JOURNAL OF COMPUTATIONAL PHYSICS **124,** 449–464 (1996)
ARTICLE NO. 0072

A Level Set Formulation of Eulerian Interface Capturing Methods for Incompressible Fluid Flows

Y. C. CHANG,* T. Y. HOU,* B. MERRIMAN,† AND S. OSHER†

*Applied Mathematics, 217-50, Caltech, Pasadena, California 91125; and †Department of Mathematics, UCLA, Los Angeles, California 90024

Received January 28, 1994; revised September 21, 1995

A level set formulation is derived for incompressible, immiscible Navier–Stokes equations separated by a free surface. The interface is identified as the zero level set of a smooth function. Eulerian finite difference methods based on this level set formulation are proposed. These methods are robust and efficient and are capable of computing interface singularities such as merging and reconnection. Numerical experiments are presented to demonstrate the effectiveness of the methods. © 1996 Academic Press, Inc.

1. INTRODUCTION

In this paper, we derive a level set formulation for incompressible, immiscible Navier–Stokes equations separated by a free surface. The flow we consider has discontinuous density and viscosity. The effect of surface tension is also included. Based on this formulation, a second-order projection method can be used to approximate the evolution equations. This approach can be considered as a method of front capturing type since no explicit information about the free surfaces is required in the solution procedure. The free surface is recovered at the end of the computation by locating the zero level set of a smooth function. This numerical method is efficient and is capable of simulating incompressible flow where change of topology in the fluid interface occurs, such as merging and reconnection.

Many physically interesting problems involve propagation of free surfaces. Water waves, boundaries between immiscible fluids, vortex sheets, and Hele–Shaw flows are examples of this kind. Numerical simulation of these free surfaces presents a great challenge to numerical analysts and computational scientists because the underlying physical problem is singular and is sensitive to small numerical perturbations [13, 21]. There are two types of numerical approaches for solving free surface problems in the context of incompressible flows. One is based on front tracking where the free interfaces are explicitly tracked. The boundary integral method and some semi-Lagrangian front tracking methods are examples of this type; see, e.g., [1–3, 12, 22, 23, 28–30, 34, 35]. The advantage of this approach is to reduce the number of nodes needed to represent the front and avoid introducing numerical diffusion which smooths out the front. In the case of irrotational flow, one can reformulate the problem in the boundary integral form so that it involves the motion of the free surface alone. In this case, it is possible to design high order accurate methods as long as the fluid interface is smooth. This is essential in studying singularity formation in fluid interfaces [18, 23, 30]. However, when the interface forms a singularity or changes its topology, the tracking methods are difficult to continue beyond the singularity time. Local surgery for the moving grid points is required. This complicates the solution procedure. Moreover, three-dimensional problems are notoriously more difficult to compute using tracking methods, especially in the presence of merging.

The second approach is based on front capturing. This is the one we adopt in this paper. The capturing method we consider is based on a level set formulation. In this formulation, the boundary of a two-fluid interface is modelled as the zero set of a smooth function ϕ defined on the entire physical domain. The boundary is then updated by solving a nonlinear equation of the Hamilton–Jacobi type on the whole domain. This level set formulation of the moving interface was introduced by Osher and Sethian in [27] and was capable of computing geometric properties of highly complicated boundaries without explicitly tracking the interface. Hence, the moving boundary can develop corners, cusps, and undergo topological changes quite naturally. Moreover, the level set formulation generalizes to three-dimensional problems easily. It eliminates the problem of grid surgery encountered in the tracking approach. One of the common difficulties in the front capturing approaches is how to keep the interface thickness finite and to preserve the mass conservation. For the level set approach, as we will see later, this difficulty can be overcome by using various fast re-initialization techniques. The first fast re-initialization technique was developed and implemented in [32].

In this paper, we derive an equivalent weak formulation of the incompressible multi-fluid flow by coupling the level set formulation to the fluid equations. The effects of discontinuous density, discontinuous viscosity, and surface ten-

0021-9991/96 $18.00

sion are taken into account. The equations governing the motion of unsteady, viscous, incompressible, immiscible two-fluid system are the Navier–Stokes equations. In conservation form, the equations are

$$\rho(\mathbf{u}_t + \nabla \cdot (\mathbf{uu})) = -\nabla p + \rho \mathbf{g} + \nabla \cdot (2\mu \mathbf{D}), \tag{1}$$

where $\mathbf{u}$ is the velocity, and ρ and μ are the discontinuous density and viscosity fields, respectively. $\mathbf{D}$ is the rate of deformation tensor, whose components are $D_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i})$. The density and viscosity are purely convected by the fluid velocity:

$$\frac{\partial}{\partial t}(\rho) + \nabla \cdot (\mathbf{u}\rho) = 0, \tag{2}$$

$$\frac{\partial}{\partial t}(\mu) + \nabla \cdot (\mathbf{u}\mu) = 0. \tag{3}$$

These equations are coupled to the incompressibility condition

$$\nabla \cdot \mathbf{u} = 0. \tag{4}$$

Denote the stress tensor by $\sigma(\mathbf{x})$, which is given by

$$\sigma(\mathbf{x}) = -p\mathbf{I} + 2\mu \mathbf{D}, \tag{5}$$

where $\mathbf{I}$ is the identity matrix, $\mathbf{D}$ is the deformation tensor, and p is the pressure. We let Γ denote the fluid interface. The effect of surface tension is to balance the jump of the normal stress along the fluid interface. This gives rise to a free boundary condition for the discontinuity of the normal stress across Γ [12, 14]

$$[\sigma_{ij} n_j]|_\Gamma = \tau \kappa n_i, \tag{6}$$

where $[p]$ denotes the jump of p across the interface, κ is the curvature of Γ, τ is the surface tension coefficient, and $\mathbf{n}$ is a unit outward normal vector along Γ. Note that in the case of inviscid flows, the above jump condition is reduced to

$$[p]|_\Gamma = \tau \kappa. \tag{7}$$

In this case, the effect of the surface tension is to introduce a discontinuity in pressure across the interface proportional to the (mean) curvature.

Our level set formulation is based on the following observation. The effect of surface tension can be expressed in terms of a singular source function which is defined by our level set function. This and other similar ideas have been used by several authors in the literature; see, e.g., [10, 28, 9, 35]. Let us denote by ϕ the level set function. The fluid interface Γ corresponds to the zero level set of ϕ. We will show that the evolution equations can be reformulated as

$$\rho(\mathbf{u}_t + \nabla \cdot \mathbf{uu}) = -\nabla p + \rho \mathbf{g} + \nabla \cdot (2\mu \mathbf{D}) + \tau \kappa(\phi) \nabla \phi \delta(\phi), \tag{8}$$

$$\frac{\partial}{\partial t}\phi + \mathbf{u} \cdot \nabla \cdot \phi = 0, \tag{9}$$

where $\delta(\phi)$ is a one-dimensional Dirac delta function and ϕ is chosen in such way that $\nabla\phi$ is in the outward normal direction when evaluated on Γ. The curvature $\kappa(\phi)$ can be expressed by ϕ and its derivatives

$$\kappa(\phi) = -\frac{\phi_y^2 \phi_{xx} - 2\phi_x \phi_y \phi_{xy} + \phi_x^2 \phi_{yy}}{(\phi_x^2 + \phi_y^2)^{3/2}}. \tag{10}$$

Assume that we have chosen the initial level set function such that $\phi < 0$ defines region 1 of the fluid and $\phi > 0$ defines region 2. Further, we assume that ρ_1 and ρ_2 are the constant densities in region 1 and region 2, respectively, and μ_1 and μ_2 are the constant viscosities in region 1 and region 2, respectively. Then we have $\rho = \rho_1 + (\rho_2 - \rho_1)H(\phi)$, where H is the Heaviside function that satisfies $H(x) = 1$ for $x > 0$ and $H(x) = 0$ for $x < 0$. Similarly we have $\mu = \mu_1 + (\mu_2 - \mu_1)H(\phi)$. The evolution equations can be solved either by a projection method or by a vorticity-based method. The convection terms can be approximated by high order ENO schemes [17] or by other high order Godunov schemes. Apparently, this level set formulation works for both two-dimensional and three-dimensional problems. There are no additional complications to extend the method to three-dimensional problems.

This formulation, after we regularize the delta function, is very similar to the one obtained by Brackbill *et al.* [9], except for the reconstruction of density and viscosity from the level set function. Sussman *et al.* used our formulation (8) and obtained interesting results in their study of gas bubbles in water with a coarse grid and a large density jump [32]. One should note that only the zero level set is physically relevant. We have a lot of freedom in extending the level set function outside the interface. Later, we will exploit this freedom in our formulation to introduce certain re-initialization of the level set function. This helps preserve mass conservation and keep the thickness of the interface non-diffusive in time.

2. DERIVATION OF THE LEVEL SET FORMULATION

Here we give a derivation of the weak equivalence between the level set formulation and the original free bound-

ary problem. Consider a volume of the fluid occupying a region Ω with boundary $\partial\Omega$. We assume that the fluid interface Γ intersects with the region Ω, dividing it into two disjoint subregions Ω_1 and Ω_2. For the body of fluid in volume Ω enclosed by the material surface $\partial\Omega$, the momentum is $\int_\Omega \rho\mathbf{u}d\mathbf{x}$ and its rate of change is

$$\int_\Omega \rho \frac{D\mathbf{u}}{Dt}\, d\mathbf{x},$$

where $D\mathbf{u}/Dt = \mathbf{u}_t + \mathbf{u}\cdot\nabla\mathbf{u}$ is the material derivative. By Newton's second law, this rate of change of momentum is balanced by the forces acting on the volume Ω and the surface Ω.

