wall. When the interface is in motion, the equilibrium contact angle should be replaced with a dynamic contact angle, depending on the local fluid and wall conditions.

One way to obtain a good estimate of the derivative of the color function is to mollify the color function ξ before determining its derivatives. An elegant way to mollify the color function is by convolution of the color function with a kernel function $\boldsymbol{K}$,

$$\tilde{\xi}(\boldsymbol{x}) = \boldsymbol{K} * \xi(\boldsymbol{x}) = \int_V \xi(\boldsymbol{x}')\boldsymbol{K}(\boldsymbol{x}' - \boldsymbol{x})d\boldsymbol{x}', \tag{18}$$

where V is the support of the kernel $\boldsymbol{K}$. Suitable kernels for this mollifying are, for instance, the Nordmark kernel (Nordmark, 1991) or a polynomial kernel (Williams et al., 1998), which we have applied

$$\boldsymbol{K}(r) = \begin{cases} A\left[1 - \left(\dfrac{r}{\epsilon}\right)^2\right]^4 & \text{if } r < \epsilon \\ 0 & \text{otherwise,} \end{cases} \tag{19}$$

in which ϵ is the support size of the kernel and A is a normalization constant. Due to the convolution, the derivative of the color function can easily be calculated, even if the color function is discontinuous. It was found by Williams et al. (1998) and in this study that applying such a well-defined mollifier can improve the representation of the continuum surface tension and drastically reduce the magnitude of parasitic currents.

A method we have found to be even more successful in determining the surface normal is an extension of the method suggested by Rider and Kothe (1998) in two-dimensions, which is Taylor series expansion formed around the point of interest to each neighboring point where the color function is stored,

$$\xi(\boldsymbol{x}_i+\boldsymbol{\delta})=\xi(\boldsymbol{x}_i)+\frac{\partial\xi(\boldsymbol{x}_i)}{\partial\boldsymbol{x}}\boldsymbol{\delta}+O(\boldsymbol{\delta}^2), \tag{20}$$

where $\boldsymbol{x}_i$ is the point where the derivative needs to be calculated, and $\boldsymbol{\delta}$ is the distance from that point to one of the surrounding points where the color function is stored. If the color function is not stored at point $\boldsymbol{x}_i+\boldsymbol{\delta}$, a higher-order interpolation is performed to obtain this value.

The Taylor series approximation can be written out toward all the neighboring points where the color function is stored. Combining these equations leads to a set of equations from which the derivative can be determined by minimizing the sum $(\xi_j-\xi_k)^2$ over the n neighbors in the least-square sense, where ξ_j denotes the values of the color function at point j, hence

$$A^TA\boldsymbol{q}=A^T\boldsymbol{b}, \tag{21}$$

with

$$A=\begin{pmatrix}\varphi_k(\boldsymbol{x}_k-\boldsymbol{x}_i)\\ \vdots\\ \varphi_j(\boldsymbol{x}_j-\boldsymbol{x}_i)\end{pmatrix}\text{ and }\boldsymbol{b}=\begin{pmatrix}\varphi_k(\xi_k-\xi_i)\\ \vdots\\ \varphi_j(\xi_j-\xi_i)\end{pmatrix} \tag{22}$$

where $\boldsymbol{\varphi}$ are the weights that should be carefully chosen, and $\boldsymbol{q}=(\partial\xi_i/\partial\boldsymbol{x})$. The normal can then be easily computed from $\boldsymbol{q}$,

$$\boldsymbol{n}=\left(\frac{\boldsymbol{q}}{|\boldsymbol{q}|}\right) \tag{23}$$

at any point of interest in the geometry. Hence, the normals can be calculated in the center of the computational cell, but can also be used to calculate the normal at the staggered velocity points, or over the interphase intersecting the computational cell. Combining the estimate of the color function derivative at different locations, gives an accurate prediction of $\nabla\xi$.

PLIC VOF method

The given volume fraction of one of the two fluids, together with an estimate of the normal $\boldsymbol{n}$ divides a computational cell into two parts, as shown in Figure 1. The volume of the computational cell below the interface is occupied by phase 1, and the remaining volume in the computational cell is occupied by phase 2. This interface between the two phases is propagated by the local fluid flow along the interface.

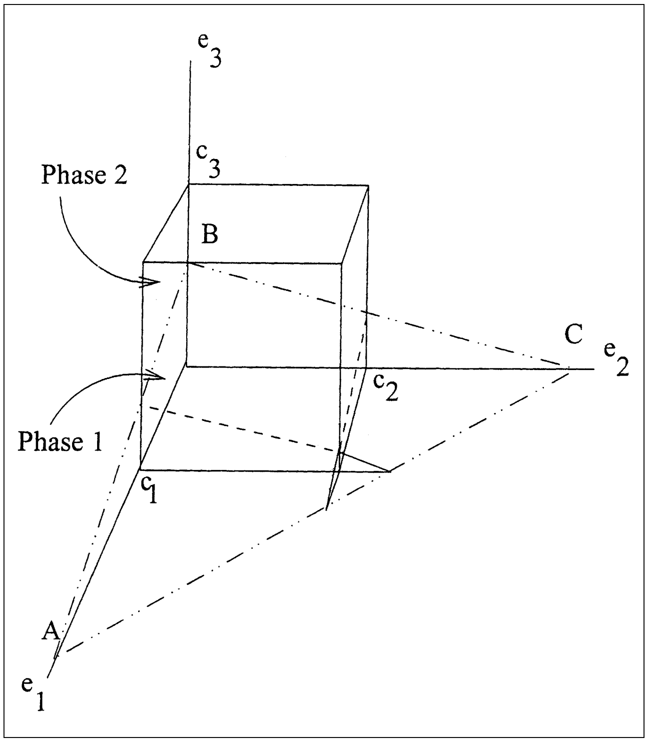

Figure 1. Volume of the computational cell cut by the interface *ABC*.

