

A Robust, Fully Adaptive Hybrid Level-Set/Front-Tracking Method for Two-Phase Flows with an Accurate Surface Tension Computation

Hector D. Cenicerros¹, Alexandre M. Roma², Aristeu Silveira-Neto³ and
Millena M. Villar^{3,*}

¹ Department of Mathematics, University of California Santa Barbara, CA 93106,
USA.

² Departamento de Matemática Aplicada, Universidade de São Paulo, Caixa Postal
66281, CEP 05311-970, São Paulo-SP, Brasil.

³ Faculdade de Engenharia Mecânica, Universidade Federal de Uberlândia, CEP
38400-902, Uberlândia-MG, Brasil.

Received 5 May 2009; Accepted (in revised version) 14 October 2009

Available online 12 February 2010

Abstract. We present a variable time step, fully adaptive in space, hybrid method for the accurate simulation of incompressible two-phase flows in the presence of surface tension in two dimensions. The method is based on the hybrid level set/front-tracking approach proposed in [H. D. Cenicerros and A. M. Roma, *J. Comput. Phys.*, 205, 391-400, 2005]. Geometric, interfacial quantities are computed from front-tracking via the immersed-boundary setting while the signed distance (level set) function, which is evaluated fast and to machine precision, is used as a fluid indicator. The surface tension force is obtained by employing the mixed Eulerian/Lagrangian representation introduced in [S. Shin, S. I. Abdel-Khalik, V. Daru and D. Juric, *J. Comput. Phys.*, 203, 493-516, 2005] whose success for greatly reducing parasitic currents has been demonstrated. The use of our accurate fluid indicator together with effective Lagrangian marker control enhance this parasitic current reduction by several orders of magnitude. To resolve accurately and efficiently sharp gradients and salient flow features we employ dynamic, adaptive mesh refinements. This spatial adaption is used in concert with a dynamic control of the distribution of the Lagrangian nodes along the fluid interface and a variable time step, linearly implicit time integration scheme. We present numerical examples designed to test the capabilities and performance of the proposed approach as well as three applications: the long-time evolution of a fluid interface undergoing Rayleigh-Taylor instability, an example of bubble ascending dynamics, and a drop impacting on a free interface whose dynamics we compare with both existing numerical and experimental data.

*Corresponding author. *Email addresses:* hdc@math.ucsb.edu (H. D. Cenicerros), roma@ime.usp.br (A. M. Roma), aristeus@mecanica.ufu.br (A. Silveira-Neto), mmvillar@mecanica.ufu.br (M. M. Villar)

AMS subject classifications: 35R35, 65M50, 68U05, 74S20, 76D45, 76T99

Key words: Front-tracking, immersed boundary method, level set method, adaptive mesh refinements, semi-implicit methods, multilevel multigrid, closest point transform, semi-backward difference formula.

1 Introduction

Multi-phase flows are the source of numerous nonlinear processes of both scientific and technological relevance. These flows are characterized by a complex motion of fluid interfaces that separate masses of fluids with different material properties and the free boundaries can undergo significant deformations and topological transitions.

Due to the multi-component nature of the flow, the fluid interfaces are subjected to surface tension and this interfacial force plays a fundamental role in nearly all multi-phase flows of physical interest. Moreover, multiphase flows are typically multi-scale; the important phenomena of drop coalescence and break-up [1–3] as well as the generation of short capillary waves [4, 5] are just a few examples that exhibit the presence of multiple length scales. Therefore, an effective numerical method for the simulation of multiphase flows is required both to accurately represent the singularly supported interfacial forces and all the physically relevant flow quantities and to faithfully capture the disparate length scales. The method we propose here responds to these requirements in a computationally efficient and robust manner.

Numerical methods for computing multiphase flows can be broadly divided into two types: capturing and tracking. In capturing methods, such as the continuum surface force (CSF) model [6], the level set approach [7, 8], the phase field method [9–15], and the volume-of-fluid (VOF) method [16–19], the fluid interface is implicitly defined through a globally specified scalar function (the mass density, a signed distance function, an order parameter, or a volume fraction) which acts as a fluid indicator. These methods capture the interface motion on an Eulerian grid and handle automatically changes in interfacial topology. Front-tracking methods [20–25] on the other hand, use a separate grid to explicitly follow the interface motion and thus can achieve, in general, an accurate representation of geometric interfacial quantities.

In an attempt to overcome some of the inherent limitations of the aforementioned methodologies, there has been in recent years an increased attention to develop hybrid approaches [19, 26–31]. These hybrid strategies seek to exploit the best features of two different approaches by merging them into one method. The method presented here follows this philosophy. It originates from the hybrid level-set/front-tracking (LeFT) setting proposed in [27], from the fast and accurate fluid indicator developed in [30], and from the adaptive immersed boundary (IB) method first introduced by Roma, Peskin, and Berger [32] (for a recent, alternative adaptive version of the IB method see [33]). Moreover, the method combines these approaches with a mixed Lagrangian/Eulerian tension force representation proposed by Shin et al. [34].

Our approach employs the IB method to couple the explicitly tracked fluid interface with the Eulerian fluid domain and uses a signed distance (level-set) function as a fluid indicator. The tension forces and the interfacial geometric quantities are computed from the tracked interface. Thus, the method retains one of the advantages of front-tracking which is an accurate computation of interfacial quantities and at the same time benefits from a continuous, geometry-based fluid indicator, the level-set function. Moreover, this signed distance function is updated only locally, in a thin neighborhood of the interface, at optimal computational cost and is computed to machine precision for a piece-wise linear representation of the fluid interface [30].

One of the main contributions of the current work is to present a variable time step, fully adaptive in space, hybrid method to resolve accurately and efficiently interfacial forces and the disparate scales that are typical of many challenging, important two-phase flows. We achieve this by endowing the LeFT approach with adaptive mesh refinement (AMR), an efficient and robust semi-implicit time discretization, and a dynamic control of the distribution of the Lagrangian nodes in the interfacial grid. In addition, the combination of the accurate level set function with the hybrid force of Shin *et al.* [34] produces an unprecedented reduction of spurious currents and the increased resolution afforded by the AMR yields an accurate area conservation for long time integrations. We restrict the presentation of the method to two dimensions here but we envision that the full approach could be extended to three dimensions.

The rest of the paper is organized as follows. In Section 2, we review the hybrid LeFT setting. This is followed by a short section (Section 3) on the hybrid interfacial force [34] that we adopt here. The numerical method is described in Section 4 and its performance and characteristic features are highlighted in Section 5 with physically relevant numerical examples. Concluding remarks are presented in Section 6.

2 Hybrid formulation for two-phase incompressible flows

To describe the new approach, let us consider a single fluid interface separating two incompressible fluids of constant but possibly different densities and viscosities and in the presence of surface tension. The method is built from the hybrid formulation [27] in which the interface or immersed boundary is explicitly tracked and the signed distance, level set function ϕ is used as a fluid indicator. Representing the tracked interface as $\mathbf{X}(\alpha, t)$, where α is a Lagrangian parameter, we have

$$\rho(\phi)[\mathbf{u}_t + (\mathbf{u} \cdot \nabla) \mathbf{u}] = \nabla \cdot [\mu(\phi)(\nabla \mathbf{u} + \nabla \mathbf{u}^\dagger)] - \nabla p + \rho(\phi) \mathbf{g} + \mathbf{f}_\sigma, \quad (2.1)$$

$$\nabla \cdot \mathbf{u} = 0, \quad (2.2)$$

$$\mathbf{X}_t(\alpha, t) = \int_{\Omega} \mathbf{u}(\mathbf{x}) \delta(\mathbf{x} - \mathbf{X}(\alpha, t)) d\mathbf{x}, \quad (2.3)$$

where $\phi > 0$ for one of the fluids, $\phi < 0$ for the other, and $\phi = 0$ along the interface between the two phases. Here, \mathbf{u} , p , \mathbf{g} , and \mathbf{f}_σ are the velocity field, the pressure, the gravitational

acceleration, and the surface tension force respectively.

The level set function ϕ is not obtained by solving

$$\phi_t + (\mathbf{u} \cdot \nabla)\phi = 0$$

as it is done usually in level set methods. Instead, it is computed directly but locally using a fast algorithm for the signed distance [35]. With this approach ϕ is, up to machine precision, the signed distance to a piece-wise linear representation of the interface at all times. It never deteriorates and thus there is no need for re-initialization. We describe in more detail the computation of ϕ in Section 4.3.

Note that the δ -distribution in (2.3) is replaced by mollified function δ_h as done originally by Peskin's [20]. We take $\delta_h(\mathbf{x}) = d_h(x)d_h(y)$ with

$$d_h(\xi) = \begin{cases} 0.5 \left[1 + \cos\left(\frac{\pi}{h}\xi\right) \right] / h, & \text{for } |\xi| \leq h, \\ 0, & \text{for } |\xi| > h, \end{cases} \quad (2.4)$$

where h is a numerical parameter depending on the Eulerian grid mesh sizes Δx and Δy . For the computations reported here, $h = 2\Delta x$ and, to simplify the exposition of the numerical methodology, it is assumed that $\Delta x = \Delta y$. This choice for d_h provides good regularization properties around the interface and it is motivated by a set of compatibility properties described by Peskin [20]. Alternative discretizations can be found in [32, 36].

Given ϕ , the material quantities are obtained by the relations

$$\rho(\phi) = \rho_1 + (\rho_2 - \rho_1)H_h(\phi), \quad (2.5)$$

$$\mu(\phi) = \mu_1 + (\mu_2 - \mu_1)H_h(\phi), \quad (2.6)$$

where ρ_1, ρ_2 and μ_1, μ_2 are the constant densities and viscosities of the bulk phases, respectively, and $H_h(\phi)$ is a mollified Heaviside function defined by

$$H_h(\xi) = \begin{cases} 0, & \text{for } \xi < -h, \\ 0.5 \left[1 + \xi/h + \sin\left(\frac{\pi}{h}\xi\right) / \pi \right], & \text{for } |\xi| \leq h, \\ 1, & \text{for } \xi > h. \end{cases} \quad (2.7)$$

We note that for this Heaviside function we have

$$\frac{dH_h}{d\xi}(\xi) = d_h(\xi).$$

In [27, 30] the surface tension force \mathbf{f}_σ was computed solely from the Lagrangian representation of the interface. In this work, we opt instead for the hybrid, Lagrangian-Eulerian force form proposed in [34] which significantly reduces spurious currents due to a mismatch in the pressure gradient and the interfacial force. For completeness, we describe this hybrid force next.

3 Hybrid interfacial force formulation

We outline here the hybrid force formulation proposed in [34]. Let us begin with the Lagrangian approach for computing the interfacial force in IB-type methods,

$$\mathbf{f}_L(\mathbf{x}) = \int_{\Gamma} \frac{\partial}{\partial \alpha} (\sigma \hat{\mathbf{t}}) \delta(\mathbf{x} - \mathbf{X}(\alpha)) d\alpha, \quad (3.1)$$

where σ is the surface tension coefficient and $\hat{\mathbf{t}}$ is a tangent unit vector. For constant surface tension, one can write (3.1) as

$$\mathbf{f}_L(\mathbf{x}) = \sigma \int_{\Gamma} \kappa(\alpha) \hat{\mathbf{n}}(\alpha) \delta(\mathbf{x} - \mathbf{X}(\alpha)) |\mathbf{X}_{\alpha}(\alpha)| d\alpha, \quad (3.2)$$

where κ is the mean curvature. Note the extra factor $|\mathbf{X}_{\alpha}(\alpha)|$ in (3.2) which accounts for the fact that α is just a Lagrangian variable and not necessarily arclength (c.f. [7]). On the other hand, the Eulerian form of the constant tension force is generally written as

$$\mathbf{f}_E(\mathbf{x}) = \sigma \kappa_E(\mathbf{x}) \nabla H_h(\phi(\mathbf{x})), \quad (3.3)$$

where the Eulerian curvature κ_E is computed from ϕ by

$$\kappa_E = -\nabla \cdot \frac{\nabla \phi}{|\nabla \phi|}. \quad (3.4)$$

Following [34], we write

$$\mathbf{f}_L(\mathbf{x}) = \sigma \kappa_L(\mathbf{x}) \hat{\mathbf{n}}_L(\mathbf{x}), \quad (3.5)$$

where

$$\hat{\mathbf{n}}_L(\mathbf{x}) = \int_{\Gamma} \hat{\mathbf{n}}(\alpha) \delta(\mathbf{x} - \mathbf{X}(\alpha)) |\mathbf{X}_{\alpha}(\alpha)| d\alpha. \quad (3.6)$$

Thus, we have that

$$\kappa_L(\mathbf{x}) = \frac{1}{\sigma} \frac{\mathbf{f}_L(\mathbf{x}) \cdot \hat{\mathbf{n}}_L(\mathbf{x})}{\hat{\mathbf{n}}_L(\mathbf{x}) \cdot \hat{\mathbf{n}}_L(\mathbf{x})}. \quad (3.7)$$

We can now use the Eulerian form of the force (3.3) to define a hybrid force:

$$\mathbf{f}_H(\mathbf{x}) = \sigma \kappa_L(\mathbf{x}) \nabla H_h(\phi(\mathbf{x})) \quad (3.8)$$

and replace in (2.1) \mathbf{f}_{σ} by \mathbf{f}_H .

4 Numerical method

The numerical method that we introduce in the current work is hybrid in two different aspects and fully adaptive in space. It is hybrid first because we exploit qualities from both the IB Method (front-tracking) and from the level set formulation (front capturing)

and second because the singular interfacial force is computed, as explained previously, using a mixed Eulerian-Lagrangian formulation.

Full adaptivity is introduced in space both through dynamic control of the Lagrangian markers, which define the interface, and through Eulerian mesh refinement, which increases locally the resolution on the grid where the flow equations are solved. For the time discretization, we employ a robust, variable time-step second order semi-implicit temporal scheme. We highlight that, in contrast to what is done in the compressible flow context where "sub-cycling in time" is adopted (e.g., [37]), here all the refinement levels of the Eulerian mesh evolve in time sharing a common time step.

In the next few subsections, we detail our method specifically designed for a robust and efficient handling of incompressible two-phase flows.

4.1 Dynamic control of the Lagrangian parametrization

It is well-known that front-tracking methods applied to multiphase flows suffer from tension-induced numerical stiffness [38–40]. High order spatial derivatives in the interfacial forces and the excessive marker (particle) clustering, typical in those interface problems, lead to prohibitively small time steps for explicit methods.

An effective approach to relax this numerical stiffness is to use a suitably chosen tangential velocity for the interface markers to control their distribution along the interface (dynamic interfacial reparametrization [39, 41, 42]) and thus to prevent their excessive clustering. This idea is a key ingredient in the non-stiff IB-based method in [41] which was successfully employed to relax the time stepping in the study of the long-time dynamics of a sheared interface with uniform surface tension. Alternatively, selective point deletion and insertion has also been used in this context [43]. However, the stiffness due to nonuniform (elastic) tension is of a different nature [40] and we do not treat this case here.

In the present work, to overcome the aforementioned difficulties associated with Lagrangian tracking and the surface tension-induced stability constraint, following the success in [41], we employ a dynamically adaptive interface tracking in the form of marker equidistribution. Following the ideas introduced by Hou, Lowengrub, and Shelley [39] in the context of boundary integral methods, we use the freedom in selecting the tangential velocity of the interface markers to control their distribution at all times (kinematically the markers are only required to move with the normal velocity of the fluid). Thus (2.3) can be modified to

$$\mathbf{X}_t(\alpha, t) = \int_{\Omega} \mathbf{u}(\mathbf{x}) \delta_h(\mathbf{x} - \mathbf{X}(\alpha, t)) d\mathbf{x} + U_A(\alpha, t) \hat{\mathbf{t}}, \quad (4.1)$$

where $U_A(\alpha, t)$ is arbitrary and determines the parametrization used to describe the interface. For example, a U_A can be found to cluster interface markers in a controlled fashion in regions of high curvature [42] or to keep the markers equidistributed [39, 41]. If the

markers are equidistributed initially then it is easy to show that

$$U_A(\alpha, t) = -U_T(\alpha, t) + \int_0^\alpha (s_\alpha \kappa U_N - \langle s_\alpha \kappa U_N \rangle) d\alpha', \quad (4.2)$$

keeps them equidistributed at all times, where

$$U_T = \hat{\mathbf{t}} \cdot \int \mathbf{u}(\mathbf{x}) \delta_h(\mathbf{x} - \mathbf{X}(\alpha, t)) d\mathbf{x}, \quad U_N = \hat{\mathbf{n}} \cdot \int \mathbf{u}(\mathbf{x}) \delta_h(\mathbf{x} - \mathbf{X}(\alpha, t)) d\mathbf{x},$$

$s_\alpha = \sqrt{X_\alpha^2 + Y_\alpha^2}$ is the arc-length metric, κ is the mean curvature, and $\langle \cdot \rangle$ stands for the spatial mean over one spatial period.