The force acting on the volume Ω is due to gravity. The total volume force on Ω is given by

$$\int_\Omega \rho\mathbf{g}\, d\mathbf{x}.$$

The i-component of the surface or contact force exerted across a surface element of area ds and the normal $\mathbf{n}$ may be represented as $\sigma_{ij}n_j ds$, where σ_{ij} is the stress tensor. The total surface force exerted on Ω by the surrounding matter is thus

$$\int_{\partial\Omega} \sigma_{ij}n_j\, ds.$$

Therefore the momentum balance for the selected portion of fluid Ω is expressed by

$$\int_\Omega \rho \frac{Du_i}{Dt}\, d\mathbf{x} = \int_\Omega \rho g_i\, d\mathbf{x} + \int_{\partial\Omega} \sigma_{ij}n_j\, ds. \tag{11}$$

Here we have used summation convention: $\sigma_{ij}n_j = \sum_j \sigma_{ij}n_j$. Denote by Γ the portion of the fluid interface which is contained inside Ω. Clearly Γ is a common boundary for Ω_1 and Ω_2. Now, decompose the boundary integral over $\partial\Omega$ into boundary integrals over $\partial\Omega_1$ and $\partial\Omega_2$, respectively. We obtain

$$\begin{aligned}\int_\Omega \rho \frac{Du_i}{Dt}\, d\mathbf{x} = \int_\Omega \rho g_i\, d\mathbf{x} + \int_{\partial\Omega_1} \sigma_{ij}n_j\, ds + \int_{\partial\Omega_2} \sigma_{ij}n_j\, ds \\ + \int_\Gamma [\sigma_{ij}n_j]\, ds,\end{aligned} \tag{12}$$

where $[\sigma_{ij}n_j]$ stands for the jump of $\sigma_{ij}n_j$ across the fluid interface Γ. For the first two surface force terms, we have from the Stokes theorem that

$$\begin{aligned}\int_{\partial\Omega_1} \sigma_{ij}n_j\, ds &= \int_{\Omega_1} \frac{\partial\sigma_{ij}}{\partial x_j}\, d\mathbf{x}, \\ \int_{\partial\Omega_2} \sigma_{ij}n_j\, ds &= \int_{\Omega_2} \frac{\partial\sigma_{ij}}{\partial x_j}\, d\mathbf{x}.\end{aligned} \tag{13}$$

For the surface force term over the free surface Γ, we use the free boundary condition (6). We obtain

$$\int_\Gamma [\sigma_{ij}n_j]\, ds = \int_\Gamma \tau\kappa n_i\, ds. \tag{14}$$

We claim that

$$\int_\Gamma \tau\kappa n_i\, ds = \int_\Omega \tau\kappa(\phi(\mathbf{x}))\delta(\phi(\mathbf{x}))\phi_{x_i}\, d\mathbf{x}, \tag{15}$$

where ϕ is the level set function with $\phi = 0$ corresponding to the fluid interface Γ.

Suppose that the above equality has been established; then the level set formulation follows easily. To see this, we substitute (13)–(15) into (12). The result is

$$\int_\Omega \left(\rho \frac{Du_i}{Dt} - \frac{\partial\sigma_{ij}}{\partial x_j} - \rho g_i - \tau\kappa\delta(\phi(\mathbf{x}))\phi_{x_i}\right) d\mathbf{x} = 0. \tag{16}$$

Since this holds for the arbitrary region Ω, we conclude that

$$\begin{aligned}\rho(\mathbf{u}_t + \mathbf{u}\cdot\nabla\mathbf{u}) - \rho\mathbf{g} + \nabla p - \nabla\cdot(2\mu\mathbf{D}) \\ - \tau\kappa\delta(\phi)\nabla\phi = 0.\end{aligned} \tag{17}$$

This gives rise to the equivalent weak formulation

$$\begin{aligned}\rho(\mathbf{u}_t + \nabla\cdot\mathbf{uu}) = -\nabla p + \rho\mathbf{g} + \nabla\cdot(2\mu\mathbf{D}) \\ + \tau\kappa\delta(\phi)\nabla\phi.\end{aligned} \tag{18}$$

Now we give a proof for (15). To prove (15), we need to introduce a transverse level set function ψ which satisfies

$$\nabla\phi\cdot\nabla\psi = 0, \quad |\nabla\psi| \neq 0. \tag{19}$$

The construction of such ψ will be given later. Introduce a change of variables

$$x' = \psi(x, y), \quad y' = \phi(x, y). \tag{20}$$

Such change of variables is well defined because

$$\det\left(\frac{\partial(\psi,\phi)}{\partial(x,y)}\right) = (\phi_y, -\phi_x)\cdot(\psi_x, \psi_y) = |\nabla\phi|\,|\nabla\psi| \neq 0, \tag{21}$$

where we have used the fact that $(\phi_y, -\phi_x)$ is parallel to (ψ_x, ψ_y), and we assume that ψ is constructed in such a way that (ψ_x, ψ_y) has the same direction as the tangent direction. With this change of variables, we have

$$\int_\Omega (\tau\kappa\delta(\phi)\nabla\phi)\,d\mathbf{x} = \int_{\Omega'} (\tau\kappa\delta(y')\nabla\phi)\frac{1}{|\nabla\phi|\,|\nabla\psi|}\,dx'\,dy' = \int_{\phi=0} \mathbf{n}\frac{\tau\kappa}{|\nabla\psi|}\,dx'. \tag{22}$$

Let $(\bar{x}(s), \bar{y}(s))$ be a parameterization of the interface Γ with s being an arclength variable. Then we have along the interface $\phi = 0$,

$$dx' = d\psi(\bar{x}(s), \bar{y}(s)) = (\psi_x\bar{x}_s + \psi_y\bar{y}_s)\,ds = (\psi_x, \psi_y)\cdot(\bar{x}_s, \bar{y}_s)\,ds. \tag{23}$$

Recall that (ψ_x, ψ_y) is parallel to the tangent vector of Γ and has the same direction as $(\bar{x}_s, \bar{y}_s)$. Further, we note that $(\bar{x}_s, \bar{y}_s)$ is the unit tangent vector since s is an arclength variable. We obtain

$$dx' = |\nabla\psi|\,ds. \tag{24}$$

Therefore, we obtain

$$\int_\Omega \tau\kappa\delta(\phi)\nabla\phi\,d\mathbf{x} = \int_{\phi=0} \tau\kappa\mathbf{n}\,ds = \int_\Gamma \tau\kappa\mathbf{n}\,ds. \tag{25}$$

This proves (15).

We now show how to construct ψ with the desired properties. Since the integrand in (22) contains a Dirac delta function $\delta(\phi)$, it is sufficient to construct the transverse level set function ψ in a small neighborhood of the interface Γ. We first define a coordinate transformation $(x(s, \tau), y(s, \tau))$ by

$$\frac{d}{d\tau}(x(s,\tau), y(s,\tau)) = \nabla\phi(x(s,\tau), y(s,\tau)), \tag{26}$$

$$(x(s,0), y(s,0)) = (\bar{x}(s), \bar{y}(s)), \tag{27}$$

where $(\bar{x}(s), \bar{y}(s))$ is a parameterization of Γ. We will solve for (26) for both $\tau > 0$ and $\tau < 0$. Let $\psi_0(s)$ be any smooth increasing function. We define ψ by

$$\psi(x(s,\tau), y(s,\tau)) = \psi_0(s). \tag{28}$$

We will show that (28) defines $\psi(x, y)$ in a small neighborhood of Γ. This amounts to proving that $(x(s, \tau), y(s, \tau))$ defines a one-to-one mapping from (s, τ) to (x, y) in a small neighborhood of Γ, or equivalently,

$$\det\left(\frac{\partial(x,y)}{\partial(s,\tau)}\right) \neq 0. \tag{29}$$

From (26), we have by Taylor expanding the solution around $\tau = 0$

$$(x_\tau, y_\tau) = \nabla\phi(x(s,\tau), y(s,\tau)) = \nabla\phi(\bar{x}(s), \bar{y}(s)) + O(\tau). \tag{30}$$

On the other hand, differentiating (26) with respect to s and integrating in τ will give

$$(x_s, y_s)(s,\tau) = (\bar{x}_s(s), \bar{y}_s(s) + \int_0^\tau \frac{d}{ds}\nabla\phi(x(s,\tau'), y(s,\tau'))\,d\tau' = \mathbf{T}(s) + O(\tau), \tag{31}$$

where $\mathbf{T}$ is the unit tangent vector of Γ. Recall that $\nabla\phi$ is orthogonal to the tangent vector $\mathbf{T}$ along Γ. Therefore, for $|\tau|$ small, we obtain

$$\det\left(\frac{\partial(x,y)}{\partial(s,\tau)}\right) = |\nabla\phi|\,|\mathbf{T}| + O(\tau) = |\nabla\phi|_{\phi=0} + O(\tau) \neq 0. \tag{32}$$

This shows that $\psi(x, y)$ is well defined in a small neighborhood of Γ. Clearly, it follows from (28) that

$$0 = \frac{d}{d\tau}\psi(x(s,\tau), y(s,\tau)) = \psi_x x_\tau + \psi_y y_\tau = \nabla\psi\cdot\nabla\phi. \tag{33}$$

This proves the orthogonality of $\nabla\phi$ and $\nabla\psi$. Moreover, we have from (28) that

$$x_s\psi_x + y_s\psi_y = \frac{d}{ds}\psi_0(s) \tag{34}$$

$$x_\tau\psi_x + y_\tau\psi_y = 0. \tag{35}$$

Since $\psi_0'(s) \neq 0$ and the coefficient matrix is nonsingular, we conclude that $(\psi_x, \psi_y) \neq 0$. This shows that $|\nabla\psi| \neq 0$ in a small neighborhood of Γ.

We remark that the orientation of $\nabla\psi$ is determined by the sign of $\psi_0'(s)$. This can be seen from (34). In our case, we choose $\psi_0(s)$ to be an increasing function of s. Therefore, $\nabla\psi$ along the interface has the same direction as the unit tangent vector $(\bar{x}_s, \bar{y}_s)$.

