A fast and practical adaptive finite difference method for the conservative Allen–Cahn model in two-phase flow system

Junxiang Yang[a], Darae Jeong[b], Junseok Kim[a,*]

[a] *Department of Mathematics, Korea University, Seoul, 02841, Republic of Korea*
[b] *Department of Mathematics, Kangwon National University, Gangwon-do 24341, Republic of Korea*

ARTICLE INFO

ABSTRACT

We present a simple and practical adaptive finite difference method for the conservative Allen–Cahn–Navier–Stokes system. For the conservative Allen–Cahn equation, we use a temporally adaptive narrow band domain embedded in the uniform discrete rectangular domain. The narrow band domain is defined as a neighboring region of the interface. The Navier–Stokes equation is solved in a fully discrete domain with the coarse grid than that for the CAC equation. Various benchmark numerical experiments, such as the pressure jump, droplet deformation in shear flow, falling droplet, and rising bubble, are performed to show that the proposed method is efficient and practical for the simulations of two-phase incompressible flow.

© 2021 Elsevier Ltd. All rights reserved.

1. Introduction

Numerical simulation of two-phase flow is interesting and important because it can be applied to various natural and technological fields. In the natural world, the droplet evaporation (Karami et al., 2017; Aguilar et al., 2014), the droplet impacting thin film (Chen et al., 2017; Burzynski and Bansmer, 2018), and the antibubble formation (Scheid et al., 2012), etc., are the classical phenomena related to the two-phase flow. In industries, some typical applications of two-phase flow are the double emulsion formation in a capillary device (Vian et al., 2018), droplet deformation in shear flow (Vu et al., 2019), pinch-off of liquid jet (Webster and Longmire, 2001), and buoyancy-driven mixing in a tilted channel (Sahu and Vanka, 2011), etc.

In the numerical study of two-phase flow, the most important issue is to accurately capture the interface between two fluid components. The volume of fluid (VOF) method is an extensively used method and it is popular in engineering field. A color function is used to distinguish various two-phase variables, such as the density ad viscosity. The change of interface is captured by the evolution of color function and mass conservation can be well guaranteed. However, the VOF method has sharp discontinuity between two fluids which usually needs an artificial reconstruction to accurately calculate the continuous surface tension. Some typical applications of the VOF method can be found in (Theodorakakos and Bergeles, 2004; Hong et al., 2013; Yan and Chen, 2020). Another useful method focusing on the interface capturing is the level-set method. The signed distance function is used to distinguish the interface and calculate the curvature. Because of the practicability of the level-set method, it has also been widely used in engineering community (Rodríguez, 2017; Tanguy and Berlemont, 2005; Balcazar et al., 2015). However, a reinitialization step is needed in each iteration and this method suffers an obvious problem of mass loss. Recently, some researchers have studied the conservative level-set method to satisfy the mass conservation (Luo et al., 2015). Different from the VOF and the level-set methods mentioned above, the immersed boundary method (IBM) is an efficient interface tracking method, which uses a series of Lagrangian points to track the position of interface and solves the flow field on the fixed Eulerian grid. The IBM method is a popular method to treat various solid-fluid interaction problems, see (Gronskis and Artana, 2016; Horng et al., 2018). In the liquid-liquid or gas-liquid two-phase flow simulations, the classical IBM method can not guarantee the mass conservation due to the violated divergence-free condition at the Lagrangian grid of the immersed boundary point. To achieve the mass conservation, some numerical corrections have been proposed, see (Li et al., 2012; 2013). Moreover, the lattice Boltzmann method (LBM) (Chen and Müller, 2020; Mitchell et al., 2018; Majidi et al., 2020) and front tracking method (FTM) (Hua and Mortensen, 2019; Pivello et al., 2014) have been extensively applied in two-phase fluid simulations.

---

* Corresponding author.
E-mail address: cfdkim@korea.ac.kr (J. Kim).
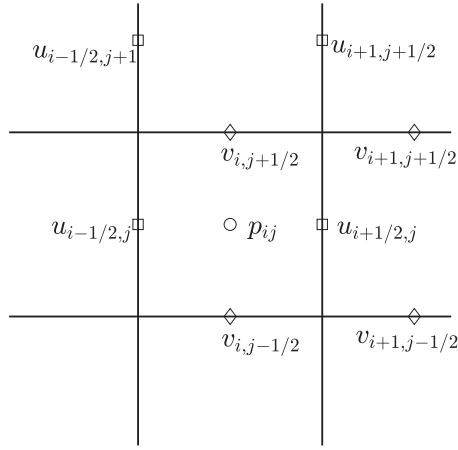*URL:* http://math.korea.ac.kr/~cfdkim (J. Kim)

Fig. 1. Schematic illustration of the MAC mesh.

In this study, we focus on the phase-field method. The change of interface is captured by the evolution of the phase-field equation. Besides that, a small and finite interface width is used as the transition layer between two fluid components. The most popular phase-field model for two-phase flow is the Cahn–Hilliard–Navier–Stokes (CHNS) system, which guarantees the basic property of mass conservation and has the physical property of energy dissipation. The surface tension can be easily imposed using the phase-field variable. For some practical applications of CHNS system in multi-phase flow simulations, please refer to (Kim, 2012; Yang and Kim, 2018; Shen et al., 2019; Bai et al., 2017). However, the classical CH equation is a fourth-order nonlinear partial differential equation which is hard to solve in an explicit manner. Another effective and conservative phase-field model is the conservative Allen–Cahn (CAC) equation which is a second-order partial differential equation and easy to solve even if the explicit scheme is used. Although the CAC equation does not possess the property of energy dissipation in theory, it still works well in capturing two-phase interface. As some typical applications, Jeong and Kim (2017) proposed the conservative Allen–Cahn–Navier–Stokes (CACNS) model for two-phase incompressible fluid flow. Joshi and Jaiman (2018) developed

an adaptive procedure with unstructured meshes for capturing the CAC fluid interfaces. Aihara et al. (2019) extended the CACNS model to simulate various multi-component fluid flows. Khan and Shah (2019) used the CAC model to numerically investigated the dynamics of Rayleigh–Taylor instability.

Since the accurate interfacial capturing is an important issue in simulation, a natural and time-saving practice is to use finer mesh size in the neighborhood of interface and use coarser mesh size in the bulk phase region. However, most existing adaptive mesh techniques are complex and not easy to implement unless one is familiar with each specific methodology. Although the classic adaptive mesh size method can be well applied to simulate the CH equation (Li et al., 2016), thin film equation (Li et al., 2014), and dendritic growth equation (Li and Kim, 2012), etc., it is still complex to apply to the two-phase fluid flows. The main advantage of the proposed method compared with the classic adaptive mesh refinement (AMR) (Li et al., 2016; 2014; Li and Kim, 2012) is its simplicity in programming because the classic AMR is very heavy in a sense that the data structure is complex and code itself is very lengthy; in particular, implementation of the phase-field model with the Navier–Stokes equation is only available to few researchers who are experts in the AMR.

In this work, we propose a simple, efficient and accurate adaptive finite difference method for the following incompressible CACNS system (Jeong and Kim, 2017) in the domain Ω:

$$\nabla \cdot \mathbf{u} = 0, \tag{1}$$

$$\rho(\phi)(\mathbf{u}_t + \mathbf{u} \cdot \nabla \mathbf{u}) = -\nabla p + \frac{1}{Re}\nabla \cdot \left[\eta(\phi)(\nabla \mathbf{u} + \nabla \mathbf{u}^T)\right] + \mathbf{SF}(\phi)$$
$$+ \frac{\rho(\phi)}{Fr^2}\mathbf{g}, \tag{2}$$

$$\phi_t + \nabla \cdot (\phi \mathbf{u}) = \frac{1}{Pe}\left(-F'(\phi) + \epsilon^2 \Delta \phi\right) + \gamma(t)\sqrt{F(\phi)}, \tag{3}$$

where $\mathbf{u}$ is the velocity filed, p is the pressure, ϕ is the phase field, $F(\phi) = 0.25(\phi^2 - 1)^2$, $\gamma(t) = \int_\Omega F'(\phi)d\mathbf{x}/[Pe\int_\Omega \sqrt{F(\phi)}d\mathbf{x}]$ is the time-dependent Lagrange multiplier, $\rho(\phi)$ and $\eta(\phi)$ are the density and viscosity, respectively, and $\mathbf{g}$ represents the gravitational
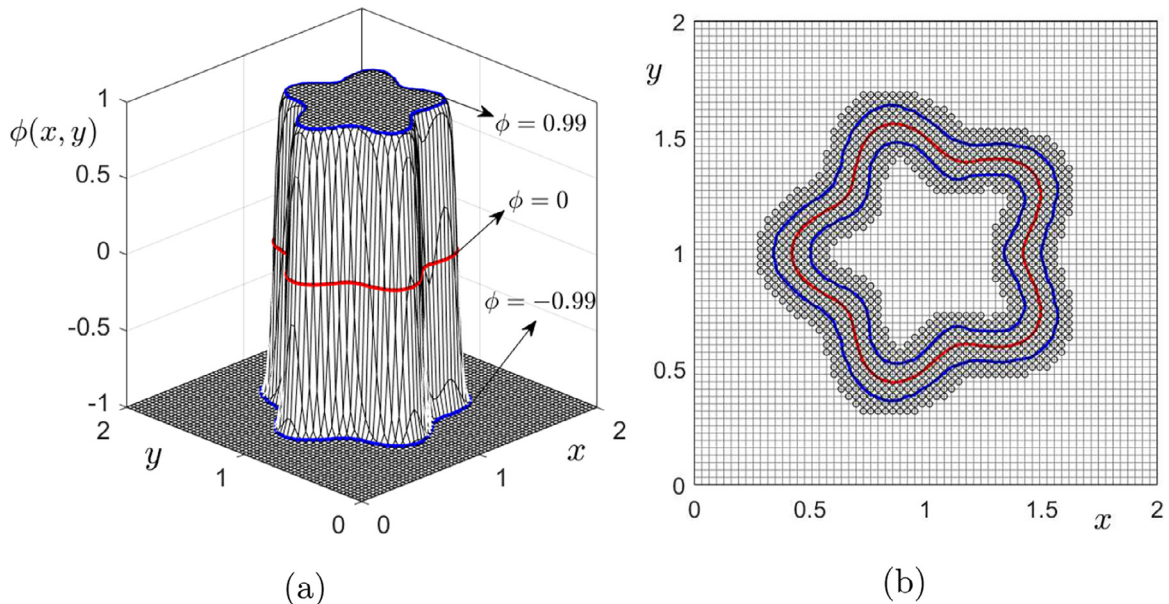