Reconstructing the Interface. In the current work, the location of the interface separating the two fluids must be found, given a volume fraction and a normal. The forward problem of finding a volume fraction, given an interface and a normal is simpler and will be derived first. Let e_1, e_2, and e_3 be the three Cartesian directions and c_1, c_2, and c_3 be the sizes of the orthogonal computational grid cell. The general equation for an interface in three dimensions is given by the equation

$$n_1x_1+n_2x_2+n_3x_3=\alpha, \tag{24}$$

where α is the shortest distance from the plane to the origin, n_i are the normals to the surface, and x_i is the coordinate in the direction of e_i. In the further derivation, all surface normals are assumed to be positive; this can always be arranged by a simple coordinate transformation within the computational cell. Then, the volume of the large tetrahedron below the interface ABC (see Figure 1) is given by

$$\frac{\alpha^3}{6n_1n_2n_3} \tag{25}$$

and the three volumes of tetrahedra that protrude outside of the original parallelepiped should be subtracted from this volume, should these volumes exist. The magnitude of each of these protruding volumes is $(1-n_ix_i/\alpha)^3$, if $n_ic_i>\alpha$. However, since the volume of the small tetrahedron in the front of Figure 1 can then be subtracted multiple times, this volume should be added again. The magnitude of this volume is $(1 - n_i c_i/\alpha - n_j c_j/\alpha)^3$, where $i \neq j$, and only if $\alpha > (n_i c_i + n_j c_j)$. Hence, the equation for the total volume under the interface, within the computational cell can be written as

$$V = \frac{\alpha^3}{6 n_1 n_2 n_3} \left\{ 1 - \sum_{i=1}^{3} H(\alpha - n_i c_i)(1 - n_i c_i/\alpha)^3 \right.$$
$$\left. + \sum_{i=1}^{3} H(\alpha - \alpha_{\max} + n_i c_i)(1 - \alpha_{\max}/\alpha + n_i c_i/\alpha) \right\}, \quad (26)$$

where H is the Heaviside step function, and

$$\alpha_{\max} = n_1 c_1 + n_2 c_2 + n_3 c_3. \quad (27)$$

Although Eq. 26 is mathematically correct for all values of n_i, solving this equation computationally can lead to large problems if one or more of the terms n_i is very close to zero, as then the large tetrahedron is very large, and in the same order of magnitude as one or more of the parallelepiped volumes that should be subtracted from this volume, leading to a very inaccurate result. In such cases, one needs to calculate the appropriate limit of Eq. 26, or apply a two-dimensional analysis of the problem and multiply the obtained area by c_i.

In this problem, we do not only need the forward Eq. 26 between the volume and the parameter α, but also the inverse, to obtain the value of α with a known volume. Since Eq. 26 is monotonic in α, solving α is straightforward, except in cases for any of the n_i being very close to zero (see earlier). To solve the inverse problem, we have applied the Newton–Raphson method, as the derivative of Eq. 26 can be easily determined.

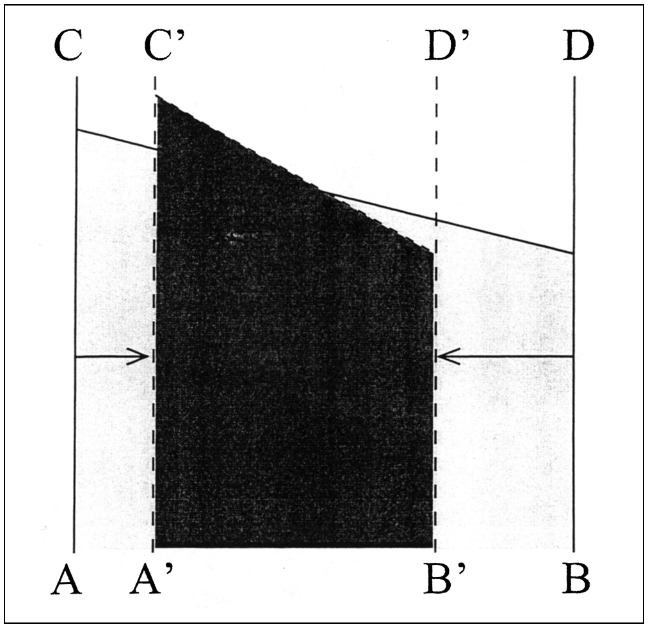

Figure 2. Calculation of volume fluxes during a time step in one direction in two dimensions.

