Numerical simulation of gas bubbles behaviour using a three-dimensional volume of fluid method

M. van Sint Annaland, N.G. Deen, J.A.M. Kuipers*

Faculty of Science and Technology, University of Twente, P.O. Box 217,7500 AE Enschede, The Netherlands

Abstract

In this paper a three-dimensional (3D) volume of fluid (VOF) method is presented featuring (i) an interface reconstruction technique based on piecewise linear interface representation, (ii) a 3D version of the CSF model of Brackbill et al. [1992, Journal of Computational Physics 100, 335]. Our model can handle a large density and viscosity ratio and a large value of the surface tension coefficient.

First the results of a number of test cases are presented to assess the correctness of the advection and interface reconstruction algorithms and the implementation of the 3D version of the CSF model. Subsequently the computed terminal Reynolds numbers and shapes of isolated gas bubbles rising in quiescent liquids are compared with data taken from the bubble diagram of Grace (1973). Finally results of two calculations are reported involving the co-axial and oblique coalescence of two gas bubbles. The computed bubble shapes compared very well with the experimentally observed bubble shapes of Brereton and Korotney [1991, In: Dynamics of Bubbles and Vortices Near a Free Surface, AMD-vol. 119. ASME, New York].

1. Introduction

Multi-fluid flows in which a sharp interface exists are frequently encountered in a variety of industrial processes. It has proven particularly difficult to accurately simulate these flows which can be attributed to (i) the fact that the interface separating the fluids needs to be tracked accurately without introducing excessive computational smearing, (ii) the necessity to account for surface tension in case of (highly) curved interfaces. In the past decade a number of techniques, each with their own particular advantages and disadvantages, have been developed to simulate complex multi-fluid flow problems. The most important techniques are summarised in Table 1 together with their main advantages and disadvantages. Subsequently these techniques will be briefly reviewed.

Front tracking methods (Unverdi and Tryggvason, 1992; Esmaeeli and Tryggvason, 1998a,b; Tryggvasson et al., 2001) make use of markers (for instance triangles), connected to a set of points, to track the interface whereas a fixed or Eulerian grid is used to solve the Navier–Stokes equations. This method is extremely accurate but also rather complex to implement due to the fact that dynamic remeshing of the Lagrangian interface mesh is required and mapping of the Lagrangian data onto the Eulerian mesh has to be carried out. Difficulties arise when multiple interfaces interact with each other as in coalescence and breakup which both require a proper sub-grid model. Contrary to most other methods, the automatic merging of interfaces does not occur in front tracking techniques due to the fact that a separate mesh is used to track the interface. This property is advantageous in case swarm effects in dispersed flows need to be studied.

Level set methods (Sussman et al., 1994, 1999; Sethian, 1996; Chang et al., 1996; Sussman and Smereka, 1997;

---

* Corresponding author. Tel.: +31 53 489 3039; fax: +31 53 489 2882.
E-mail address: J.A.M.Kuipers@ct.utwente.nl (J.A.M. Kuipers).

Table 1
Overview of techniques for multi-fluid flows with sharp interfaces

Method	Advantages	Disadvantages
Front tracking	Extremely accurate Robust Account for substantial topology changes in interface Merging and breakage of interfaces does not occur automatically	Mapping of interface mesh onto Eulerian mesh Dynamic remeshing required Merging and breakage of interfaces requires sub-grid model
Level set	Conceptually simple Easy to implement	Limited accuracy Loss of mass (volume)
Shock capturing	Straightforward implementation Abundance of advection schemes are available	Numerical diffusion Fine grids required Limited to small discontinuities
Marker particle	Extremely accurate Robust Accounts for substantial topology changes in interface	Computationally expensive Re-distribution of marker particles required
SLIC VOF	Conceptually simple Straightforward extension to three dimensions	Numerical diffusion Limited accuracy Merging and breakage of interfaces occurs automatically
PLIC VOF	Relatively simple Accurate Accounts for substantial topology changes in interface	Difficult to implement in three dimensions Merging and breakage of interfaces occurs automatically
Lattice Boltzmann	Accurate Accounts for substantial topology changes in interface	Difficult to implement Merging and breakage of interfaces occurs automatically

Sussman and Fatemi, 1999; Fedkiw and Osher, 2001) are designed to minimise the numerical diffusion hampering shock-capturing methods and typically define the interface as the zero level set of a distance function from the interface. The advection of this distance function evolves through the solution of the following equation:

$$\frac{\mathrm{D}F}{\mathrm{D}t} = \frac{\partial F}{\partial t} + (\bar{u} \cdot \nabla F) = 0, \qquad (1.1)$$

expressing that the interface property is advected with the local fluid velocity. Level set methods are conceptually simple and relatively easy to implement and yield accurate results when the interface is advected parallel to one of the co-ordinate axis. However, in flow fields with appreciable vorticity or in cases where the interface is significantly deformed, level set methods suffer from loss of mass (volume) and hence loss of accuracy.

In *shock capturing* methods (Ida, 2000) high order shock-capturing schemes are used to treat the convective terms in the governing equations. The advantage of this method is that explicit reconstruction of the interface is circumvented which offers advantages for unstructured grids. Although state-of-the-art shock-capturing methods are quite sophisticated, they work less well for the sharp discontinuities usually encountered in multi-fluid flows. Moreover, they require

relatively fine grids to obtain accurate solutions. Rider and Kothe (1995) used a high order Godunov method and conducted several numerical tests and concluded that "In all cases the use of shock-capturing methods was inadequate".

In *marker particle* methods (Welch et al., 1965; Rider and Kothe, 1995) marker particles are assigned to a particular fluid and are used to track the motion (and hence the interface) of this fluid. From the instantaneous positions of the marker particles the relevant Eulerian fluid properties, required to solve the Navier–Stokes equation, are retrieved. Marker particle methods are extremely accurate and robust and can be used successfully to predict the topology of an interface subjected to considerable shear and vorticity in the fluids sharing the interface. However, this method is computationally very expensive, especially in 3D. Moreover, difficulties arise when the interface stretches considerably which necessitates the addition of fresh marker particles during the flow simulation. Similar difficulties arise when the interface shrinks. Also merging and breakup of interfaces poses a problem; again a proper sub-grid model needs to be invoked.

Volume of fluid (VOF) methods (Hirt and Nichols, 1981; Youngs, 1982; Rudman, 1997; Rider and Kothe, 1998; Scardovelli and Zaleski, 1999; Popinet and Zaleski, 1999; Bussman et al., 1999) employ a colour function $F(x, y, z, t)$ that indicates the fractional amount of fluid present at a

certain position (x, y, z) at time t. The evolution equation for F is again Eq. (1.1) which is usually solved using special advection schemes (such as geometrical advection, a pseudo-Lagrangian technique), in order to minimise numerical diffusion. In addition to the value of the colour function the interface orientation needs to be determined, which follows from the gradient of the colour function. Roughly two important classes of VOF methods can be distinguished with respect to the representation of the interface, namely simple line interface calculation (SLIC) and piecewise linear interface calculation (PLIC). Earlier work is generally typified by the SLIC algorithm due to Noh and Woodward (1976) and the donor–acceptor algorithm published by Hirt and Nichols (1981). Modern VOF techniques include the PLIC method due to Youngs (1982). The accuracy and capabilities of the modern PLIC VOF algorithms greatly exceeds that of the older VOF algorithms such as the Hirt and Nichols VOF method (Rudman, 1997). A drawback of VOF methods is the so-called artificial (or numerical) coalescence of gas bubbles which occurs when their mutual distances is less than the size of the computational cell.