(a)   (b)

Fig. 2. Schematic illustration of (a) ϕ and (b) narrow band domain Ω_a (open circles) and the contour line by a given zero-level set of ϕ (red line). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

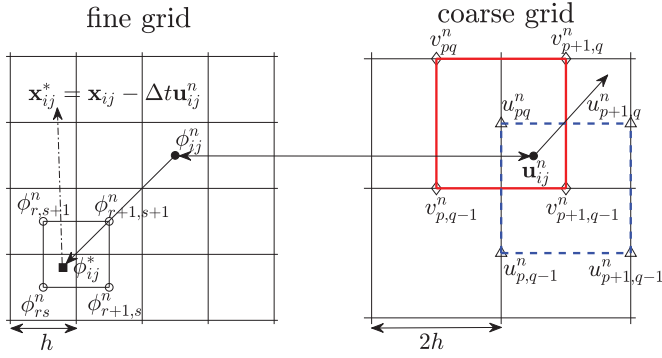fine grid   coarse grid

Fig. 3. Schematic illustration for the computation of ϕ_{ij}^* in Step 1.

acceleration. To close the system (1)–(3), the periodic boundary condition or the following boundary conditions are used

$$\nabla \phi \cdot \mathbf{n} = \nabla p \cdot \mathbf{n} = \mathbf{u} \cdot \mathbf{n} = 0 \text{ on } \partial \Omega, \tag{4}$$

where $\mathbf{n}$ is the outward unit normal vector on the boundary $\partial \Omega$. We find that Eq. (3) becomes to the classical AC equation if we drop the last term and it is well-known that the AC equation does not satisfy the mass conservation. The continuous surface tension is expressed as:

$$\mathbf{SF}(\phi) = -\frac{1}{2We} \nabla \cdot \left(\frac{\nabla \phi}{|\nabla \phi|} \right) \nabla \phi = \frac{1}{We} \nabla \cdot \left(-\frac{\nabla \phi}{|\nabla \phi|} \right) \frac{|\nabla \phi|}{2} \frac{\nabla \phi}{|\nabla \phi|}, \tag{5}$$

where $\nabla \cdot (-\nabla \phi / |\nabla \phi|)$ is the curvature, $|\nabla \phi|/2$ is the smoothed Dirac delta function (Lee and Kim, 2012), $\nabla \phi / |\nabla \phi|$ is the unit normal vector to the interface. Various dimensionless parameters are the Reynolds number: $Re = \rho_c U_c L_c / \eta_c$, the Weber number: $We = \rho_c L_c U_c^2 / \sigma$, the Froude number $Fr = U_c / \sqrt{gL_c}$, and the Peclet number: $Pe = U_c L_c / (M \mu_c)$. Here, L_c, U_c, ρ_c, η_c, σ, M, and μ_c are the characteristic length, velocity, density, viscosity, surface tension coefficient, constant mobility, and chemical potential, respectively (Kim, 2012).

The main purpose of this work is to develop a temporally adaptive narrow band domain around the interface to calculate the binary CAC equation. The discrete CAC equation is solved by using a fast explicit operator splitting method. The NS equation is solved in a discrete domain with coarser grid than that for the CAC equation. Therefore, the proposed method is fast. Different from the classic adaptive mesh grid methods, the proposed method is easy to implement without complex techniques. Another primary advantage of the proposed method is that it can be easily extended to efficiently solve an arbitrary multi-component incompressible fluid system.

The contents of this article are as follows. In Section 2, we present the proposed numerical scheme and its solution algorithm for the CACNS system. We present numerical results using the proposed method in Section 3. Conclusions are drawn in Section 4.

2. Numerical method

We now describe the proposed adaptive finite difference method for Eqs. (1)–(3) on the two-dimensional computational domain $\Omega = (a, b) \times (c, d)$. The extension to three-dimensional space is straightforward. First, the discrete global domain for the CAC equation is defined as: $\Omega_h = \{(x_i, y_j) | x_i = a + (i - 0.5)h, y_j = c + (j - 0.5)h$ for $1 \le i \le N_x, 1 \le j \le N_y\}$ with respect to an uniform spatial mesh size $h = (b - a)/N_x = (d - c)/N_y$, where N_x and N_y are even positive integers. We let ϕ_{ij}^n indicate the approximation of the solution $\phi(x_i, y_j, n\Delta t)$ of Eq. (3), where Δt is the time step. Next, we define the discrete global domain for the NS equation: $\Omega_{\tilde{h}} = \{(\tilde{x}_i, \tilde{y}_j) | \tilde{x}_i = a + (i - 0.5)\tilde{h}, \tilde{y}_j = c + (j - 0.5)\tilde{h}$ for $1 \le$

$i \le N_x/2, 1 \le j \le N_y/2\}$ according to a uniform spatial mesh size $\tilde{h} = 2h = 2(b - a)/N_x = 2(d - c)/N_y$, which means a coarser grid is used for the NS equation. Then, we let p_{ij}^n be the approximation of the solution $p(\tilde{x}_i, \tilde{y}_j, n\Delta t)$. The velocities $u_{i+\frac{1}{2}, j}$ and $v_{i, j+\frac{1}{2}}$ are stored at the cell edges $(\tilde{x}_{i+\frac{1}{2}, j}, \tilde{y}_j) = (a + i\tilde{h}, c + (j - 0.5)\tilde{h})$ and $(\tilde{x}_i, \tilde{y}_{j+\frac{1}{2}}) = (a + (i - 0.5)\tilde{h}, c + j\tilde{h})$, respectively. We use the staggered marker-and-cell (MAC) mesh (Harlow and Welch, 1966) (see, Fig. 1), where the pressure is stored at the cell centers and the velocities are stored at cell edges.

Now, we briefly describe the numerical solution of the NS equation in one time step on the discrete global domain $\Omega_{\tilde{h}}$. For the given values of $\mathbf{u}^n = (u^n, v^n)$ and ϕ^n, we want to find $\mathbf{u}^{n+1}$ and p^{n+1}.

Step 1. Compute the intermediate velocity field $\hat{\mathbf{u}}$:

$$\rho(\tilde{\phi}^n) \left(\frac{\hat{\mathbf{u}} - \mathbf{u}^n}{\Delta t} \right) = \frac{1}{Re} \nabla \cdot \left[\eta(\tilde{\phi}^n) (\nabla_d \mathbf{u} + \nabla_d \mathbf{u}^T)^n \right]$$
$$+ \mathbf{SF}(\tilde{\phi}^n) + \frac{\rho(\tilde{\phi}^n)}{Fr^2} \mathbf{g} - \rho(\tilde{\phi}^n)(\mathbf{u}^n \cdot \nabla_d \mathbf{u}^n), \tag{6}$$

where the subscript d represents the discrete operator. $\rho(\tilde{\phi}) = \rho_1 (1 + \tilde{\phi})/2 + \rho_2 (1 - \tilde{\phi})/2$, $\eta(\tilde{\phi}) = \eta_1 (1 + \tilde{\phi})/2 + \eta_2 (1 - \tilde{\phi})/2$. Here, ρ_1 and ρ_2 are the densities of fluid 1 and fluid 2, respectively. η_1 and η_2 are the viscosities of fluid 1 and fluid 2, respectively. The gravitational acceleration $\mathbf{g} = (0, -1)$. The phase-field in a coarser grid is defined as $\tilde{\phi}_{ij} = 0.25(\phi_{2i-1,2j-1} + \phi_{2i,2j-1} + \phi_{2i-1,2j} + \phi_{2i,2j})$, where $1 \le i \le N_x/2, 1 \le j \le N_y/2$. The discretizations of Eq. (6) take the following forms:

$$\hat{u}_{i+\frac{1}{2}, j} = u_{i+\frac{1}{2}, j}^n - \Delta t(uu_x + vu_y)_{i+\frac{1}{2}, j}^n + \frac{\Delta t \mathbf{SF}_x(\tilde{\phi}^n)_{i+\frac{1}{2}, j}}{0.5(\rho(\tilde{\phi}_{i+1,j}^n) + \rho(\tilde{\phi}_{i,j}^n))}$$

$$+ \frac{\Delta t \left[2 \left(\eta(\tilde{\phi}_{i+1,j}^n)(u_{i+\frac{3}{2}, j}^n - u_{i+\frac{1}{2}, j}^n) - \eta(\tilde{\phi}_{i,j}^n)(u_{i+\frac{1}{2}, j}^n - u_{i-\frac{1}{2}, j}^n) \right) \right]}{0.5\tilde{h}^2 Re(\rho(\tilde{\phi}_{i+1,j}^n) + \rho(\tilde{\phi}_{i,j}^n))}$$