4.2 Non-stiff temporal discretization

The incompressible Navier-Stokes equations are solved by a projection method. Its time discretization is based on a second order *semi-backward difference formula* (SBDF or Gear's method) and it is inspired by that employed in [13] with some notable differences. Namely, modifications to deal with variable density, adaptive time-stepping, and a different pressure update which includes solving the pressure Poisson's equation with homogeneous, Neumann boundary conditions.

We employ a linearly implicit approach to discretize the viscous term with variable viscosity as suggested in [13]. To illustrate the idea of this discretization we consider the variable (perhaps non-linear) coefficient diffusion equation $u_t = \nabla \cdot (\chi \nabla u)$, where $\chi > 0$. We rewrite that diffusion equation as

$$\frac{\partial u}{\partial t} = \lambda \nabla^2 u + f(u), \quad (4.3)$$

where $f(u) = \nabla \cdot (\chi \nabla u) - \lambda \nabla^2 u$ and λ is a positive constant to be conveniently chosen. We now perform the temporal discretization of (4.3) employing the second-order, two-step Gear's method for which we treat the first term of the right hand side implicitly and we extrapolate the second term in time (explicit treatment). If we allow for variable time steps, the resulting two-step scheme for (4.3) can be written then as

$$\frac{a_2 u^{n+1} + a_1 u^n + a_0 u^{n-1}}{\Delta t} = \lambda \nabla^2 u^{n+1} + b_1 f^n + b_0 f^{n-1}, \quad (4.4)$$

where $a_0 = \Delta t^2 / (\Delta t_0 \Delta t_1)$, $a_1 = -\Delta t_1 / \Delta t_0$, and $a_2 = (\Delta t_0 + 2\Delta t) / \Delta t_1$, $b_0 = -\Delta t / \Delta t_0$ and $b_1 = \Delta t_1 / \Delta t_0$, with $\Delta t = t^{n+1} - t^n$, $\Delta t_0 = t^n - t^{n-1}$, and $\Delta t_1 = \Delta t_0 + \Delta t$. Note that for fixed time steps the coefficients above assume their usual, constant values $a_0 = 1/2$, $a_1 = -2$, $a_2 = 3/2$, and $b_0 = -1$, $b_1 = 2$. To obtain u^1 , the numerical solution at time $t = t^1$, we simply employ forward Euler's method which is enough to achieve second-order in time, as the numerical validation and other results, included further ahead, show. This linearly implicit strategy for diffusion equations such as (4.3) can be found in [44–46]. We discuss later the selection of the parameter λ .

Based on the ideas above, we first rewrite the momentum equation

$$\rho[\mathbf{u}_t + (\mathbf{u} \cdot \nabla)\mathbf{u}] = \nabla \cdot [\mu(\nabla\mathbf{u} + \nabla\mathbf{u}^\dagger)] - \nabla p + \rho\mathbf{g} + \mathbf{f}_H \quad (4.5)$$

as

$$\mathbf{u}_t = \lambda \frac{\nabla^2 \mathbf{u}}{\rho} - \frac{\nabla p}{\rho} + \mathbf{g} + \frac{\mathbf{f}}{\rho}, \quad (4.6)$$

where

$$\mathbf{f} = \nabla \cdot \mu(\nabla\mathbf{u} + \nabla\mathbf{u}^\dagger) - \lambda \nabla^2 \mathbf{u} - \rho(\mathbf{u} \cdot \nabla)\mathbf{u} + \mathbf{f}_H.$$

Next, for the discretization in time, we employ the extrapolated Gear's method with variable time stepping (4.4), treating implicitly all the terms in the right hand side of (4.6), except for \mathbf{f} which we extrapolate in time. The result, along with the incompressibility constraint, express the coupling between the pressure and the velocity which can be translated into a set of equations given by

$$\frac{a_2 \mathbf{u}^{n+1} + a_1 \mathbf{u}^n + a_0 \mathbf{u}^{n-1}}{\Delta t} = \lambda \frac{\nabla^2 \mathbf{u}^{n+1}}{\rho^{n+1}} - \frac{\nabla p^{n+1}}{\rho^{n+1}} + \mathbf{g} + \frac{b_1 \mathbf{f}^n + b_0 \mathbf{f}^{n-1}}{\rho^{n+1}}, \quad (4.7)$$

$$\nabla \cdot \mathbf{u}^{n+1} = 0. \quad (4.8)$$

To handle the pressure-velocity coupling (4.7) and (4.8), we propose the projection method

$$\frac{a_2 \mathbf{u}^* + a_1 \mathbf{u}^n + a_0 \mathbf{u}^{n-1}}{\Delta t} = \lambda \frac{\nabla^2 \mathbf{u}^*}{\rho^{n+1,0}} - \frac{\nabla p^{n+1,0}}{\rho^{n+1,0}} + \mathbf{g} + \frac{b_1 \mathbf{f}^n + b_0 \mathbf{f}^{n-1}}{\rho^{n+1,0}}, \quad (4.9)$$

$$\mathbf{u}^* = \mathbf{u}^{n+1} + \frac{\Delta t}{a_2} \frac{\nabla q}{\rho^{n+1,0}}, \quad (4.10)$$

$$\nabla \cdot \mathbf{u}^{n+1} = 0, \quad (4.11)$$

where $p^{n+1,0}$ and $\rho^{n+1,0}$ are given approximations of the pressure and of the density at $t = t^{n+1}$, respectively. Once the *provisional velocity* \mathbf{u}^* is computed from (4.9) imposing $\mathbf{u}^* \doteq \mathbf{u}^{n+1}$ on the boundaries, it is *projected* onto the space of divergence-free vector fields. The projection step is accomplished by solving the Poisson equation for q defined by (4.10) and (4.11) along with homogeneous Neumann boundary conditions $\partial q / \partial \mathbf{n} = 0$, where \mathbf{n} is the unit normal exterior to the domain (see [47, 48] for a discussion of the numerical boundary conditions for the Poisson equation during the projection step).

After solving the Poisson equation defined by (4.10) and (4.11), the new velocity is then given by

$$\mathbf{u}^{n+1} = \mathbf{u}^* - \frac{\Delta t}{a_2} \frac{\nabla q}{\rho^{n+1,0}}. \quad (4.12)$$

Substituting (4.10) into (4.9) and subtracting the result from (4.7), after rearranging the terms, we have an expression for updating the pressure given by

$$\begin{aligned} \nabla p^{n+1} = & \nabla p^{n+1,0} + \nabla q - \lambda \frac{\Delta t}{a_2} \nabla^2 \frac{\nabla q}{\rho^{n+1,0}} \\ & + (\rho^{n+1} - \rho^{n+1,0}) \left(\mathbf{g} - \frac{a_2 \mathbf{u}^{n+1} + a_1 \mathbf{u}^n + a_0 \mathbf{u}^{n-1}}{\Delta t} \right). \end{aligned} \quad (4.13)$$

Next, we discuss the selections of $p^{n+1,0}$ and $\rho^{n+1,0}$. Like in many other projection methods, we simply choose for the pressure the first order approximation in time $p^{n+1,0} \doteq p^n$. For the selection of $\rho^{n+1,0}$, there is a variety of second order possible approximations in time, e.g. based on the extrapolation of the density $\rho^{n+1,0} = b_1 \rho^n + b_0 \rho^{n-1}$ or on the extrapolation of the interface location via the fluid indicator ϕ or the interface position variable \mathbf{X} . However, as clearly indicated by the numerical experiments reported later, the first order approximation in time, $\rho^{n+1,0} \doteq \rho^n = \rho(\phi^n)$, suffices to furnish a second order approximation for the velocity. Considering the aforementioned approximations for the pressure and for the density and neglecting the $\mathcal{O}(\Delta t)$ terms in (4.13), we obtain the simpler expression for updating the pressure

$$p^{n+1} = p^n + q, \quad (4.14)$$

from which we see that q may be interpreted as a *pressure increment*. The pressure obtained in this form in general will not be second order accurate (except under very special circumstances such as periodic boundary conditions on uniform grids).

The SBDF-based, pressure-increment projection method defined by (4.9)-(4.12) with the updating for the pressure given by (4.14) has some distinct features when compared to other related second order projection methods, especially for multiphase flows. Most notably, the linearly implicit treatment of the viscous term results in an effective decoupling of the velocity components. This relaxes the viscous term-induced stability constraint for small to moderate viscosity jumps and tied with the multi-step strategy reduces one velocity update to the solution of two linear, constant coefficient parabolic problems along with a linear elliptic problem for the pressure. With the intent of providing some background for the proposed projection method we review briefly some related approaches.

The idea of a decoupled implicit discretization can already be found in the early second order, one-step method of Goda [49] which is based on the alternating direction implicit (ADI) approach. In contrast to our proposed SBDF-based method, Goda's scheme requires in 2D the solution of seven partial differential equations (three for each velocity component plus one for the Poisson equation).

Several multi-step projection methods have also been proposed [50–52]. The SBDF approach can be found in the methods considered by Karniadakis, Israeli, and Orszag [50] and by Azaiez [52]. However, these methods were only considered for constant viscosity Stokes flows with a fixed time-step and without extrapolation of the forcing term. A closer relative to the method proposed here is that introduced in [13] as remarked above.

Finally, we note that due to our geometrical approach to the computation of the fluid indicator, a partial differential equation has been eliminated from the traditional multi-phase flow setting. As a result, only three partial differential equation solves are required in two dimensions even in the presence of variable material properties. To our knowledge, this is the only multiphase flow method that achieves such an economy.

To finish the presentation of the discretization in time, we turn our attention now to the evolution of the interface in time. We apply to (4.1) once more the automatic time stepping, extrapolated Gear's method

$$\frac{a_2 \mathbf{X}^{n+1} + a_1 \mathbf{X}^n + a_0 \mathbf{X}^{n-1}}{\Delta t} = b_1 \mathbf{U}^n + b_0 \mathbf{U}^{n-1}, \quad (4.15)$$

with \mathbf{U}^k , $k = n, n-1$, the modified interface velocity

$$\mathbf{U}^k \doteq \int \mathbf{u}^k \delta_h(\mathbf{x} - \mathbf{X}^k) d\mathbf{x} + U_A^k \hat{\mathbf{t}}^k, \quad (4.16)$$

where U_A is the tangential velocity (4.2) selected to dynamically control the fluid interface parametrization (to keep Lagrangian markers equidistributed), and $\hat{\mathbf{t}}$ is the tangent vector

$$\hat{\mathbf{t}} = \frac{\partial \mathbf{X} / \partial \alpha}{\|\partial \mathbf{X} / \partial \alpha\|}. \quad (4.17)$$

To complete a typical time step, we update the fluid indicator $\phi^{n+1} \doteq \phi(\mathbf{X}^{n+1})$ employing an optimal procedure based on an algorithm stemming from Computational Geometry. This is followed by an update of the material properties

$$\rho^{n+1} = \rho_1 + (\rho_2 - \rho_1) H_h(\phi^{n+1}), \quad \mu^{n+1} = \mu_1 + (\mu_2 - \mu_1) H_h(\phi^{n+1}),$$

where H_h is given by (2.7). Next, we explain the optimal procedure used to update the fluid indicator.

4.3 Geometry-based fluid indicator and its fast computation

Several approaches that employ the interface position at a given time have been proposed to update material properties, such as the viscosity and density, which are constant in the bulk phases [21, 53, 54]. Although clever and functional for many situations, those approaches were not robust or computationally efficient enough to handle more general flows or interface geometries.

Since density and viscosity are constants in each of the bulk phases and since the interface motion is limited by the CFL condition to less than a mesh size in each time step, it is computationally appealing to update these quantities only in a vicinity of the interface. In [30], a robust and computationally efficient approach for achieving this was introduced. The basic idea is to make use of a fast algorithm from Computational Geometry [35] to compute the signed distance to the interface. With this approach, the material

properties are accurately updated and the computational work of this task is confined to a narrow band around the fluid-fluid interfaces. Thus, the procedure has optimal cost (linear in the number of Lagrangian markers) and is easy to implement.

To explain that approach briefly, let the interface Γ be represented by a non self-intersecting, piecewise linear curve. We define ϕ as the *signed distance function* only in T_γ , a narrow band centered at Γ and of width $2\gamma > 0$. Outside this band, ϕ is continuously defined to be $\pm\gamma$, that is

$$\phi(\mathbf{x}) = \begin{cases} -\gamma, & \text{if } d(\mathbf{x}) < -\gamma, \\ d(\mathbf{x}), & \text{if } |d(\mathbf{x})| \leq +\gamma, \\ +\gamma, & \text{if } d(\mathbf{x}) > +\gamma, \end{cases} \quad (4.18)$$

where the signed distance function $d(\mathbf{x})$ is the Euclidean distance from the given point \mathbf{x} to the fluid interface Γ for which a sign is chosen according to the direction of the normal to the interface.

To keep ϕ as a local signed distance function (up to machine precision) around Γ at all times, we employ a fast algorithm to compute the *Closest Point Transform* (CPT) due to Mauch [35]. The CPT finds the closest point on Γ and determines the Euclidean distance to Γ for all the Eulerian grid points within a specified distance $\epsilon > 0$ from Γ .

Since Γ is a piecewise linear curve, the closest point ζ on Γ to a given point \mathbf{x} , either lies on one of the “edges” (links) or at one of the “vertices” (interface points). If ζ lies on a edge, the vector from ζ to \mathbf{x} is orthogonal to the edge. Thus, the set of closest points to a given edge must lie within a strip defined by the edge itself and by its normal vector. In particular, the set of closest points within a specified distance $\epsilon > 0$ to a given edge is given by the polygon P seen in Fig. 1.

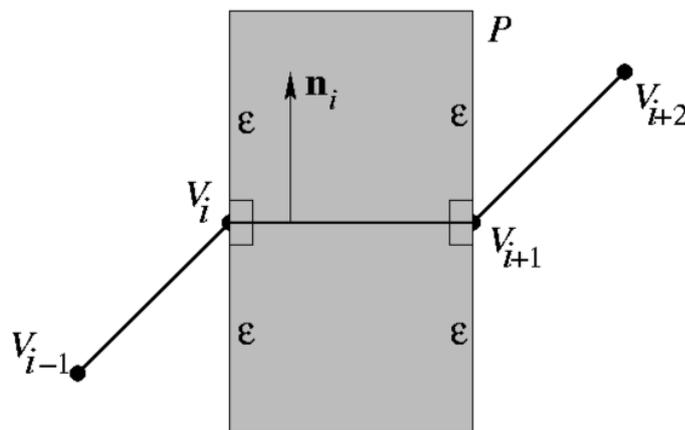


Figure 1: Polygon P : set of points within distance ϵ to Γ for which the closest point on Γ lies on the edge $V_i \rightarrow V_{i+1}$.

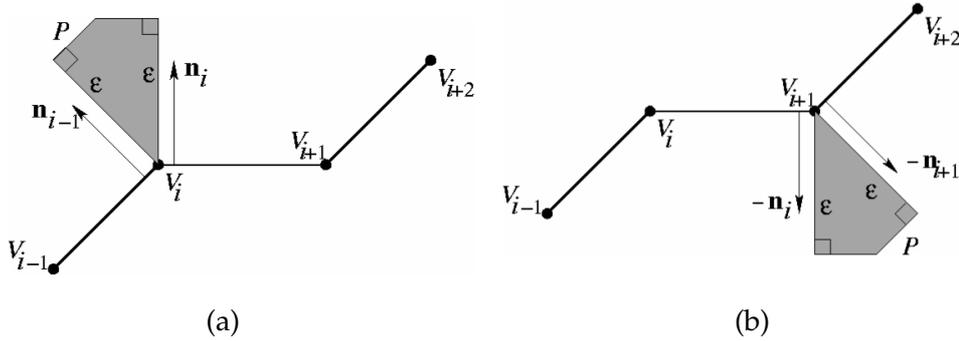


Figure 2: Polygons containing the set of points within distance ϵ to Γ for which the closest point on Γ lies on either the vertex V_i or V_{i+1} [(a) or (b), respectively].

When ζ lies on a vertex, the vector from ζ to \mathbf{x} must lie between the normal vectors to the two adjacent edges at the vertex. Thus, the closest points to a vertex must lie in a wedge. If the outside (inside) angle formed by the two adjacent edges is less than π then there are no points of positive (negative) distance from the vertex. The set of closest points, within a specified distance $\epsilon > 0$, is *contained* by polygons P like those shown in Fig. 2. The vertex opposite to V_i in Fig. 2(a) is determined by taking the intersection between the lines which are perpendicular to the edges intersecting at V_i (similarly, at V_{i+1} in Fig. 2(b)). Note that, given ϵ such that $0 < \gamma < \epsilon$, the union of all polygons constructed as above will contain the band T_γ .