The *lattice Boltzmann method* (LBM) can be viewed as a special, particle-based discretisation method to solve the Boltzmann equation. This method is particularly attractive in case multiple moving objects (particles, bubbles or droplets) have to be treated and avoids, contrary to the classical finite difference and finite-element methods, the dynamic remeshing which becomes prohibitive for a large number of moving objects. Ladd (1994a,b) has used the LBM successfully to compute the effective gas–particle drag in particulate suspensions whereas Sundaresan and co-workers (Sankaranarayanan et al., 2002a; Sankaranarayanan and Sundaresan, 2002b) recently extended this technique to deformable interfaces and applied this technique sucessfully to study the dynamics of isolated gas bubbles rising in quiescent liquids. However, similar to VOF methods in this method problems may arise due to the artificial coalescence of the dispersed elements (gas bubbles).

In this study we have adopted the VOF method based on a piecewise linear interface representation (PLIC VOF). The VOF method, constitutes powerful and efficient direct numerical simulation (DNS) technique for complex free surface problems and was originally proposed by Hirt and Nichols (1981). Subsequently many improvements and extensions were embedded in the original VOF method; for an excellent overview the interested reader is referred to the review paper by Rider and Kothe (1998). Our model is based on Youngs' VOF method which gave the best overall performance in standard (two-dimensional (2D)) advection tests and simulations of (2D) Rayleigh–Taylor instability as reported by Rudman (1997).

Within the context of this paper we will focus on its application to dispersed two-phase flows. Similar to other DNS techniques the temporal and spatial resolution is such that all details of the flow fields are captured which enables a priori prediction of the drag, lift and added mass forces experi-

enced by the dispersed elements moving either in isolation or in (dense) swarms in a continuous phase. As such it can provide vital information on the closures for the phase interactions required for either Euler–Lagrange or Euler–Euler type of models which can in principle be applied to engineering problems. We have developed a full 3D VOF model for *two-material* flows embedding a 3D version of the CSF model of Brackbill et al. (1992) for the computation of the surface tension force.

In our model relatively high values for the density and viscosity ratio (typically one hundred) can be used without an adverse effect on the stability and the required computational effort. Traditionally systems with a high density and viscosity ration have proven difficult to simulate as reported by Scardovelli and Zaleski (1999) and Sabisch et al. (2001). The latter authors typically used a density ratio of 0.5 in their computations and reported a steep increase in the required computational time at low density ratios. However, Rudman (1998) presented a 2D VOF method for multifluid flows with large density variations by incorporating a piecewise linear interface reconstruction on a grid twice as fine as the velocity–pressure grid used to solve the Navier–Stokes equations. The extension of his dual-grid method to three dimensions seems possible but has not been reported so far.

In this paper, we apply our VOF model (i) to conduct a systematic comparison between simulation and experiment (Grace diagram) over a wide range of physical properties using a high density and viscosity ratio which are characteristic for gas–liquid systems, (ii) assess its capability to simulate co-axial and oblique coalescence of two rising gas bubbles, a process in which substantial changes in the interface topology occur.

2. Governing equations

For incompressible multi-material flows the Navier–Stokes equations can be combined into a single equation for the fluid velocity $\bar{u}$ in the entire domain of interest taking into account surface tension through a local volumetric surface tension force $\bar{F}_\sigma$ accounting for the presence of the dispersed phases. The governing conservation equations for unsteady, incompressible, Newtonian, multi-fluid flows are given by the following expressions:

$$(\nabla \cdot \bar{u}) = 0, \tag{2.1}$$

$$\frac{\partial}{\partial t}(\rho \bar{u}) + (\nabla \cdot \rho \bar{u} \otimes \bar{u})$$
$$= -\nabla p + \rho \bar{g} + \left(\nabla \cdot \mu \left[(\nabla \bar{u}) + (\nabla \bar{u})^{\mathrm{T}}\right]\right) + \bar{F}_\sigma, \tag{2.2}$$

where the local averaged density ρ and viscosity μ are evaluated from the local distribution of the phase indicator or colour function F which is governed for two-fluid flows by

$$\frac{DF}{Dt} = \frac{\partial F}{\partial t} + (\bar{u} \cdot \nabla F) = 0, \tag{2.3}$$

expressing that the interface property is advected with the local fluid velocity. For the local average density ρ linear weighing of the densities of the continuous (2) and dispersed phase (1) is used:

$$\rho = F\rho_1 + (1 - F)\rho_2. \tag{2.4}$$

Similarly, the local average dynamic viscosity can also be obtained via linear averaging of the dynamic viscosities of the continuous (2) and dispersed phase (1). As an alternative, more fundamental approach recently proposed by Prosperetti (2001), the local average viscosity can be calculated via harmonic averaging of the kinematic viscosities of the involved phases according to the following expression:

$$\frac{\rho}{\mu} = F\frac{\rho_1}{\mu_1} + (1 - F)\frac{\rho_2}{\mu_2}. \tag{2.5}$$

In all computations reported in this paper Eq. (2.5) was used to compute the local average viscosity. The volumetric surface tension force appearing in the momentum Eq. (2.2) acts only in the vicinity of the interface.

3. Numerical solution method

3.1. Computation of the flow field

The Navier–Stokes equations can be solved with a standard finite volume technique on a staggered rectangular 3D grid using a two-step projection-correction method with an implicit treatment of the pressure gradient and explicit treatment of the convection and diffusion terms. A second-order flux delimited Barton-scheme (Centrella and Wilson, 1984) is used for the discretisation of the convection terms and standard second-order central finite differences for the diffusion terms. In order to be able to simulate systems with very large density ratios, the Navier–Stokes equations are rewritten in the following non-conservative form using the continuity equation

$$\rho\left[\frac{\partial\bar{u}}{\partial t} + (\nabla \cdot \bar{u} \otimes \bar{u})\right]$$
$$= -\nabla p + \rho\bar{g} + \left(\nabla \cdot \mu\left[(\nabla\bar{u}) + (\nabla\bar{u})^{\mathrm{T}}\right]\right) + \bar{F}_\sigma, \tag{3.1}$$

where the density in the inertial terms is treated at the old time level. We use a robust and very efficient incomplete Cholesky conjugate gradient (ICCG) algorithm to solve the pressure Poisson equation (PPE).

3.2. Computation of the surface tension force

In the CSF model (Brackbill et al., 1992) the surface tension force acts via a source term $\bar{F}_\sigma$ in the momentum equation which only acts in the vicinity of the interface. The expression for $\bar{F}_\sigma$ is given by

$$\bar{F}_\sigma = 2F\sigma\kappa\bar{m}, \tag{3.2}$$
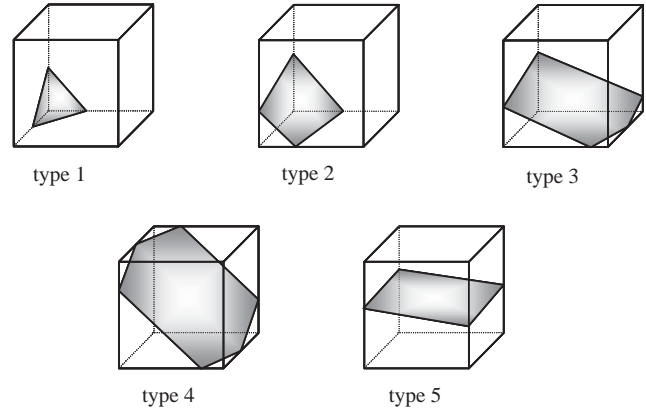

Fig. 1. Five generic types of interface configurations considered in the computation of the fluxes through the cell faces.

where the expression for the curvature is obtained from the divergence of the unit normal vector to the interface:

$$\kappa = -\left(\nabla \cdot \frac{\bar{m}}{|\bar{m}|}\right) = -(\nabla \cdot \bar{n})$$
$$= \frac{1}{|\bar{m}|}\left[\frac{\bar{m}}{|\bar{m}|} \cdot \nabla|\bar{m}| - (\nabla \cdot \bar{m})\right]. \tag{3.3}$$

The normal to the interface is computed from the gradient of the smoothed colour function. The smoothing technique used in this paper will be discussed later.