$$+ \frac{\Delta t \left[\eta_a(u_{i+\frac{1}{2}, j+1}^n - u_{i+\frac{1}{2}, j}^n) - \eta_b(u_{i+\frac{1}{2}, j}^n - u_{i+\frac{1}{2}, j-1}^n) \right]}{0.5\tilde{h}^2 Re(\rho(\tilde{\phi}_{i+1,j}^n) + \rho(\tilde{\phi}_{i,j}^n))}$$

$$+ \frac{\Delta t \left[\eta_a(v_{i+1, j+\frac{1}{2}}^n - v_{i, j+\frac{1}{2}}^n) - \eta_b(v_{i+1, j-\frac{1}{2}}^n - v_{i, j-\frac{1}{2}}^n) \right]}{0.5\tilde{h}^2 Re(\rho(\tilde{\phi}_{i+1,j}^n) + \rho(\tilde{\phi}_{i,j}^n))}, \tag{7}$$

$$\hat{v}_{i, j+\frac{1}{2}} = v_{i, j+\frac{1}{2}}^n - \Delta t(uv_x + vv_y)_{i, j+\frac{1}{2}}^n + \frac{\Delta t \mathbf{SF}_y(\tilde{\phi}^n)_{i, j+\frac{1}{2}}}{0.5(\rho(\tilde{\phi}_{i,j+1}^n) + \rho(\tilde{\phi}_{i,j}^n))} - \frac{\Delta t}{Fr^2}$$

$$+ \frac{\Delta t \left[2 \left(\eta(\tilde{\phi}_{i,j+1}^n)(v_{i, j+\frac{3}{2}}^n - v_{i, j+\frac{1}{2}}^n) - \eta(\tilde{\phi}_{i,j}^n)(v_{i, j+\frac{1}{2}}^n - v_{i, j-\frac{1}{2}}^n) \right) \right]}{0.5\tilde{h}^2 Re(\rho(\tilde{\phi}_{i,j+1}^n) + \rho(\tilde{\phi}_{i,j}^n))}$$

$$+ \frac{\Delta t \left[\eta_c(v_{i+1, j+\frac{1}{2}}^n - v_{i, j+\frac{1}{2}}^n) - \eta_d(v_{i, j+\frac{1}{2}}^n - v_{i-1, j+\frac{1}{2}}^n) \right]}{0.5\tilde{h}^2 Re(\rho(\tilde{\phi}_{i,j+1}^n) + \rho(\tilde{\phi}_{i,j}^n))}$$

$$+ \frac{\Delta t \left[\eta_c(u_{i+\frac{1}{2}, j+1}^n - u_{i+\frac{1}{2}, j}^n) - \eta_d(u_{i-\frac{1}{2}, j+1}^n - u_{i-\frac{1}{2}, j}^n) \right]}{0.5\tilde{h}^2 Re(\rho(\tilde{\phi}_{i,j+1}^n) + \rho(\tilde{\phi}_{i,j}^n))}, \tag{8}$$

where $\mathbf{SF}_x$ and $\mathbf{SF}_x$ are the components of $\mathbf{SF}$ along x- and y-directions, respectively. The definitions of η_a, η_b, η_c, and η_d are as follows:

$$\eta_a = \eta_c = 0.25 \left(\eta(\tilde{\phi}_{i,j}^n) + \eta(\tilde{\phi}_{i+1,j}^n) + \eta(\tilde{\phi}_{i,j+1}^n) + \eta(\tilde{\phi}_{i+1,j+1}^n) \right),$$
$$\eta_b = 0.25 \left(\eta(\tilde{\phi}_{i,j}^n) + \eta(\tilde{\phi}_{i+1,j}^n) + \eta(\tilde{\phi}_{i,j-1}^n) + \eta(\tilde{\phi}_{i+1,j-1}^n) \right),$$
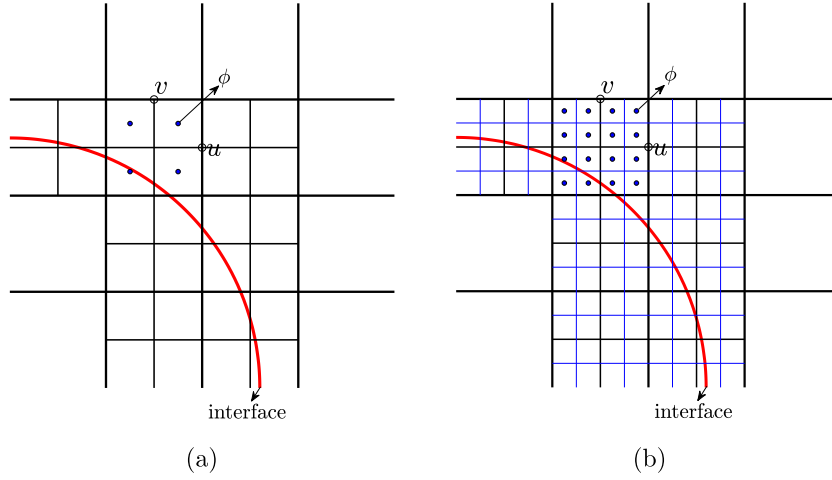
Fig. 4. Schematic illustration of refinement of narrow band domain: (a) before refinement, (b) after refinement.
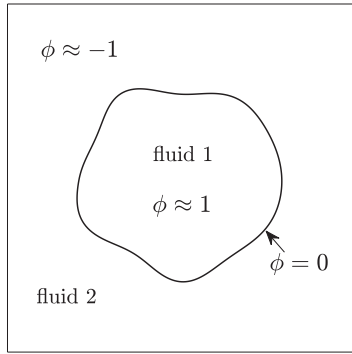

Fig. 5. Schematic illustration of a two-phase flow system, where the interface is represented by the zero level set of phase-field function, i.e., $\phi = 0$.

$$\eta_d = 0.25\big(\eta(\tilde{\phi}_{i,j}^n) + \eta(\tilde{\phi}_{i-1,j}^n) + \eta(\tilde{\phi}_{i,j+1}^n) + \eta(\tilde{\phi}_{i-1,j+1}^n)\big).$$

The following upwind scheme is used to treat the advection terms in Eqs. (7) and (8)

$$(uu_x + vu_y)_{i+\frac{1}{2},j}^n = u_{i+\frac{1}{2},j}^n \bar{u}_{x_{i+\frac{1}{2},j}}^n + \frac{v_{i,j-\frac{1}{2}}^n + v_{i+1,j-\frac{1}{2}}^n + v_{i,j+\frac{1}{2}}^n + v_{i+1,j+\frac{1}{2}}^n}{4} \bar{u}_{y_{i+\frac{1}{2},j}}^n,$$

$$(uv_x + vv_y)_{i,j+\frac{1}{2}}^n = \frac{u_{i-\frac{1}{2},j}^n + u_{i-\frac{1}{2},j+1}^n + u_{i+\frac{1}{2},j}^n + u_{i+\frac{1}{2},j+1}^n}{4} \bar{v}_{x_{i,j+\frac{1}{2}}}^n + v_{i,j+\frac{1}{2}}^n \bar{v}_{y_{i,j+\frac{1}{2}}}^n,$$

where $\bar{u}_{x_{i+\frac{1}{2},j}}^n$ and $\bar{u}_{y_{i+\frac{1}{2},j}}^n$ are computed as follows

$$\bar{u}_{x_{i+\frac{1}{2},j}}^n = \begin{cases} \frac{u_{i+\frac{1}{2},j}^n - u_{i-\frac{1}{2},j}^n}{\tilde{h}} & \text{if } u_{i+\frac{1}{2},j}^n > 0, \\ \frac{u_{i+\frac{3}{2},j}^n - u_{i+\frac{1}{2},j}^n}{\tilde{h}} & \text{otherwise.} \end{cases}$$

and

$$\bar{u}_{y_{i+\frac{1}{2},j}}^n = \begin{cases} \frac{u_{i+\frac{1}{2},j}^n - u_{i+\frac{1}{2},j-1}^n}{\tilde{h}} & \text{if } v_{i,j-\frac{1}{2}}^n + v_{i+1,j-\frac{1}{2}}^n + v_{i,j+\frac{1}{2}}^n + v_{i+1,j+\frac{1}{2}}^n > 0, \\ \frac{u_{i+\frac{1}{2},j+1}^n - u_{i+\frac{1}{2},j}^n}{\tilde{h}} & \text{otherwise.} \end{cases}$$

Other terms are similarly computed. From the definition of upwind scheme, we can observe that it has first-order spatial accuracy. The spatial discretization of surface tension force (Kim, 2012) is defined

to be

$$\mathbf{SF}(\tilde{\phi}_{ij}) = -\frac{1}{2We} \nabla_d \cdot \left(\frac{\mathbf{n}}{|\mathbf{n}|}\right)_{ij} \nabla_d \tilde{\phi}_{ij}. \tag{9}$$

Here, the normal vector at the top right vertex $(\tilde{x}_{i+\frac{1}{2}}, \tilde{y}_{j+\frac{1}{2}}) = (i\tilde{h}, j\tilde{h})$ is given by

$$\mathbf{n}_{i+\frac{1}{2},j+\frac{1}{2}} = \left(n_{i+\frac{1}{2},j+\frac{1}{2}}^x, n_{i+\frac{1}{2},j+\frac{1}{2}}^y\right)$$
$$= \left(\frac{\tilde{\phi}_{i+1,j+1} + \tilde{\phi}_{i+1,j} - \tilde{\phi}_{i,j+1} - \tilde{\phi}_{ij}}{2\tilde{h}}, \frac{\tilde{\phi}_{i+1,j+1} + \tilde{\phi}_{i,j+1} - \tilde{\phi}_{i+1,j} - \tilde{\phi}_{ij}}{2\tilde{h}}\right).$$