From the previous considerations, a simple procedure for updating ϕ in a neighborhood of T_γ can be devised: In a first pass, after updating the location of Γ , we tag all Eulerian grid points \mathbf{x}_{ij} belonging to the union of all strips and wedges (by setting $(\mathbf{x}_{ij}) = +\infty$). Then, in a second pass, we compute the CPT for all such points \mathbf{x}_{ij} , and simultaneously apply the cutoff given by (4.18).

This procedure is executed at every time step, after the interface position has been updated. At $t=0$, we must have a fluid indicator function ϕ satisfying (4.18) so that only a local correction is needed subsequently. This initial ϕ can be obtained by computing at $t=0$ the signed distance at every point in the Eulerian grid and then by applying the cutoff to it. We emphasize that this procedure has linear computational complexity in the number of interfacial markers and, since it takes into account the full extent of the interface to compute the signed distance, it correctly handles situations when two disparate interface segments lie too close to each other, such as in the near merging or in the near self-intersection cases.

Finally, we point out the hybrid aspect of our approach: The fluid indicator, efficiently computed from tracked Lagrangian markers, is a continuous function whose *zero level set* is given, clearly, by the fluid interface. Details on the computation, including an algorithm and an efficient procedure to determine whether or not an Eulerian grid point belongs to each polygon P , can be found in [30].

4.4 Adaptive spatial discretization

The presence of a fluid interface acting with a singular force leads to large gradients localized in a vicinity of the free boundary. Moreover, surface tension can induce the production of focused centers of vorticity and to other small scale phenomena whose adequate capturing often demands computationally prohibitive fine resolutions to uniform grid approaches. This problem can be overcome with a judicious use of a *local mesh refinement technique*.

Following the adaptive version of the IB method introduced by Roma *et al.* [32], we employ here the hierarchical grid structure proposed by Berger and Colella [55]. In our fully adaptive computational scheme, regions of the flow bearing special interest are covered by block-structured grids, defined as a hierarchical sequence of nested, progressively finer levels (*composite grids*). Each level is formed by a set of disjoint rectangular grids and the refinement ratio between two successive refinement levels are constant and equal to two. Ghost cells are employed around each grid, for all the levels, and underneath fine grid patches to formally prevent the finite difference operators from being redefined at grid borders and at interior regions which are covered by finer levels. Values defined in these cells are obtained from interpolation schemes, usually with second or third order accuracy, and not from solving the equations of the problem. The description of composite grids is given in greater details in [55].

Composite grid generation depends on the *flagging step*, i.e., on determining first the cells whose collection gives the region where refinement is to be applied. Here, we mark for refinement a neighborhood of all interface points (*uniform covering of the interface*). Once the collection of flagged cells is obtained, grids in each level are generated by applying the algorithm for point clustering due to Berger and Rigoutsos [56]. Regridding is performed as often as regions of high vorticity or interface points are “about to leave” the finest level.

We use a staggered (composite) grid, i.e., pressure and other scalar variables are placed at the centers of the computational cells, the first component of vector variables are placed at the middle of the vertical edges, and the second component are placed at the middle of the horizontal ones. The discretizations of the Laplacian, gradient, and divergence differential operators are performed by standard compact second-order difference stencils which are centered respecting the locations where variables are placed. Variables are interpolated to the middle of cell edges or to cell corners as required by the difference stencils by simple averaging their values from the nearest computational cells (e.g. to compute the non-linear term by the centered, standard second-order discretization).

We employ multilevel-multigrid methods to solve both for the provisional vector field, \mathbf{u}^* in (4.9), and for the pressure increment q , computed from (4.10) and (4.11). We employ a V-cycle schedule from the finest to the base level of the composite grid, switching to a W-cycle schedule from the base level to the coarsest multigrid level underneath, with one relaxation on each level, upwards and downwards. Detailed descriptions of the Eulerian composite grid and of the basic ideas of the multilevel-multigrid methodology

employed can be found in [32,57].

After generating new grid hierarchy, pressure, velocity components, and forces, for all needed time steps, are either interpolated from the old grid hierarchy, where old and new grid hierarchies do not overlap, or directly copied from it, otherwise. Before taking the next time step, we perform a projection to ensure that the discrete incompressibility constraint holds on the new grid hierarchy.

We note that, although in gas dynamics problems refinement in time comes naturally along with refinement in space, this is not the approach we employ in the current work. All grids, in all levels, evolve together in time, with the time step of the finest level. In the incompressible case, there is not a finite limit to the speed at which disturbances can propagate in the flow. Since each part of the incompressible flow influences all other parts instantaneously, it is not clear how different time steps could be used on the different grids.

4.5 Pressure-increment projection method

We end this section with a summary of a typical time step. For simplicity, we drop the i, j spatial indices. From computed values of p^n , the pressure at time $t = t^n$, of \mathbf{X}^k and of \mathbf{u}^k , $k = n, n-1$, the interface position and the velocity field at times $t = t^n$ and $t = t^{n-1}$ respectively, we update the pressure, the interface position, and the velocity field, by solving

$$\frac{a_2 \mathbf{u}^* + a_1 \mathbf{u}^n + a_0 \mathbf{u}^{n-1}}{\Delta t} = \lambda \frac{L \mathbf{u}^*}{\rho^n} - \frac{G p^n}{\rho^n} + \mathbf{g} + \frac{b_1 \mathbf{f}^n + b_0 \mathbf{f}^{n-1}}{\rho^n}, \quad (4.19)$$

$$\mathbf{u}^* = \mathbf{u}^{n+1} + \frac{\Delta t}{a_2} \frac{G q}{\rho^n}, \quad (4.20)$$

$$D \cdot \mathbf{u}^{n+1} = 0, \quad (4.21)$$

$$p^{n+1} = p^n + q, \quad (4.22)$$

$$\frac{a_2 \mathbf{X}^{n+1} + a_1 \mathbf{X}^n + a_0 \mathbf{X}^{n-1}}{\Delta t} = b_1 \mathbf{U}^n + b_0 \mathbf{U}^{n-1}, \quad (4.23)$$

$$\phi^{n+1} \doteq \phi(\mathbf{X}^{n+1}), \quad (4.24)$$

where L , G , and D are the centered second order Laplacian (at the cell edges), gradient (at the cell edges), and divergence (at the cell centers) difference operators, respectively, and the parameters a_i , $i = 2, 1, 0$, and b_j , $j = 1, 0$, are as in (4.4). The forcing term \mathbf{f} in (4.19) is a second order discretization of its continuous counterpart appearing in (4.6). The modified interface velocity \mathbf{U}^k , $k = n, n-1$, is given by

$$\mathbf{U}^k = h^2 \sum_{\mathbf{x}} \mathbf{u}^k \delta_h(\mathbf{x} - \mathbf{X}^k) + U_A^k \hat{\mathbf{t}}^k, \quad (4.25)$$

and the geometry-based fluid indicator at time $t = t^{n+1}$, ϕ^{n+1} , is obtained by application of the CPT algorithm (Section 4.3). The updated material properties are given by $\rho^{n+1} = \rho_1 + (\rho_2 - \rho_1)H_\epsilon(\phi^{n+1})$ and $\mu^{n+1} = \mu_1 + (\mu_2 - \mu_1)H_\epsilon(\phi^{n+1})$, where H_ϵ is given by (2.7).

It is well known [54], that the IB setting produces small amplitude mesh-scale oscillations in the interface position variables. When derivatives are computed from the interface position to obtain geometric quantities and tension forces, these oscillations are amplified by numerical differentiation and if left unattended can quickly lead to numerical instability. To eliminate the growth of these small amplitude mesh-scale oscillations we apply the fourth order filter [58]

$$\mathbf{X}_k \leftarrow \frac{1}{16}(-\mathbf{X}_{k-2} + 4\mathbf{X}_{k-1} + 10\mathbf{X}_k + 4\mathbf{X}_{k+1} - \mathbf{X}_{k+2}). \quad (4.26)$$

The filter is applied every ten time steps to the fluid interface markers. As noted in [59], the fourth order filter (4.26) effectively eliminates mesh-scale oscillations without compromising the accuracy of the solution. In contrast, the second order 1-2-1 filter, which is frequently used in the literature, introduces excessive numerical diffusion.

4.6 Time-stepping strategy

The time step is constrained by advection (CFL), gravity, surface tension, and interface motion. The SBDF-based projection method employed here removes the viscous constraint as we further explain below. By similar arguments to those presented by Kang, Fedkiw, and Liu [60] we write

$$\frac{\Delta t}{2} \left(v_a + \sqrt{v_a^2 + (2v_g)^2 + (2v_s)^2} \right) < 1, \quad (4.27)$$

where the constraints from advection, gravity, and surface tension are given by

$$v_a = \frac{|u|_{\max}}{\Delta x} + \frac{|v|_{\max}}{\Delta y}, \quad v_g = \sqrt{\frac{g}{\Delta y}}, \quad \text{and} \quad v_s = \sqrt{\frac{\sigma|\kappa|_{\max}}{\min\{\Delta x, \Delta y\}^2 \rho_{\min}}}.$$

From the interface motion, we have the constraint

$$\Delta t v_k = \Delta t \frac{\max\{|U|, |V|\}}{\min\{\Delta x, \Delta y\}} \leq c_k < 1, \quad (4.28)$$

which prevents Lagrangian points from moving more than one mesh cell per time step. Observe that (4.27) will be satisfied if we select $\Delta t v_a \leq c_a$, $\Delta t v_g \leq c_g$, and $\Delta t v_s \leq c_s$ with constants such that $c_a + c_g + c_s < 1$. This follows immediately from the fact that

$$\sqrt{v_a^2 + (2v_g)^2 + (2v_s)^2} < v_a + 2v_g + 2v_s.$$

From the observations above, we take the time step given by

$$\Delta t = \min \left\{ c_a v_a^{-1}, c_g v_g^{-1}, c_s v_s^{-1}, c_k v_k^{-1} \right\}, \quad (4.29)$$

where $c_a + c_g + c_s < 1$.

Due to the equidistribution of Lagrangian points, the time step constraint induced by *uniform* surface tension gets milder because the usually excessive Lagrangian clustering is prevented [41]. Only when surface tension is extremely large (as it is the case in one of the tests presented here, large Laplace number flows), the surface tension-induced constraint becomes dominant and Δt could be as small as $\mathcal{O}(\max\{\Delta x, \Delta y\}^2)$. On the other hand, under mild to moderate tension forces the time step is $\mathcal{O}(\max\{\Delta x, \Delta y\})$ which also keeps the errors in space and in time of the same order. Aided by extensive numerical experimentation, we select conservative values of the constants c in (4.29) so that we comply with the stability constraints even when Δt is kept fixed for a few (typically 10) time steps at time (to avoid the degradation of the order in time of the scheme).

We have observed in our computations that the proposed SBDF-based projection method, with the decoupled, linearly implicit treatment of the viscous term is effective in relaxing the stability constraint induced by the viscous dissipation for small to moderate viscosity differences. Assuming, as we may here, that the fluid interface is diffuse so that the velocity remains smooth across it (albeit with large gradients) and using incompressibility we have that $v_{xy} = -u_{xx}$ and $u_{xy} = -v_{yy}$. Employing these identities and $u_x = -v_y$ we can write the viscous term $(V_1, V_2) = \nabla \cdot [\mu(\nabla \mathbf{u} + \nabla \mathbf{u}^\dagger)]$ in component form as

$$V_1 = \nabla \cdot (\mu \nabla u) - \mu_x v_y + \mu_y v_x, \quad (4.30)$$

$$V_2 = \nabla \cdot (\mu \nabla v) - \mu_y u_x + \mu_x u_y. \quad (4.31)$$

As remarked earlier, the linearly implicit approach can effectively eliminate the quadratic stability constraint that the terms $\nabla \cdot (\mu \nabla u)$ and $\nabla \cdot (\mu \nabla v)$ induce in an explicit method [44]. With an explicit treatment of the first order derivative terms in (4.30)-(4.31), as in effect is the case for the proposed the time-discretization, one would expect a CFL condition of the form $\Delta t \leq C |\nabla \mu|^{-1} h$. This is a mild constraint if $|\nabla \mu|$ is small. However, in the interfacial region of a two-phase flow $|\nabla \mu| = \mathcal{O}(|\mu_1 - \mu_2|/h)$ which would imply a quadratic stability constraint. But the level curve $\phi = 0$ is also a level curve of μ and hence the first derivative terms in (4.30)-(4.31) evaluated at $\phi = 0$ are actually tangential derivatives of v and u , respectively. That is, in the interfacial region we can write

$$V_1 = \nabla \cdot (\mu \nabla u) + |\nabla \mu| \mathbf{t} \cdot \nabla v, \quad (4.32)$$

$$V_2 = \nabla \cdot (\mu \nabla v) - |\nabla \mu| \mathbf{t} \cdot \nabla u, \quad (4.33)$$

where

$$\mathbf{t} = \frac{(\mu_y, -\mu_x)}{|\nabla \mu|}, \quad (4.34)$$

defines a tangent vector to the level curve $\phi = 0$. In contrast to the normal derivatives, $\mathbf{t} \cdot \nabla u$ and $\mathbf{t} \cdot \nabla v$ are smooth and small so long as the curvature is not too high. This much smaller tangential derivatives might explain why we observe a milder stability constraint for small to moderate viscosity differences.

Finally, we note that the parameter λ in (4.19) is chosen as $\lambda = 2 \|\mu\|_\infty$. For this particular choice we were guided by both our own numerical experimentation and by a recent analysis of Xu and Tang [61] of a related, linear semi-implicit method for continuum epitaxial growth models.

5 Numerical results

We present in this section results from numerical experiments designed to test the capabilities of the proposed adaptive, hybrid method.

First, we perform an accuracy test in the presence of a fluid interface in a two-phase flow with different material properties. We then look at the ability of the method to reduce spurious currents. We do this with two tests: i) we depart from rest with a circular interface with and without equidistributed Lagrangian markers and ii) we depart from rest with an initially elliptical interface and we measure the spurious currents after the steady state has been reached. The former is for assessing the effects of the hybrid force formulation while the latter is meant to evaluate the combined effects of both the dynamic Lagrangian equidistribution and the hybrid force formulation when there is an initial transient phase and a non-circular interface.

We continue probing the capability of the proposed adaptive tracking method with a popular test for multiphase flow methods; a straining, vortex flow affecting an initially circular interface [18]. The given flow field produces substantial interfacial stretching and the formation of a thin, high curvature tail. We monitor closely the loss of mass (area) during the interfacial deformation. The area is computed in this and in all the numerical examples from the interface position variable \mathbf{X} by applying Stokes theorem and approximating the resulting integral (with periodic integrand) with the trapezoidal rule. For consistency with the proposed methodology, the given flow field is interpolated on the interface via Peskin's delta interpolation and evolved according to (4.23) and (4.25). Both dynamic Lagrangian equidistribution and several levels of refinement are employed.

Following the accuracy and performance tests above, we present a simulation of Rayleigh-Taylor instability. The fluid interface in an unstably stratified flow develops the characteristic mushroom-like structure of this fundamental instability. However, the proposed adaptive, hybrid method is able to capture with unprecedented detail and accuracy a much more complex evolution which includes near pinch-off events and the formation of extremely thin filaments. The high-resolution computation presented here indicates that the apparent topological singularity that occurs in the inviscid case [62] is prevented in the presence of viscous effects.

The rising motion of a single buoyant bubble is then simulated, providing an excellent

test for more quantitative comparisons involving two-phase flows. The last numerical experiment is a simulation of an impinging drop onto a free surface. We compare our numerical results for this simulation with both experimental data reported in [63] and recent numerical results [31].

In the numerical simulations that follow, we refer to the composite adaptive meshes using the notation $m \times nLp$ which stands for a mesh with an $m \times n$ coarsest (base) level and p levels of refinement (total, from the coarsest to the finest level). We also take the number of interface (Lagrangian) markers N_b to be $m2^p$ for a given $m \times nLp$ adaptive Eulerian mesh, unless stated otherwise. Dynamically, with equidistribution of Lagrangian markers, N_b is doubled whenever the average Lagrangian spacing, Δs , becomes larger than $h/2$. This would be the case for example of an interface that becomes highly stretched.