The faces of the cell (the lines $A-B$ and $C-D$) are translated with the locally interpolated fluid velocities to the lines $A'-B'$ resp. $C'-D'$. The volumes on the right- and lefthand sides, between $A-B$ and $A'-B'$, and $C-D$ and $C'-D'$, are moved to the left resp. the right cell. Since the three-dimensional velocity field is divergence-free, the compression in this direction should be compensated by a stretching along at least one of the other two directions.

Table 1. Properties of the Gas and Liquid During Experiments and Simulations

Parameter	Description	Value
ρ_g (kg/m^3)	Gas density	1.29
ρ_l (kg/m^3)	Liquid density	998
μ_g (Pa·s)	Gas viscosity	1.71×10^{-5}
μ_l (Pa·s)	Liquid viscosity	1.00×10^{-3}
σ (kg/s^2)	Surface tension of water against air	73.0×10^{-3}

Lagrangian Propagation of the Interface. Once the location of the interface has been elucidated in each computational cell, the motion of the interface should follow the motion of the fluid accurately. The Lagrangian propagation of the interface can be best described by how the equation for describing the interface (Eq. 24) is changed due to the movement of the flow. This change has two possible contributions:

1. The values of α and n_i change due to the fluid flow and lead to a movement of the interface within the computational cell.

2. Due to the movement of the rectangular sides of the volume, the values of c_i may change and the origin to the interface may shift.

The first contribution is found by updating Eq. 24 with new values for n_i and α. These new values are found by integration of the local fluid velocity over the interface. In one direction, this local velocity can be written as

$$u_i(x) = U_0 \left(1 - \frac{x_i}{c_i} \right) + U_h \frac{x_i}{c_i}, \quad (28)$$

where U_0 is the fluid velocity, either gas or liquid, where the interface cuts the left face of the computational cell, U_h is the fluid velocity where the interface cuts the right face of the computational cell, and x_i is the coordinate into the com-

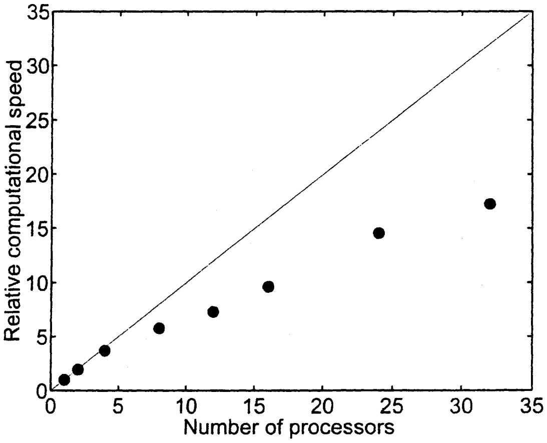

Figure 3. Parallel scale-up of the simulation program on a Silicon Graphics CRAY Origin2000 of a representative case.

putational cell, being zero at the left edge and c_i at the right. Now, the new coordinate, x_i^n, becomes

$$x_i^{(n)} = x_i + u_i(x_i)\Delta t = \left(1 + \left\{\frac{U_h - U_0}{c_i}\right\}\Delta t\right)x_i + U_o \Delta t. \quad (29)$$

After substituting the equation for x_i into Eq. 24, new values for α and n_i can be found, because the equation of the new interface is given by

$$n_i^{(n)} x_i^{(n)} + n_j x_j + n_k x_k = \alpha^{(n)}, \quad (30)$$

where $i \neq k \neq j$. Hence,

$$n_i^{(n)} = \frac{n_i}{1 + \left(\frac{U_h - U_0}{c_i}\right)\Delta t} \quad (31)$$

$$\alpha^{(n)} = \alpha + \frac{n_i U_0 \Delta t}{1 + \left(\frac{U_h - U_0}{c_i}\right)\Delta t}. \quad (32)$$

This process is done sequentially for all spatial directions.

The second contribution occurs both on the left- and right-hand sides of each direction due to protrusion of the interface into neighboring cells. At the left face of the computational cell, the interface protrudes into the lefthand side cell if $U_{\text{left}} < 0$. The volume shifted from the current cell into the lefthand cell is determined by considering the intersection of a cell with dimensions $-U_{\text{left}}\Delta t$, c_j, and c_k, where $i \neq k \neq j$, with an interface at which α is shifted by $-n_i U_{\text{left}} \Delta t$.

The interface protrudes into the righthand side cell if $U_{\text{right}} > 0$ and $\alpha / n_i > c_i$, as the interface has to intersect with the right face of the computational cell. The volume shifted from the current cell into the righthand side cell is determined by considering the intersection of a cell with dimensions $U_{\text{right}}\Delta t$, c_j and c_k, where $i \neq k \neq j$, with an interface intersecting this cell of which α is shifted by $-n_i c_i$. Again, moving these volumes is done sequentially for all spatial directions.

Both contributions to the changing of a certain fluid phase volume in a cell are depicted in Figure 2.