The other normal vectors can be defined in a same manner. Then, the curvature term at cell centers can be defined to be

$$\nabla_d \cdot \left(\frac{\mathbf{n}}{|\mathbf{n}|}\right)_{ij} = \frac{1}{2\tilde{h}}\left(\frac{n_{i+\frac{1}{2},j+\frac{1}{2}}^x + n_{i+\frac{1}{2},j+\frac{1}{2}}^y}{|\mathbf{n}_{i+\frac{1}{2},j+\frac{1}{2}}|} + \frac{n_{i+\frac{1}{2},j-\frac{1}{2}}^x - n_{i+\frac{1}{2},j-\frac{1}{2}}^y}{|\mathbf{n}_{i+\frac{1}{2},j-\frac{1}{2}}|}\right.$$
$$\left. - \frac{n_{i-\frac{1}{2},j+\frac{1}{2}}^x - n_{i-\frac{1}{2},j+\frac{1}{2}}^y}{|\mathbf{n}_{i-\frac{1}{2},j+\frac{1}{2}}|} - \frac{n_{i-\frac{1}{2},j-\frac{1}{2}}^x + n_{i-\frac{1}{2},j-\frac{1}{2}}^y}{|\mathbf{n}_{i-\frac{1}{2},j-\frac{1}{2}}|}\right).$$

The discrete gradient at cell centers is given by

$$\nabla_d \tilde{\phi}_{ij} = \left(\mathbf{n}_{i+\frac{1}{2},j+\frac{1}{2}} + \mathbf{n}_{i+\frac{1}{2},j-\frac{1}{2}} + \mathbf{n}_{i-\frac{1}{2},j+\frac{1}{2}} + \mathbf{n}_{i-\frac{1}{2},j-\frac{1}{2}}\right)/4.$$

Step 2. We solve the following Eqs. (10) and (11) for the pressure at $(n + 1)$ time level.

$$\frac{\mathbf{u}^{n+1} - \hat{\mathbf{u}}}{\Delta t} = -\frac{1}{\rho^n}\nabla_d p^{n+1}, \tag{10}$$

$$\nabla_d \cdot \mathbf{u}^{n+1} = 0. \tag{11}$$

Here, we use $\rho^n = \rho(\tilde{\phi}^n)$ for convenience. Taking the discrete divergence operator to Eq. (10) and using Eq. (11), we can update the pressure field by solving the following Poisson equation. Here, a multigrid algorithm is used (Kim, 2012).

$$\nabla_d \cdot \left(\frac{1}{\rho^n}\nabla_d p^{n+1}\right) = \frac{1}{\Delta t}\nabla_d \cdot \hat{\mathbf{u}}^n. \tag{12}$$

Here, the discretizations of terms are defined as follows:

$$\nabla_d \cdot \left(\frac{1}{\rho^n}\nabla_d p_{ij}^{n+1}\right) = \frac{\frac{1}{\rho_{i+\frac{1}{2},j}^n}p_{i+1,j}^{n+1} + \frac{1}{\rho_{i-\frac{1}{2},j}^n}p_{i-1,j}^{n+1} + \frac{1}{\rho_{i,j+\frac{1}{2}}^n}p_{i,j+1}^{n+1} + \frac{1}{\rho_{i,j-\frac{1}{2}}^n}p_{i,j-1}^{n+1}}{\tilde{h}^2}$$
$$- \frac{\frac{1}{\rho_{i+\frac{1}{2},j}^n} + \frac{1}{\rho_{i-\frac{1}{2},j}^n} + \frac{1}{\rho_{i,j+\frac{1}{2}}^n} + \frac{1}{\rho_{i,j-\frac{1}{2}}^n}}{\tilde{h}^2}p_{ij}^{n+1}, \tag{13}$$
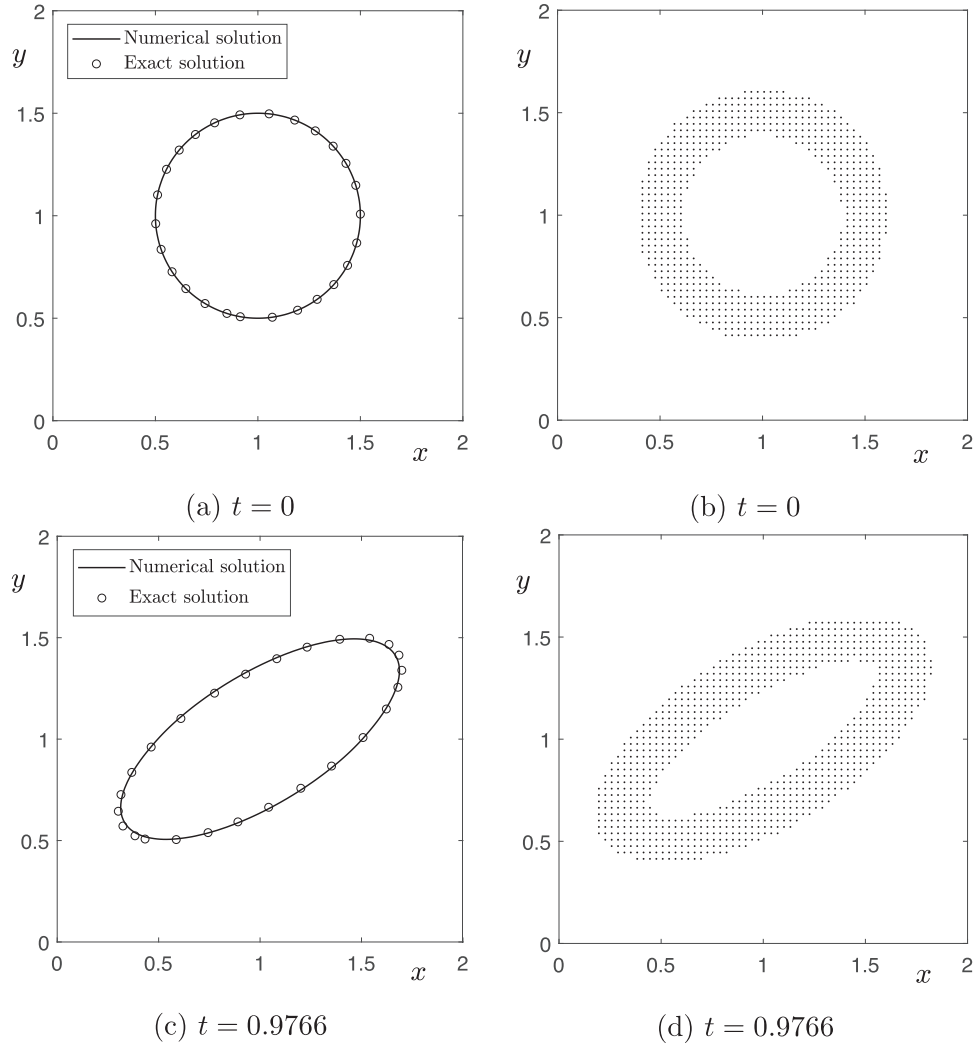
(a) $t = 0$

(b) $t = 0$

(c) $t = 0.9766$

(d) $t = 0.9766$

Fig. 6. Temporal evolutions of the interface and its adaptive domain Ω_a in a background flow.

$$\nabla_d \cdot \hat{\mathbf{u}}_{ij}^n = \frac{\hat{u}_{i+\frac{1}{2},j}^n - \hat{u}_{i-\frac{1}{2},j}^n}{\tilde{h}} + \frac{\hat{v}_{i,j+\frac{1}{2}}^n - \hat{v}_{i,j-\frac{1}{2}}^n}{\tilde{h}}, \tag{14}$$

where $\rho_{i+\frac{1}{2},j}^n = 0.5(\rho_{ij}^n + \rho_{i+1,j}^n)$ and other terms are similarly defined.

Step 3. Update the velocity field which satisfies the divergence-free condition.

$$\mathbf{u}^{n+1} = \hat{\mathbf{u}} - \frac{\Delta t}{\rho(\tilde{\phi}^n)} \nabla_d p^{n+1}. \tag{15}$$

The steps $1 - 3$ complete the solution of the NS equation in one time step. Next, we describe the numerical solution of the CAC equation in an adaptive narrow band domain. More details about the numerical solution of the NS equation can be found in (Kim, 2012). As have been mentioned, ϕ is the order parameter having the value between -1 and 1 as shown in Fig. 2(a). We define the interface of two phases as the zero-level set of ϕ. Using these properties of ϕ, we define new computational domain.