5.1 Accuracy test

We begin with a test to measure the numerical rate of convergence of the proposed method in the case of a two-phase flow with different material properties in an unstably stratified setting. We will look at both a region surrounding the fluid interface and a region far from it. Initially, the unstably stratified two-phase flow is at rest and the interface is given by

$$\mathbf{X}(0) = \left(\alpha, 0.025 \cos(2\pi\alpha) \right), \quad 0 \leq \alpha \leq 1. \quad (5.1)$$

At the boundaries of the rectangular domain $\Omega = [0,1] \times [-1.5,1.5]$, the velocity satisfies the homogeneous Dirichlet condition at the north and at the south borders, and periodic condition in the horizontal direction. The acceleration of gravity is taken to be $g = 10^3 \text{ cm/s}^2$. The heavier fluid, at the top, has mass density $\rho_1 = 1.2 \text{ g/cm}^3$. The fluid at the bottom has mass density $\rho_2 = 1.1 \text{ g/cm}^3$. Thus, the Atwood number A defined as

$$A = \frac{\rho_1 - \rho_2}{\rho_1 + \rho_2} \quad (5.2)$$

is equal to 0.1. Both fluids have the same viscosity $10^{-4} \text{ Pa}\cdot\text{s}$ and the surface tension coefficient is $\tau = 0.005 \text{ dyn/cm}^2$.

The accuracy of the method is investigated separately both on a neighborhood of the fluid interface, $\Omega_1 = [0,1] \times [-0.3,0.3]$, and on a region $\Omega_2 = [0,1] \times [-1.3,-0.6] \cup [0.6,1.3]$, placed in the interior of the domain where the bulk flow is relatively smooth, away from the fluid interface.

We compute the numerical solution up to $t = 0.2\text{s}$ in a sequence of progressively finer composite grids with two refinement levels. Both regions Ω_1 and Ω_2 are located away from coarse-fine level (grid) interfaces to avoid interpolation errors at those locations. A typical composite grid employed is shown in the Fig. 3.

For each run, the spatial grid is obtained by doubling the number of computational cells in all the refinement levels of the composite grid used in the previous case, keeping the region of the domain covered by the second level fixed. The composite grids range

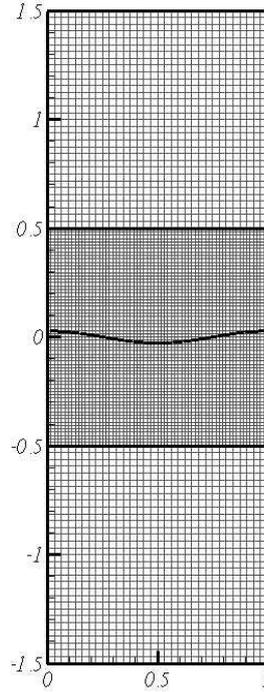


Figure 3: Typical adaptive mesh employed in the accuracy test.

from $64 \times 192L2$ to $512 \times 1536L2$, where in the notation $m \times n L2$ the symbol “L2” means that one level of refinement is used over a base level which has m by n computational cells (a total of two refinement levels). On each of the subdomains Ω_1 and Ω_2 , we compute the difference between the numerical approximations obtained in two successive runs. We do this by first averaging the finer composite grid solution at the staggered-grid velocity locations of the next coarser composite grid, and then we compute the difference between that average and the computed coarser grid solution. These differences are employed to estimate the numerical ratios of convergence, given in Table 1.

Table 1: L_∞ error approximations and numerical convergence ratios in the strips Ω_1 e Ω_2 on a sequence of progressively finer adaptive meshes.

	Ω_1	r_{Ω_1}	Ω_2	r_{Ω_2}
$\ \mathbf{u}_{128 \times 384L2} - \mathbf{u}_{64 \times 192L2}\ _{\infty, \Omega_i}$	2.55675×10^{-2}	-	7.13599×10^{-4}	-
$\ \mathbf{u}_{256 \times 768L2} - \mathbf{u}_{128 \times 384L2}\ _{\infty, \Omega_i}$	5.52438×10^{-3}	4.63	5.13459×10^{-4}	1.39
$\ \mathbf{u}_{512 \times 1536L2} - \mathbf{u}_{256 \times 768L2}\ _{\infty, \Omega_i}$	2.68285×10^{-3}	2.05	1.74846×10^{-4}	2.95

As the results in Table 1 suggest, the overall scheme presents a first order behavior close to the fluid interface (as inherited from the IB Method) and a tendency toward second order away from it. Fig. 4 depicts the interface locations at the initial and final computation times ($t=0s$ and $0.2s$).

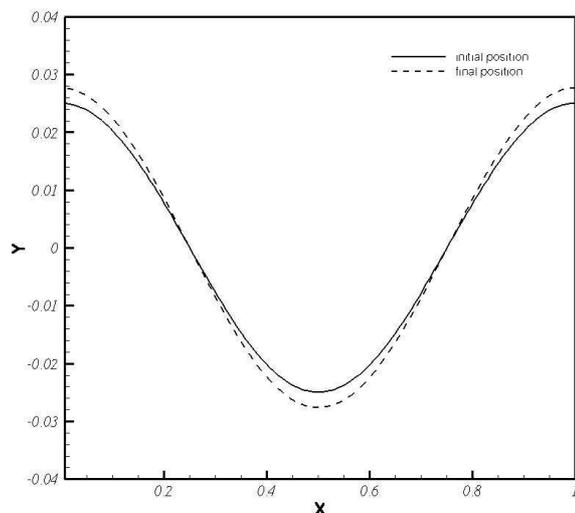


Figure 4: Initial and final interface locations.

5.2 Reduction of spurious currents

A common test for the accuracy of the surface tension representation in multiphase flow methods is a static cylindrical droplet [19,34,54,64]. That is, the exact solution is $\mathbf{u}=\mathbf{0}$ and thus the droplet should not move. However, when the surface tension is sufficiently high and there is significant, numerically introduced anisotropy in its representation, artificial nonzero velocities can be generated. These spurious currents, while small in magnitude, could affect the accuracy of computations for large tension forces and relatively coarse grids. This numerical problem has been well documented for both tracking methods (see for example [54]) and volume-of-fluid methods [65].

This simple flow is characterized by a Laplace number, $La = \sigma \rho D / \mu^2$, where D is the diameter of the drop, and by ρ_1 / ρ_2 , and μ_1 / μ_2 . Here we fix the viscosity and density ratios to one, following [19, 34]. A dimensionless measure of the strength of the spurious currents is the capillary number $Ca = U_{\max} \mu / \sigma$, where U_{\max} is the magnitude of the maximum velocity for a given simulation. The computational domain is a square of size $2D$. The drop is placed at the center of the square and periodic boundary conditions are adopted.

To evaluate the effectiveness of the proposed hybrid approach in reducing spurious currents we perform three series of tests. Two of them start with the traditional setting of a circular interface at rest; one test is done with equidistribution of Lagrangian points and the other one without. The third series corresponds to an initially elliptical interface configuration, also at rest initially. This last and more stringent series of tests allows us to investigate also the onset and the intensity of the spurious currents when the steady state is reached after the transient dynamics and when the interface is non-circular. In all the cases $\rho_1 / \rho_2 = \mu_1 / \mu_2 = 1$.

5.2.1 Case one: Circular droplet with equidistributed points

We perform two runs starting with a circular interface with equidistributed Lagrangian markers and the equidistribution is maintained dynamically, via (4.1)-(4.2). In the first run, we use a composite grid 16×16 L2 and vary La in multiples of 10 from 1.2 to 12000. We measure the strength of the spurious currents at time $t = 250t_c$, where $t_c = \mu D / \sigma$ is the characteristic time scale, by determining $\|\mathbf{u}\|_\infty$. Table 2 displays the maximum magnitude of the spurious currents, $Ca = \mu / \sigma \|\mathbf{u}\|_\infty$, for the different La . The corresponding Ca 's obtained using the standard Lagrangian force are also included in the middle column as a reference. While Ca is about 5×10^{-4} with the Lagrangian force at high La , the magnitude of the spurious currents are virtually eliminated when the hybrid force formulation is used in conjunction with dynamic Lagrangian equidistribution in this particular example. Ca is close to machine precision and is, for some cases, over six orders of magnitude smaller than that reported in [19, 34, 64] for the same values of La and grids with equivalent resolution.

For our second run, we fix $La = 12000$ and increase the spatial resolution by using the adaptive capability of the proposed method (respecting the time stepping constraints). The composite grids have a base level (coarsest level) of 16×16 and the finest level covers entirely the circular interface. We consider from one up to five levels of refinement which give equivalent uniform grids of sizes 16×16 to 256×256 . The results summarized in Table 3 show that the delicate balance between the pressure gradient and the tension force is maintained up to machine precision on the adaptive composite grid, even at modest resolution, with the hybrid force setting. On the other hand, the use of the standard

Table 2: Magnitude Ca of the spurious currents on a composite grid 16×16 L2 for different La . Lagrangian equidistribution.

$La = \frac{\sigma \rho D}{\mu^2}$	$Ca = \frac{\mu}{\sigma} \ U\ _\infty$ (Lagrangian)	$Ca = \frac{\mu}{\sigma} \ U\ _\infty$ (Hybrid)
1.2	5.11×10^{-3}	6.81×10^{-9}
12	5.20×10^{-5}	1.50×10^{-10}
120	5.08×10^{-4}	2.22×10^{-12}
1200	5.08×10^{-4}	9.57×10^{-13}
12000	5.16×10^{-4}	7.73×10^{-14}

Table 3: Magnitude Ca of the spurious currents on a sequence of adaptive grids for $La = 12000$. Lagrangian equidistribution.

Mesh	$Ca = \frac{\mu}{\sigma} \ U\ _\infty$ (Lagrangian)	$Ca = \frac{\mu}{\sigma} \ U\ _\infty$ (Hybrid)
16×16 L1	5.22×10^{-4}	8.37×10^{-13}
16×16 L2	4.45×10^{-4}	7.73×10^{-14}
16×16 L3	2.91×10^{-4}	3.08×10^{-14}
16×16 L4	1.53×10^{-4}	9.50×10^{-15}
16×16 L5	9.02×10^{-5}	5.55×10^{-15}

Lagrangian formulation (3.2) in this case results in spurious currents whose magnitude is $\mathcal{O}(10^{-4})$ as Table 3 demonstrates.

The two previous runs show that the spurious currents are in effect eliminated by the proposed approach for this special case of an initially circular interface. The Lagrangian equidistribution is preserved dynamically, via (4.1)-(4.2), and thus the required pressure-force balance is achieved at machine precision.

To measure the percentage of area loss, we define

$$A_{\text{loss}}(t) = \frac{A(t) - A(0)}{A(0)} \times 100, \quad (5.3)$$

where $A(t)$ is the area of the circular droplet at time t . Table 4 compares the area loss at $t = 250t_c$ for the hybrid force (HF) and the standard Lagrangian force (LF) formulations using dynamic Lagrangian equidistribution for both and for $La = 12000$. The hybrid force formulation yields consistently a smaller area loss than the Lagrangian force formulation and at high resolutions that difference can be about one order of magnitude or better. This behavior is the same for the set of La tested (1.2–12000).

Table 4: Percentage of area loss $A_{\text{loss}}(250t_c)$ for $La = 12000$ for hybrid and Lagrangian force formulations using Lagrangian equidistribution for both.

Mesh	Hybrid (%)	Lagrangian (%)
16×16 L1	1.905	2.240
16×16 L2	0.292	0.557
16×16 L3	0.048	0.230
16×16 L4	0.009	0.099
16×16 L5	0.002	0.049

5.2.2 Case two: Circular droplet with non-equidistributed points

We start with Lagrangian markers distributed *non-uniformly* along the circular droplet according to

$$\mathbf{x}_k(0) = 0.2 \begin{pmatrix} \cos \theta_k \\ \sin \theta_k \end{pmatrix}, \quad 0 \leq k < N_b,$$

where $\theta_k = k\theta + 0.4\sin(2k\theta)$ with $\theta = 2\pi / (N_b - 1)$.

We perform two series of computations which mirror those of the previous case, 5.2.1. The results are summarized in Table 5 and Table 6.

A comparison of Table 2 to Table 5 and Table 3 to Table 6 shows the impact of the non-uniformly distributed Lagrangian points on the intensity of the spurious currents for this case; Ca increases at least five orders of magnitude. This dramatic difference could be attributed to the unusual precision with which equidistribution yields the curvature (tension force) for this special, constant curvature case. But even without equidistribution, the hybrid force formulation produces currents which are between one and two orders of magnitude smaller than those obtained with the standard Lagrangian force.

Table 5: Magnitude Ca of the spurious currents on a composite grid 16×16 L2 for different La . Non-uniformly distributed Lagrangian markers.

$La = \frac{\sigma D}{\mu^2}$	$Ca = \frac{\mu}{\sigma} \ U\ _{\infty}(\text{Lagrangian})$	$Ca = \frac{\mu}{\sigma} \ U\ _{\infty}(\text{Hybrid})$
1.2	1.53×10^{-2}	4.07×10^{-4}
12	1.00×10^{-4}	1.23×10^{-5}
120	1.86×10^{-3}	4.25×10^{-5}
1200	2.04×10^{-3}	4.14×10^{-5}
12000	1.60×10^{-3}	9.18×10^{-5}

Table 6: Magnitude Ca of the spurious currents on a sequence of adaptive grids for $La = 12000$. Non-uniformly distributed Lagrangian markers.

Mesh	$Ca = \frac{\mu}{\sigma} \ U\ _{\infty}(\text{Lagrangian})$	$Ca = \frac{\mu}{\sigma} \ U\ _{\infty}(\text{Hybrid})$
16×16 L1	2.22×10^{-3}	4.28×10^{-4}
16×16 L2	1.60×10^{-3}	5.24×10^{-5}
16×16 L3	5.26×10^{-4}	1.72×10^{-5}
16×16 L4	2.06×10^{-4}	5.53×10^{-6}
16×16 L5	6.69×10^{-5}	1.59×10^{-6}

There is no apparent difference vis a vis area conservation with respect to the results presented in Table 4. That is, the hybrid force formulation yields a smaller area loss and at high resolutions, the area (mass) conservation is at least one order of magnitude better than that with the standard Lagrangian formulation. It is clear that the hybrid force formulation alone makes a significant difference in the reduction of spurious currents and can also improve area conservation appreciably.

5.2.3 Case three: an initially elliptical interface

We present now a more stringent test to better evaluate the combined effectiveness of the hybrid force formulation and the equidistribution of the proposed methodology. Initially, the interface has an elliptical shape with horizontal and vertical semi-axes equal to $0.6D$ and $0.4D$, respectively. The domain is again a square of size $2D$. We fix the Laplace number to 250 and measure the spurious currents at time $t = 500t_c$, when $\|\mathbf{u}^{n+1}\|_{\infty} - \|\mathbf{u}^n\|_{\infty}$ is $\mathcal{O}(10^{-6})$ (our definition of steady state) where $t_c = \mu D / \sigma$.

We perform two series of runs both with initially equidistributed Lagrangian points but in one series equidistribution is enforced dynamically, as proposed in the method, whereas in the other it is not and the Lagrangian points move freely according to the flow's tangential velocity. Tables 7 and 8 show the results for this test with the dynamic Lagrangian equidistribution and without it, respectively. In these tables, sd stands for the standard deviation of the mean Lagrangian particle distance. With equidistribution enforced dynamically, the spurious currents are two orders of magnitude smaller than those without. There does not seem to be an appreciable difference in area conservation for these two cases. As a reference, Ca when the standard Lagrangian force is used is

Table 7: Amplitude of the spurious currents for an initially elliptical interface on a sequence of adaptive meshes for $La=250$ with dynamic Lagrangian equidistribution.

Mesh	sd ($t=0.0$)	sd ($t=500t_c$)	$A_{\text{loss}}(500t_c)\%$	$Ca = \frac{\mu}{\sigma} \ U\ _{\infty}$
24×24 L1	8.21×10^{-7}	1.40×10^{-5}	0.3760	4.97×10^{-7}
24×24 L2	2.76×10^{-8}	1.94×10^{-6}	0.0525	2.14×10^{-7}
24×24 L3	8.79×10^{-10}	2.45×10^{-7}	0.0067	7.53×10^{-8}
24×24 L4	2.76×10^{-11}	3.24×10^{-8}	0.0010	4.33×10^{-8}

Table 8: Amplitude of the spurious currents for an initially elliptical interface on a sequence of adaptive meshes for $La=250$ without enforcing dynamic Lagrangian equidistribution.