Simulations

Simulations of the injection of air in water were performed in a geometry of a 200-cm-high, 30-cm-wide, and 15-mm-deep column. In the simulations, the geometry of the semi twodimensional column in which experiments were carried out has been met as accurately as possible. The physical properties of the gas and liquid are shown in Table 1. The mesh employed was between 34 by 18 by 100 and 53 by 29 by 161 cells. Mesh refinement in each direction was performed to establish confidence in the numerical results. Both a serial code as well as a parallel code were developed, the latter employing the parallel numerical software library of PETSc (Balay et al., 1997, 2000). Both versions of the software codes employed a biconjugate gradient stabilized (Bi-CGstab) solver, preconditioned with an incomplete LU decomposition (ILUD) (Bruaset, 1995) to solve the sets of linear equations (Eqs. 6 and 7) at each time step. The velocity and pressure solutions were checked at each time step by making sure the error in the continuity equation is very small ($< 10^{-12}$). Simulation runs were done on an AMD Athlon 1 Ghz and on a Silicon Graphics CRAY Origin2000. A representative simulation took on the order of a few days to a week per processor on both machines. The parallel scale-up of the computer code for a representative calculation is shown in Figure 3.

Experimental Setup

Experiments of the injection of air in water were carried out to precisely study the behavior of individual bubbles. The experiments were carried out in a two-dimensional Plexiglas rectangular column, 200 cm high, 30 cm wide, and 15 mm deep. A needle was located at 49 cm above the bottom of the column, and centered in the other two directions.

The air flow through the needle was controlled by accurate variable area flow meters and pressure-reducing valves that are controlled by computer. Hence, an accurately specified amount of dry air ($\rho = 1.29$ kg·m^{-3} and $\mu = 1.71 \times 10^{-5}$ Pa·s) can be fed through the needle at a specified time for a specified duration. Bubbles with a diameter between 1.5 cm and 8 cm could be accurately injected.

Video images of the flow of the bubbles were taken with an image size of 0.15 m × 0.15 m, using a Dalsa CA-D6 high-

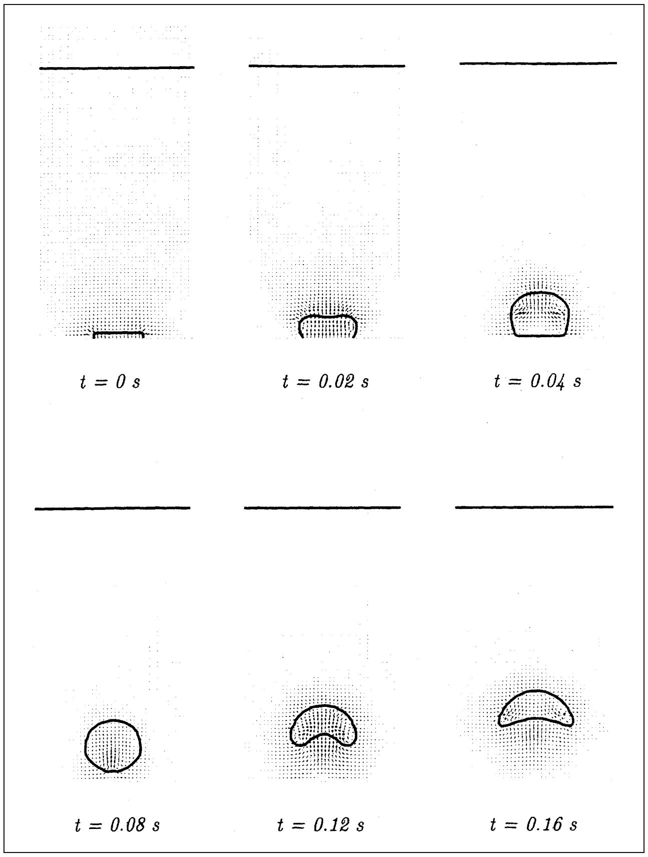

Figure 4. 2-D cross section of a 3-D simulation result of the bubble formation at an orifice.

The geometry in the figures is 20 cm wide.

speed camera, taking 955 frames per second. Due to the high shutter speed, severe illumination of the imaged surface is required.

A Drück PTX pressure sensor was connected in the center of the column, 84 cm above the bottom of the column. The pressure fluctuations were measured with a sampling frequency of 25 Hz.

Results

An important feature of the present Lagrangian volume of the fluid model is its capability of accurately resolving the shape of a gas bubble under the prevailing conditions of the physical properties of the gas and liquid. Also, the model resolves both the gas and liquid flow fields associated with the rising bubble.

Simulations of the bubble formation and rise of different bubbles sizes are compared with detailed experiments. In the simulation, a slightly different orifice size compared to the experiment was used to prevent very high Reynolds and CFL numbers near the orifice. Due to this, the duration of the injection in the simulation is almost twice as long compared to the experiment. Figure 4 shows a two-dimensional cross section of the simulated bubble formation process at three different times. After 0.05 s, the injection of air stops and the bubble detaches from the orifice. In the experiments, the injection time is smaller. Figure 5 shows a three-dimensional representation of a bubble rising in the column, and the cross section of this bubble with the velocity field around it, compared to the bubble image captured in the experiment. The diameter of the bubble in both the simulation and the experiment is around 6 cm. The shape of the simulated bubble closely matches the experimental one. In the experiment, some very small bubbles are formed during the injection period, but do not influence the big bubble.

A three-dimensional representation of a bubble with a diameter of approximately 1 cm is shown in Figure 6, compared to the bubble captured in the experiment. Because the bubble diameter is close to the column width, the image of the experiment is not very sharp. The experimentally found shape, however, is very similar to the shape predicted by the simulation.