Before we conclude this section, we would like to point out that Tryggvason's formulation [35] can be derived by an argument similar to that given above. If $\mathbf{x}(s, t)$ is a parameterization of the fluid interface, with s being the arclength variable and t being the time variable, and $\delta(\mathbf{x})$ is the two-dimensional Dirac delta function, then it is easy to show that

$$\begin{aligned}\int_\Omega \int_\Gamma \tau\kappa(\mathbf{x}(x,t))\mathbf{n}\delta(\mathbf{x}-\mathbf{x}(s,t))\,ds\,d\mathbf{x} &= \int_\Gamma \tau\kappa(\mathbf{x}(s,t))\mathbf{n}\,ds \\ &\quad \int_\Omega \delta(\mathbf{x}-\mathbf{x}(s,t))\,d\mathbf{x} \\ &= \int_\Gamma \tau\kappa(\mathbf{x}(s,t))\mathbf{n}\,ds.\end{aligned}$$

This implies that

$$\begin{aligned}\rho(\mathbf{u}_t + \nabla\cdot(\mathbf{uu})) = &-\nabla p + \rho\mathbf{g} + \nabla\cdot(2\mu\mathbf{D}) \\ &+ \int_\Gamma \tau\kappa(\mathbf{x}(s,t))\delta(\mathbf{x}-\mathbf{x}(s,t))\mathbf{n}\,ds.\end{aligned} \tag{36}$$

Note that the singular source term in this formulation is nonlocal. This is in contrast with our local expression for the singular source term in the level set formulation.

To complete the evolution equation in Tryggvason's formulation (also see Peskin [28]), we need to convect the interface position $\mathbf{x}(s, t)$ by the fluid velocity:

$$\frac{d}{dt}\mathbf{x}(s,t) = \mathbf{u}(\mathbf{x}(s,t),t), \tag{37}$$

$$\mathbf{x}(s,0) = \mathbf{x}(s). \tag{38}$$

Here s is a purely Lagrangian variable; it is not the same as the arclength variable for $t > 0$. Thus we need to modify Eq. (36) by taking the arclength metric $|\mathbf{x}_s|$ into account in the singular source term:

$$\begin{aligned}\rho(\mathbf{u}_t + \nabla\cdot\mathbf{uu}) = &-\nabla p + \nabla\cdot(2\mu\mathbf{D}) + \rho\mathbf{g} \\ &+ \int_\Gamma \tau\kappa(\mathbf{x}(s,t))\delta(\mathbf{x}-\mathbf{x}(s,t))\mathbf{n}|\mathbf{x}_s|\,ds.\end{aligned} \tag{39}$$

This subtle fact has not been clearly stated in the literature.

3. FINITE DIFFERENCE DISCRETIZATIONS

In this section, we describe finite difference discretizations for the level set formulation derived in the previous section. First, we need to introduce a regularization for the singular Dirac delta function, δ, and the discontinuous Heaviside function, H. As in [28], we define the regularized delta function δ_ε as

$$\delta_\varepsilon(x) \equiv \begin{cases} \frac{1}{2}(1+\cos(\pi x/\varepsilon))/\varepsilon & \text{if } |x| < \varepsilon, \\ 0 & \text{otherwise,}\end{cases} \tag{40}$$

and we define a corresponding regularized Heaviside function H_ε as

$$H_\varepsilon(x) \equiv \begin{cases} 0 & \text{if } x < -\varepsilon \\ (x+\varepsilon)/(2\varepsilon) + \sin(\pi x/\varepsilon)/(2\pi) & \text{if } |x| \le \varepsilon, \\ 1 & \text{if } x > \varepsilon.\end{cases} \tag{41}$$

The above Heaviside function satisfies the relation $dH_\varepsilon(x)/dx = \delta_\varepsilon(x)$. Using the regularized Heaviside function H_ε, we can define the corresponding regularized density function ρ and the regularized viscosity μ as

$$\rho_\varepsilon(x) = \rho_1 + (\rho_2 - \rho_1)H_\varepsilon(\phi(x)), \tag{42}$$

$$\mu_\varepsilon(x) = \mu_1 + (\mu_2 - \mu_1)H_\varepsilon(\phi(x)), \tag{43}$$

where ϕ is the level set function. With this regularization, the resulting evolution equations are well posed. We can consider ρ_ε and μ_ε as smooth variable density and variable viscosity. Then the second-order projection method for variable density problems introduced in [5] can be used to discretize the momentum equations. We refer the reader to [5] for a detailed description of the projection method for variable density problems. Some implementation issues for applying the projection method to the level set formulation are discussed in [32]. We will not repeat the discussion here.

From now on, we will focus our attention on a simpler problem: the Bousinesq approximation to the Navier–Stokes equations with variable densities. This corresponds to the case where the variation in the density is small. In this case, the momentum equations are reduced to

$$\begin{aligned}\mathbf{u}_t + \nabla\cdot(\mathbf{uu}) = &-\nabla p + \nabla\cdot(2\mu\mathbf{D}) + \rho'\mathbf{g} \\ &+ \tau\kappa\delta(\phi(x))\nabla\phi,\end{aligned} \tag{44}$$

where ρ' describes the density variation of the fluids [36]. ρ' is convected by the fluid velocity

$$\rho_t' + \mathbf{u}\cdot\nabla\rho' = 0. \tag{45}$$

For clarity, we will drop the prime in ρ'. With our regularization, we can rewrite the momentum equations as

$$\mathbf{u}_t + \nabla \cdot (\mathbf{uu}) = -\nabla p + \nabla \cdot (2\mu_\varepsilon \mathbf{D}) + \rho_\varepsilon \mathbf{g} + \tau\kappa\delta_\varepsilon(\phi(x))\nabla\phi. \tag{46}$$

Using a MAC grid and the projection method [15, 11, 6], it is easy to discretize the above equations. We remark that a numerical study of interface motion with a large density ratio has been carried out successfully by Sussman *et al.* in [32] using our formulation.

We can also formulate the problem in the vorticity-stream function variables. Assume that viscosity difference is sufficiently small so that we can assume the viscosities for the two fluids are the same. Consider the two-dimensional case. Let $\mathbf{x} = (x, y)$ and $\mathbf{u} = (u, v)$. Define the vorticity variable $\omega = \text{curl}(\mathbf{u}) = v_x - u_y$. In terms of the vorticity formulation, we obtain

$$\omega_t + \mathbf{u} \cdot \nabla\omega = \mu\Delta\omega - g\rho_x + \tau(\kappa\delta_\varepsilon(\phi)\phi_y)_x - \tau(\kappa\delta_\varepsilon(\phi)\phi_x)_y.$$

By direct calculations, we can show that

$$(\kappa\delta_\varepsilon(\phi)\phi_y)_x - (\kappa\delta_\varepsilon(\phi)\phi_x)_y) = (\kappa_x\delta_\varepsilon\phi_y - \kappa_y\delta_\varepsilon\phi_x).$$

Therefore, the vorticity equations can be rewritten as

$$\omega_t + \mathbf{u} \cdot \nabla\omega = \mu\Delta\omega - g\rho_x + \tau(\kappa_x\delta_\varepsilon\phi_y - \kappa_y\delta_\varepsilon\phi_x). \tag{47}$$

And the velocity field is related to the vorticity field through a stream function ψ. That is,

$$u = \psi_y, \quad v = -\psi_x, \tag{48}$$

and ψ satisfies

$$\Delta\psi = -\omega. \tag{49}$$

Now it is a trivial matter to discretize (47)–(49). To present the method more easily, some simplifying assumptions are made on the domain and the boundary conditions. First, we consider flows in a unit square domain Ω and assume the flow is periodic in the x-direction. The boundary conditions at $y = 0$ and $y = 1$ are assumed to be no-slip and no-flow boundary conditions; i.e., $u = 0$, $v = 0$. A corresponding vorticity boundary condition can be obtained in terms of the stream function. More general domains can be considered by mapping the domain into a rectangular or a circular domain; see [19].

3.1. *Centered Difference Approximations*

In this subsection, we consider centered difference approximations to the level set formulation in the vorticity stream function formulation. An $N \times N$ grid (with spacing $h = 1/N$) is laid on the domain Ω. We consider continuous time approximations $\psi_{i,j}(t)$ to $\psi(ih, jh; t)$. Approximations $\omega_{i,j}$, $u_{i,j}$, $v_{i,j}$, $\rho_{i,j}$, and $\phi_{i,j}$ are defined similarly. Introduce the following difference operators:

$$D_0^x f_{i,j} = (f_{i+1,j} - f_{i-1,j})/2h \quad \text{(centered)},$$
$$D_-^x f_{i,j} = (f_{i,j} - f_{i-1,j})/h \quad \text{(backward)},$$
$$D_+^x f_{i,j} = (f_{i+1,j} - f_{i,j})/h \quad \text{(forward)}.$$

The operators D_0^y, D_0^y, and D_+^y are defined similarly. The centered difference approximation to the Laplacian is denoted by Δ_h which can be written as

$$\Delta_h = D_-^x D_+^x + D_-^y D_+^y. \tag{50}$$

In terms of these difference operators, we approximate Eqs. (47)–(49) by

$$\frac{d\omega_{i,j}}{dt} = -u_{i,j}D_0^x\omega_{i,j} - v_{i,j}D_0^y\omega_{i,j} + \mu\Delta_h\omega_{i,j} - gD_0^x\rho_{i,j} + \tau(D_0^x\kappa_{i,j}D_0^y\phi_{i,j} - D_0^y\kappa_{i,j}D_0^x\phi_{i,j})\delta_\varepsilon(\phi_{i,j}), \tag{51}$$

$$\frac{d\phi_{i,j}}{dt} = -u_{i,j}D_0^x\phi_{i,j} - v_{i,j}D_0^y\phi_{i,j} + \nu\Delta_h\phi_{i,j}, \tag{52}$$

$$\kappa_{i,j} = -\frac{(D_0^y\phi_{i,j})^2 D_-^x D_+^x\phi_{i,j} - 2D_0^x\phi_{i,j}D_0^y\phi_{i,j}D_0^x D_0^y\phi_{i,j} + (D_0^x\phi_{i,j})^2 D_-^y D_+^y\phi_{i,j}}{(D_0^x\phi_{i,j})^2 + (D_0^y\phi_{i,j})^2)^{3/2}}, \tag{53}$$

$$\rho_{i,j} = \rho_1 + (\rho_2 - \rho_1)H_\varepsilon(\phi_{i,j}), \tag{54}$$

$$\Delta_h\psi_{i,j} = -\omega_{i,j}, \tag{55}$$

$$u_{i,j} = D_0^y\psi_{i,j}, \tag{56}$$

$$v_{i,j} = -D_0^x\psi_{i,j}, \tag{57}$$

where ν is a numerical viscosity for the level set equation. In practice, we take $\nu = O(h^2)$. We need to introduce the numerical viscosity only when the interface Γ forms a singularity and changes its topology. In general, if a discretization of upwinding type is used in the convection terms, it will introduce some amount of numerical viscosity. Here, we want to illustrate that a second-order numerical viscosity is sufficient to stabilize a centered difference discretization when the interface forms a singularity. Actually, since the discontinuity in the fluid velocity is a contact discontinuity, the usual upwinding discretization for the convection terms may be more diffusive than the high order centered difference discretization using second-order numerical diffusion. The reason for choosing a second-order diffusion is because the smallest scale associated with the second-order diffusion is of the order $(\nu)^{1/2}$ and we

need to have a few grid points per wavelength to resolve the smallest scales. In our calculations using centered difference approximations, we use $\nu \approx (4h)^2$. The results compare favorably with the corresponding calculations using high order ENO schemes for the inviscid level set equation for ϕ.