Mesh	sd ($t=0.0$)	sd ($t=500t_c$)	$A_{\text{loss}}(500t_c)\%$	$Ca = \frac{\mu}{\sigma} \ U\ _{\infty}$
24×24 L1	8.21×10^{-7}	2.05×10^{-3}	0.1060	1.06×10^{-5}
24×24 L2	2.76×10^{-8}	1.01×10^{-3}	0.0229	4.27×10^{-6}
24×24 L3	8.79×10^{-10}	5.02×10^{-4}	0.0035	2.75×10^{-6}
24×24 L4	2.76×10^{-11}	2.52×10^{-4}	0.0011	1.53×10^{-6}

$\mathcal{O}(10^{-4})$ for this test. Thus, the combined hybrid force-equidistribution yields spurious currents nearly four orders of magnitude smaller in this more general case.

In summary, the combined hybrid force formulation and the dynamic Lagrangian equidistribution yield a significant reduction of the spurious currents, of at least four orders of magnitude, over the standard Lagrangian force computation and can also improve area conservation appreciably at high resolutions. Interestingly, for the standard Lagrangian force computation the uniformity or the lack thereof in the initial distribution of interface points does not seem to affect the order of magnitude of the spurious currents.

5.3 Deforming vortex flow

We consider now a test that has become prototypical for multiphase flow methods [18, 66–69]. It consists of an initially circular interface which is affected by a given straining flow. This test was first considered by Rider and Kothe [18] and the velocity field is one proposed by Bell, Colella, and Glaz [70]. Introducing a cosine time factor to reverse the flow as in [66–69], the straining flow is given in terms of the stream function

$$\psi(t, x, y) = \frac{1}{\pi} \sin^2(\pi x) \sin^2(\pi y) \cos(\pi t/T), \quad (5.4)$$

where T is the period after which the interface returns to its initial shape and $u = -\psi_y$, $v = \psi_x$. The computational domain is $\Omega = [0, 1] \times [0, 1]$, the initially circular interface has a radius of 0.15 and its disk center is placed at $(0.50, 0.75)$, and $T = 8$. The straining velocity field gives rise to a highly stretched and deformed interface with thin pointed tail. As the interface continues to be affected by the flow it evolves into an elongated, thin

Table 9: Standard deviation from Lagrangian equidistribution for $\alpha = 4.0$

Mesh	Standard Deviation (t=0)	Standard Deviation (t=1)
32×32 L5	2.49×10^{-17}	6.02×10^{-5}
32×32 L6	2.17×10^{-17}	2.13×10^{-5}
32×32 L7	3.00×10^{-17}	7.51×10^{-6}

coiled filament. The purpose of this test is usually two-fold: to assess the ability of the given method to capture accurately the thin elongated filaments and the high curvature regions and to track the extent of area loss under such a severe interfacial deformation. Naturally, an accurate tracking method such as ours outperforms capturing-based approaches in the former. Thus, our main intent in this test is really to show the power of the proposed adaptive approach. With it, it is also possible to efficiently reduce mass loss to unprecedented low levels with a number of computational cells equivalent to those of a relatively coarse resolution.

We evolve the interface according to (4.23) and (4.25) with $U^A \equiv 0$ as there is absence of surface tension and Lagrangian clustering in this case provides natural adaption at high curvature regions. We employ four Lagrangian points per Eulerian mesh cell. Table 9 shows that the Lagrangian point equidistribution is maintained in time up to truncation error.

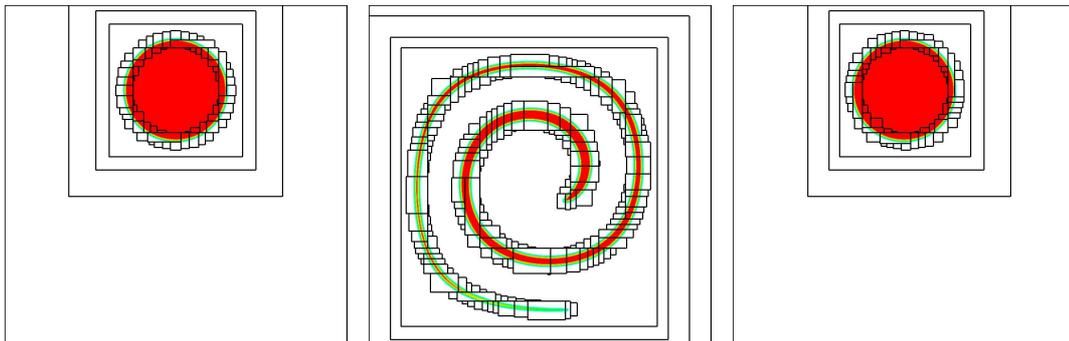
Figure 5: Interface and 32×32 L5 composite grid at $t=0$, $t=4$, and $t=8$.

Fig. 5 shows the interface as well as the composite grid structure (32×32 L5) at $t=0$, at $t=4$ when it reaches its maximum deformation, and at the final time $t=8$ when it completes a full period. The high curvature tip and tail of the elongated interface are accurately and efficiently resolved with adaptive strategy. The adaptive mesh 32×32 L5 provides a fine resolution equivalent to that of a 512×512 uniform grid but at a fraction of the uniform grid cost. The area is preserved remarkably well despite the extreme interfacial deformation even with the relatively coarse 32×32 L4 as Fig. 6 indicates. With 32×32 L4 the area variation throughout the entire period is less than 0.007% while that for 32×32 L5 is well below 0.001%.

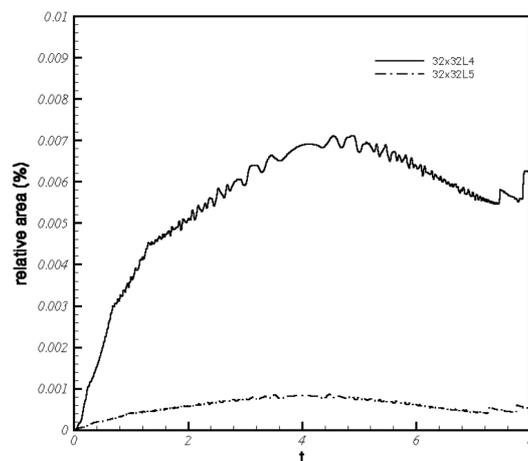


Figure 6: Percentage of relative area loss for composite grids 32×32 L4 and 32×32 L5.

5.4 Rayleigh-Taylor instability

The Rayleigh-Taylor (RT) instability [71, 72] which occurs at the interface of two fluids with different densities when the heavier fluid is accelerated into the lighter fluid is one of the fundamental instabilities in fluid dynamics [73]. The RT instability is believed to play a crucial role in the process of mixing which is important to a wide variety of applications, including industrial fuels, micro-emulsions, explosions and implosions. Due to its fundamental relevance and applications, several numerical studies have been performed [24, 74–83] and the RT instability has also served as a test case for numerical methods of multiphase flows [17, 19, 62, 84, 85].

In the presence of surface tension, the simulations of the RT instability are particularly challenging. To date there is only a handful of numerical studies of the effect of interfacial forces on the long time dynamics of a fluid free boundary undergoing RT instability. For inviscid 2D flows, at small to moderate Atwood numbers, the interface develops thin fluid fingers that subsequently roll-up and appear to eventually collapse with the adjacent fluid interface giving rise to a topological singularity [62]. A relevant question is how small but finite viscosity would affect such a singular event and to what extent commonly observed drop formation in RT simulations is numerically induced rather than a consequence of the underlying model. To try to answer these questions, we consider the same single-mode setup and flow parameters used in [62] but with nonzero, small viscosity.

The flow parameters are given (in cgs units) as follows [62]: the mass density of the heavier fluid (at the top) is $\rho_1 = 1.2$ while that of the lighter fluid (at the bottom) is $\rho_2 = 1.1$. The Atwood number (5.2) is -0.1 . The two fluids are viscosity matched and $\mu_1 = \mu_2 = 10^{-4}$. The surface tension coefficient is $\sigma = 0.005$ and the “gravity acceleration” is chosen

as $g = 10$ (100 times smaller than its actual value). Initially, the flow is at rest and the interface is given by

$$\mathbf{X}(0) = (\alpha, 0.1 \cos(2\pi\alpha)), \quad 0 \leq \alpha \leq 1. \quad (5.5)$$

At the boundaries of the rectangular domain $\Omega = [0,1] \times [-1.5,1.5]$, the velocity satisfies the homogeneous Dirichlet condition at the north and at the south borders, and periodic condition in the horizontal direction.

Fig. 7 shows snapshots of the fluid interface undergoing RT instability. As the free boundary is accelerated downward, symmetric high curvature points develop and subsequently evolve into two small fluid fingers. After this, the fingers begin to roll-up and the interface develops the mushroom-shape structure characteristic of the RT instability. The interface profile up to about $t = 1.40$ is remarkably close to that of the inviscid flow, Fig. 4 in [62] and is in accord with single-mode experiments for a similar Atwood number [86]. Also, as in the inviscid case [62], there are traces of capillary waves emanating from the fingers. These small-scale structures are accurately captured by the proposed approach. Two protuberances opposite the fingers develop at about $t = 1.36$ (more noticeable at $t = 1.63$). The evidence presented in [62] for the corresponding inviscid flow suggests that the fingers collapse with these protuberances in finite time, giving rise to the formation of trapped bubbles. However, the interface motion in the viscous flow is much more complex than its inviscid counterpart. The fingers undergo a more pronounced deformation and a stretching leading up to formation of thin, filament-like structures ($t = 2.28$).

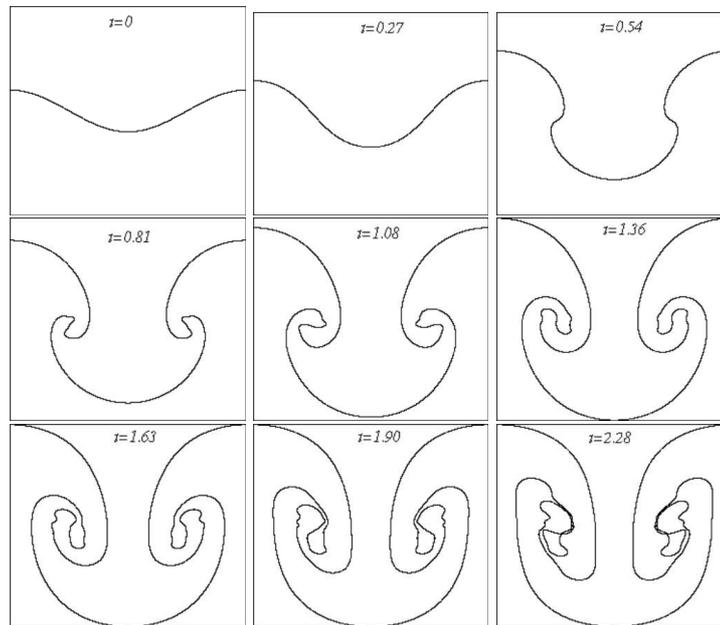


Figure 7: Time evolution of the fluid interface undergoing RT instability in the presence of surface tension.

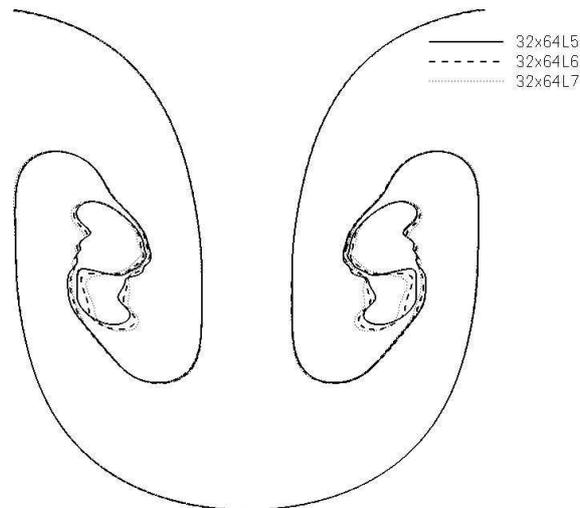


Figure 8: Comparison of interface positions for different adaptive mesh resolutions at $t=2.13$.

Fig. 8 presents a convergence-under-refinement study at time $t = 2.13$. Three resolutions are compared, 32×64 L5, 32×64 L6, and 32×64 L7. The corresponding average distance between two Lagrangian points as well as the standard deviation from this average (a measure of equidistribution) is provided in Table 10. The fluid interface appears to be accurately resolved with the 32×64 L7 composite grid and the Lagrangian equidistribution is maintained up to truncation error. Further indication of the accuracy is given in Fig. 9 which displays the percentage of relative area variation in time. As the fingers develop, there is an evident area loss but throughout the entire long-time computation the area variation is well below 0.0055%.

Table 10: Average Lagrangian point distance and its standard deviation at $t=2.23$.

Mesh	Standard Deviation ($t=2.13$)	Average Distance
32×64 L5	1.23×10^{-5}	1.03×10^{-3}
32×64 L6	4.04×10^{-6}	5.12×10^{-4}
32×64 L7	2.02×10^{-6}	2.55×10^{-4}

The dynamics leading to the last interface profile in Fig. 7, $t = 2.28$, suggests a possible pinch-off scenario as that in the inviscid case [62]. However, a close look at the time behavior of the minimum distance between adjacent, opposite interfacial segments, Fig. 10, reveals a contrasting outcome. Slightly before $t = 2.26$, the decrease of the minimum distance saturates, as clearly indicated by the two highest resolutions, 32×64 L6 and 32×64 L7. At that instant, the minimum distance achieved is about 5 computational cells which starts to increase afterwards. This is reminiscent of the near pinching roll-up observed in a 2D viscous interface undergoing Kelvin-Helmholtz instability where vis-

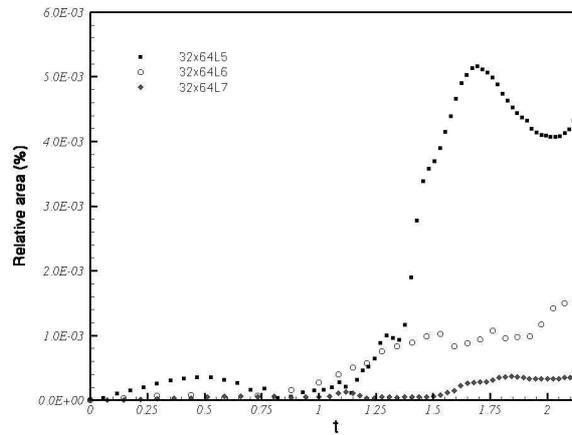


Figure 9: Evolution in time of the percentage of relative area loss with for all composite grids.

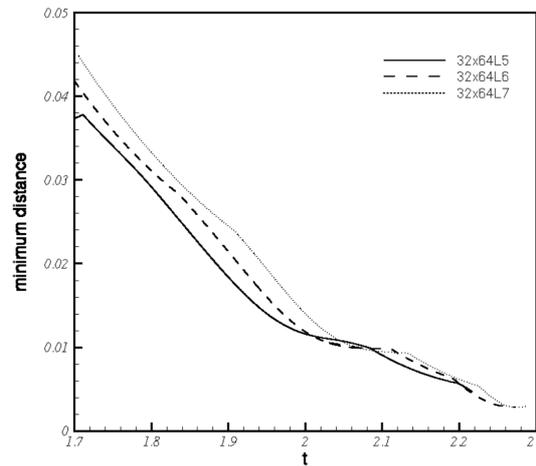


Figure 10: Evolution of the minimum distance from disparate interface segments.

cous effects appear to prevent finite-time pinching [41,87]. Fig. 11 shows a sequence of close-ups around the near pinch-off region. In particular, the plots in the second row in Fig. 11 provide details of the adaptive mesh structure at its finest level which effectively covers the fluid boundary and adequately resolves the thin interfacial gap. Note that the mollified delta function spans its support inside the light band showed in the figure (approximately four cells wide, about the same as the width of the support of the discrete delta function). That band is the diffused interface itself. At the time when the minimum distance is reached, the near pinch-off region has width larger than four computational cells, indicating that the (diffused) interface did not touch itself - hence, computations are still physically relevant.

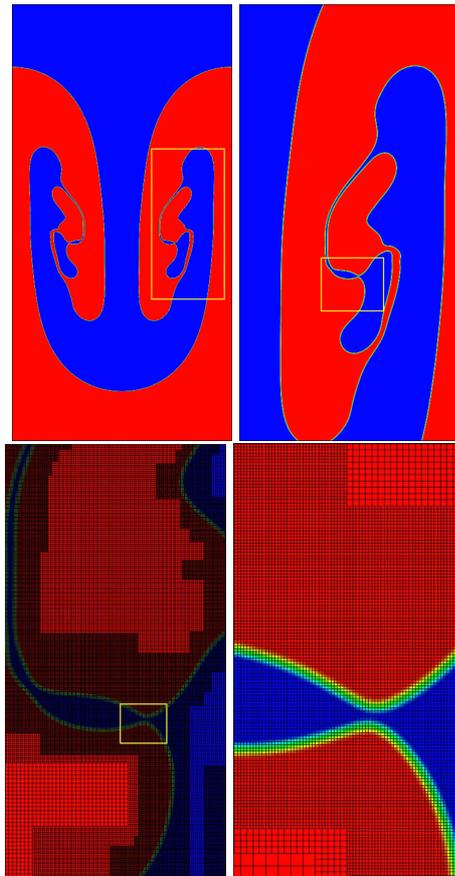


Figure 11: Rayleigh-Taylor instability at $t=2.28$. Fluid indicator, interface, grid details are depicted on a seven level composite grid.