3.3. Solution of the F-advection equation

The integration of the hyperbolic F-advection equation is the most critical part of the VOF model and is based on geometrical advection which can be viewed as a pseudo-Lagrangian advection step. The advantage of the geometrical advection is given by the fact that a very sharp interface is maintained during the simulations. First for each Eulerian cell containing an interface the unit normal vector to the interface is estimated from the gradient of the colour function F:

$$\bar{n} = \frac{\nabla F}{|\nabla F|}. \tag{3.4}$$

In our model first the components of the normal vector are computed at the vertices of the Eulerian cell and then the cell-centered components are computed by averaging over the eight surrounding vertices. To minimise the number of possible interface conditions which need to be considered, first a number of transformations at the level of computational cells are carried out to achieve that the components n_1, n_2 and n_3 of the normal to the interface are always positive and satisfy the following inequality:

$$n_1 < n_2 < n_3, \tag{3.5}$$

where the subscript refers to the co-ordinate direction. These transformations involve (i) change of co-ordinate directions,

Table 2
Criteria for the determination of the type of interface cell

Type	
1	$6n_1 n_2 n_3 F < n_1^3$
2	$n_1^3 < 6n_1 n_2 n_3 F < n_2^3 - (n_2 - n_1)^3$
3	if $n_1 + n_2 > n_3$ $n_2^3 - (n_2 - n_1)^3 < 6n_1 n_2 n_3 F < n_3^3 - (n_3 - n_1)^3 - (n_3 - n_2)^3$ if $n_1 + n_2 < n_3$ $n_2^3 - (n_2 - n_1)^3 < 6n_1 n_2 n_3 F < (n_1 + n_2)^3 - n_1^3 - n_2^3$
4	$n_1 + n_2 > n_3$ and $n_3^3 - (n_3 - n_1)^3 - (n_3 - n_2)^3 < 6n_1 n_2 n_3 F < (n_1 + n_2)^3 - n_1^3 - n_2^3 - (n_1 + n_2 - n_3)^3$
5	$n_1 + n_2 < n_3$ and $6n_1 n_2 n_3 F > (n_1 + n_2)^3 - n_1^3 - n_2^3$

(ii) mutual interchange of co-ordinate directions and (iii) interchange of the dispersed and continuous phase. Through the use of these transformations it is possible to reduce the total number of possible interface configurations from 64 to five generic ones which are schematically shown in Fig. 1. From these five generic interface types the particular type prevailing in a certain Eulerian cell needs to be determined on basis of the known interface orientation (i.e. the normal vector to the interface) and the F-value of the interface cell. The criteria for determining the type of interface cell are listed in Table 2. For the computation of the fluxes through the cell faces the equation for the planar interface segment cutting through the Eulerian cell needs to be considered. This equation is given by

$$n_1 \xi_1 + n_2 \xi_2 + n_3 \xi_3 = d, \tag{3.6}$$

where ξ_i $(i = 1, \ldots, 3)$ represents the dimensionless co-ordinate in direction i given by

$$\xi_i = \frac{x_i}{\Delta x_i}, \tag{3.7}$$

where Δx_i represents the grid-spacing in co-ordinate direction x_i $(i = 1, \ldots, 3)$. The value of the plane constant d can be determined by equating the expression for the dimensionless liquid volume (volume below the planar interface segments shown in Fig. 1) to the known fractional amount of liquid or the F-value in the interface cell which leads to the equations listed in Table 3. The value of d can be obtained readily from the root of these non-linear equations using the Newton–Raphson method which needs however to be done with care in order to find the correct root of the cubic equations. As an alternative the Regula Falsi method can be used, which requires however an interval in which the root can be found. This interval can be obtained on basis of the known interface orientation (i.e. components of the

Table 3
Equations for the plane constant d

Type	
1	$d^3 = 6n_1 n_2 n_3 F$
2	$d^3 - (d - n_1)^3 = 6n_1 n_2 n_3 F$
3	$d^3 - (d - n_1)^3 - (d - n_2)^3 = 6n_1 n_2 n_3 F$
4	$d^3 - (d - n_1)^3 - (d - n_2)^3 - (d - n_3)^3 = 6n_1 n_2 n_3 F$
5	$d^3 - (d - n_1)^3 - (d - n_2)^3 + (d - n_1 - n_2)^3 = 6n_1 n_2 n_3 F$

normal to the interface) and the fractional amount of liquid in the interface cell (i.e. the F-value) using simple geometrical considerations. One should keep in mind here that the solution of the non-linear equation needs to be carried out only for the interface cells.

Once the aforementioned steps have been taken, finally the amount of liquid fluxed through each of the faces of the Eulerian cells during a time step Δt can be computed. The F-advection equation is discretised with an explicit treatment of the convections terms, where a straightforward generalisation of the 2D geometrical advection method given by Delnoij (1999) is used (also see Scardovelli and Zaleski, 1999). In our implementation of this method we have adopted the split advection scheme. Because the expressions for the fluxes through the cell faces are quite lengthy they are not given here. Finally, the computed new F-values are corrected for (small) non-zero divergence of the velocity field due to the iterative solution of the PPE.

3.4. Smoothening of the colour function F

As indicated before the interface orientation (i.e. the normal to the interface) is computed from the gradient of the colour function F according to Eq. (3.4). Basically this involves numerical differentiation of a discontinuous function

leading in practice to (small) inaccuracies. This problem can be overcome however by making use of a smoothed colour function $\tilde{F}$ for the computation of the unit normal to the interface using Eq. (3.4) with F replaced by $\tilde{F}$ obtained from:

$$\tilde{F}(x, y, z) = \sum_m D(x - x_m)D(y - y_m)$$
$$\times D(z - z_m)F(x_m, y_m, z_m), \quad (3.8)$$

where the smoothening function D is given by the function proposed by Peskin (1977):

$$D(x) = \frac{1}{2h}\left(1 + \cos\left(\pi \frac{x}{h}\right)\right) \quad (3.9)$$

or as an alternative by a suitable polynomial expression as the one proposed by Deen et al. (2004):

$$D(x) = \frac{15}{16}\frac{1}{h}\left[\left(\frac{x}{h}\right)^4 - 2\left(\frac{x}{h}\right)^2 + 1\right], \quad (3.10)$$

where h represents the width of the computational stencil used for the smoothening. The summation in Eq. (3.8) only involves the grid points with distance (in each separate coordinate direction) equal or less than the smoothening or filter width h.

We typically use $h = 2\Delta$ where Δ represents the Eulerian grid size and, unless otherwise stated. The width of the computational stencil for the smoothening should be selected carefully. When the width is too small numerical instabilities may arise, especially in case the coefficient of surface tension is high. On the other hand when the width of the computational stencil is chosen too large, excessive smoothening ("thickening" of the interface) is obtained which is undesirable. For the simulations reported in this paper we used Eq. (3.10) and additionally we used the smoothed colour function $\tilde{F}$ instead of F in Eq. (3.2). It should be stressed here that this smoothed colour function is only used in conjunction with the estimation of the unit normal to the interface and not in the computation of the material fluxes through the faces of the computational cells for which the unsmoothed colour function was used.

4. Results

As a first step our VOF model was subjected to several tests to verify the correctness of the computer implementation. Our first test is a so-called standard advection test (Rider and Kothe, 1998) in which basically the advection and reconstruction of the interface is tested in a prescribed flow field. Similar tests (in two dimensions) have been conducted by Rudman (1997) in his benchmark study of well-known volume tracking techniques. We have conducted a rather severe test using a flow field in which considerable stretching and deformation of the interface occurs. In our second test the implementation of the surface tension model was tested by comparing the computed pressure jump with the well-known Young–Laplace equation.