The simulated bubble rise velocities are depicted in Figure 7 and compared with experimentally determined ones. Also, the equation for a single bubble rising in a two-dimensional infinite medium is shown,

$$V_b = \varphi\sqrt{g d_b}\,, \tag{33}$$

where Pyle and Harrison (1967) derive $\varphi = 0.54$ for a two-dimensional geometry. There is only a small difference between the rise velocity of bubbles determined in the experiment compared to the rise velocity determined from simulations. The friction of the liquid and the gas against the column wall is probably higher in the experiment than in the simulation, for instance, due to wall roughness and surface effects.

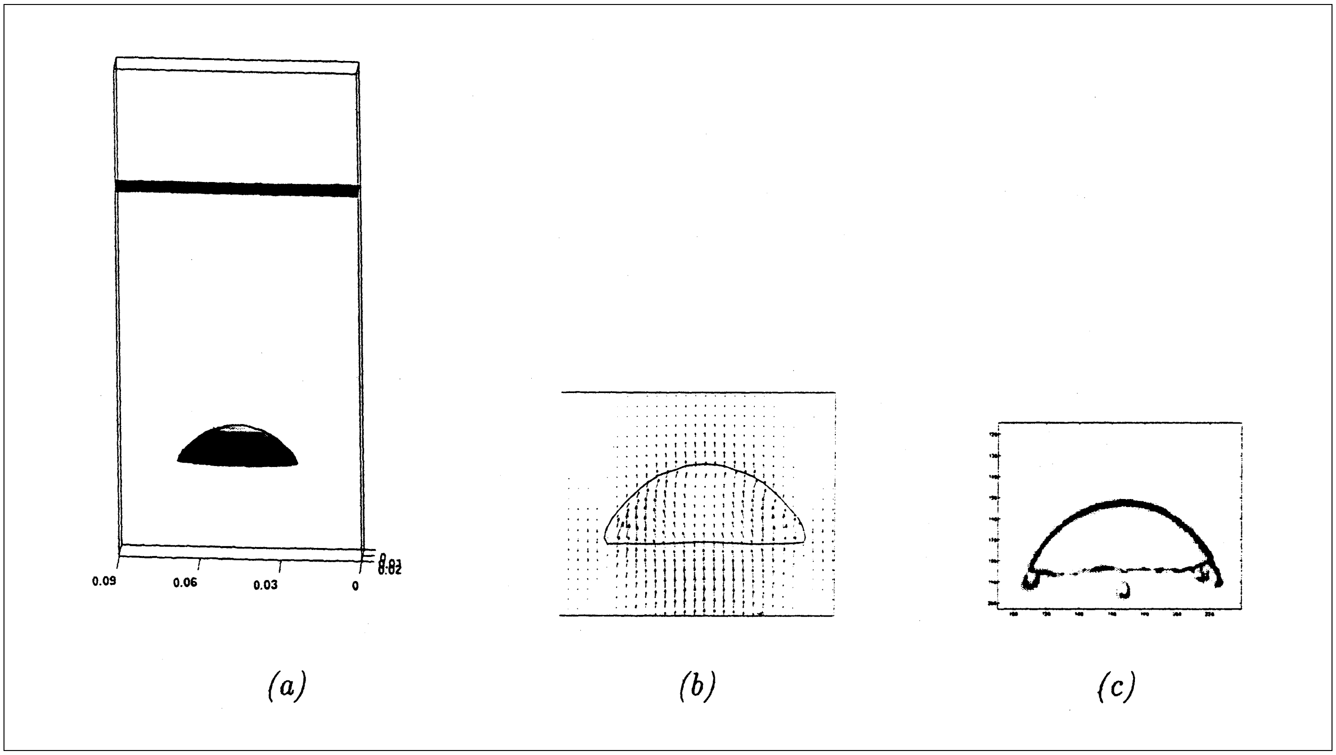

Figure 5. Visual comparison between (a) the 3-D simulation, (b) 2-D cross section, and (c) the experiment after $t = 0.24$ s after bubble injection.

The column diameter is 8 cm wide. The bubble diameter is 4 cm.