This simulation provides only a glimpse of the complex rheology of a viscous fluid interface undergoing RT instability in the presence of surface tension and which can be accurately and efficiently captured with the proposed methodology. There is a number of important questions that remain open which include the limiting behavior as a) viscosity goes to zero and b) as surface tension goes to zero, as well as the effects of viscosity stratification. We intend to conduct a thorough investigation of these problems and will report on the results elsewhere.

5.5 Ascending bubble dynamics

The rising motion of a single buoyant bubble in an initially quiescent ambient fluid is a classical fluid dynamics problem which has been well studied both experimentally and theoretically [88]. As such, it provides an excellent test for two-phase flow methods. Experimental results are generally presented in terms of the following dimensionless

groups [89]

$$Re = \frac{\rho_2 DU}{\mu_2}, \quad Eo = \frac{\rho_2 g D^2}{\sigma}, \quad \text{and} \quad M = \frac{g \mu_2^4}{\sigma^3 \rho_2},$$

where ρ_2 and μ_2 are the density and viscosity of the ambient fluid, respectively, D is the initial diameter of the bubble, and U is the norm of the rise terminal velocity. Re , Eo , M , are referred to as the rise Reynolds number, Eötvös number, and Morton number, respectively.

In our test, we take $Eo = 1$, $M = 10^{-2}$ (with a corresponding $\sigma = 9N/m$), $\rho_1/\rho_2 = 0.1$, and $\mu_1/\mu_2 = 1$. Here ρ_1 (ρ_2) and μ_1 (μ_2) are the density and viscosity of the bubble (ambient fluid), respectively. With these parameters, it is expected that the bubble will rise essentially preserving its circular shape with a small rise Re [88]. This is an appropriate case to test the ability of the method to accurately resolve interfacial forces. A circular interface is also a stringent test for AMR composite grids [32].

Our computational domain is $\Omega = [0, \frac{10}{3}D] \times [0, \frac{20}{3}D]$, where $D = 0.03m$ is the initial diameter of the bubble. Initially, the center of the bubble is located at $(\frac{5}{3}D, D)$ and the velocity field is zero. We applied for the velocity a boundary condition which was periodic in x and Dirichlet (no-slip) in y .

Fig. 12 displays snapshots of the density field along with streamlines at different characteristic times of the evolution. The characteristic (dimensionless) time is $t^* = t/\sqrt{D/g}$. The flow, as depicted by the streamlines inside and outside the bubble, matches that of an circular rising bubble through a quiescent flow [88] with no apparent effects from the domain's boundary. Also shown in Fig. 12 are the nested grids (rectangular patches) for a $32 \times 64L5$ composite mesh whose finest level covers entirely the fluid interface at all times, guaranteeing that the material properties stay sharply defined throughout the computation. As an illustration of the computational efficiency of our adaptive strategy in terms of memory usage, an equivalent uniform grid of 512×1024 would employ 733863 computational cells while our adaptive $32 \times 64L5$ mesh only uses 57800. Per time step, the adaptive method is about six times faster than the uniform grid one at this resolution. Naturally, the computational savings increase with higher resolutions (i.e., as more levels of refinement were added).

A look at the vorticity field, shown in Fig. 13, reveals that after a short transient period the motion is dominated by a vortex pair whose strength and shape remain largely unchanged. This steady motion is confirmed by a plot of the rise speed of the bubble (that is, of its center) or equivalently the rise Re against time. This is shown in Fig. 14. The steady, rise Re obtained from this simulation is 0.48 which is fairly close to the value of 0.5 from the shape-velocity diagram in [88] for the same Eo and M but for an unbounded fluid. As pointed in [90], it is expected that for finite volume fractions the bubble's rise will be slower.

We now examine the accuracy of this simulation by looking at the pressure, the mean curvature, and the bubble's area variation for different resolutions. Fig. 15 shows the pressure jump across the bubble's boundary for fixed y corresponding to the bubble's

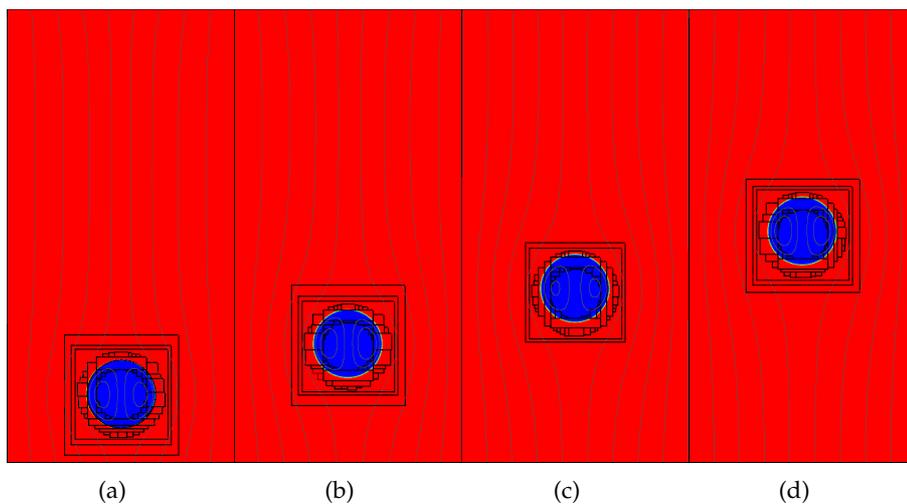


Figure 12: Evolution of the density field superimposed with streamlines for a $32 \times 64L5$ composite grid simulation. The nested adaptive grid patches are also shown. $Eo=1$ and $M=10^{-2}$. (a) $t^*=0.18$, (b) $t^*=5.3$, (c) $t^*=10.6$, and (d) $t^*=16.0$.

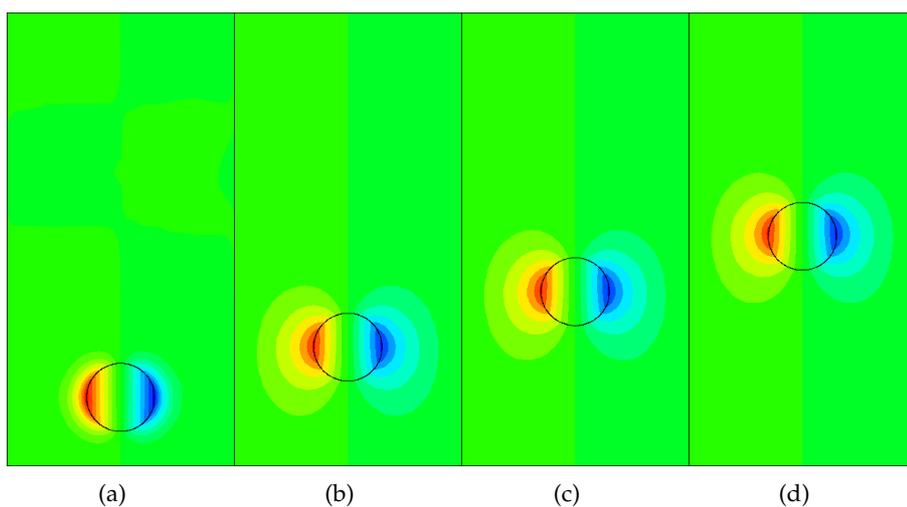


Figure 13: Evolution of the vorticity field for a $32 \times 64L5$ composite grid simulation. $Eo=1$ and $M=10^{-2}$. (a) $t^*=0.18$, (b) $t^*=5.3$, (c) $t^*=10.6$, and (d) $t^*=16.0$.

centroid, for four different resolutions. The sharp pressure profiles obtained with four and five levels of refinement are indistinguishable. Both resolutions give a pressure jump equal to $591KPa$ while the theoretical value from the Laplace-Young condition is $2\sigma/D=600KPa$, i.e. there is a variation of less than 1.5%. We also compare the theoretical value of the mean curvature ($2/D$) with that obtained from different resolutions in Table 11. Here, the curvature was evaluated at the point corresponding to the angle π along the

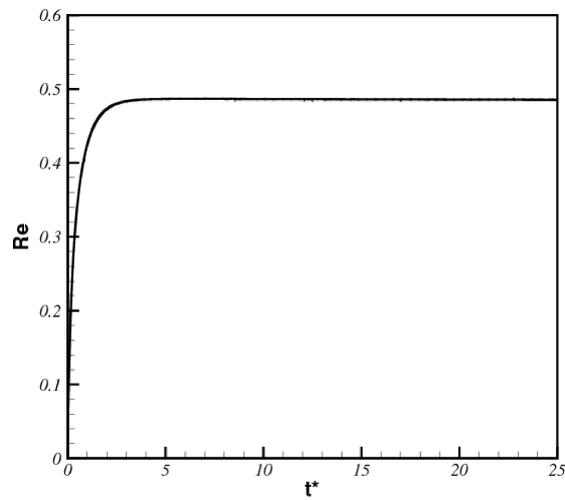


Figure 14: Rise Reynolds number against non-dimensional time on 32×64 L5 grid cells.

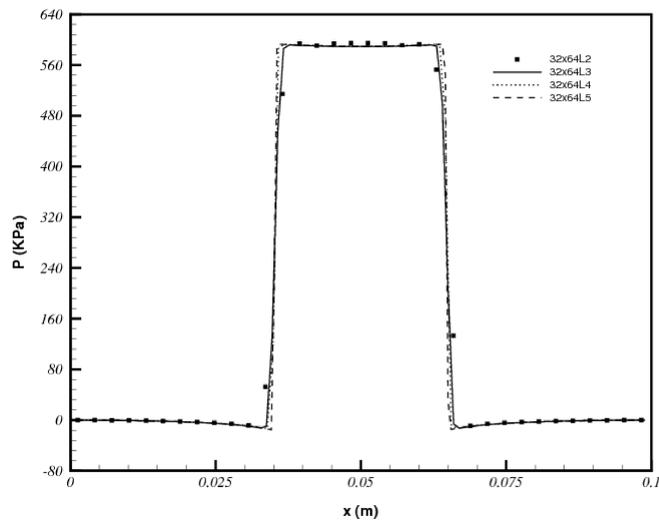


Figure 15: Capillary pressure of a circular bubble with $Eo=1$ and $M=10^{-2}$.

circular interface at $t^* = 25$. As Table 11 shows, the percentage error is quite small and approaches 0.176% for the four and five level adaptive grids.

Due to the interpolations at the interface, it is expected that the area (volume) of the bubble is only approximately conserved. Typically, the area loss grows linearly in time and with the mesh size. However, for this special case when the bubble keeps its shape unchanged, the area conservation of our IB-based method is much better than expected

Table 11: Mean curvature κ at $\theta = \pi$ and $t^* = 25$, and percentage error when compared to the analytic value $\kappa = 2/D = 66.6666$.

Curvature error		
Mesh	κ at $\theta = \pi$	Error %
$32 \times 64L3$	66.5461	0.181%
$32 \times 64L4$	66.5484	0.177%
$32 \times 64L5$	66.5492	0.176%

Table 12: Relative area variation per characteristic time ($t^* = t/\sqrt{D/g}$) for $Eo = 1$ and $M = 10^{-2}$.

Relative area variation (%/ t^*)		
Mesh	Hybrid force formulation	Lagrangian force formulation
$32 \times 64L2$	$1.92 \cdot 10^{-2}$	$4.75 \cdot 10^{-3}$
$32 \times 64L3$	$1.32 \cdot 10^{-3}$	$1.97 \cdot 10^{-3}$
$32 \times 64L4$	$1.64 \cdot 10^{-4}$	$8.86 \cdot 10^{-4}$
$32 \times 64L5$	$9.16 \cdot 10^{-6}$	$4.03 \cdot 10^{-4}$

for both the hybrid force and the Lagrangian force formulations as Table 12 shows. The area is computed from the interface position variable \mathbf{X} by applying Stokes theorem and approximating the resulting integral (with periodic integrand) with the trapezoidal rule. For this particular flow, the adaptive method with the hybrid force formulation preserves area with astonishing accuracy giving a relative area variation per characteristic time for the five-level grid of less than $10^{-5}\%$. This unprecedented accuracy in the area conservation appears to be directly linked to the special, constant curvature nature of the fluid interface. It is interesting to note however that the adaptive, nonsymmetric grid was able to preserve that interface geometry and pressure-surface tension balance so accurately.

5.6 Impact and rebound of a deformable drop onto a fluid interface

Our final example to highlight the capabilities of the proposed adaptive, hybrid method is the numerical simulation of a falling drop in an ambient fluid which impinges upon a free fluid surface and bounces back due to inertia. This simulation is motivated by the experiments of Mohamed-Kassim and Longmire [63] and by the level-set simulations of Zheng, Lowengrub, Anderson, and Cristini [31].

The dynamics is dependent on the density and viscosity ratios, and three dimensionless groups: the Reynolds number Re , the Weber number We , and the Froude number Fr . Letting D be the initial diameter of the circular drop and U the drop's terminal velocity (i.e. the impact velocity), we have [63]

$$Re = \frac{\rho_2 U D}{\mu_2}, \quad We = \frac{\rho_2 U^2 D}{\sigma}, \quad Fr = \left(\frac{t_g}{t_i} \right)^2, \quad (5.6)$$

where $\rho_2(\mu_2)$ is the density (viscosity) of the ambient fluid, t_g and t_i are the gravity and

impact velocity time scales respectively, which are defined as

$$t_g = \frac{D}{U}, \quad t_i = \sqrt{\frac{D}{2g(\rho_1 - \rho_2)/(\rho_1 + \rho_2)}}, \quad (5.7)$$

with g the acceleration of gravity and ρ_1 is the density of the drop and of the bottom fluid.

We adopted a 30 cm by 30 cm square as our computational domain which is slightly narrower than the 3D tank used in the experiments [63] but three times wider and three times taller than that in the level-set simulations [31]. We made this domain choice to keep a good approximation of the experimental set up and reduce boundary effects while at the same time keeping the computations affordable. The rest of the computational parameters match those reported by Zheng *et al.* [31]: $D=1$ cm, $\rho_1/\rho_2=1.189$ and $\mu_1/\mu_2=0.33$. Note that both the density and the viscosity ratios are the experimental parameters employed by Mohamed-Kassim and Longmire [63] (referred to as “Combination 1” in their work). With that particular choice of parameters the Reynolds, Weber and Froude numbers are approximately 68, 7 and 1, respectively, close to those of the experiments [63] and exactly those in the level set simulations [31].

Initially, the flat fluid interface was placed at a height of 13 cm as in the experiments [63] and the circular drop had a radius of $a=0.5$ cm.

We employed for the velocity periodic boundary conditions in x and Dirichlet (no-slip) boundary conditions in y . The adaptive mesh for this simulation was $32 \times 32L9$, that is, it had a total of 9 refinement levels, beginning with a base level of 32×32 . The finest resolution was equivalent to that of a 8192×8192 uniform mesh.

When releasing the drop from rest at a height of 23 cm as in [63], we were unable to reproduce the impact conditions (terminal velocity and drop interface shape) reported in the experiments. This is not entirely surprising as the dynamics in the experiments is three dimensional. Instead, the circular drop center was placed at y_c , where $y_c = 13 + w$ with w , the actual distance between the initially flat interface and the center of the circular drop, satisfying the ratio $w/a = 2.5$ (a is the circular drop radius), as in the level-set simulations [31]. Following Zheng *et al.* [31], we defined the initial velocity as the divergence free projection of the field given by $(0, -U)$ inside the drop and zero elsewhere, where $U = 13.2$ cm/s is the reported [63] impact (terminal) speed.

The dynamics of the impacting drop is illustrated in the sequence of snapshots of Fig. 16. The time scale is chosen as in [31], $t^* = tU/a$, where a is the radius of the drop. Following Mohamed-Kassim and Longmire [63], we define the impact time ($t_i^* = 0$) as the time when lower drop surface crosses the quiescent fluid interface level. The trailing edge of the drop quickly flattens and the drop elongates significantly in the spanwise direction prior to the collision and immediately after it [Fig. 16(d)]. This large deformation is followed by a rebound occurring at about $t^* - t_i^* = 2.9$ which is clearly an inertia effect. A look at the vorticity in Fig. 17 shows how this field is produced and subsequently shed off the drop surface. The vorticity deposited in the wake contributes to the large drop deformation. The vorticity field shown in Fig. 17 is in qualitative agreement with that reported in the experiments [63].