It is not difficult to modify the above discretizations to make a fourth-order centered difference discretizations. As was pointed out in [24, 8], a fourth-order method is more effective than the second-order discretization. It requires fewer points per wavelength to resolve the smallest scales.

3.2. *High-Order ENO Discretizations for the Convection Terms*

Now we describe a high-order ENO discretization for the convection terms. Since we are interested in computing accurately the convection of the interface position, we will use the non-conservative form of the ENO scheme. Let us illustrate how to discretize the convection terms by a second-order ENO scheme. Define a minmod function as

$$\text{minmod}(u, v) = \begin{cases} \text{sgn}(u)\min(|u|, |v|) & \text{if } u \cdot v > 0, \\ 0 & \text{otherwise.} \end{cases} \quad (58)$$

Here sgn is the sign function. Then the second-order discretization of the convection term $u\phi_x$ is given by

$$u_{i,j}(D^x_- \phi_{i,j} + \frac{h}{2}\text{minmod}(D^x_- D^x_+ \phi_{i,j}, D^x_- D^x_+ \phi_{i-1,j})), \quad \text{if } u_{i,j} > 0, \quad (59)$$

$$u_{i,j}(D^x_+ \phi_{i,j} - \frac{h}{2}\text{minmod}(D^x_- D^x_+ \phi_{i,j}, D^x_- D^x_+ \phi_{i+1,j})), \quad \text{otherwise.}$$

Similarly we can define the second-order ENO discretization for the convection term in the y-direction, $v\phi_y$. A third-order ENO discretization has also been used in our calculations. For a derivation of the discretization, we refer the reader to [17, 31].

The finite difference method which uses the ENO scheme for the convection terms differs from the centered difference scheme described in the previous subsection only in the way the convection terms are discretized. The discretizations of all other terms are the same as before.

3.3. *Re-Initialization of Level Set Functions and Mass Conservation*

In general, even if we initialize the level set function ϕ as a signed distance from the front, the level set function will not remain a distance function at later times. For large time computation, it would be desirable to keep the level set function as a distance function. This will ensure that the front has a finite thickness of order ε for all time. In [32], an iterative procedure was proposed to re-initialize the level set function at each time step so that the re-initialized level set function remains a distance function from the front. Specifically, given a level set function, ϕ_0, at time t, solve for the steady state solution of the equation

$$\frac{\partial}{\partial t}\phi = \text{sgn}(\phi_0)(1 - |\nabla\phi|)$$

$$\phi(\mathbf{x}, 0) = \phi_0(\mathbf{x}),$$

where sgn is the sign function. The solution ϕ will have the same sign and the same zero level set as ϕ_0, and it satisfies $|\nabla\phi| = 1$, and so it is a distance function for the front.

Another important issue is mass conservation. For incompressible flows, the total mass is conserved in time. However, the numerical discretization of the level set formulation does not preserve this property in general. Even with the above reinitialization procedure for the level set function, it has been found that a considerable amount of total mass is lost in time. To overcome this difficulty, Hou [20] recently proposed to introduce another re-initialization procedure aimed at preserving the total mass in time. Motivated by the observation that numerical diffusion introduces a normal motion proportional to the interface's local curvature (see, e.g., [25]), he introduces a re-initialization procedure to remedy this effect. The re-initialization procedure involves solving the following perturbed Hamilton–Jacobi equation to a steady state,

$$\frac{\partial}{\partial t}\phi + (A_0 - A(t))(-P + \kappa)|\nabla\phi| = 0$$

$$\phi(\mathbf{x}, 0) = \phi_0(\mathbf{x}),$$

where A_0 denotes the total mass for the initial condition at $t = 0$ and $A(t)$ denotes the total mass corresponding to the level set function $\phi(t)$ in the above re-initialization process. P is a positive constant. It helps stabilize this re-initialization procedure. The above perturbed Hamilton–Jacobi equation may look ill-posed by itself. But since we will solve this perturbed Hamilton–Jacobi equation with the governing level set equation and the solution procedure for the governing level set equation introduces numerical viscosity, the combined fractional step method can be shown to be stable. Moreover, since we will perform this re-initialization at every time step, the difference in the mass, $(A_0 - A(t))$, is of order $O(h^2)$ in the smooth region. So this re-initialization step does not change the overall

second-order accuracy of the method. More discussions and numerical examples can be found in [20]. In principle, this allows conservation of the total mass up to an arbitrary accuracy, depending on the stopping criteria for obtaining a steady state solution. A numerical example is included in this paper to demonstrate the effect of this mass-preserving re-initialization procedure. The result is quite encouraging.

4. NUMERICAL RESULTS

In this section, we will apply our numerical method to several problems. The first one is the merging of two fluid bubbles with the same density. The second one is the interaction of three-density interfaces. The last one is the study of the Rayleigh–Taylor instability for a periodic jet under the influence of gravity. We also perform a resolution study to verify convergence of the method. In the first numerical example, we use a second-order projection method in solving the Navier–Stokes equations and we use a second-order ENO scheme in the discretization of the convection terms. In the second and third examples, we use a fourth-order centered difference scheme to approximate the inviscid Euler equations in the vorticity stream function formulation. A second-order explicit numerical viscosity is used to avoid introducing numerical oscillations. From our computational experiences, we find that the second-order ENO scheme is more robust, especially when we have a large density ratio. On the other hand, the fourth-order centered difference scheme with a second-order numerical viscosity is more accurate and less diffusive.

The purpose of our numerical experiments is to demonstrate that detailed interfacial structures during a topological transition can be captured accurately using our level set formulation. For this reason, we have performed a series of resolution studies. There are a few numerical parameters that might affect the convergence of the solution. One is the numerical grid size, h. One is the smoothing parameter, ε, for the regularized delta function. In our calculations, we relate these two parameters by the relation, $\varepsilon = 2.5h$. The numerical viscosity is also related to the grid size h. For the second-order ENO scheme, it will produce a numerical viscosity proportional to $O(h^2)$ in the smooth region. When a topological change takes place in a free interface, it signals the formation of a singularity. In that case, it is natural to expect that these numerical regularization parameters would have a strong effect for the fine structure of the interface. On the other hand, as we refine the mesh size, we expect that the numerical solution converges to a weak viscosity solution.

The numerical solutions we present here are mostly carried out using a 256×256 grid. These numerical solutions are well resolved. When we compare these solutions with the corresponding 512×512 grid calculations, they are almost indistinguishable. A coarser grid calculation gives a similar result qualitatively. But there is a difference in the detailed structure such as the time of merging, and the fine structure in the rollup region. If one is only interested in the gross features of the interface motion for large times, then a coarse grid calculation would be sufficient. But if the detailed information during the topological transition has a great impact on the solution structure at later times, then a fine grid calculation is necessary. Here we focus our attention to test the capability of the method in capturing the fine scale structure when we resolve the physical solution. For this purpose, we make sure that we have enough numerical resolution so that the physical effects of the viscosity and surface tension are accurately captured. Coarse grid calculations for the large time dynamics of water drops in the air with large density jumps have been carried out by Sussman, Smereka, and Osher using our formulation [32].

4.1. *Merging Two Bubbles with the Same Density*

In our first example, we compute the interaction of two fluid bubbles of the same density under the influence of gravity. The fluid is at rest initially. Viscosity for the fluid inside the two bubbles is equal to $\mu = 0.00025$. Viscosity for fluid outside the bubbles is equal to $\mu = 0.0005$. The surface tension is set to zero in this example. The effect of surface tension will be considered in the next example. The initial positions of the two bubbles correspond to two circles. The lower one is centered at $(0.5, 0.35)$ with radius 0.1. The upper one is centered at $(0.5, 0.65)$ with radius 0.15.

We take the density inside the two bubbles to be 1 and the density outside the bubbles to be 10. Since the two bubbles have a lighter density than that of the background fluid, they will rise in time. In this study, it is reasonable to impose simple periodic boundary conditions in both directions.