Table 4
Data used for the standard advection test

Computational grid	$80 \times 80 \times 80$	(-)
Grid size	0.01	m
Number of time steps	1000	(-)
Time step	0.001	s
Bubble radius	0.15	m
Initial bubble position	$(x, y, z) = (0.4, 0.55, 0.4)$	m
Liquid density	1000	kg/m^3
Liquid viscosity	0.1	kg/(m s)
Gas density	10	kg/m^3
Gas viscosity	0.001	kg/(m s)

Following these test calculations computed shapes and rise velocities of gas bubbles were compared with the corresponding data obtained from the bubble diagram published by Grace (1973). As far as the authors know this study is the first attempt to make a systematic comparison between simulation and experiment (Grace diagram) over a wide range of physical properties using at the same time a relatively high density and viscosity ratio. However, prior to these calculations the effect of the domain size and the typical number of required computational cells inside the gas bubble were determined. Finally the results of 3D simulations are presented involving the co-axial and oblique coalescence of two initially spherical gas bubbles. The results of these 3D simulations are compared with the experimental results reported by Brereton and Korotney (1991) for axial and oblique coalescence of two rising gas bubbles.

4.1. Standard advection test

Following Rider and Kothe (1998), in our first test a gas bubble was positioned in a box and subjected to a prescribed vortical flow field with corresponding stream function Ψ given by

$$\Psi = \sin^2\left(\frac{x}{a}\right)\sin^2\left(\frac{y}{b}\right), \quad (4.1)$$

where a and b represent the box size in, respectively, the x- and y-direction. Here x and y represent, respectively, the horizontal and vertical direction. The velocity components are related to the stream function Ψ by

$$u_x = -\frac{\partial \Psi}{\partial y}, \quad u_y = \frac{\partial \Psi}{\partial x}, \quad u_z = 0. \quad (4.2)$$

By comparing the initial bubble shape with its shape obtained when the flow is advanced for n steps and then reversed for the same number of time steps the accuracy of our VOF model can be assessed. In Table 4 the data used for this test are summarised.

In Fig. 2 the shape of the bubble is computed at several moments during the advection. A perfect computational technique would lead to exactly the same spatial distribution of F-values at the beginning (Fig. 2a) and the end (Fig. 2c) of the integration process. Clearly small differences can be

Fig. 2. Computed bubble shapes for the advection test: (left) initial bubble shape at $t = 0.0$ s, (centre) bubble shape at the end of the forward integration in time at $t = 1.0$ s and (right) bubble shape at the end of the backward integration in time at $t = 0.0$ s.

observed between the initial shape of the bubble and its final shape. It should be noted here that in Fig. 2 (and all subsequent figures) we show the true interface representation which is used in the computations and not the (smoothed) colour function. A quantitative measure for the associated computational error E_1 can be obtained from ($F = 0$ corresponds to the bubble phase; $F = 1$ corresponds to the liquid phase):

$$E_1 = \frac{\sum_{i,j,k} |(1 - F_{i,j,k}^{\text{end}}) - (1 - F_{i,j,k}^{\text{initial}})|}{\sum_{i,j,k} (1 - F_{i,j,k}^{\text{initial}})}, \qquad (4.3a)$$

where the summation is performed over the entire computational domain. Another error is associated with the volume conservation which can be quantified using the following expression:

$$E_2 = \frac{\sum_{i,j,k} (1 - F_{i,j,k}^{\text{end}}) - \sum_{i,j,k} (1 - F_{i,j,k}^{\text{initial}})}{\sum_{i,j,k} (1 - F_{i,j,k}^{\text{initial}})}. \qquad (4.3b)$$

For our test E_1 equals 0.037 which indicates a relatively small error (see e.g. Rider and Kothe, 1998 for the 2D case) arising from the advection and reconstruction of the interface whereas $E_2 = 2.5 \times 10^{-4}$ indicating an excellent volume conservation. From inspection of Fig. 2 and the computed E-values it can be concluded that the advection and interface reconstruction algorithm work properly.

4.2. Surface tension model

It is well known that surface tension induces an excess pressure inside a bubble which for a spherical bubble can be calculated from the Youngs–Laplace equation given by

$$\Delta p = \frac{2\sigma}{R}, \qquad (4.4)$$

where σ is the surface tension coefficient and R the bubble radius. In Fig. 3 the computed pressure distribution in a vertical plane cutting through the centre of the bubble is given and compared to the analytical solution where the pressure far from the bubble has been taken as the reference pressure. For this simulation the bubble was positioned in the centre of the simulation box in a zero gravity field. The data given in Table 5 were used for this simulation. Contrary to
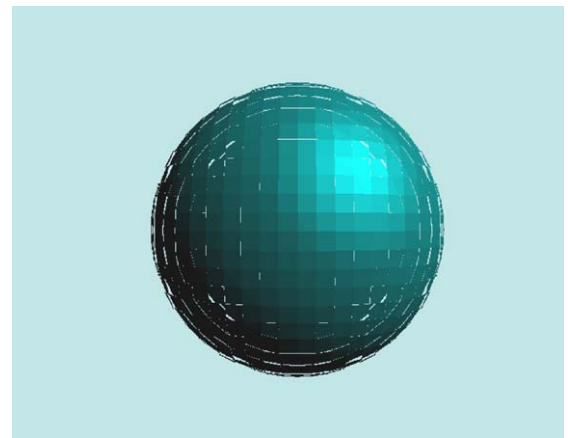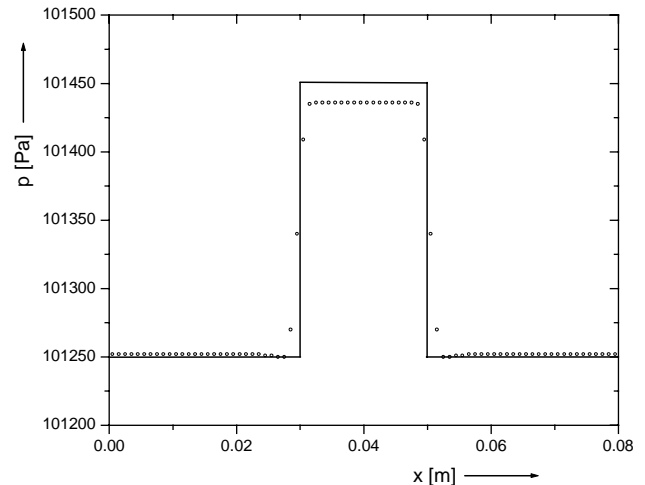

Fig. 3. Top: Computed (open circles) pressure distribution in a central plane cutting through the bubble together with the analytical solution (solid line), the Young–Laplace equation (4.4); Bottom: Computed shape of the bubble at $t = 0.05$ s.

Table 5
Data used for the test of the surface tension model

Computational grid	$80 \times 80 \times 80$	(-)
Grid size	0.001	m
Number of time steps	1000	(-)
Time step	0.0001	s
Bubble radius	0.01	m
Initial bubble position	$(x, y, z) = (0.04, 0.04, 0.04)$	m
Liquid density	1000	kg/m^3
Liquid viscosity	0.1	kg/(m s)
Gas density	10	kg/m^3
Gas viscosity	0.001	kg/(m s)
Surface tension	1.0	N/m

the previous test case here the pressure (and velocity) distribution is computed from the solution of the Navier–Stokes equations. From Fig. 3 it can be seen that the excess pressure inside the bubble is predicted with reasonable accuracy (deviation from the analytical solution is 6%) which indicates that the surface tension model has been implemented
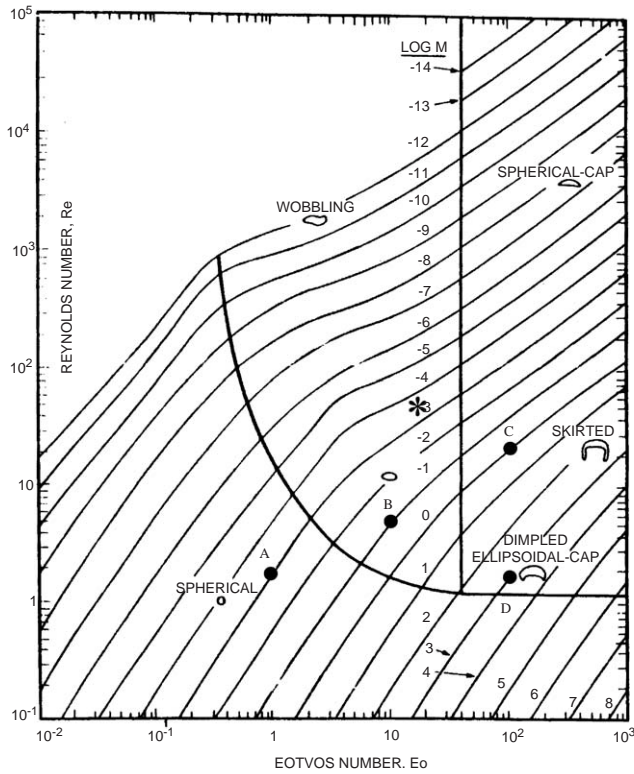
Fig. 4. Bubble diagram of Grace (1973) for the shape and terminal rise velocity of gas bubbles in quiescent viscous liquids. The simulated cases (see Table 7) are indicated with capitals.

correctly. Moreover the spherical bubble shape is nicely maintained. The small discrepancy in the calculated excess pressure can be further decreased by increasing the number of computational cells inside the bubble.