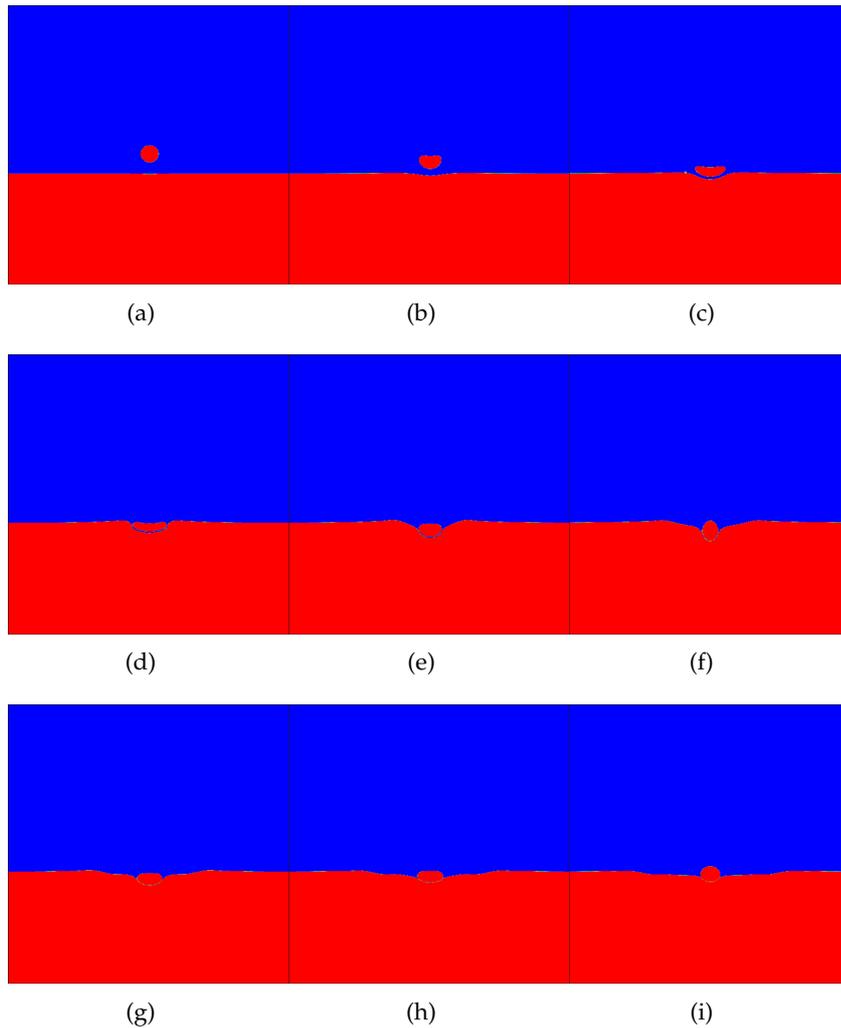


Figure 16: Evolution of the density field for a $32 \times 32L9$ simulation. (a) $t^* - t_i^* = -1.22$, (b) $t^* - t_i^* = -0.56$, (c) $t^* - t_i^* = 0.32$, (d) $t^* - t_i^* = 1.20$, (e) $t^* - t_i^* = 2.07$, (f) $t^* - t_i^* = 2.95$, (g) $t^* - t_i^* = 3.83$, (h) $t^* - t_i^* = 4.71$, and (i) $t^* - t_i^* = 5.59$.

Fig. 18 shows positions of the free interface, and the drop's lower and upper surfaces and compares these with the numerical results of Zheng *et al.* [31]. Despite the differences in the domain size, resolutions, and numerical approaches there is substantial agreement between the two numerical simulations. The interfacial dynamics predicted by our numerics is also qualitatively similar to that in the experiments but, as indicated by Zheng *et al.* [31], the 2D results have slower dynamics as observed in Fig. 19.

The relative area variation of the drop in our numerical simulation is displayed in Fig. 20. The maximum difference coincides with the event of largest drop deformation at about $t^* - t_i^* = 1.10$ but the area is maintained within 0.5% for most of the simulation. The

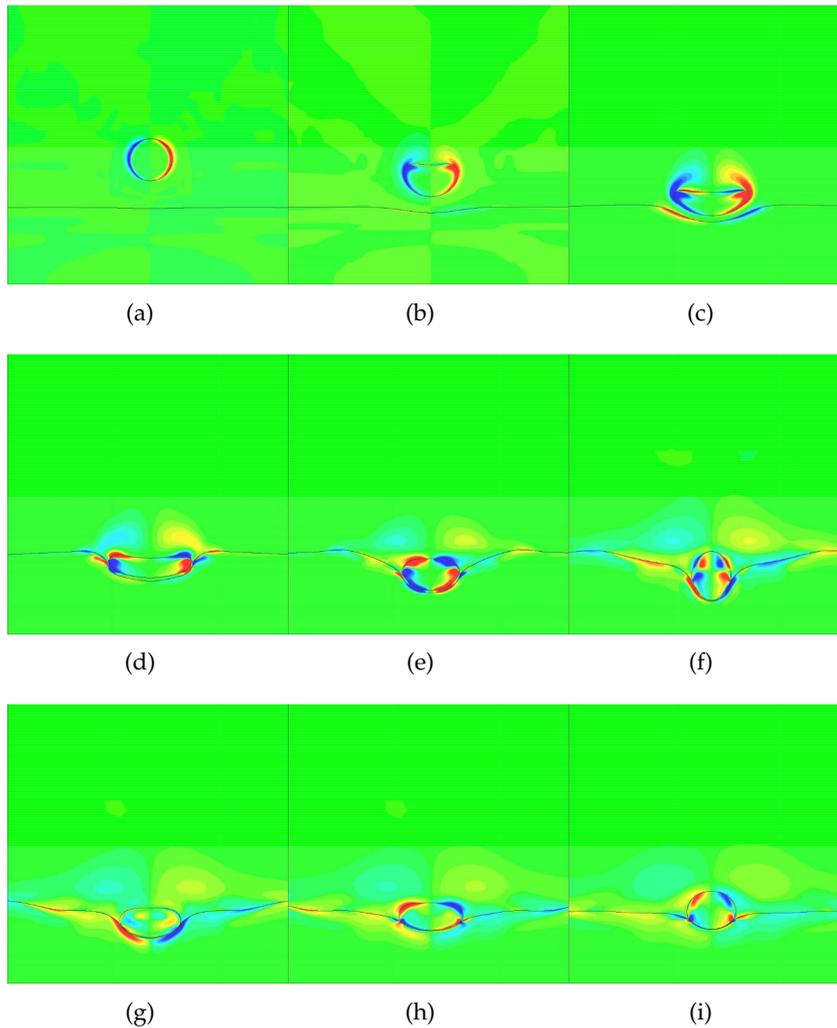


Figure 17: Evolution of the vorticity field for a $32 \times 32L9$ simulation. (a) $t^* - t_i^* = -1.22$, (b) $t^* - t_i^* = -0.56$, (c) $t^* - t_i^* = 0.32$, (d) $t^* - t_i^* = 1.20$, (e) $t^* - t_i^* = 2.07$, (f) $t^* - t_i^* = 2.95$, (g) $t^* - t_i^* = 3.83$, (h) $t^* - t_i^* = 4.71$, and (i) $t^* - t_i^* = 5.59$.

finest level of the composite adaptive mesh accompanies the drop boundary and the free liquid interface at all times. Close-ups of the actual meshes around the drop and at the thinnest fluid gap region at $t^* - t_i^* = 3.4$ are presented in Fig. 21. Note that the thin film created by the drop-free surface interaction is well resolved at this time with more than four grid cells covering the thinnest region. The separating film slowly begins to drain, as can be observed in Figs. 18 and 19, which would lead to an eventual coalescence.

With our fully adaptive method we are able to efficiently resolve length scales spanning nearly four orders of magnitude and both fast and slow dynamics. Finally, to highlight the computational savings for this case, we report that at the final simulation time,

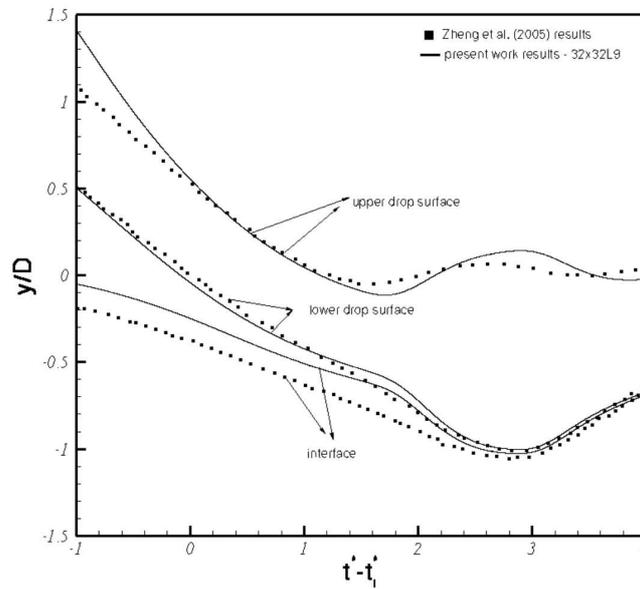


Figure 18: Interface position during impact of drop onto an interface. Comparison between Zheng et al. [31] results on a unstructured mesh with $32 \times 32L9$ adaptive mesh. Dimensionless parameters: $Re = 68$, $We = 7$, $Fr = 1$, $\mu_1/\mu_2 = 0.33$, $\rho_1/\rho_2 = 1.189$ and $h/D = 0.0035$. The y axis has been shifted to coincide with the initial position of the free fluid interface.

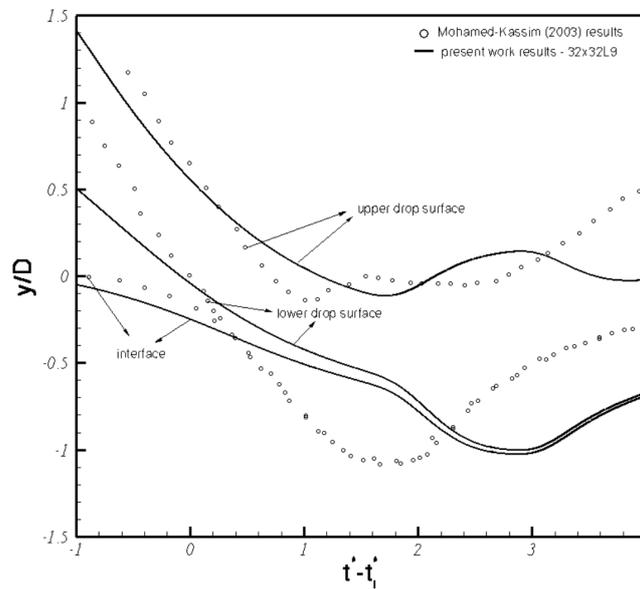


Figure 19: Interface position during impact of drop onto an interface. Comparison between Mohamed-Kassim and Longmire [63] results with $32 \times 32L9$ adaptive mesh. Dimensionless parameters: $Re = 68$, $We = 7$, $Fr = 1$, $\mu_1/\mu_2 = 0.33$, $\rho_1/\rho_2 = 1.189$ and $h/D = 0.0035$.

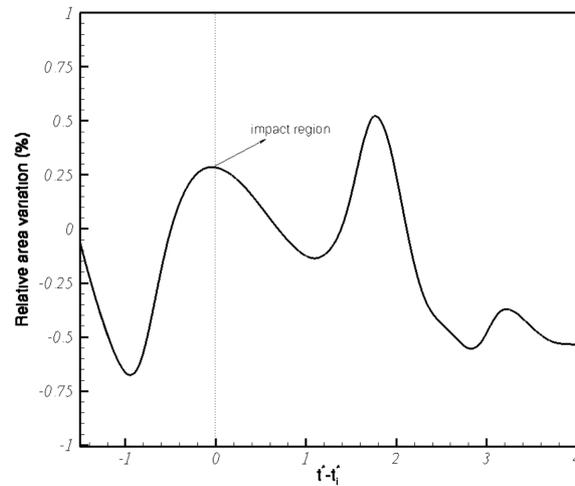


Figure 20: Relative area variation of the drop for 32x32L9.

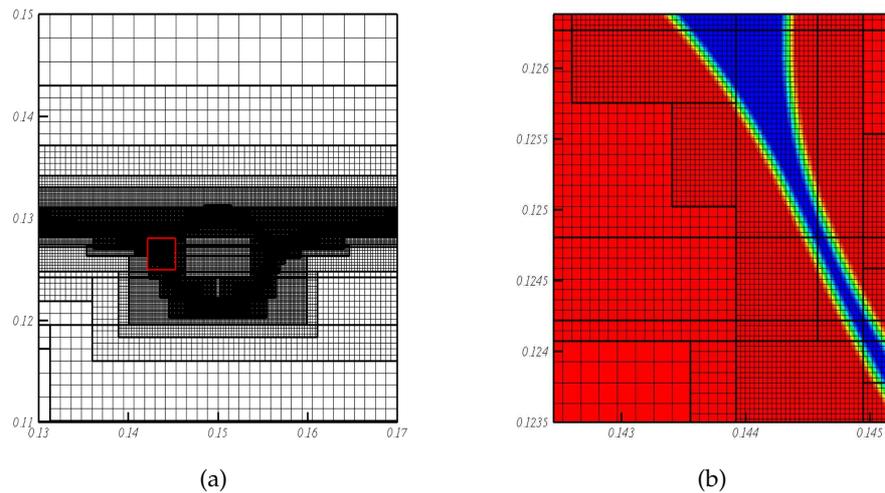


Figure 21: Close-ups of the 32x32L9 adaptive mesh at $t^* - t_i^* = 3.4$. (a) A region enclosing the drop and part of the free boundary and (b) a close-up of the smallest gap region.

the total number of computational cells was less than 6.3×10^5 while an equivalent uniform grid would have employed close to 6.7×10^7 computational cells, i.e. we reduce the number of grid cells by two orders of magnitude (implying in memory savings).

6 Conclusions

We presented a variable time step, fully adaptive in space, hybrid level-set/front tracking method for the simulation of 2D two-phase flows in the presence of surface tension.

It is hybrid first because we exploit qualities from both the IB Method (front tracking) and from the level set formulation (front capturing) and, second, because the singular interfacial force is computed employing a mixed Eulerian-Lagrangian formulation. Full adaptivity is introduced in space both through dynamic control of the Lagrangian markers, which define the interface, and through Eulerian mesh refinement, which increases locally the resolution of the fluid solver with a sequence of nested, progressively finer refinement levels. For the time discretization, we employ a variable time-step, second order semi-implicit temporal scheme imbedded in a projection method of pressure-increment type (no "sub-cycling in time" is adopted).

Fluid interfaces are efficiently covered at all times with the finest grid resolution. An effective time integration is attained with the proposed time discretization allowing for stable simulations with only mild stability constraints for the flow conditions considered. The use of the hybrid Lagrangian-Eulerian tension force representation in concert with the our accurate signed-distance fluid indicator (level set) and the dynamic Lagrangian equidistribution reduce the magnitude of the spurious currents by several orders with respect to that obtained by other existing approaches. The increased resolution afforded by the AMR yields also an accurate area conservation for long time integrations, as demonstrated by the numerical examples. We restrict the presentation of the method to two dimensions here but we envision that the full approach could be directly extended to three dimensions.

Acknowledgments

Partial support for this research was provided by the National Science Foundation under Grant number DMS 0609996 (HDC), by the Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) under Grant numbers 04/13781-1 and 06/57099-5 (AMR), and by the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) under Grant number 155491/2006-7 (MMV). H.D.C. would like to acknowledge the hospitality of the Departamento de Matemática Aplicada at USP during a three-month visit in the course of which part of this research was conducted.

References

- [1] H. A. Stone, Dynamics of drop deformation and breakup in viscous fluids, *Annu. Rev. Fluid Mech.* 26 (1994) 65–102.
- [2] Y. T. Hu, D. J. Pine, L. G. Leal, Drop deformation, breakup, and coalescence with compatibilizer, *Phys. Fluids* 12 (3) (2000) 484–489.
- [3] H. Yang, C. C. Park, Y. T. Hu, L. G. Leal, The coalescence of two equal-size drops in two-dimensional linear flow, *Phys. Fluids* 13 (5) (2001) 1087–1105.
- [4] J. H. Duncan, H. Qiao, V. Philomin, A. Wenz, Gentle spilling breakers: crest profile evolution, *J. Fluid Mech.* 379 (1999) 191–222.