For given $\beta > 0$ and integer $m_p > 0$, we introduce the following time-dependent adaptive narrow band domain Ω_a as

$$\Omega_a = \left\{ (x_{i+k}, y_{j+l}) \,\middle|\, |\phi_{ij}^n| \leq \beta \ \text{ for } \ \forall i, j \text{ and } -m_p \leq k, l \leq m_p \right\}.$$

Here, the narrow band domain Ω_a is embedded in the discrete computational domain Ω_h. From the definition above, we can
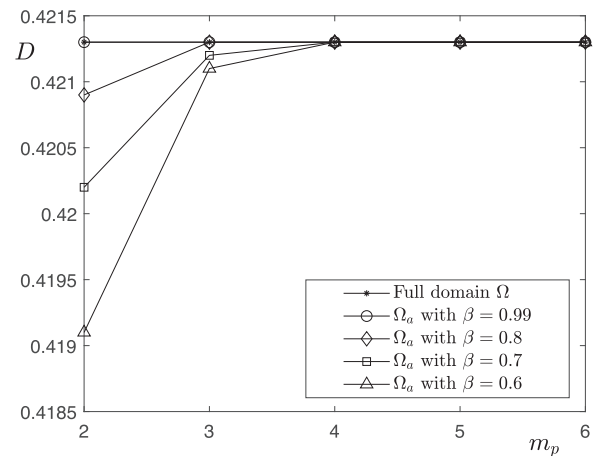

Fig. 7. Comparison of the deformation parameter D of a droplet with respect to different values of β and m_p.

find that the narrow band domain adaptively updates with the evolution of ϕ. Fig. 2(b) illustrates the narrow band domain Ω_a which is marked by the open circles. Note this adaptive narrow band approach has been successfully applied to two-phase physical
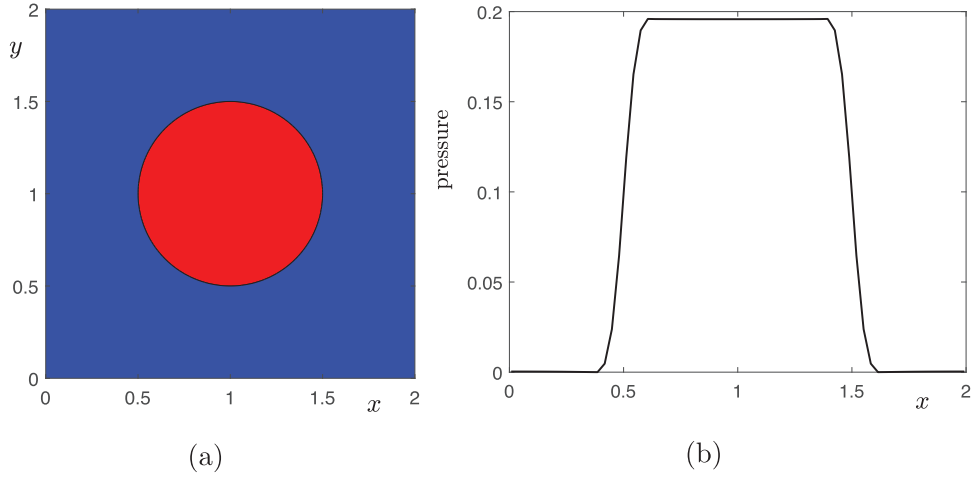
Fig. 8. Pressure jump test. (a) Initial condition. (b) Cross section of the pressure at $y = 1$.
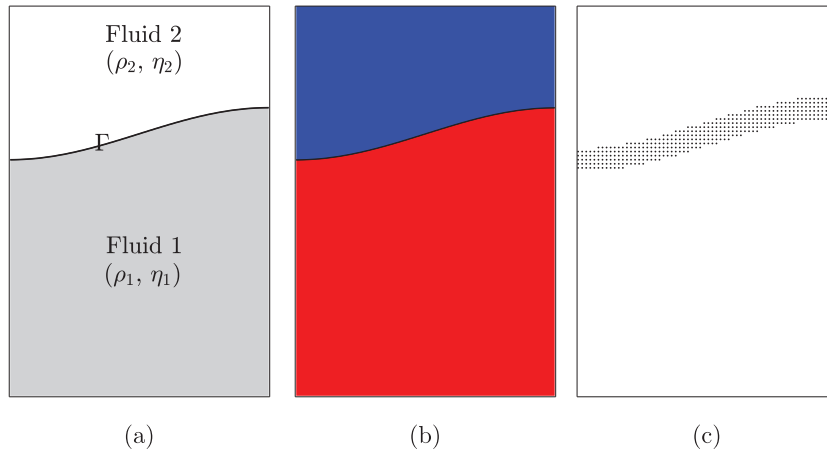

Fig. 9. Local views of two-phase flow in a sloshing tank. (a) schematic diagram, (b) initial state, (c) initial narrow band domain.

problem with interface, such as the dendritic growth (Jeong and Kim, 2019).

Now, we solve the governing Eq. (3) on the narrow band domain Ω_a. First, we initialize ϕ^0 on the computational domain Ω_h. Then, we define the narrow band domain Ω_a with the given ϕ^n. By taking the operator splitting method (Kim et al., 2014), Eq. (3) is split into three parts:

$$\frac{\phi^* - \phi^n}{\Delta t} = \nabla_d \cdot (\phi \mathbf{u})^n, \tag{16}$$

$$\frac{\phi^{**} - \phi^*}{\Delta t} = \frac{1}{Pe}\left(-F'(\phi^*) + \epsilon^2 \Delta_d \phi^*\right), \tag{17}$$

$$\frac{\phi^{n+1} - \phi^{**}}{\Delta t} = \gamma(t)\sqrt{F(\phi^{**})}. \tag{18}$$

To solve Eq. (16), we first find the velocity of ϕ_{ij} at the position (x_i, y_j) by using the bilinear interpolation on the coarse grid $\Omega_{\tilde{h}}$. The schematic illustration of this process is shown in Fig. 3. The convection Eq. (16) is solved by using a similar idea in "stable fluids" method of Stam (1999). To compute the convective value ϕ^*_{ij} of ϕ^n at (x_i, y_j), a new position $\mathbf{x}^*_{ij} = \mathbf{x}_{ij} - \Delta t \mathbf{u}^n_{ij}$ is first calculated, then the ϕ^*_{ij} is obtained by the bilinear interpolation on the fine grid Ω_h, see Fig. 3.

Next, we solve Eq. (17) by using the following explicit scheme:

$$\frac{\phi^{**}_{ij} - \phi^*_{ij}}{\Delta t} = \frac{1}{Pe}\left(-F'(\phi^*_{ij}) + \epsilon^2 \Delta_d \phi^*_{ij}\right), \tag{19}$$

where the discrete Laplacian operator takes the standard five-point discretization $\Delta_d \phi_{ij} = (\phi_{i+1,j} + \phi_{i-1,j} + \phi_{i,j+1} + \phi_{i,j-1} - 4\phi_{ij})/h^2$.

Finally, Eq. (18) is discretized as:

$$\frac{\phi^{n+1}_{ij} - \phi^{**}_{ij}}{\Delta t} = \gamma^{**}\sqrt{F(\phi^{**}_{ij})}. \tag{20}$$

Using Eq. (20) and the mass conservation, we have

$$\sum_{\mathbf{x}_{ij} \in \Omega_a} \phi^0_{ij} = \sum_{\mathbf{x}_{ij} \in \Omega_a} \phi^{n+1}_{ij} = \sum_{\mathbf{x}_{ij} \in \Omega_a} \left(\phi^{**}_{ij} + \Delta t \gamma^{**}\sqrt{F(\phi^{**}_{ij})}\right). \tag{21}$$

Thus,

$$\gamma^{**} = \frac{1}{\Delta t} \sum_{\mathbf{x}_{ij} \in \Omega_a} \left(\phi^0_{ij} - \phi^{**}_{ij}\right) \Big/ \sum_{\mathbf{x}_{ij} \in \Omega_a} \sqrt{F(\phi^{**}_{ij})}. \tag{22}$$

We finally get ϕ^{n+1} by $\phi^{n+1}_{ij} = \phi^{**}_{ij} + \Delta t \gamma^{**}\sqrt{F(\phi^{**}_{ij})}$.

Remarks. In this work, we only focus on developing a simple, fast, and practical adaptive finite difference method for the simulation of two-phase incompressible fluid flow. The temporally first-order scheme is used for the purpose of convenience and smaller time steps (less than $O(h^2)$) are used in all simulations for the purpose of accuracy. Note that the temporally second-order scheme
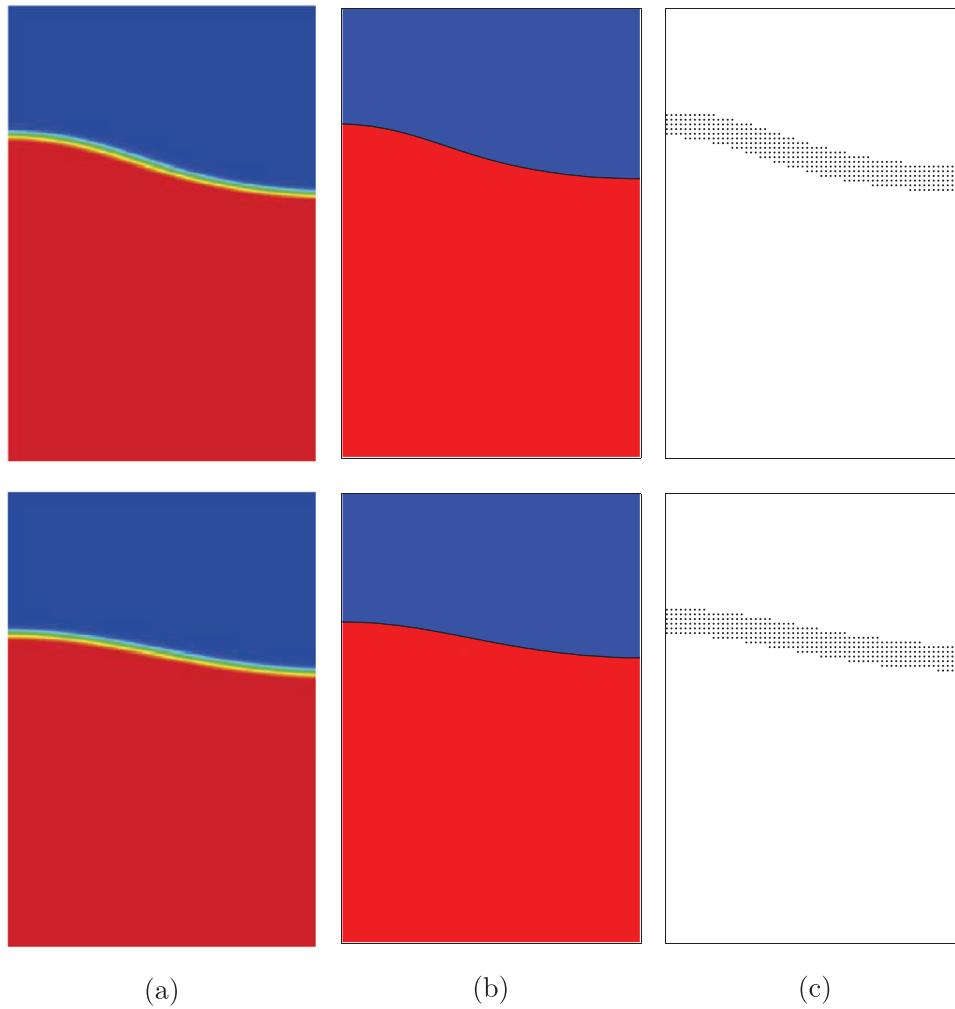
(a)   (b)   (c)