4.3. Comparison with the bubble diagram of Grace

Grace (1973) has analysed a large body of experimental data on shapes and rise velocities of bubbles in quiescent viscous liquids and has shown that this data can be condensed into one diagram, provided that an appropriate set of dimensionless numbers is used. A copy of this diagram, taken from Clift et al. (1978) is reproduced in Fig. 4 where the dimensionless Morton (M), Eotvos (Eo) and Reynolds (Re) are given by

$$M = \frac{g \mu_l^4 \Delta \rho}{\rho_l^2 \sigma^3}, \tag{4.5}$$

$$Eo = \frac{g \Delta \rho d_e^2}{\sigma}, \tag{4.6}$$

$$Re = \frac{\rho_l v_\infty d_b}{\mu_l}, \tag{4.7}$$

where the effective diameter d_e is defined as the diameter of a spherical bubble with the same volume as the bubble
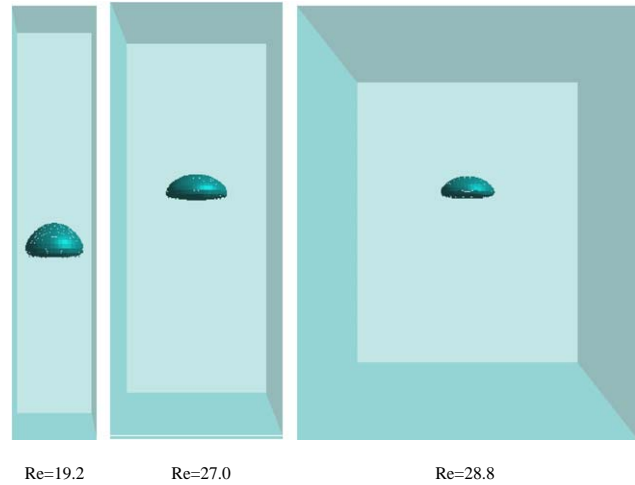

Fig. 5. Computed bubble shapes and rise velocities for computational domains with increasing lateral dimensions (see Table 6). Terminal Reynolds number (Re) for the bubble according to the Grace diagram equals 30.

Table 6
Data used for the simulations to assess the size of the computational domain in the lateral directions

Computational grids	$20 \times 20 \times 100$ (grid a)	(-)
	$40 \times 40 \times 100$ (grid b)	(-)
	$80 \times 80 \times 100$ (grid c)	(-)
Grid size	0.001	m
Number of time steps	5000	(-)
Time step	0.0001	s
Bubble radius	0.006	m
Initial bubble position	$(x_0, y_0, z_0) = (0.01, 0.01, 0.01)$ (grid a)	m
	$(x_0, y_0, z_0) = (0.02, 0.02, 0.01)$ (grid b)	m
	$(x_0, y_0, z_0) = (0.04, 0.04, 0.01)$ (grid c)	m
Liquid density	1000	kg/m³
Liquid viscosity	0.1	kg/(m s)
Gas density	10	kg/m³
Gas viscosity	0.001	kg/(m s)
Surfacetension	0.1	N/m

Table 7
Morton (M) and Eotvos (Eo) numbers for simulations of bubbles in different regimes according to the bubble diagram of Grace

Bubble regime	M	Eo	Re_G	Re_C	Case in Fig. 4
Spherical	1.26×10^{-3}	0.971	1.7	1.6	A
Ellipsoidal	0.10	9.71	4.6	4.3	B
Skirted	0.971	97.1	20	18	C
Dimpled/l ellipsoidal	10^3	97.1	1.5	1.7	D

Re_G and Re_C represent, respectively, the bubble Reynolds number obtained from the Grace diagram and the computed bubble Reynolds number.

under consideration. In the Reynolds number v_∞ appears which represents the terminal rise velocity of the bubble.

For the simulations presented in this paper a fixed density and viscosity ratio of one hundred was used (viscosity and

(A) Spherical: Eo=0.971, M=1.26.10^{-3}

(B) Ellipsoidal: Eo=9.71, M=0.1

(C) Skirted: Eo=97.1, M=0.971

(D) Dimpled/Ellipsoidal: Eo=97.1, M=10^3

(C) Skirted: Eo=97.1, M=0.971
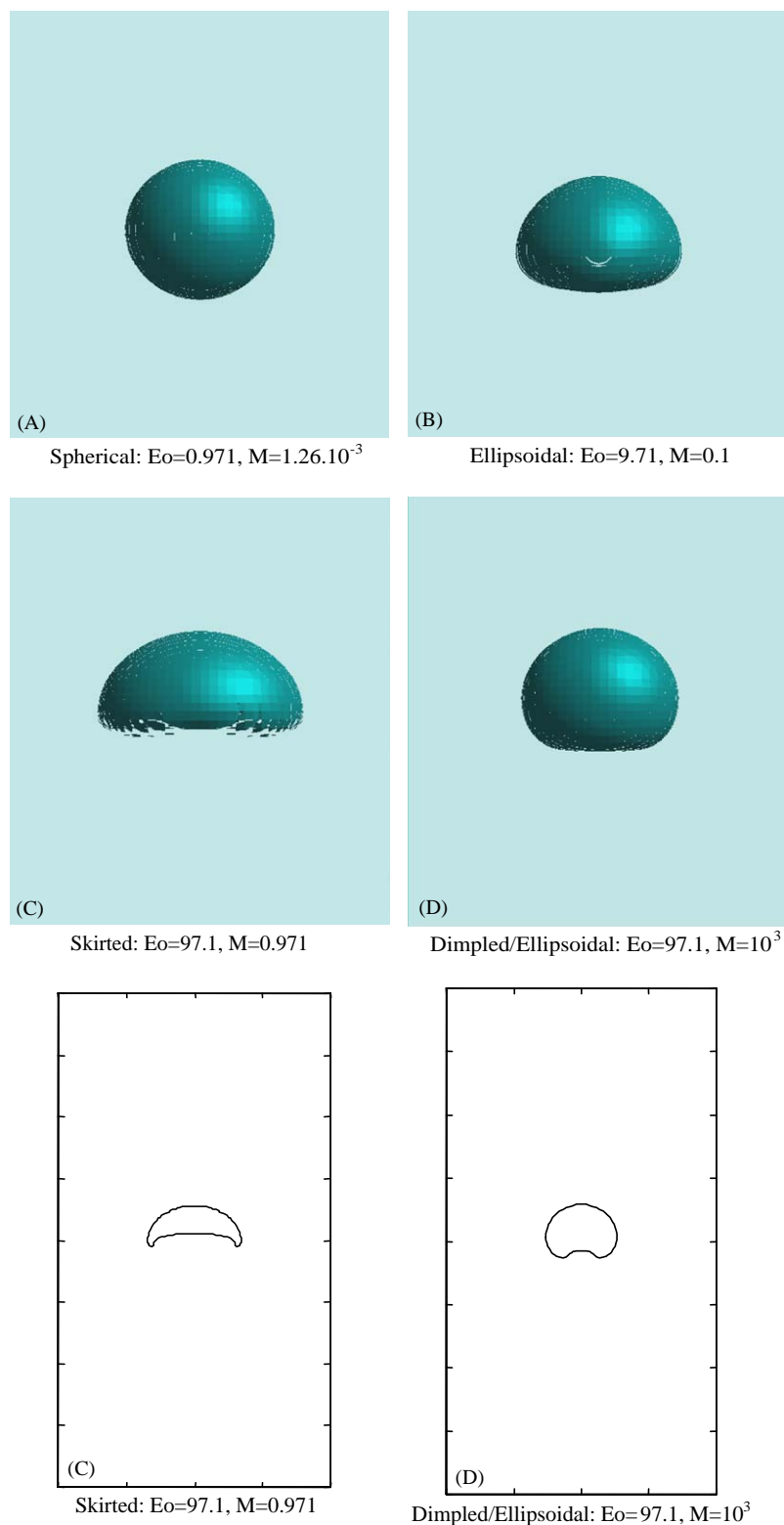
(D) Dimpled/Ellipsoidal: Eo=97.1, M=10^3