- [5] X. Liu, J. H. Duncan, The effects of surfactants on spilling breaking waves, *Nature* 421 (2003) 520–523.
- [6] J. U. Brackbill, D. B. Kothe, C. Zemach, A continuum model for modeling surface tension, *J. Comput. Phys.* 100 (1991) 335.
- [7] Y. C. Chang, T. Y. Hou, B. Merriman, S. Osher, A level set formulation of Eulerian interface capturing methods for incompressible fluid flows, *J. Comput. Phys.* 124 (1996) 449–464.
- [8] M. Sussman, P. Smereka, S. Osher, A level set approach for computing solutions to incompressible two-phase flow, *J. Comput. Phys.* 114 (1994) 146–159.
- [9] R. Chella, V. Viñals, Mixing of a two-phase fluid by a cavity flow, *Phys. Rev. E* 53 (1996) 3832.
- [10] D. M. Anderson, G. B. McFadden, A. A. Wheeler, Diffuse-interface methods in fluid mechanics, *Annu. Rev. Fluid Mech.* 30 (1998) 139.
- [11] D. Jacqmin, Calculation of two-phase Navier-Stokes flows using phase-field modeling, *J. Comput. Phys.* 155 (1999) 96–127.
- [12] F. Boyer, A theoretical and numerical model for the study of incompressible mixture flows, *Computer & Fluids* 31 (1) (2002) 41.
- [13] V. E. Badalassi, H. D. Cenicerós, S. Banerjee, Computation of multiphase systems with phase field models, *J. Comput. Phys.* 190 (2003) 371–397.
- [14] J. Kim, K. Kang, J. Lowengrub, Conservative multigrid methods for Cahn-Hilliard fluids, *J. Comput. Phys.* 193 (2004) 511–543.
- [15] P. Yue, J. J. Feng, C. Liu, J. Shen, A diffuse-interface method for simulating two-phase flows of complex fluids, *J. Fluid Mech.* 515 (2004) 293–317.
- [16] R. Scardovelli, S. Zaleski, Direct numerical simulation of free-surface and interfacial flow, *Annu. Rev. Fluid Mech.* 31 (1999) 567–603.
- [17] E. G. Puckett, A. S. Almgren, J. B. Bell, D. L. Marcus, W. J. Rider, A high order projection method for tracking fluid interfaces in variable density incompressible flows, *J. Comput. Phys.* 100 (1997) 269–282.
- [18] W. J. Rider, D. B. Kothe, Reconstructing volume tracking, *J. Comput. Phys.* 141 (1998) 112.
- [19] S. Popinet, S. Zaleski, A front-tracking algorithm for accurate representation of surface tension, *Int. J. Numer. Meth. Fluids* 30 (1999) 775–793.
- [20] C. S. Peskin, Numerical analysis of blood flow in the heart, *J. Comput. Phys.* 25 (1977) 220–252.
- [21] S. O. Unverdi, G. Tryggvason, A front-tracking method for viscous, incompressible, multi-fluid flows, *J. Comput. Phys.* 100 (1992) 25–37.
- [22] J. Glimm, J. Grove, X.-L. Li, K. M. Shyue, Y. Zeng, Q. Zhang, Three-dimensional front tracking, *SIAM J. Sci. Comput.* 19 (3) (1998) 703–727.
- [23] J. Glimm, J. Grove, X.-L. Li, D. C. Tan, Robust computational algorithms for dynamic interface tracking in three dimensions, *SIAM J. Sci. Comput.* 21 (6) (2000) 2240–2256.
- [24] J. Glimm, J. W. Grove, X. Li, W. Oh, D. H. Sharp, A critical analysis of Rayleigh-Taylor growth rates, *J. Comput. Phys.* 169 (2001) 652–677.
- [25] A. L. F. Lima, A. Silveira-Neto, J. J. R. Damasceno, Numerical simulation of two-dimensional flows over a circular cylinder using the immersed boundary method, *J. Comput. Phys.* 189 (2003) 351–370.
- [26] M. Sussman, E. G. Puckett, A coupled level set and volume-of-fluid method for computing 3d and axisymmetric incompressible two-phase flows, *J. Comput. Phys.* 162 (2000) 301–337.
- [27] H. D. Cenicerós, The effect of surfactants on the formation and evolution of capillary waves, *Phys. Fluids* 15 (1) (2003) 245–256.
- [28] S. M. E. Aulisa, R. Scardovelli, A mixed markers and volume-of-fluid method for the re-

- construction and advection of interfaces in two-phase and free-boundary flows, *J. Comput. Phys.* 188 (2003) 611–639.
- [29] F. S. de Sousa, N. Mangiavacchi, L. G. Nonato, A. Castelo, M. F. Tomé, V. G. Ferreira, J. A. Cuminato, S. Mckee, A front-tracking/front-capturing method for the simulation of 3D multi-fluid flows with free surfaces, *J. Comput. Phys.* 198 (2004) 469–499.
- [30] H. D. Cenicerros, A. M. Roma, A multi-phase flow method with a fast, geometry-based fluid indicator, *J. Comput. Phys.* 205 (2005) 391–400.
- [31] X. Zheng, J. Lowengrub, A. Anderson, V. Cristini, Adaptive unstructure volume remeshing-II: Application to two- and three-dimensional level-set simulations of multiphase flow., *J. Comput. Phys.* 208 (2005) 626–650.
- [32] A. M. Roma, C. S. Peskin, M. J. Berger, An adaptive version of the immersed boundary method, *J. Comput. Phys.* 153 (1999) 509–534.
- [33] B. E. Griffith, R. D. Hornung, D. M. McQueen, C. S. Peskin, An adaptive, formally second order accurate version of the immersed boundary method, *J. Comput. Phys.* 223 (2007) 10–49.
- [34] S. Shin, S. Abdel-Khalik, V. Daru, D. Juric, Accurate representation of surface tension using the level contour reconstruction method, *J. Comput. Phys.* 203 (2005) 493–516.
- [35] S. Mauch, Efficient algorithms for solving static Hamilton-Jacobi equations, Ph.D. thesis, California Institute of Technology (2003).
- [36] K. M. Arthurs, L. C. Moore, C. S. Peskin, E. B. Pitman and H. E. Layton, Modeling arteriolar flow and mass transport using the immersed boundary method, *J. Comput. Phys.* 147 (1998) 402–440.
- [37] M. J. Berger, P. Colella, Local adaptive mesh refinement for shock hydrodynamics, *J. Comput. Phys.* 82 (1989) 64–84.
- [38] A. A. Mayo, C. S. Peskin, An implicit numerical method for fluid dynamics problems with immersed elastic boundaries, in: A. Y. Cheer, C. P. V. Dam (Eds.), *Fluid Dynamics in Biology: Proceedings of the AMS-IMS-SIAM Joint Summer Research Conference on Biofluidynamics*, American Mathematical Society, 1993, pp. 261–277.
- [39] T. Y. Hou, J. S. Lowengrub, M. J. Shelley, Removing the stiffness from interfacial flows with surface tension, *J. Comput. Phys.* 114 (1994) 312–338.
- [40] J. M. Stockie, B. R. Wetton, Analysis of stiffness in the immersed boundary method and implications for time-stepping schemes, *J. Comput. Phys.* 154 (1999) 41–64.
- [41] H. D. Cenicerros, A. M. Roma, Study of long-time dynamics of a viscous vortex sheet with a fully adaptive non-stiff method, *Phys. Fluids* 16 (2004) 4285–4318.
- [42] T. Y. Hou, J. S. Lowengrub, M. J. Shelley, The long-time motion of vortex sheets with surface tension, *Phys. Fluids* 9 (7) (1997) 1933–1954.
- [43] W. Tauber, S. O. Unverdi, G. Tryggvason, The nonlinear behavior of a sheared immiscible fluid interface, *Phys. Fluids* 14 (8) (2002) 2871–2885.
- [44] J. J. Douglas, T. Dupont, Alternating-direction galerkin methods on rectangles, in: B. Hubbard (Ed.), *SYNSPADE-1970, Numerical Solution of Partial Differential Equations-II*, Academic Press, New York, 1971, pp. 133–213.
- [45] D. Gottlieb, B. Gustafsson, Generalized Du Fort-Frankel methods for parabolic initial-boundary value problems, *SIAM J. Numer. Anal.* 13 (1) (1976) 129–144.
- [46] D. Gottlieb, S. A. Orszag, *Numerical Analysis of Spectral Methods: Theory and Applications*, CBMSNSF Regional conference series in applied mathematics, SIAM Press, 1977.
- [47] W. E, J.-G. Liu, Projection method I: convergence and numerical boundary layers, *SIAM J. Numer. Anal.* 32 (4) (1995) 1017–1057.

- [48] D. Brown, R. Cortez, M. Minion, Accurate projection methods for the incompressible navier-stokes equations, *J. Comput. Phys.* 168 (2001) 464–499.
- [49] K. Goda, A multistep technique with implicit difference schemes for calculating two- or three-dimensional cavity flows, *J. Comput. Phys.* 30 (1979) 76–95.
- [50] G. E. Karniadakis, M. Israeli, S. A. Orszag, High-order splitting methods for the incompressible navierstokes equations, *J. Comput. Phys.* 97 (1991) 414.
- [51] O. Botella, On the solution of the navierstokes equations using chebyshev projection schemes with thirdorder accuracy in time, *Computers & Fluids* 26 (1997) 107–116.
- [52] M. Azaiez, A spectral element projection scheme for incompressible flow with application to the unsteady axisymmetric stokes problem, *J. Sci. Comp.* 17 (1–4) (2002) 573–584.
- [53] H. S. Udaykumar, H.-C. Kan, W. Shyy, R. Tran-Son-Tay, Multiphase dynamics in arbitrary geometries on fixed cartesian grids, *J. Comput. Phys.* 137 (1997) 366–405.
- [54] G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nas, Y.-J. Jan, A front-tracking method for computations of multiphase flow, *J. Comput. Phys.* 169 (2001) 708–759.
- [55] M. J. Berger and P. Colella, Local adaptive mesh refinement for shock hydrodynamics, *J. Comput. Phys.* 82 (1989) 64–84.
- [56] M. J. Berger and I. Rigoutsos, An algorithm for point clustering and grid generation, *IEEE Transactions on Systems, Man, and Cybernetics* 21 (5) (1991) 1278–1286.
- [57] A. M. Roma, A multilevel self-adaptive version of the immersed boundary method, Ph.D. thesis, Courant Institute of Mathematical Sciences – New York University, University Microfilms # 9621828. (January 1996).
- [58] M. S. Longuet-Higgins, E. D. Cokelet, The deformation of steep surface waves on water I. A numerical method of computation, *Proc. R. Soc. Lond. A.* 350 (1976) 1–26.
- [59] H. D. Cenicerros, T. Y. Hou, An efficient dynamically adaptive mesh for potentially singular solutions, *J. Comput. Phys.* 172 (2001) 609–639.
- [60] M. Kang, R. Fedkiw, X.-D. Liu, A boundary condition capturing method for multiphase incompressible flow, *J. Sci. Computing* 15 (2000) 323–360.
- [61] C. Xu, T. Tang, Stability analysis of large time-stepping methods for epitaxial growth models, *SIAM J. Numer. Anal.* 44 (4) (2006) 1759–1779.
- [62] H. D. Cenicerros, T. Y. Hou, Convergence of a non-stiff boundary integral method for interfacial flows with surface tension, *Math. Comput.* 67 (1998) 137–182.
- [63] Z. Mohamed-Kassim, E. K. Longmire, Drop impact on a liquid-liquid interface, *Phys. Fluids* 15 (11) (2003) 3263–3273.
- [64] M. Sussman, K. M. Smith, M. Y. Hussaini, M. Ohta, R. Zhi-Wei, A sharp interface method for incompressible two-phase flows, *J. Comput. Phys.* 221 (2007) 469–505.
- [65] B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, G. Zanetti, Modeling merging and fragmentation in multiphase flows with SURFER, *J. Comput. Phys.* 113 (1994) 134–147.
- [66] D. Enright, R. Fedkiw, J. Ferziger, O. Mitchell, A hybrid particle level set method for improved interface capturing, *J. Comput. Phys.* 183 (2002) 83–116.
- [67] D. Enright, F. Losasso, R. Fedkiw, A fast and accurate semi-lagrangian particle level set method, *Computers and Structures* 83 (2005) 479–490.
- [68] Y. Sun, C. Berkmann, Sharp interface tracking using the phase-field equation, *J. Comput. Phys.* 220 (2007) 626–653.
- [69] C. Min, F. Gibou, A second order accurate level set method on non-graded adaptive cartesian grids, *J. Comput. Phys.* 225 (2007) 330–321.
- [70] J. Bell, P. Colella, H. Glaz, A second order projection method for the incompressible navier-

- stokesaccurate, *J. Comput. Phys.* 85.
- [71] L. Rayleigh, Investigation of the character of equilibrium of an incompressible heavy fluid of variable density, *Proceedings of the London Mathematical Society* 14 (1883) 170–177.
 - [72] G. I. Taylor, The instability of liquid surfaces when accelerated in a direction perpendicular to their planes, *Proceedings of the Royal Society of London, Series A, Mathematical and Physical Sciences* 201 (1950) 192–196.
 - [73] P. G. Drazin, W. H. Reid, *Hydrodynamic stability*, Cambridge monographs on mechanics and applied mathematics, Cambridge University Press, New York, 1981.
 - [74] B. J. Daly, Numerical study of two fluid Rayleigh-Taylor instability, *Phys. Fluids* 10 (2) (1967) 297–307.
 - [75] G. R. Baker, D. I. Meiron, S. A. Orszag, Vortex simulations of the Rayleigh-Taylor instability, *Phys. Fluids* 23 (1980) 1485–1490.
 - [76] D. I. Pullin, Numerical studies of surface-tension effects in nonlinear Kelvin-Helmholtz and Rayleigh-Taylor instability, *J. Fluid Mech.* 119 (1982) 507–532.
 - [77] G. R. Baker, D. I. Meiron, S. A. Orszag, Boundary integral methods for axisymmetric and three dimensional Rayleigh-Taylor instability problems, *Physica D* 12 (1984) 19–31.
 - [78] G. Tryggvason, Numerical simulations of the Rayleigh-Taylor instability, *J. Comput. Phys.* 75 (1988) 253–282.
 - [79] A. Elgowany, N. Ashgriz, Rayleigh-Taylor instability of viscous fluid layers, *Phys. Fluids* 9 (6) (1997) 1635–1649.
 - [80] X. He, R. Zhang, S. Chen, G. D. Doolen, On the three-dimensional Rayleigh-Taylor instability, *Phys. Fluids* 11 (5) (1999) 1143–1152.
 - [81] S. V. Weber, G. Dimonte, M. M. Marinak, Arbitrary Lagrange-Eulerian code simulations of turbulent Rayleigh-Taylor instability in two and three dimensions, *Laser and Particle Beams* 21 (2003) 455–461.
 - [82] S. M. Asida, E. Livne, J. Stein, L. Metzker, 3-D simulations of Rayleigh-Taylor instability using "VULCAN/3D", *Astrophysics and Space Science* 298 (2005) 363–367.
 - [83] M. Horikoshi, K. Nishihara, Vortex generation and deformation of the interface in three-dimensional Rayleigh-Taylor instability, *J. Phys. IV France* 133 (2006) 209–212.
 - [84] F. H. Harlow, J. E. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with a free surface, *Phys. Fluids* 8 (1965) 2182–2189.
 - [85] J. B. Bell, D. L. Marcus, A second-order projection method for variable density flows, *J. Comput. Phys.* 101 (1992) 334–348.
 - [86] J. T. Waddell, C. E. Niederhaus, J. W. Jacobs, Experimental study of Rayleigh-Taylor instability: low Atwood number liquid systems with single-mode initial perturbations, *Phys. Fluids* 13 (5) (2001) 1263–1273.
 - [87] T. Y. Hou, Z. Shi, An efficient semi-implicit immersed boundary method for the Navier-Stokes equations, *J. Comput. Phys.* 227 (2008) 8968–8991.
 - [88] R. Cliff, J. R. Grace, M. E. Webber, *Bubbles, Drops, and Particles*, Academic Press, San Diego, 1978.
 - [89] D. Bhaga, M. E. Weber, Bubbles in viscous liquids: shapes, wakes and velocities, *J. Fluid Mech.* 105 (1981) 61–85.
 - [90] A. Esmaeeli, G. Tryggvason, Direct numerical simulations of bubbly flows. Part 1. Low Reynolds number arrays, *J. Fluid Mech.* 377 (1998) 313–345.