Our first calculation uses a second-order projection method. We use the second-order ENO scheme in the discretization of the convection term. The time discretization is performed using a second Adams–Bashforth method. In Fig. 1a, we plot the evolution of the two bubbles at time $t = 0.0, 0.1, 0.2, 0.3, 0.325, 0.4, 0.45, 0.5$, using 256×256 grid points. We can see that the bottom portions of the bubbles travel the fastest. As time evolves, the lower bubble produces a jet, moving upward. In the process, two opposite signed vorticity fields are created in the wake of the large bubble. This produces a lower pressure region behind the large bubble and generates flow streaming into the symmetry line of the flow. As a result, the front portion of the small bubble becomes narrower and sharper. At time $t = 3$, we see that the head of the small bubble almost catches up with the bottom of the large bubble. In the next moment, $t = 0.325$, the two bubbles merge into a single

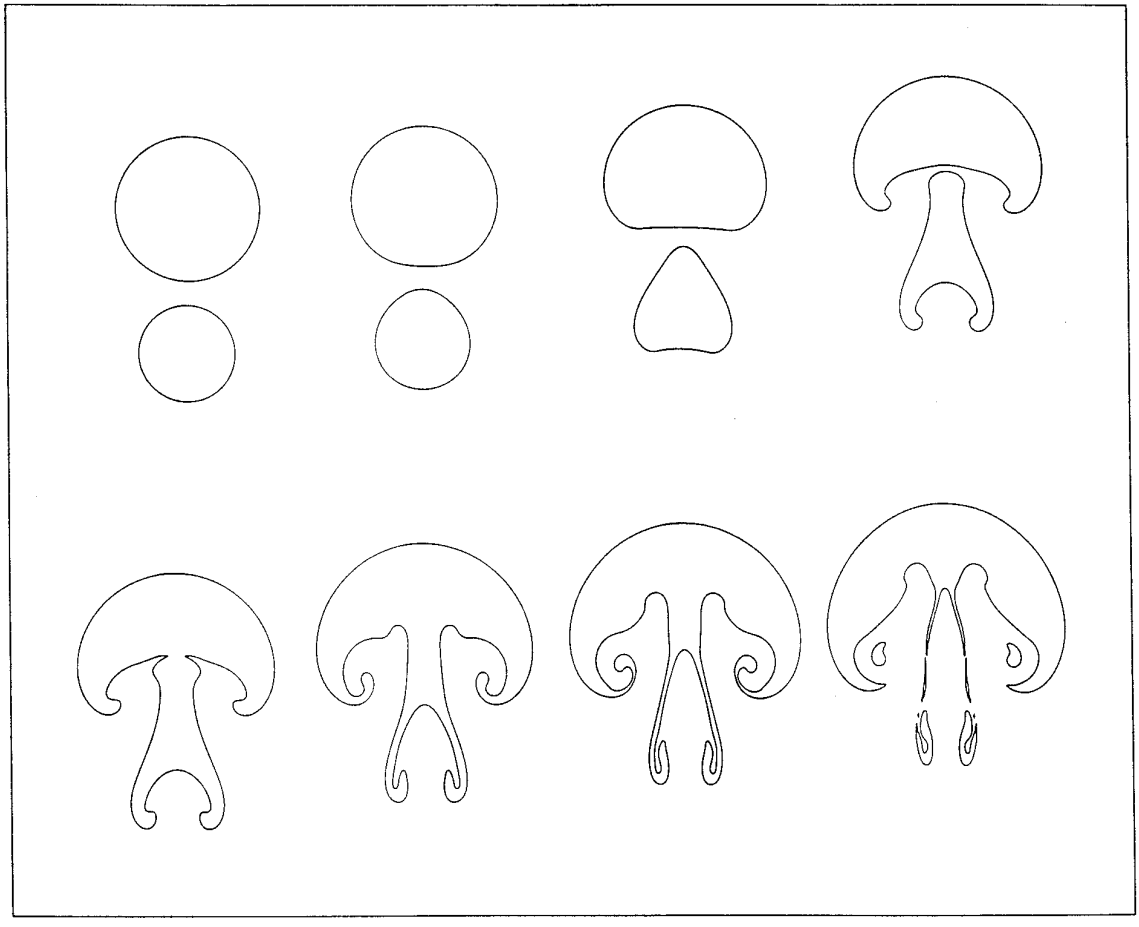


FIG. 1. Second-order ENO approximations for two bubbles of the same density. The density ratio between the bubbles and the background is 1 : 10. Here $N = 256$, $\mu_1 = 0.0005$, $\mu_2 = 0.00025$, $\varepsilon = 0.01$, and $\tau = 0$; $t = 0, 0.1, 0.2, 0.3$ for the first row, and $t = 0.325, 0.4, 0.45, 0.5$ for the second row.

bubble. At this time, the interface conjunction forms a cusp singularity. This sharp cusp is smoothed out by viscosity in time. At time $t = 0.4$, the interface becomes smooth again. On the other hand, the vorticity created in the bottom of the large bubble generates a rollup. The bottom part of the small bubble forms a jet which tries to penetrate through the stem of the merged bubble. By $t = 0.5$, we observe that a secondary topological change takes place in several regions. The drops in the rollup region of the large bubble have pinched off.

Next, we consider the effect of surface tension. In Fig. 2, we compute the interaction of two bubbles using the second-order ENO discretization and 256×256 grid points. Again a second-order projection method is used in the discretization of the Navier–Stokes equations and the second-order Adams–Bashforth method is used in the time integration. The surface tension is taken to be $\tau = 0.005$. With the other parameters being the same, our numerical experiments show that it takes a longer time for the two bubbles to merge in the case with surface tension. The initial bubble interfaces are still the same as in Fig. 1. In Fig. 2, we plot the interface positions at $t = 0.0, 0.1, 0.2, 0.3, 0.375, 0.4, 0.45, 0.5$. We observe that the bubbles merge right before $t = 0.4$. The cusp singularity at the time of merging is regularized quickly by the surface tension and the fluid viscosity. We can see that the head of the small bubble is rounder than that without surface tension. Also the "legs" of the small bubble are relatively straight and do not roll up. This is because the surface tension has an additional regularization effect near the region of large curvature.

4.2. *Interaction of Bubbles of Three-density Interfaces*

We now illustrate how to generalize the level set formulation to include the interaction of three-density interfaces. Let us denote the density of the bubble on the top by ρ_1, the density of the bubble on the bottom by ρ_2, and the background density by ρ_3. Furthermore, we denote the interface position of the bubble on the top as $\phi = d_1$, the

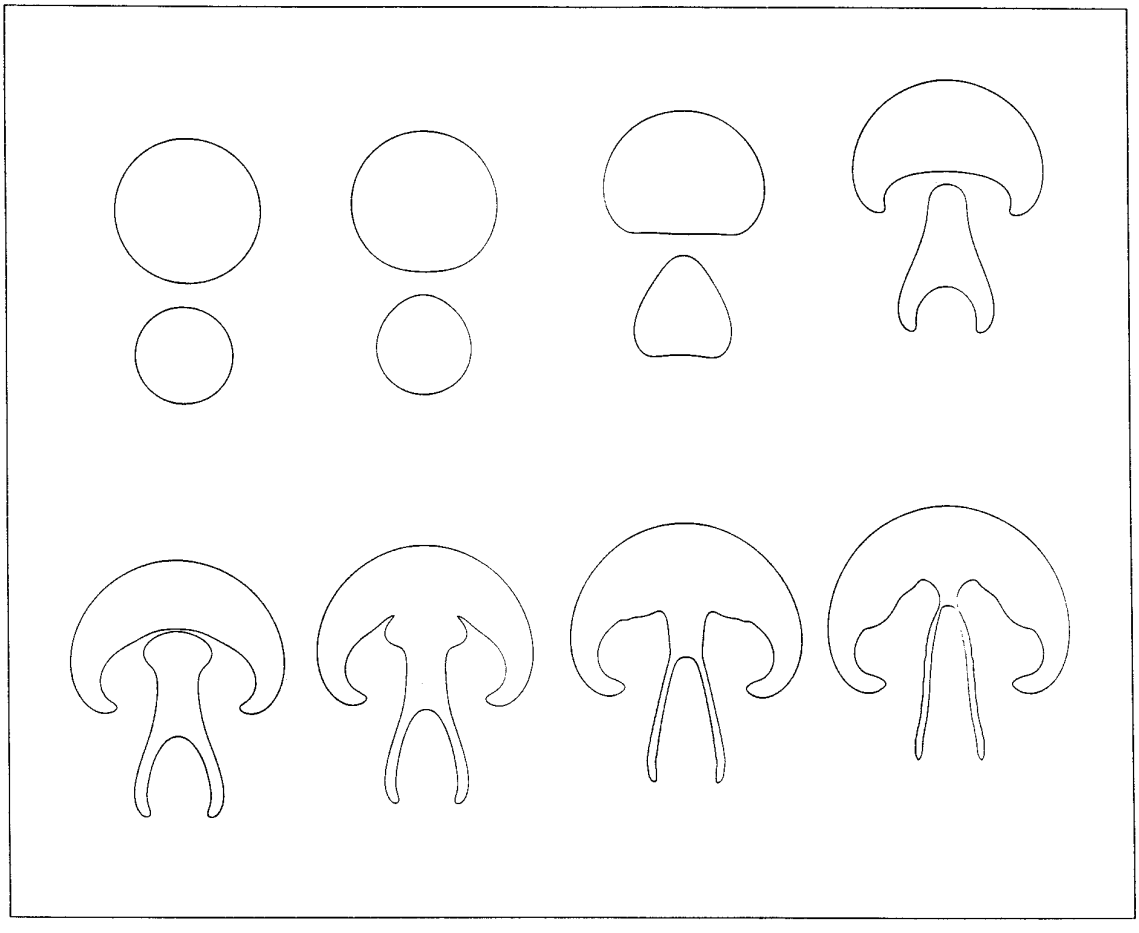


FIG. 2. Second-order ENO approximations for two bubbles of the same density. The density ratio between the bubbles and the background is 1:10. Here $N = 256$, $\mu_1 = 0.0005$, $\mu_2 = 0.00025$, $\varepsilon = 0.01$, and $\tau = 0.005$; $t = 0, 0.1, 0.2, 0.3$ for the first row, and $t = 0.375, 0.4, 0.45, 0.5$ for the second row.

interface position of the bubble on the bottom as $\phi = d_2$, with $d_1 > d_2$. The level set function ϕ is chosen such that $\phi > d_1$ corresponds to the region occupied by the bubble on the top; $\phi \leq d_2$ corresponds to the region occupied by the bubble on the bottom; and $d_1 > \phi > d_2$ corresponds to the background fluid. Then it is easy to show that the density function $\rho(\mathbf{x})$ can be expressed as

$$\rho(\mathbf{x}) = (\rho_1 - \rho_3)H(\phi(\mathbf{x}) - d_1) + (\rho_3 - \rho_2)H(\phi(\mathbf{x}) - d_2) + \rho_2. \tag{60}$$

The singular source term for the surface tension now becomes

$$\kappa(\delta(\phi(\mathbf{x}) - d_1) + \delta(\phi(\mathbf{x}) - d_2))\nabla\phi. \tag{61}$$

The curvature κ is computed in the same way as before. At the end of the calculations, the interface position for the bubble on the top is given by the level set $\phi = d_1$, the interface for the bubble in the bottom is given by $\phi = d_2$.