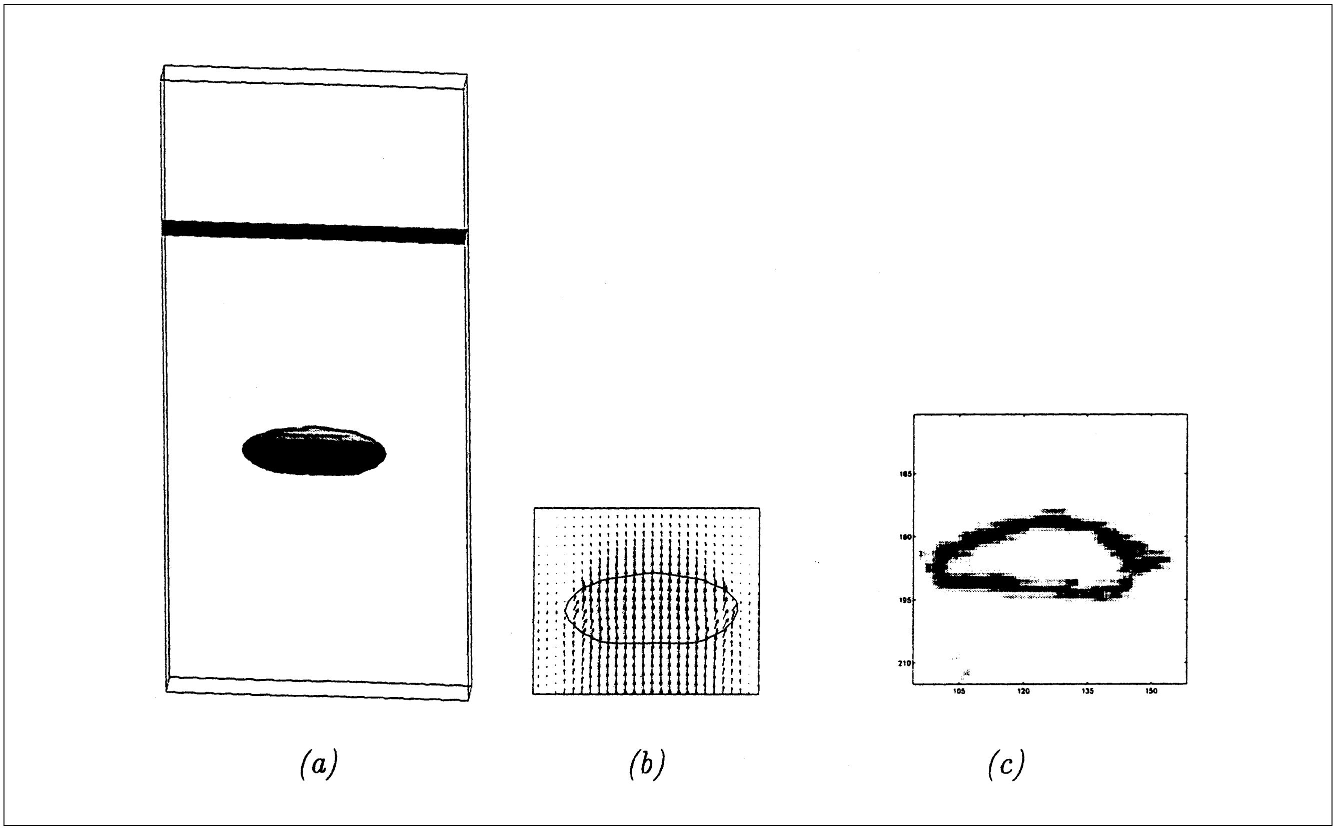

Figure 6. Visual comparison between (a) the 2-D simulation, (b) 2-D cross section, and (c) the experiment after $t = 0.4$ s after bubble injection.

The column diameter is 3.5 cm wide. The bubble diameter is 1 cm.

A property of VOF and level-set methods often encountered is the lack of mass conservation of one or two of the phases. This leads to bubbles decreasing or increasing in size during the simulation, due to smearing of the interface, unsuitable discretization schemes, or numerical diffusion. Figure 8 shows the relative mass of the gas phase of the simulated bubble during the simulation. There is essentially no loss of volume. The current piecewise-linear interface calcu-

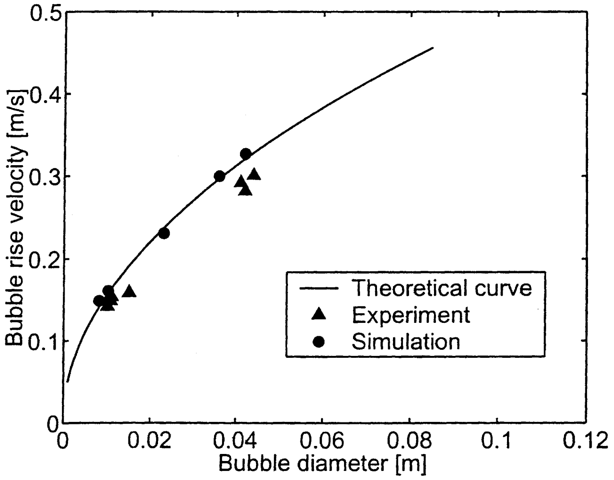

Figure 7. Experimentally determined vs. simulated bubble rise velocity.

Also shown is the rise velocity of a single bubble in an infinite, two-dimensional geometry (Pyle and Harrison, 1967).

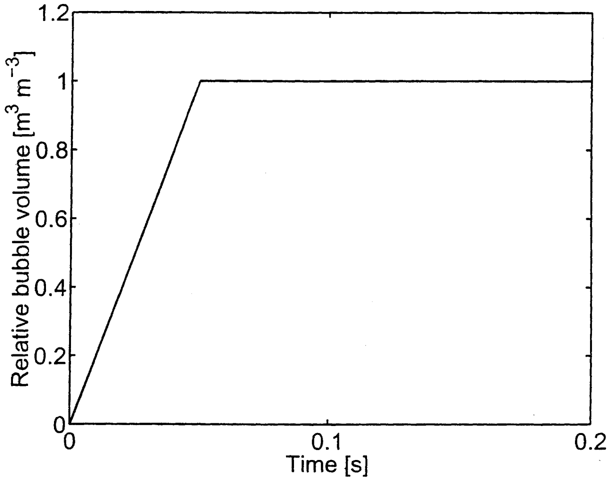

Figure 8. Dimensionless volume of bubble vs. time in the simulation.

It can be clearly seen that no gas volume in the bubble is lost.