Fig. 10. The snapshots of two-phase flow in a sloshing tank. The left column shows the previous results. Adapted form Joshi and Jaiman (2018) with the permission of Elsevier Science. The middle and right columns show the present results and the corresponding narrow band domains.
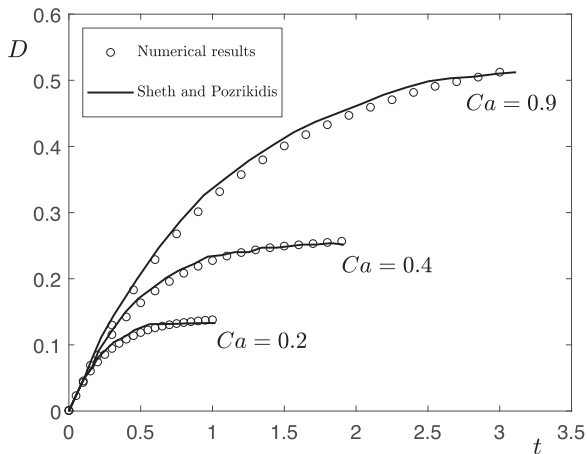

Fig. 11. Temporal evolution of the deformation parameter $D = (L - B)/(L + B)$ of a droplet for $Re = 1$, $Ca = 0.2$, 0.4, and 0.9.

can be straightforwardly extended in a same manner, we will leave this for the interested readers. Different from the classic adaptive mesh technique, our proposed method only calculates phase-field in an adaptive narrow band region containing the interface so that the computation is very fast. We can easily refine the grid in the

narrow band domain to achieve a finer resolution, schematic illustrations are shown in Fig. 4(a) and (b). To achieve multilevel grids, a similar approach of classical adaptive mesh technique must be used, it is out of the scope of this study.

3. Numerical experiments

In the following parts, we perform several numerical experiments, such as the droplet deformation in background flow, pressure jump, droplet deformation in shear flow, heavier droplet falling, and lighter bubble raising, etc, to investigate the efficiency and the practicability of the proposed method. In 2D cases, the periodic boundary condition along x-direction is used, the no-slip and homogeneous-Neumann boundary conditions along y-direction are used for the shear flow and the buoyancy flow, respectively. In 3D cases, the periodic boundary condition is used along x- and y-directions, the no-slip and homogeneous-Neumann boundary conditions along z-direction are used for the shear flow and the buoyancy flow, respectively. Here, we define the interfacial parameter to be (Yang and Kim, 2018):

$$\epsilon = \epsilon_m = \frac{mh}{2\sqrt{2}\tanh^{-1}(0.9)}, \tag{23}$$

which indicates the transition layer approximately occupies m grids. Although the transition layer appears in the phase-field model, the interface is usually represented by the zero level set
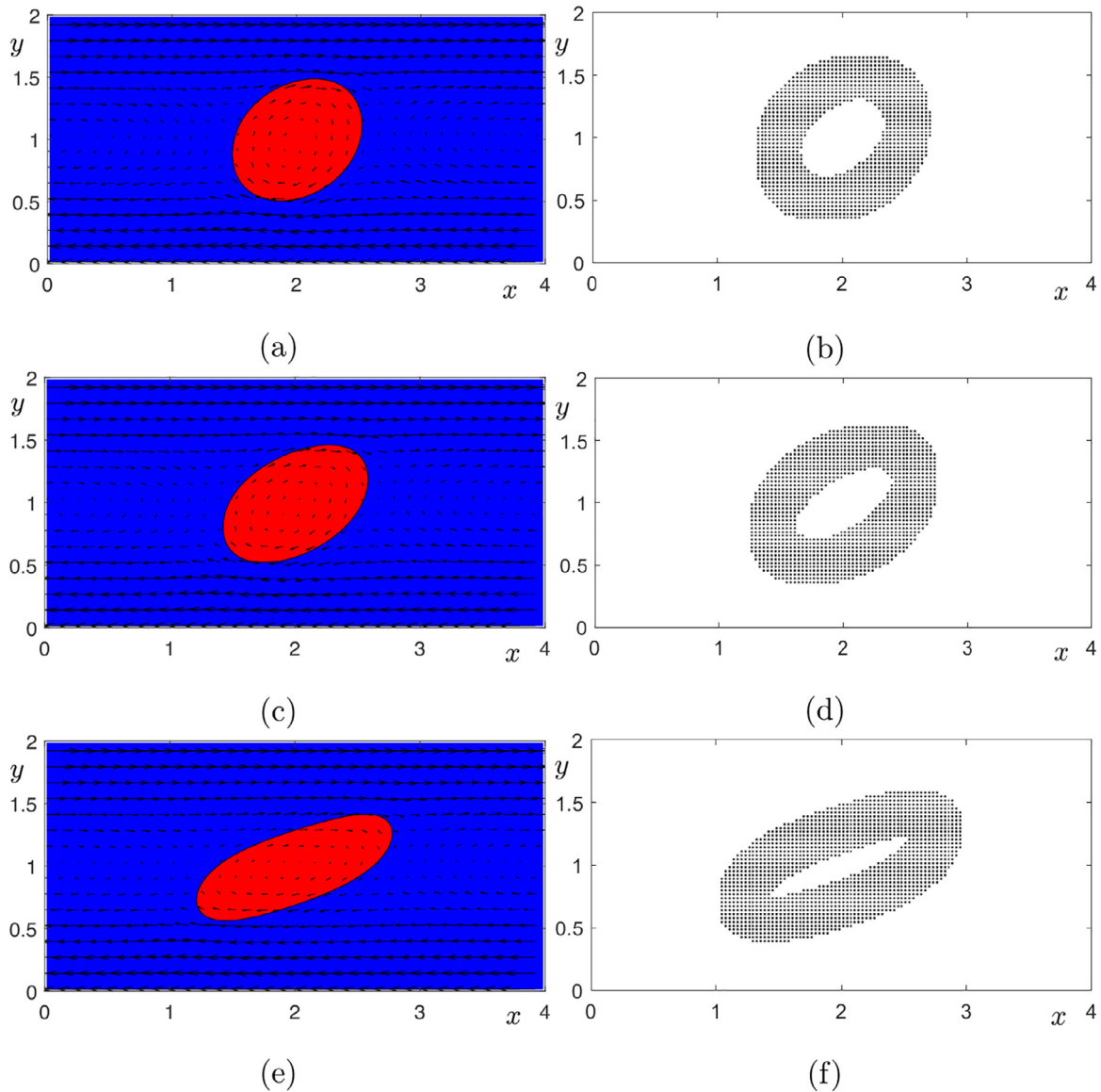
Fig. 12. Profiles of the droplet for (a) $Ca = 0.2$, (c) $Ca = 0.4$, and (e) $Ca = 0.9$. The corresponding times are $t = 1$, 1.9, and 3, respectively. Their corresponding adaptive domains are shown in (b), (d), and (f), respectively.

of phase-field function, i.e., $\phi = 0$ in many studies (Sui et al., 2014; Mu et al., 2018; Aland and Voigt, 2012). A schematic illustration is shown in Fig. 5. In the following simulations, we will plot the zero level set of phase-field function instead of the transition layer and different colors indicate that the regions are approximately occupied by two different fluids.

3.1. Droplet deformation in a background flow

We first consider the droplet deformation in a background velocity field with the absence of the NS equation. The initial condition is defined as:

$$\phi(x, y, 0) = \tanh\left(\frac{0.5 - \sqrt{(x-1)^2 + (y-1)^2}}{\sqrt{2}\epsilon}\right) \qquad (24)$$

in the domain $\Omega = (0, 2)^2$. The background velocity field is set to be: $u = y - 1$, $v = 0$, i.e., a shear flow. The numerical parameters: $h = 1/64$, $\Delta t = 0.2h^2$, ϵ_5, $Pe = 0.002/\epsilon_5$ are used. Note that the exact solution exists in this kind of background flow, which has the following form at time t: $x = x_0 + ut$, $y = y_0$, where x and y are the

positions of points at the interface which is defined by the zero level-set ($\phi = 0$), x_0 and y_0 are the initial positions. Fig. 6(a) and (b) show the initial profile and its adaptive domain Ω_a, respectively. The evolutions of profile and adaptive domain at $t = 0.9766$ are shown in Fig. 6(c) and (d), respectively. We can observe that the exact solutions and numerical results are in good agreement. The convection term works well to transfer the interface to a desired position.