Fig. 6. Computed bubble shapes and rise velocities for the bubble regimes indicated in Table 7 and Fig. 4.

density of the continuous liquid phase equal one hundred times the viscosity and density of the dispersed gas phase). This density and viscosity ratio is believed to be sufficiently high to mimic gas–liquid systems with sufficient accuracy and much higher than the ratio used by Sabisch et al. (2001).
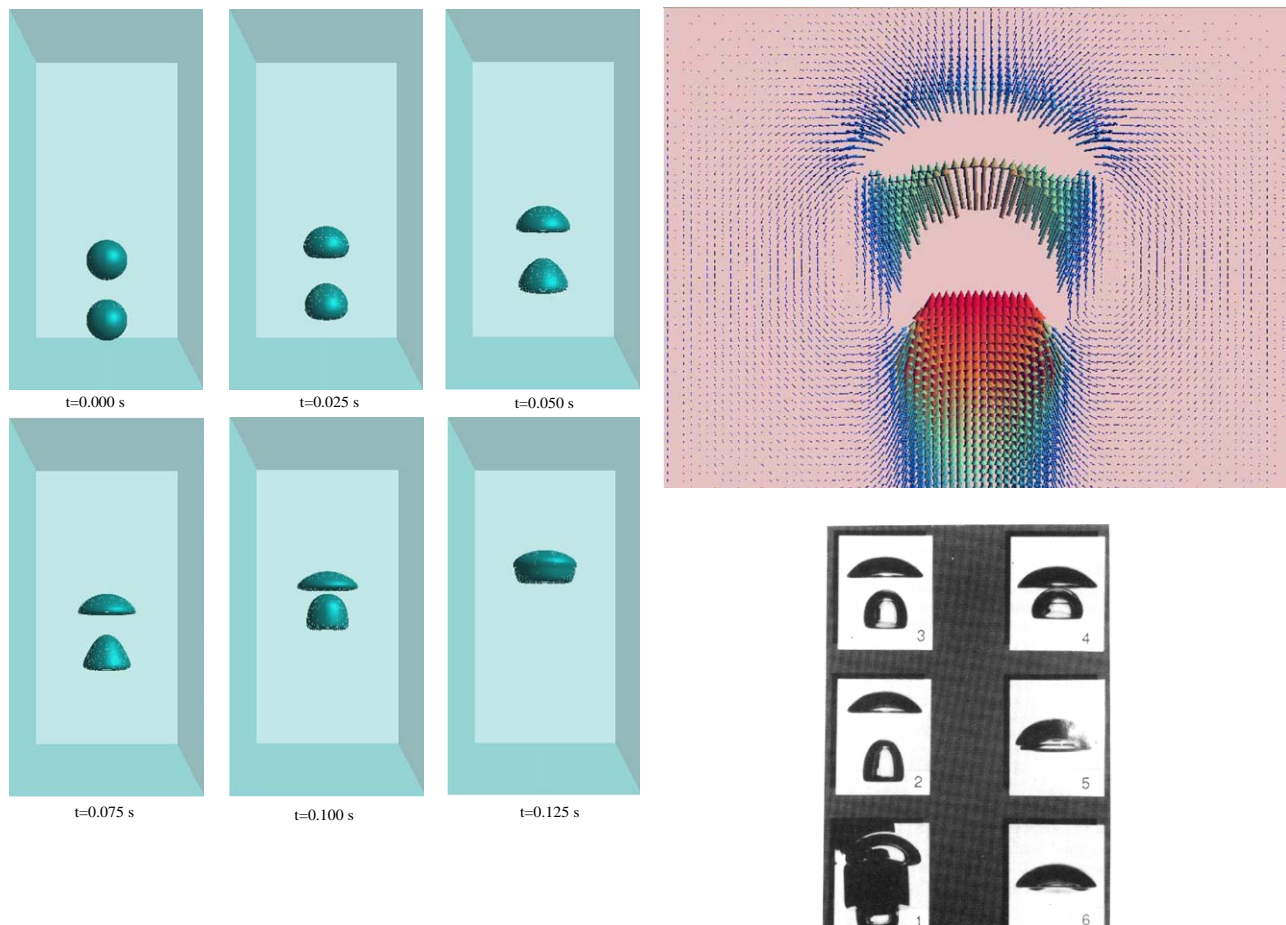
Fig. 7. Snapshots at different times of the co-axial coalescence of two initially spherical bubbles of 0.010 m diameter released from positions (0.020, 0.020, 0.010 m) and (0.020, 0.020, 0.025 m) in an initially quiescent liquid in a square column of 0.04 m × 0.04 m × 0.08 m, using a 80 × 80 × 160 grid and a time step of 5×10^{-5} s. $Eo = 16$ and $M = 2 \times 10^{-4}$. Included are a liquid phase velocity map in the central vertical plane at $t = 0.100$ s and the photographs (time difference between subsequent photographs is 0.03 s) taken from Brereton and Korotney (1991).

First a couple of simulations were carried out to warrant that the size of the computational domain was sufficiently large to mimic the condition of an infinite quiescent liquid. In Fig. 5 the computed shapes of the bubbles are shown together with the computed terminal Reynolds numbers for a number of sizes of the computational domain. Free-slip boundary conditions were applied at all confining walls. The data used for these simulations are given in Table 6. It can be seen that the terminal rise velocity is not affected anymore in case the lateral dimensions of the box in the horizontal directions (i.e. the *x*- and *y*-direction) exceed 3–4 times the (initial) bubble diameter. Another issue is concerned with the required number of computational cells initially present inside the gas bubble. For the simulations shown in Fig. 5 the gas bubble initially contained 12 computational cells in each direction. It was verified that the computed rise velocities were not affected by this relatively low number of computational cells.

Subsequently the simulation results for a number of important regimes given in the bubble diagram of Grace will

be presented. In Table 7 the values of the selected Morton and Eotvos numbers are given for simulations of bubbles in different regimes according to this diagram (also indicated in Fig. 4). In this table Re_G and Re_C represent, respectively, the bubble Reynolds number obtained from the diagram and the computed bubble Reynolds number. In Fig. 6 snapshots are given of the computed shapes of the bubbles. It can be seen that the computed Reynolds numbers and bubble shapes compare very well with the data obtained from the bubble diagram.