In Fig. 3, we illustrate the method by considering the interaction of two fluid bubbles with different densities using 256×256 grid points. The density for the bubble on the top is 60, the density for the bubble on the bottom is 1, and the background density is 3600. This example is designed to produce a strong interaction among different interfaces within a short time. This particular choice of density ratio is clearly motivated by the purpose of testing our numerical method rather than from a physical consideration, for the density ratio is beyond the validity of the Bousinesq approximation. Here we consider the inviscid Euler equations and use a fourth-order centered difference approximation to the vorticity stream-function formulation. A second-order numerical viscosity is used for the Euler equations and for the level set equations, respectively. In our calculation, the numerical viscosity is taken to be $\mu = 0.000125$ for the Euler equations, and the

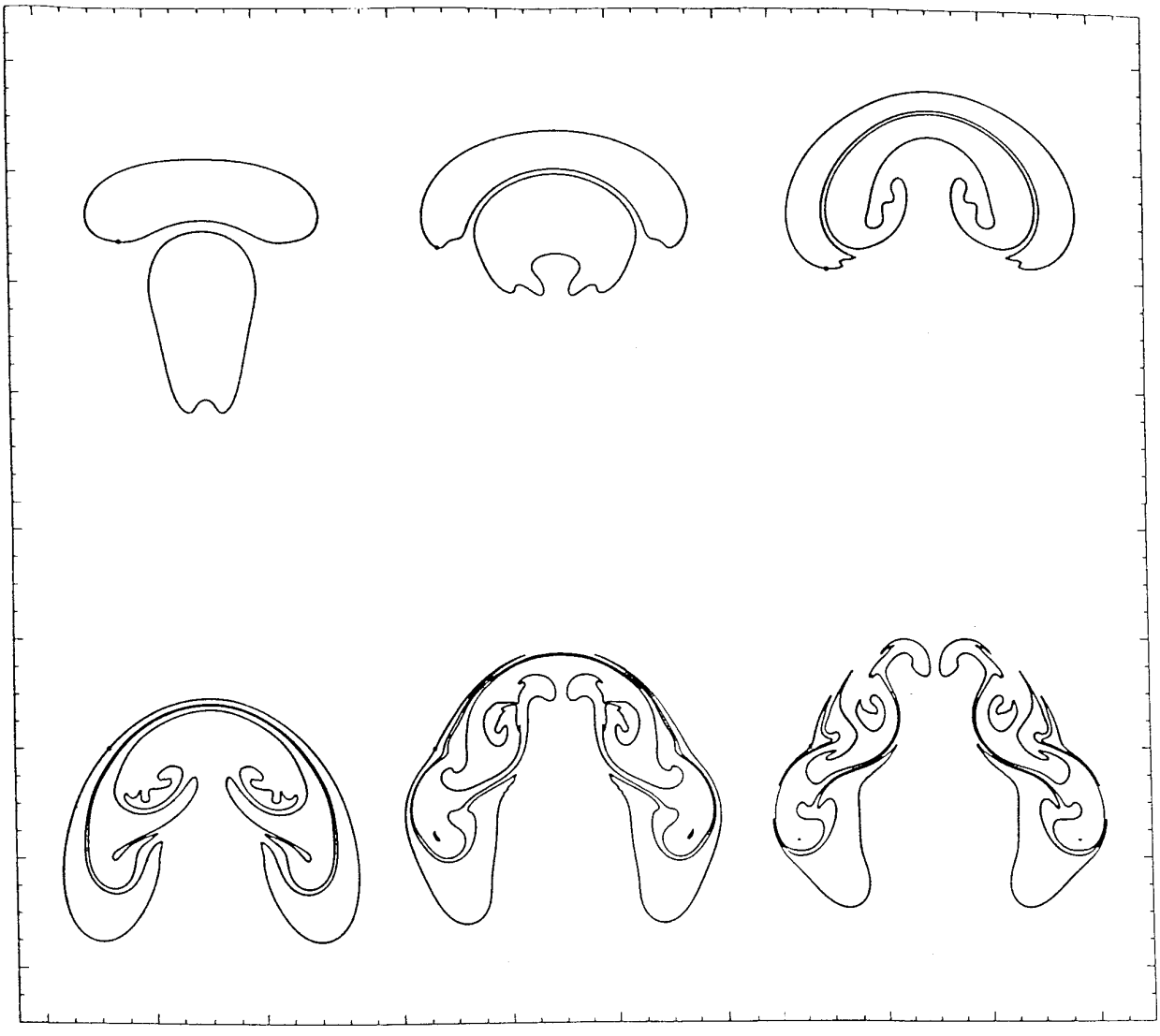


FIG. 3. Fourth-order difference approximations for two fluid bubbles with different densities. The density ratio is 1:60:3600 with the bottom bubble being the lightest. Numerical viscosity for the Euler equations is equal to 0.00025. Numerical viscosity for the level set equations is equal to 0.000125; $\tau = 0$ and $\varepsilon = 0.015$; $t = 0.1, 0.15, 0.2$ for the first row ($N = 256$), and $t = 0.275, 0.325, 0.35$ for the second row ($N = 512$).

numerical viscosity is taken to be $\nu = 0.00025$ for the level set equation. The time integration is carried out using a fourth-order Runge–Kutta method. The initial interfaces of the bubbles are of elliptical shapes. The problem is set up in such a way that both bubbles will rise in time, and the bottom bubble rises the fastest. As the bottom bubble rises in time, we see that the top portions of the bubble interfaces are almost in contact. But they cannot merge into a single bubble in this case because the densities are different for these two bubbles. In the meantime, the bubble in the bottom forms a strong jet and develops a rollup. We plot the solutions at $t = 0.1, 0.15, 0.2, 0.275, 0.325, 0.35$. We increase our numerical resolutions to 512×512 for times larger than $t = 0.2$. Part of the interface that has rolled up pinches off before $t = 0.275$; two smaller bubbles are detached from the bottom bubble and have their own dynamics. As the region in between the top portions of two bubbles becomes thinner and thinner in time, they eventually pinch off at $t = 0.325$ and $t = 0.35$, respectively. In the process, many small scale structures are produced due to the unstable stratification of the fluids.

There is no re-initialization used in this numerical example. Also the level set function can develop a steep gradient when two bubbles with different densities approach each other. Alternative approaches to motion of multiple junc-

tions which include merging have been studied in [37, 33]. These methods use several level set functions.

4.3. *Vortex Sheet Rollup in a Periodic Jet*

In this example, we would like to illustrate that the level set formulation can be used naturally to compute vortex sheet rollup due to the Rayleigh–Taylor instability. It is well known that without additional regularization, the underlying physical problem is ill-posed. Small numerical errors can be amplified rapidly in time [1, 23, 30]. In the level set formulation, the regularization comes from the numerical viscosity in the discretization of the convection terms. As we will see from our numerical example, this second-order numerical viscosity is sufficient to stabilize the method. The method can compute naturally beyond the curvature singularity due to the Rayleigh–Taylor instability and produces a nice rollup. We remark that since we keep the level set function as a distance function in time, the viscous regularization in the level set equation is in fact a curvature regularization. This curvature regularization is an intrinsic geometrical regularization. When we apply this curvature regularization to the Lagrangian boundary integral formulation, it also gives a stable discretization which can compute beyond the curvature singularity in vortex sheets and produces a rollup solution similar to those obtained by Krasny [23].

In Fig. 4, we study the rollup of a periodic jet due to the Rayleigh–Taylor instability. The fluid in between the two fluid interfaces has constant density d_1, and the rest of the fluid has density d_2. In this example, $d_1 = 10$ and $d_2 = 20$. The initial interface positions consist of sinusoidal perturbations of flat interfaces. The inviscid Euler equations are discretized with $N = 256$. A fourth-order central difference scheme is used with an explicit second-order numerical viscosity $\nu = 0.00025$ in the convection terms for both the Euler equations and the level set equation. ε is chosen to be 0.02. The time integration is carried out using a fourth-order Runge–Kutta method. As before, the problem is double-periodic in two directions. The flow is at rest initially. The motion is driven by gravity only. As time increases, we see the effect of gravity forms a bottle neck. The formation of the bottle neck becomes very clear at $t = 0.3$. For $t \geq 0.35$, we see that the bottle neck turns into a jet that falls down very rapidly and produces a strong rollup due to the Rayleigh–Taylor instability. As time increases further, the flow generates tighter rolls. In the mean time, the small perturbations in the outer arms produce secondary rollups; see the figure at $t = 0.65$. This process repeats itself and generates more and more small-scale structures in time. A level set formulation for vortex motion including vortex sheets in two and three dimensions has been proposed and tested by Harabetian, Osher, and Shu [16].

4.4. *An Area-Preserving Re-initialization*

In this subsection, we present some preliminary results of the level set method with area-preserving re-initialization. As we mentioned before, this amounts to re-initializing the level set function at every time step until the level set function satisfies the following perturbed Hamilton–Jacobi equation to a steady state,

$$\frac{\partial}{\partial t}\phi + (A_0 - A(t))(-P + \kappa)|\nabla\phi| = 0$$

$$\phi(\mathbf{x}, 0) = \phi_0(\mathbf{x}),$$

where A_0 denotes the total mass for the initial condition at $t = 0$, $A(t)$ denotes the total mass corresponding to the level set function $\phi(t)$ in the above re-initialization process. We took $P = 1$ in our test. The total mass $A(t)$ is computed from the formula

$$A(t) = \sum_{i,j} (d_2 + (d_1 - d_2)H_\varepsilon(\phi(i,j)))h^2,$$

where d_1 and d_2 are the densities inside and outside the fluid interface, respectively. To accelerate the convergence, we took a larger time step in solving this equation. In Fig. 5a, we repeat the calculation presented in Fig. 1, using the area-preserving re-initialization. We stop the re-initialization procedure when the relative error between the total mass at the current time and the initial mass is less than 10^{-5}. Throughout the calculation, the number of iterations required to satisfy this error tolerance is about 5 or 6. For most of the time, it is within 2 or 3 iterations. The total mass as a function of time is plotted in Figure 6a. We clearly observe an excellent conservation of total mass in time. In comparison, a considerable amount of mass is lost towards the end of the calculation *without* using such a re-initialization; see Fig. 6b. Note that the distance function re-initialization has been used in the calculation presented in Fig. 6b. This shows that the distance function re-initialization itself is not enough to conserve the total mass. It is designed mainly to maintain the finite thickness of the interface. On the other hand, since we use a relatively fine mesh in this calculation, the loss of total mass has not yet polluted the accuracy of the fluid interface. If we compare Fig. 5a with Fig. 1, they look almost the same, except for a small difference in the last picture at $t = 0.5$. We overlap the two solutions with and without the area-preserving re-initialization at $t = 0.4$ in Fig. 5b. They are almost indistinguishable from each other.