3.2. Verification of the adaptive domain Ω_a

In this work, the temporally adaptive narrow band domain is related to the parameters β and m_p. To investigate the effect of β and m_p on the computational results, we calculate the deformation parameter $D = (L - B)/(L + B)$ of a droplet in shear flow, where L and B are the maximum and minimum distances from the interface to the central point, respectively. Here, the initial condition, background flow, and numerical parameters are unchanged like those in Section 3.1. We consider different values of β: 0.99, 0.8, 0.7, and 0.6 for the adaptive domain Ω_a and per-
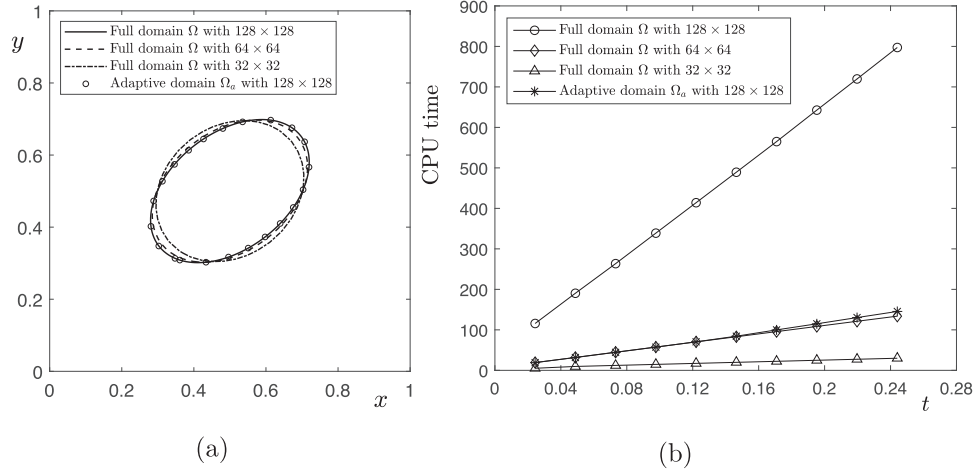
Fig. 13. (a) Contour line $\phi = 0$ at $t = 0.2441$ and (b) evolution of CPU time with respect to different domain.
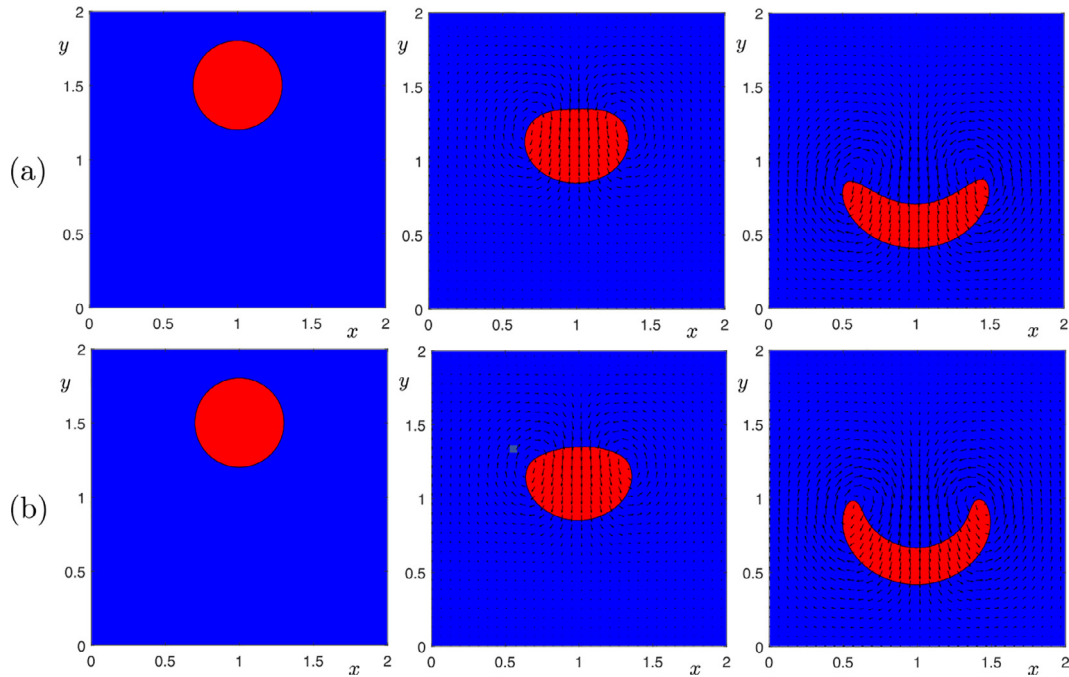

Fig. 14. Temporal evolutions of the droplet falling with respect to different Weber numbers, (a) $We = 20$ and (b) $We = 200$. The computational time from the left to right in each row are $t = 0$, 1.4648, and 2.4414.

form the same simulation in the full domain Ω as a comparison. Fig. 7 illustrates the change of D versus different values of m_p at $t = 0.9766$. As we can see, extremely small values of both β and m_p can not guarantee the good results. However, the good results can be obtained when we use a large enough value of β or m_p. Therefore, $\beta = 0.99$ and $m_p = 2$ is an appropriate choice to obtain efficient and accurate results.

3.3. Pressure jump test

We now investigate the test of pressure jump, which is important to verify the numerical treatment of the surface tension term. The pressure jump around a circular droplet after one time step is considered. The initial conditions are defined as:

$$\phi(x, y, 0) = \tanh\left(\frac{0.5 - \sqrt{(x-1)^2 + (y-1)^2}}{\sqrt{2}\epsilon}\right), \tag{25}$$

$$u(x, y, 0) = v(x, y, 0) = 0 \tag{26}$$

in the domain $\Omega = (0, 2)^2$, which means the radius R of the droplet is 0.5. The numerical parameters are: $h = 1/64$, $\Delta t = 0.1h^2$, ϵ_8, $Pe = 1.0/\epsilon_8$, $Re = 10$, $We = 10$. In this simulation, the density- and viscosity-matched conditions (i.e., $\rho_1 = \rho_2$, $\eta_1 = \eta_2$) are considered and the gravity effect is omitted (i.e., $Fr = \infty$). Fig. 8(a) is the initial condition of a droplet, (b) illustrates the cross section of the pressure along $y = 1$. Theoretically, the pressure jump across the circular disk is (Kim, 2012):

$$[p]_t = \frac{1}{RWe}, \tag{27}$$

where R is the radius of the circle and We is the Weber number. In this test, the theoretical pressure jump $[p]_t$ is 0.2 because we set $R = 0.5$ and $We = 10$. The numerical pressure jump is $[p]_n = p_{\max} - p_{\min} = 0.1964$. Therefore, the numerical result and the theoretical value are in good agreement.
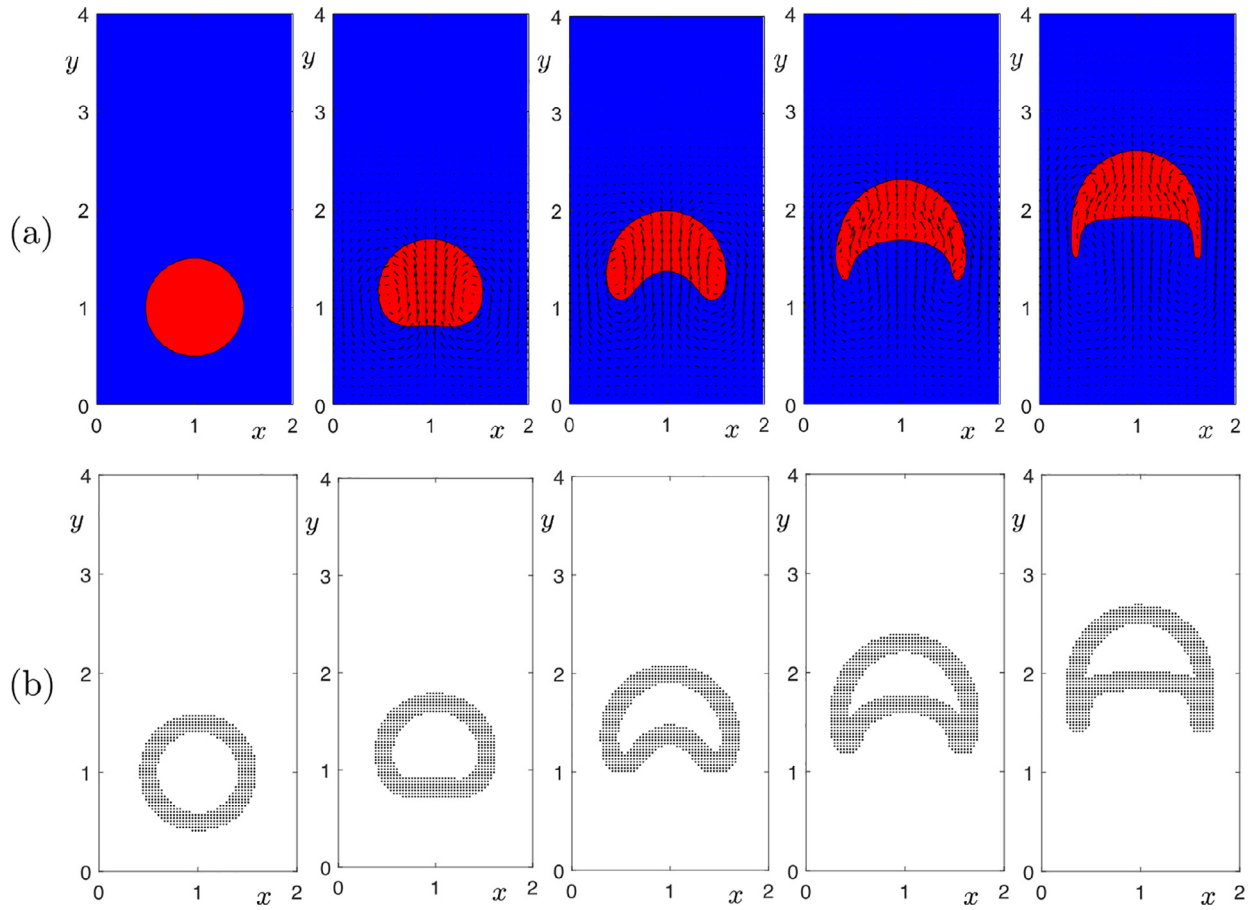
9