4.4. Multiple bubbles

The first example involves the co-axial coalescence of two gas bubbles in an initially quiescent liquid where the bubbles are initially spherical with their centres separated by three-bubble radii. In Fig. 7 the computed evolution of the bubble shapes is shown together with photographs of the experimentally observed bubble shapes just before and
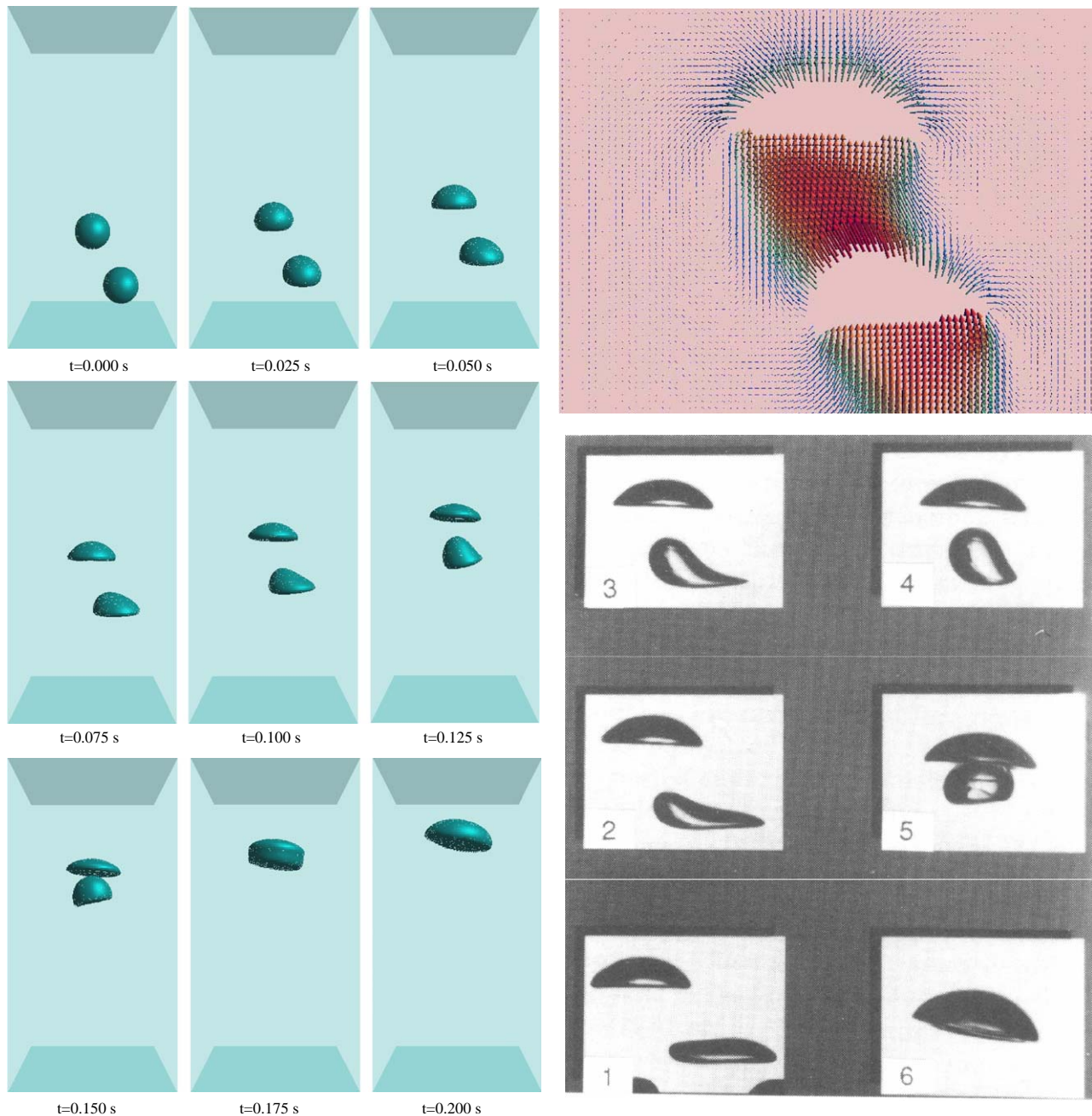
Fig. 8. Snapshots at different times of the oblique coalescence of two initially spherical bubbles of 0.010 m diameter released from positions (0.020, 0.020, 0.025 m) and (0.028, 0.020, 0.010 m) in an initially quiescent liquid in a square column of 0.04 m × 0.04 m × 0.08 m, using a 80 × 80 × 160 grid and a time step of 5×10^{-5} s. $Eo = 16$ and $M = 2 \times 10^{-4}$. Included are a liquid phase velocity map in the central vertical plane at $t = 0.100$ s and the photographs (time difference between subsequent photographs is 0.03 s) taken from Brereton and Korotney (1991).

after the coalescence process as reported by Brereton and Korotney (1991). The Morton and Eotvos number for this case are, respectively, equal to 2×10^{-4} and 16 (based on the individual bubbles) which, according to the Grace diagram, would correspond with a terminal Reynolds number of 50 (case indicated with an asterisk in Fig. 4) which agrees reasonably well with the experimental value of 43 and the computed value of 40. Apparently the leading bubble expe-

riences very little effect of the trailing bubble. However, the behaviour of the trailing bubble is completely different as can be seen clearly from Fig. 7. The trailing bubble catches up with the leading bubble and attains a distinctly different shape than the leading bubble when it enters the wake region of the leading bubble which is in excellent agreement with the experimental observations. In Fig. 7 also a velocity map is included just before the coalescence of the bubbles

($t = 0.10$ s) for the central (vertical) plane cutting through the column.

The second example involves the oblique coalescence of two gas bubbles in an initially quiescent liquid where the bubbles are initially spherical with their centres separated again by three-bubble radii in the vertical direction. Compared to the first case the position of the lower bubble was shifted to the right (*x*-direction) over 1.6 bubble radii. In Fig. 8 the computed evolution of the bubble shapes is shown together with photographs of the experimentally observed bubbles shapes just before and after the coalescence process. The Morton and Eotvos number are the same as for the first case. Similar to the previous case the leading bubble behaves as an isolated bubble although its major (vertical) axis attains a small angle with the (vertical) *z*-axis when the trailing bubble enters its wake region. Again the trailing bubble catches up with the leading bubble and experiences considerable shape deformation when it enters the wake region of the leading bubble which is in excellent agreement with the experimental observations. The liquid phase velocity map is shown for the central (vertical) plane cutting through the column just before coalescence of the bubbles commences. In this case the velocity field is clearly asymmetrical.

5. Conclusions

In this paper a 3D volume of fluid (VOF) method has been presented featuring (i) an interface reconstruction technique based on piecewise linear interface representation, (ii) a 3D version of the CSF model of Brackbill et al. (1992). Several (severe) tests were conducted which revealed that the advection and interface reconstruction algorithms work properly and the model for the surface tension forces has been implemented correctly. It was also assessed that the computed terminal rise velocities were not affected by the size of the computational domain in case the lateral dimensions of the box in the horizontal directions (i.e. the *x*- and *y*-directions) exceed 3–4 times the (initial) bubble diameter.

Subsequently the computed terminal Reynolds numbers and shapes of isolated gas bubbles rising in quiescent liquids were compared with data taken from the bubble diagram of Grace. It was demonstrated that the computed shapes and rise velocities agreed very well with the data taken from this diagram. Finally the model was successfully applied to a case where the interface experiences substantial changes, i.e. co-axial and oblique coalescence of two gas bubbles rising in a viscous liquid. In all cases studied the volume conservation, quantified by Eq. (4.3b), was very good.