We also perform a convergence test for our level set method. In the following table, we illustrate the numerical errors for the density, the level set function, and the velocity in the discrete L^2 norm for $N = 64, 128, 256$, $\varepsilon = 2.5h$. This convergence test is performed for the calculation

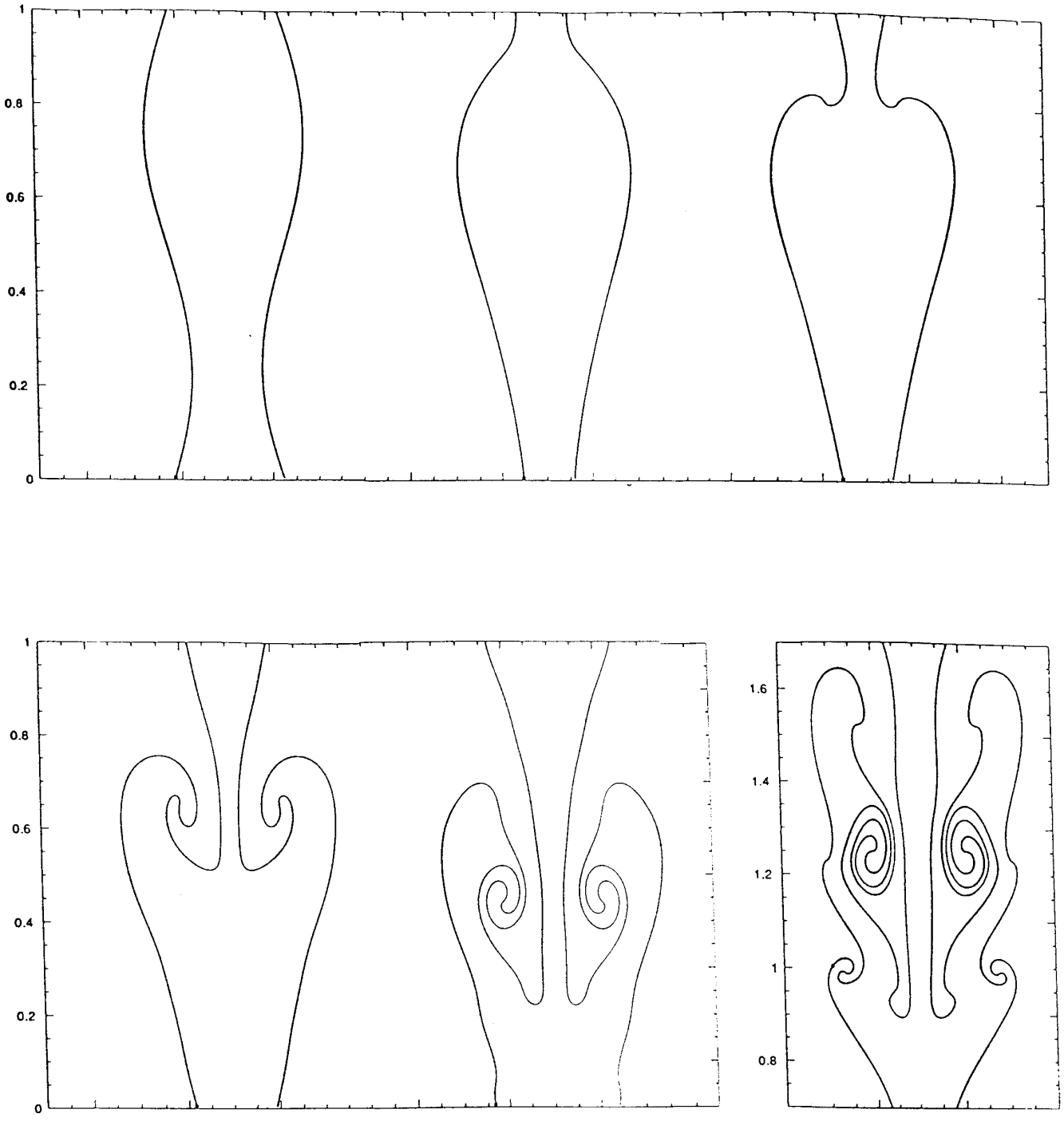
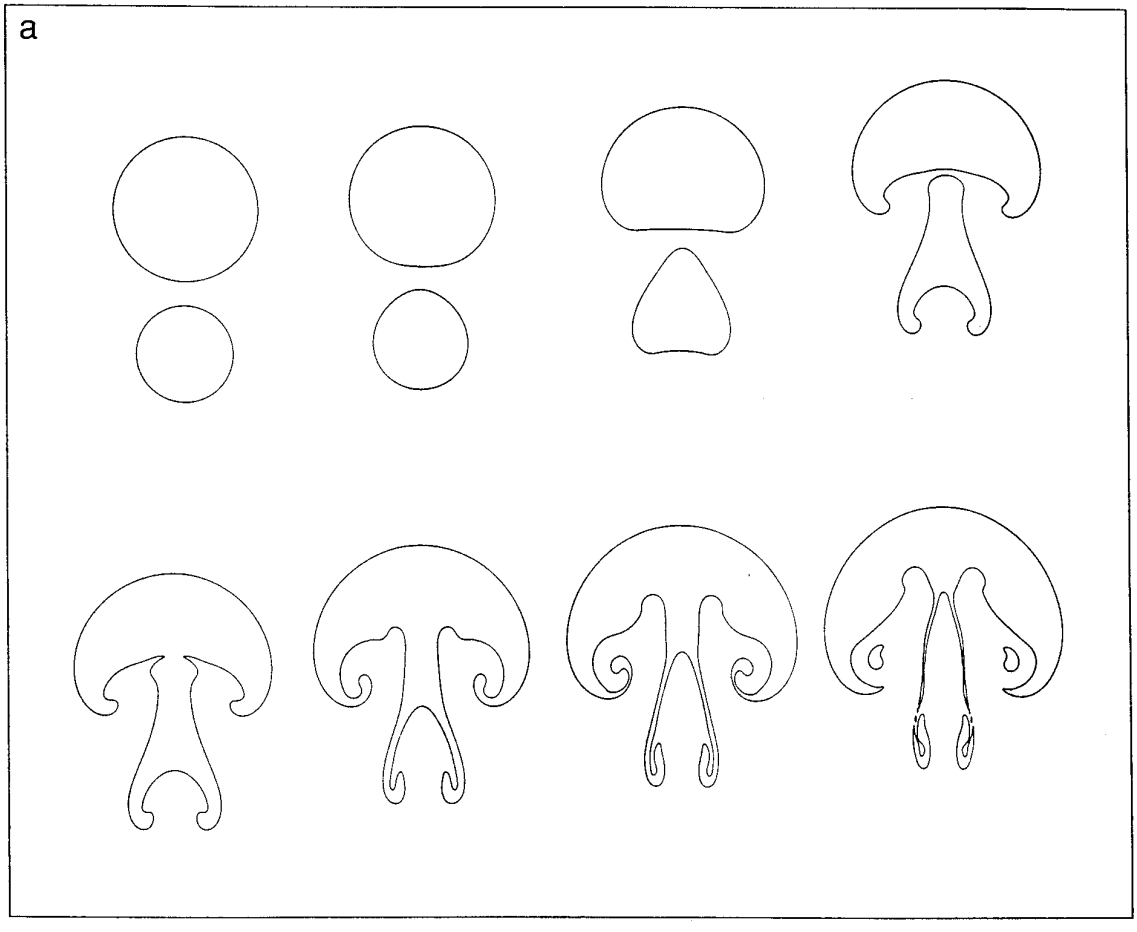


FIG. 4. Fourth-order centered difference approximations for a periodic jet. The density is equal to 10 in the region bounded by the two interfaces and equal to 20 elsewhere. Here $N = 256$, $\nu = \mu = 0.00025$, $\tau = 0$, and $\varepsilon = 0.02$; $t = 0.1, 0.3, 0.35$ for the first row and $t = 0.45, 0.55, 0.65$ for the second row.



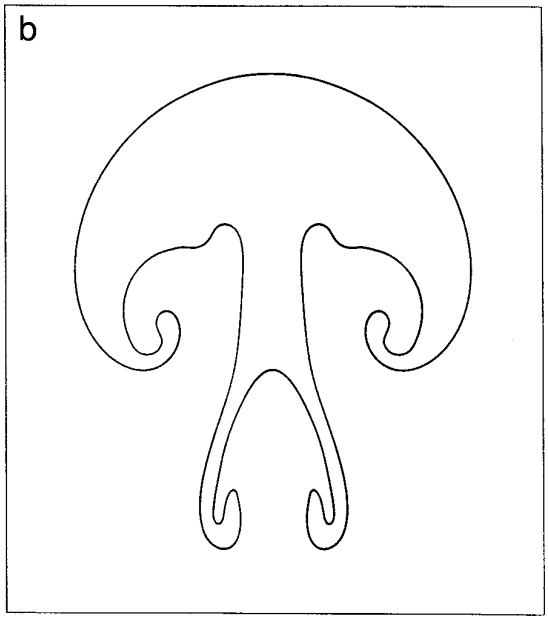


FIG. 5. (a) The same calculation as in Fig. 1 with area-preserving re-initialization. The total area is conserved up to 5 digits. (b) The same calculation as in Fig. 1. We overlap the solution at $t = 0.4$ obtained with area-preserving re-initialization on top of the solution obtained *without* area-preserving re-initialization. The two pictures are almost indistinguishable.