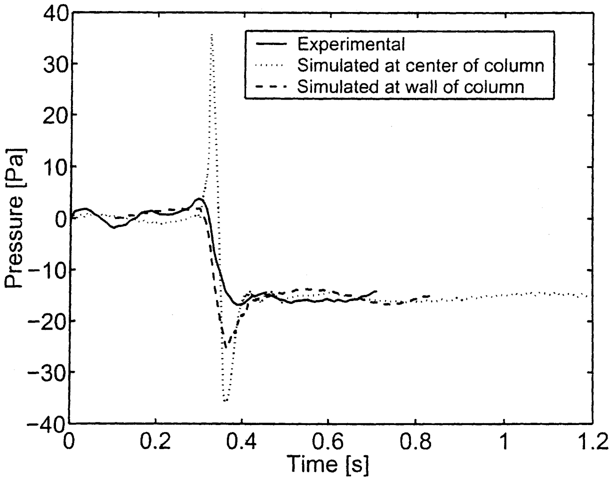

Figure 9. Experimentally determined vs. simulated pressure signal at the column wall and at the middle of the column.

lation (PLIC) VOF model presented in this article conserves volume to a high degree of accuracy, even in cases involving significant deformation of the interface.

The pressure signal of a passing bubble with a diameter of 6 cm is shown in Figure 9. The pressure sensors in the experiment are located at the back of the column and flush with the wall, so they will interfere as little as possible with the hydrodynamics in the column. The simulated pressure signals, both at the wall of the column and in the center of the column, are compared to the experimentally determined pressure signal. The simulated pressure signal in the center of the column, which causes the interface of the bubble to bulge, shows high peaks when the interface causes the point where the pressure is recorded to protrude. This is in good agreement with the pressure field determined by, for instance, Collins et al. (1978). The simulated pressure signal at the wall of the column is in good agreement with the experimentally determined one.

Conclusions

In this article, we have applied a novel numerical method for three-dimensional interface tracking using a Lagrangian volume-of-fluid approach with piecewise-linear reconstruction of the interface. We have combined this method with a novel least-square method to get an accurate estimate of the normal of the interface to achieve an accurate reconstruction at the interface. Moreover, we have applied a kernel function to determine the addition of the surface tension to minimize parasitric currents. The computational model shows efficient scale-up on the Cray Origin 2000 with the numerical library PETSc.

Results obtained with this model showed it to be capable of accurately predicting the rising of a single gas bubble in a column with initially quiescent water. The formation and the shape of the simulated bubble are in very good agreement with bubbles observed in the experiment. Moreover, the predicted rise velocity of bubbles with different sizes is in good agreement with the experimentally determined ones. Finally, the pressure fluctuation of a single rising bubble can be accurately predicted with this model. Therefore, we believe that this Lagrangian piecewise-linear interface-calculation volume-of-fluid model offers a promising tool for studying the hydrodynamic aspects of gas–liquid flows.

Notation

A = normalization constant of kernel function
c_i = size of computational grid in direction i, m
d_b = bubble diameter, m
f = source terms and boundary conditions, $\mathrm{m \cdot s^{-2}}$
g = boundary conditions influence on continuity equation, $\mathrm{m \cdot s^{-2}}$
$\boldsymbol{g}$ = gravity acceleration, $\mathrm{m \cdot s^{-2}}$
G = discretization of pressure-gradient operator, $\mathrm{m^2 \cdot kg^{-1}}$
H = Heaviside step function
$\boldsymbol{K}$ = kernel
N = discretization of inertia and viscous operator terms, $\mathrm{m \cdot s^{-2}}$
$\boldsymbol{n}$ = normal interface
p = fluid pressure, Pa
r = flux-limiter constant
r = distance to kernel center, m
$\boldsymbol{S}$ = source term, $\mathrm{kg \cdot m^{-2} \cdot s^{-2}}$
t = time, s
$\boldsymbol{u}$ = fluid velocity, $\mathrm{m \cdot s^{-1}}$
U = fluid velocity at intersect of computational cell with interface, $\mathrm{m \cdot s^{-1}}$
V = total volume under interface, $\mathrm{m^3}$
V_b = bubble rise velocity, $\mathrm{m \cdot s^{-1}}$
x, e_1 = Cartesian coordinate direction
y, e_2 = Cartesian coordinate direction
z, e_3 = Cartesian coordinate direction

Greek letters

α = shortest distance to origin, m
Γ = interface indicator, $\mathrm{m^{-1}}$
δ = short distance, m
δp = pressure correction, Pa
Δt = time step, s
ϵ = support size of kernel, m
φ = constant
ϕ = estimate of velocity component, $\mathrm{m \cdot s^{-1}}$
κ = flux-limiter constant
κ = interface curvature, $\mathrm{m^{-1}}$
σ = surface-tension constant, $\mathrm{kg \cdot s^{-2}}$
Ψ = flux-limiter function
θ = contact angle between fluid and wall
ρ = fluid density, $\mathrm{kg \cdot m^{-3}}$
$\bar{\bar{\tau}}$ = fluid viscous stress tensor, Pa
μ = fluid viscosity, Pa·s
ξ = color function

Superscripts and subscripts

1 = x component of vector
2 = y component of vector
3 = z component of vector
b = bubble
eq = equilibrium value
g = gas phase
h = right side of computational cell
i = center of computational cell
$i + 1/2$ = face of computational cell
l = liquid phase
n = current time level
$n - 1$ = previous time level
o = left side of computational cell
t = between interface and wall
wall = of the wall
zz = (3,3) component of tensor
$+$ = outward
$-$ = inward
∞ = extrapolated toward infinity

Literature Cited

Balay, S., W. D. Gropp, L. Curfman McInnes, and B. F. Smith, "Efficient Management of Parallelism in Object Oriented Numerical Software Libraries," *Modern Software Tools in Scientific Computing*, E. Arge, A. M. Bruaset, and H. P. Langtangen, eds., Birkhäuser Press, Basel, Switzerland, p. 163 (1997).