Notation

a	computational domain in *x*-direction, m
b	computational domain in *y*-direction, m
d	plane constant for interface segment cutting through Eulerian cell, dimensionless
d_e	equivalent bubble diameter, m
D	distribution or smoothening function, dimensionless
E_1	relative advection error, dimensionless
E_2	relative volume conservation error, dimensionless
Eo	Eotvos number, dimensionless
F	fractional amount of liquid, dimensionless
$\tilde{F}$	smoothed colour function, dimensionless
F_σ	volumetric surface tension force, N/m^3
g	gravitational acceleration, m/s^2
h	width of the computational stencil for the smoothening function, m
M	Morton number, dimensionless
n_i	*i*th component of the unit normal vector, dimensionless
p	pressure, N/m^2
R	bubble radius, m
Re	Reynolds number, dimensionless
t	Time, s
v_∞	terminal bubble rise velocity, m/s
x_i	*i*th co-ordinate direction, m
x	*x*-co-ordinate, m
y	*y*-co-ordinate, m
z	*z*-co-ordinate, m

Greek letters

$\Delta\rho$	density difference, kg/m^3
Δp	pressure difference, N/m^2
Δx_i	grid spacing in *i*th co-ordinate direction, m
ΔV	volume of computational cell, m^3
μ	dynamic viscosity, kg/(m s)
ρ	density, kg/m^3
σ	surface tension, N/m
Ψ	stream function, m^2/s

Vectors

$\bar{m}$	normal vector
$\bar{n}$	unit normal vector
$\bar{u}$	velocity, m/s

Subscripts and superscripts

1,2	phase number
l	liquid phase
x	*x*-direction
y	*y*-direction
z	*z*-direction

Operators

$\otimes$	dyadic vector product
$\dfrac{\partial}{\partial t}$	partial time derivative, s^{-1}

$\dfrac{D}{Dt}$ substantial derivative, s^{-1}

∇ gradient operator, m^{-1}

$\nabla\cdot$ divergence operator, m^{-1}

T   transpose of a tensor

References

Brackbill, J.U., Kothe, D.B., Zemach, C., 1992. A continuum method for modeling surface tension. Journal of Computational Physics 100, 335.

Brereton, G., Korotney, D., 1991. Coaxial and oblique coalescence of two rising bubbles. In: Sahin, I., Tryggvason, G. (Eds.), Dynamics of Bubbles and Vortices Near a Free Surface, AMD-vol. 119 ASME, New York.

Bussman, M., Mostaghimi, J., Chandra, S., 1999. On a three-dimensional volume tracking model of droplet impact. Physics of Fluids 11, 1406–1417.

Centrella, J., Wilson, J., 1984. Planar numerical cosmology. II. The difference equations and numerical tests. Astrophysical Journal Supplement Series 54, 229.

Chang, Y.C., Hou, T.Y., Merriman, B., Osher, S., 1996. A level set formulation of Eulerian interface capturing methods for incompressible fluid flows. Journal of Computational Physics 124, 449–464.

Clift, R., Grace, J.R., Weber, M., 1978. Bubbles, Drops and Particles. Academic Press, New York.

Deen, N.G., van Sint Annaland, M., Kuipers, J.A.M., 2004. Multi-level modelling of dispersed gas–liquid two-phase flow. Chemical Engineering Science 59, 1853–1861.

Delnoij, E., 1999. Fluid dynamics of gas–liquid bubble columns: a theoretical and experimental study. Ph.D. Thesis, Twente University, The Netherlands.

Esmaeeli, A., Tryggvason, G., 1998a. Direct numerical simulation of bubble flows. Part I. Low Reynolds number arrays. Journal of Fluid Mechanics 377, 313–345.

Esmaeeli, A., Tryggvason, G., 1998b. Direct numerical simulation of bubble flows. Part II. Moderate Reynolds number arrays. Journal of Fluid Mechanics 385, 325–358.

Fedkiw, R., Osher, S., 2001. Level-set methods: an overview and some recent results. Journal of Computational Physics 169, 463.

Grace, J.R., 1973. Shapes and velocities of bubbles rising in infinite liquids. Transactions of the Institution of Chemical Engineering 51, 116–120.

Hirt, C.W., Nichols, B.D., 1981. Volume of fluid (VOF) method for the dynamics of free boundaries. Journal of Computational Physics 39, 201.

Ida, M., 2000. An improved unified solver for compressible and incompressible fluids involving free surfaces. Part I. Convection. Computer Physics Communications 132, 44–65.

Ladd, A.J.C., 1994a. Numerical simulations of particulate suspensions via a discretised Boltzmann equation. Part 1. Theoretical foundation. Journal of Fluid Mechanics 271, 285–309.

Ladd, A.J.C., 1994b. Numerical simulations of particulate suspensions via a discretised Boltzmann equation. Part 2. Numerical results. Journal of Fluid Mechanics 271, 311–339.

Noh, W.F., Woodward, P.R., 1976. SLIC (Simple Line Interface Calculation) method. In: van de Vooren, A.I., Zandbergen, P.J. (Eds.), Lecture Notes in Physics. p. 330.

Peskin, C.S., 1977. Numerical analysis of blood flow in the heart. Journal of Computational Physics 25, 220–252.

Popinet, S., Zaleski, S., 1999. A front-tracking algorithm for accurate representation of surface tension. International Journal of Numerical Methods in Fluids 30, 775–793.

Prosperetti, A., 2001. Navier–Stokes numerical algorithms for free-surface flow computations: an overview. Drop Surface Interactions, 21.

Rider, W.J., Kothe, D.B., 1995. Stretching and tearing interface tracking methods, Los Alamos National Laboratory. Available on World Wide Web at: http://laws.lanl.gov/XHM/personnel/wjr/Web_papers/pubs.html.

Rider, W.J., Kothe, D.B., 1998. Reconstructing volume tracking. Journal of Computational Physics 141, 112–152.

Rudman, M., 1997. Volume-tracking methods for interfacial flow calculations. International Journal of Numerical Methods in Fluids 24, 671–691.

Rudman, M., 1998. A volume-tracking method for incompressible multifluid flows with large density variations. International Journal of Numerical Methods in Fluids 28, 357–378.

Sabisch, W., Worner, M., Grotzbach, G., Cacuci, D.G., 2001. Driedimen-sionale numerische Simulation von aufsteigenden Einzelblasen und Blasenschwarmen mit einer Volume-of Fluid-Methode. Chemie Ingenieur Technik 73, 368–373.

Sankaranarayanan, K., Shah, X., Kevrekidis, I.G., Sundaresan, S., 2002b. Analysis of drag and added mass forces in bubbly suspensions using an implicit formulation of the lattice Boltzmann method. Journal of Fluid Mechanics 452, 61–96.

Sankaranarayanan, K., Sundaresan, S., 2002a. Lift force in bubble suspensions. Chemical Engineering Science 57, 3521–3542.

Scardovelli, S., Zaleski, S., 1999. Direct numerical simulation of free-surface and interfacial flow. Annual Review of Fluid Dynamics 31, 567–603.

Sethian, J.A., 1996. Level Set Methods. Cambridge University Press, Cambridge, UK.

Sussman, M., Fatemi, E., 1999. An efficient interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow. SIAM Journal on Scientific Computing 20, 1165–1191.

Sussman, M., Smereka, P., 1997. Axi-symmetric free boundary problems. Journal of Fluid Mechanics 341, 269–294.

Sussman, M., Smereka, P., Osher, S., 1994. A level set approach for computing solutions to incompressible two-phase flow. Journal of Computational Physics 114, 146–159.

Sussman, M., Almgren, A.S., Bell, J.B., Colella, P., Howell, L.H., Welcome, M.L., 1999. An adaptive level set approach for incompressible two-phase flows. Journal of Computational Physics 148, 81–124.

Tryggvasson, G., Bunner, B., Esmaeeli, A., 2001. A front tracking method for the computations of multiphase flow. Journal of Computational Physics 169 (2), 708–759.

Unverdi, S.O., Tryggvason, G., 1992. A front-tracking method for viscous, incompressible multi-fluid flows. Journal of Computational Physics 100, 25–37.

Welch, J.E., Harlow, F.H., Shannon, J.P., Daly, B.J., 1965. The MAC method: a computing technique for solving viscous incompressible transient fluid flow problems involving free surfaces. Los Alamos Scientific Laboratory Report LA-3425.

Youngs, D.L., 1982. Time-dependent multi-material flow with large fluid distortion. In: Morton, K.W., Bianes, M.J. (Eds.), Numerical Methods for Fluid Dynamics. Academic Press, New York, pp. 273–285.