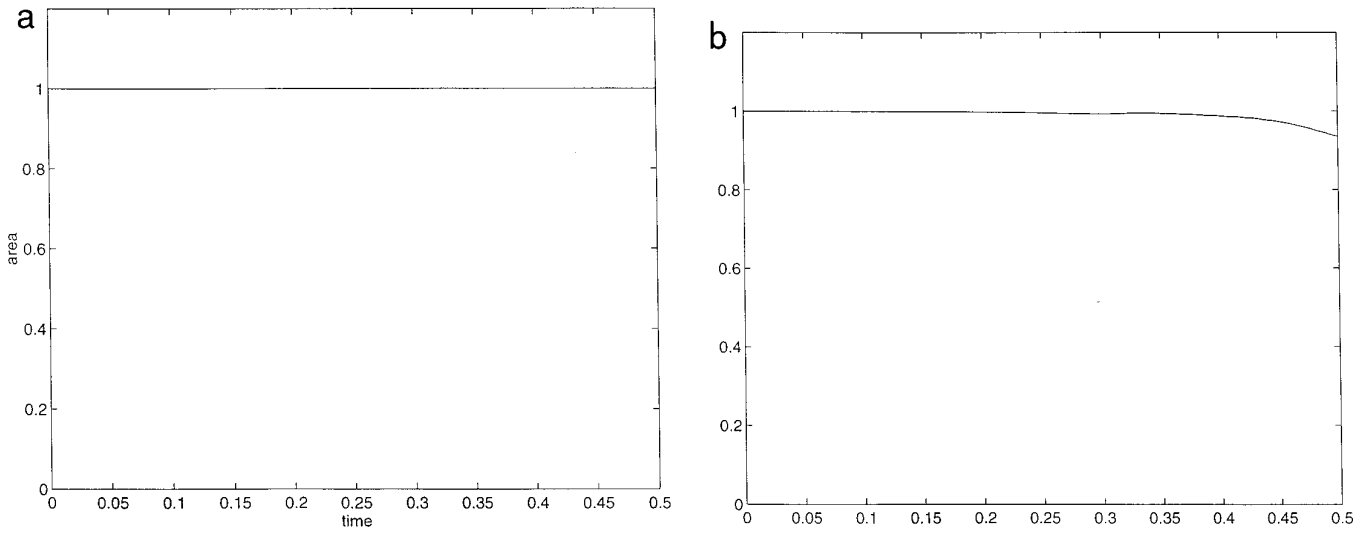


FIG. 6. (a) The same calculation as in Fig. 1. The total area is a function of time. This is *with* area-preserving re-initialization. The normalized area is a function of time, $N = 256$, del = 0.01, err = $1.d - 5$. (b) The same calculation as in Fig. 1. The total area is a function of time. This is *without* area-preserving re-initialization. The normalized area is a function of time, $N = 256$, del = 0.01, reinitialization.

presented in Fig. 2 with surface tension and the area-preserving re-initialization. Since the way we approximate the discrete delta function is relatively crude, in general we can expect only first-order convergence in the presence of surface tension; see, e.g., [7]. We clearly observe convergence of first-order accuracy.

Table I illustrates the convergence before the change of topology takes place at $t = 0.3$. Table II illustrates the convergence after the change of topology takes place at $t = 0.4$.

5. CONCLUSION

In summary, we have derived a level set formulation for incompressible, immiscible multi-fluid flow. A second-order projection method or vorticity-based method can be used to discretize the fluid equations in the level set formulation. The numerical method is purely Eulerian. There is no explicit tracking of the fluid interfaces. The fluid interface is recovered at the end of the calculation by locating the zero level set of a smooth function. The effects of discontinuous density, discontinuous viscosity, and surface tension can all be taken into account naturally. Using various re-initialization techniques, it is possible to keep the level set function as a distance function and to enforce mass conservation in time. As in [35], the front has a finite thickness of order $O(h)$ which does not change in time. The numerical diffusion introduced in the convection step for the smooth level set function does not diffuse the front. This would not have been the case if we used a capturing scheme of Godunov type to solve for the density equation directly, since the density has a contact discontinuity across the interface. From our computational experiences, we find that the second-order ENO scheme is more robust than a corresponding centered difference scheme with explicit numerical viscosity, especially when we have a large density ratio. On the other hand, the fourth-order centered difference scheme with a second-order numerical viscosity is more accurate and less diffusive. In both cases, the method is efficient and is capable of handling topological change in the fluid interfaces, such as merging and reconnection. It can be generalized to three-dimensional problems fairly easily.

TABLE I

Convergence Study at $t = 0.3$.

	Coarse 64 vs 256	Fine 128 vs 256
ρ	0.0810	0.0352
u	0.0620	0.0242
ϕ	0.00598	0.00183

TABLE II

Convergence Study at $t = 0.4$.

	Coarse 64 vs 256	Fine 128 vs 256
ρ	0.0892	0.0399
u	0.0614	0.0286
ϕ	0.00998	0.00310

In the future, we would like to generalize the method to study the interaction of three-dimensional fluid bubbles with a solid wall. In another direction, we would like to generalize the method to some non-Newtonian flows. In that case, a transverse level set function may be needed to compute the tangential directional derivatives which appear in the elastic tension force law for the interface. One example of such an application is to compute the nonlinear elastic tension force in Peskin's immersed boundary problem [28].

ACKNOWLEDGMENTS

Hou acknowledges partial support from NSF Grant DMS-9407030 and ONR Grant N00014-94-1-0310. Merriman and Osher acknowledge partial support by DARPA/ONR Grants N0014-92-J-1890, ARO DAA 203-91-G0162, and NSF DMS-91-03104.

REFERENCES

1. C. Anderson, *J. Comput. Phys.* **61,** 4 (1985).
2. G. R. Baker, D. I. Meiron, and S. A. Orzag, *J. Fluid Mech.* **123,** 477 (1982).
3. G. R. Baker and M. Shelley, *J. Fluid Mech.* **215,** 116 (1990).
4. J. T. Beale, T. Y. Hou, and J. Lowengrub, *SIAM J. Numer. Anal.,* to appear.
5. J. B. Bell and D. Marcus, *J. Comput. Phys.* **101,** 334 (1992).
6. J. B. Bell, P. Colella, and H. M. Glaz, *J. Comput. Phys.* **85,** 257 (1989).
7. R. P. Beyer and R. J. LeVeque, *SIAM J. Numer. Anal.* **29,** 332 (1992).
8. Y. C. Chang and H. O. Kreiss, preprint.
9. J. U. Brackbill, D. B. Kothe, and C. Zemach, *J. Comput. Phys.* **100,** 335 (1992).
10. S. Chandrasekhar, *Hydrodynamic and Hydromagnetic Stability* (Dover, New York, 1961).
11. A. J. Chorin, *Math. Comput.* **22,** 745 (1968).
12. K. A. Cliffe, S. J. Tavener, and A. A. Wheeler, *Int. J. Numer. Methods Fluids* **15,** 1243 (1992).
13. P. G. Drazin and W. H. Reid, *Hydrodynamic Stability* (Cambridge Univ. Press, Cambridge, (1981).
14. M. Gunzburger, *Finite-Element Methods for Viscous Incompressible Flows* (Academic Press, London, 1989).
15. F. H. Harlow and J. E. Welsh, *Phys. Fluids* **8,** 2181 (1965).
16. E. Harabetian, S. J. Osher, and C. W. Shu, *J. Comput. Phys.*, to appear.
17. A. Harten, B. Engquist, S. Osher, and S. Chakravarthy, *J. Comput. Phys.* **71,** 231 (1987).
18. T. Y. Hou, J. Lowengrub, and M. Shelley, *J. Comput. Phys.* **114,** 312 (1993).
19. T. Y. Hou and B. Wetton, *SIAM J. Numer. Anal.* **29,** 615 (1992).
20. T. Y. Hou, in preparation.
21. D. D. Joseph and J. C. Saut, *Theoret. Comput. Fluid Dyn.* **1,** 191 (1990).
22. R. M. Kerr, *J. Comput. Phys.* **76**(1), 48 (1988).
23. R. Krasny, *J. Comput. Phys.* **65,** 292 (1986).
24. H. O. Kreiss, *Tellus* **24**(3), 199 (1972).
25. B. Merriman, J. K. Bence, and S. J. Osher, *J. Comput. Phys.* **112,** 334 (1994).
26. S. A. Orszag and M. Israeli, *Annu. Rev. Fluid Mech.* **6,** 281 (1974).
27. S. Osher and J. A. Sethian, *J. Comput. Phys.* **79**(1), 12 (1988).
28. C. Peskin, *J. Comput. Phys.* **25,** 220 (1977).
29. D. Pullin and R. Grimshaw, *Phys. Fluids* **26,** 1731 (1983).
30. M. Shelley, *J. Fluid Mech.* **244,** 493 (1992).
31. C. W. Shu and S. Osher, *J. Comput. Phys.* **83,** 32 (1989).
32. M. Sussman, P. Smereka, and S. Osher, *J. Comput. Phys.* **114,** 146 (1994).
33. M. Sussman, E. Fatemi, P. Smereka, and S. Osher, ''A Level Set Approach for Computing Solutions to Incompressible Two-Phase Flow, II,'' in Proceedings, Sixth International Symposium on CFD, Lake Tahoe, Vol III, edited by M. Hafez, p. 1216.
34. G. Tryggvason, *J. Comput. Phys.* **75,** 253 (1988).
35. S. O. Unverdi and G. Tryggvason, *J. Comput. Phys.* **100,** 25 (1992).
36. J. S. Turner, *Buoyancy Effects in Fluids* (Cambridge Univ. Press, Cambridge, 1973).
37. H. K. Zhao, T. Chan, B. Merriman, and S. J. Osher, *J. Comput. Phys.,* submitted.