Balay, S., W. D. Gropp, L. Curfman McInnes, and B. F. Smith, PETSc home page. http://www.mcs.anl.gov/petsc (2000).

Brackbill, J. U., D. B. Kothe, and C. Zemach, "A Continuum Method for Modeling Surface Tension," *J. Comput. Phys.*, **100**, 335 (1992).

Bruaset, A. M., *A Survey of Preconditioned Iterative Methods*, Longman Group, England (1995).

Collins, R., F. F. De-Moraes, J. F. Davidson, and D. Harrison, "The Motion of a Large Gas Bubble Rising Through Liquid Flowing in a Tube," *J. Fluid Mech.*, **89**, 497 (1978).

Gueyffier, D., J. Li, R. Scardovelli, and S. Zaleski, "Volume-of-Fluid Interface Tracking with Smoothed Surface Stress Methods for Three-Dimensional Flows," *J. Comput. Phys.*, **152**, 423 (1999).

Harlow, F. H., and A. A. Amsden, "Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with a Free Surface," *Phys. Fluids*, **8**, 2182 (1965).

Harvie, D. J. E., and D. F. Fletcher, "A New Volume of Fluid Advection Algorithm: The Stream Scheme," *J. Comput. Phys.*, **162**, 1 (2000).

Hirt, C. W., and B. D. Nicholls, "Volume of Fluid (VOF) Method for the Dynamics of Free Boundaries," *J. Comp. Phys.*, **39**, 201 (1981).

Koren, B., and A. C. J. Venis, "A Fed Back Level-Set Method for Moving Material-Void Interfaces," *J. Comp. Appl. Math*, **101**, 131 (1999).

Kothe, D. B., and W. J. Rider, "Comment on Modeling Interfacial Flows with Volume-of-Fluid Methods," Tech. Rep. LA-UR-3384, Los Alamos National Laboratory, http://public.lanl.gov/mww/HomePage.html (1995).

Kothe, D. B., M. W. Williams, K. L. Lam, D. R. Korzekwa, P. K. Tubesing, and E. G. Puckett, "A Second-Order Accurate, Linearity-Preserving Volume Tracking Algorithm for Free Surface Flows on 3-D Unstructured Meshes," *Proc. ASME/JSME Joint Fluids Engineering Conf.*, San Fransisco (1999).

Lafaurie, B., C. Nardone, R. Scardovelli, S. Zaleski, and G. Zanetti, "Modelling Merging and Fragmentation in Multiphase Flows with Surfer," *J. Comput. Phys.*, **113**, 134 (1993).

Nordmark, H. O., "Rezoning for Higher Order Vortex Methods," *J. Comput. Phys.*, **97**, 366 (1991).

Patankar, S. V., *Numerical Heat Transfer and Fluid Flow*, Hemisphere, New York (1980).

Pyle, D. L., and D. Harrison, "The Rising Velocity in 2-d Fluidized Beds," *Chem. Eng. Sci.*, **22**, 531 (1967).

Rider, W. J., and D. B. Kothe, "Reconstructing Volume Tracking," *J. Comput. Phys.*, **141**, 112 (1998).

Rudman, M., "Volume-Tracking Methods for Interfacial Flow Calculations," *Int. J. Numer. Methods Fluids*, **24**, 671 (1997).

Sussman, M., and E. G. Puckett, "A Coupled Level Set and Volume-of-Fluid Method for Computing 3d and Axisymmetric Incompressible Two-Phase Flows," *J. Comput. Phys.*, **162**, 301 (2000).

Sussman, M., P. Smereka, and S. J. Osher, "A Level Set Approach for Computing Solutions to Incompressible Two-Phase Flow," *J. Comput. Phys.*, **114**, 146 (1994).

Wesseling, P., *Principles of Computational Fluid Dynamics*, Springer-Verlag, Berlin (2000).

Williams, M. W., D. B. Kothe, and E. G. Puckett, "Accuracy and Convergence of Continuum Surface Tension Models," *Tech. Rep.*, LA-UR-98-2268, Los Alamos National Laboratory, http://public.lanl.gov/mww/HomePage.html (1998).

Youngs, D. L., "Time-Dependent Multimaterial Flow with Large Fluid Distortion," *Numerical Methods for Fluid Dynamics*, K. Morton and M. Baines, eds., Academic Press, New York, p. 273 (1982).

Zijlema, M., *Computational Modeling of Turbulent Flow in General Domains*, PhD Thesis, Delft Univ. of Technology, Delft, The Netherlands (1996).

Manuscript received Dec. 27, 2001, and revision received May 20, 2002.