Fig. 15. Temporal evolutions of (a) rising bubble and (b) its corresponding adaptive domain. The computational time from the left to right in each row are $t = 0$, 1, 2, 3, and 4.
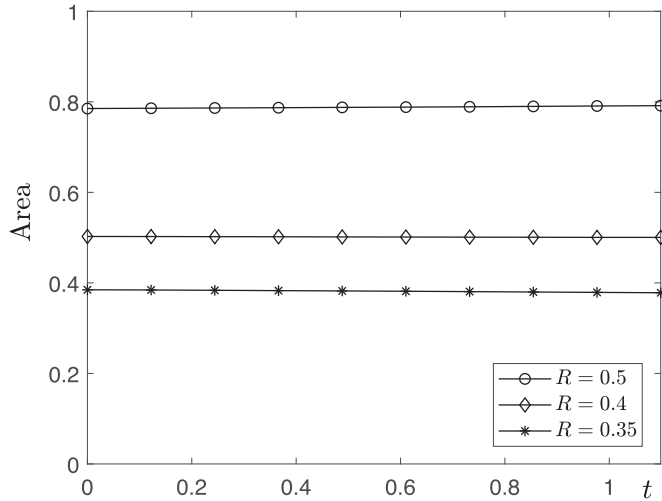

Fig. 16. Temporal evolutions of (a) falling droplets of various sizes and (b) its corresponding adaptive domain. The computational time from the left to right in each row are $t = 0$, 0.2441, 0.4883, 0.8545, and 1.0986.

3.4. Comparison with the previous results (Joshi and Jaiman, 2018) in a sloshing tank

In a previous work, Joshi and Jaiman (2018) developed an accurate adaptive finite element method for the CACNS system. They verified their method by performing various numerical tests, such as the the phase separation, evolutions of two circles without fluid flow, and binary fluid flow in a sloshing tank, etc. In this part, we consider a same simulation in a sloshing tank. Fig. 9(a)–(c) illustrate the local views in $(0, 1) \times (0, 1.5)$ of schematic diagram, initial state, and initial narrow band domain, respectively. We take the following definitions of initial condition:

$$\phi(x, y, 0) = - \tanh \left(\frac{y - 1.01 - 0.1 \sin(2\pi x)}{\sqrt{2}\epsilon} \right), \tag{28}$$

$$u(x, y, 0) = v(x, y, 0) = 0. \tag{29}$$

For the simulation in $\Omega = (0, 1) \times (0, 2)$, we take $h = 1/128$, $\Delta t = 0.2h^2$, $\epsilon = \epsilon_4$, $Pe = 0.01$, $Re = 10^4$, and $Fr = 1$. The no-slip boundary conditions are used for velocity field. The left column in Fig. 10 illustrates the results in (Joshi and Jaiman, 2018), the middle column illustrates the present results, and the corresponding narrow band domains are shown in the right column. The top and bottom rows in Fig. 10 show the results at $t = 1.98$ and $t = 12.96$, respectively. We observe that the similar results are obtained by using the present method.

3.5. Comparison with the previous results (Aland and Voigt, 2012) in shear flow

In this work, the NS equation is solved in a discrete domain with coarser mesh size for the purpose of efficiency. To verify the practicability of our approach for the CACNS system under the effect of surface tension, we compare the droplet deformation
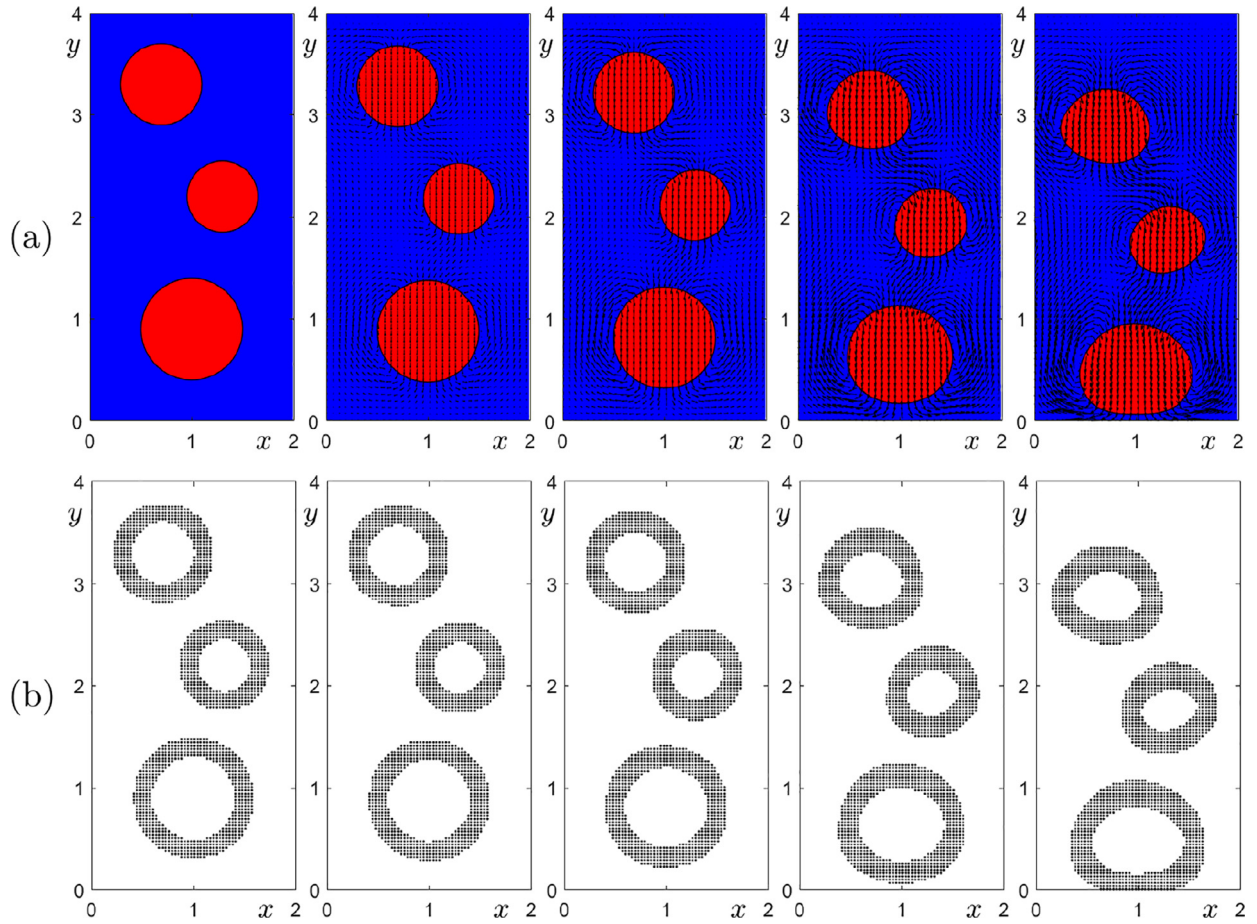
Fig. 17. Temporal evolutions of the area of three droplets.

in shear flow with the previous results performed by Aland and Voigt (2012). The initial conditions are the circular droplet and the Couette flow velocity in the domain $\Omega = (0, 4) \times (0, 2)$:

$$\phi(x, y, 0) = \tanh\left(\frac{0.5 - \sqrt{(x-1)^2 + (y-1)^2}}{\sqrt{2}\epsilon}\right), \tag{30}$$

$$u(x, y, 0) = y - 1, \quad v(x, y, 0) = 0. \tag{31}$$

The numerical parameters used are: $h = 1/32$, $\Delta t = 0.2h^2$, ϵ_4, $Re = 1$, and $Fr = \infty$. The density- and viscosity-matched conditions are considered. In Fig. 11, the solid black lines represent the deformation parameter $D = (L - B)/(L + B)$ in the work from Aland and Voigt (2012). The open circles are the results obtained from the proposed method. We can find that they are in good agreement. Here, we use $Pe = 0.025$, $Pe = 0.05$, and $Pe = 0.1125$ for the cases $Ca = We/Re = 0.2$, $Ca = 0.4$, and $Ca = 0.9$. We observe that the Peclet number Pe is proportional to the Capillary number Ca. Fig. 12(a), (c), and (e) illustrate the profiles of the droplet for $Ca = 0.2$, 0.4, and 0.9, respectively. The corresponding adaptive domains are shown in Fig. 12(b), (d), and (f), respectively.

3.6. Efficiency of the proposed method

Next, we investigate the efficiency of the proposed method by considering the droplet deformation under shear flow in the domain $\Omega = (0, 1)^2$. The mesh size 128×128 is used for the calculation in the adaptive domain. To do the comparison, the simulations in the full domain with mesh sizes: 32×32, 64×64, and

128×128 are considered. The initial conditions are defined as:

$$\phi(x, y, 0) = \tanh\left(\frac{0.2 - \sqrt{(x-0.5)^2 + (y-0.5)^2}}{\sqrt{2}\epsilon}\right), \tag{32}$$

$$u(x, y, 0) = 2(y - 0.5), \quad v(x, y, 0) = 0. \tag{33}$$

The numerical parameters: $\Delta t = 2.44 \times 10^{-5}$, $\epsilon = 0.015$, $Pe = 0.03$, $Re = 500$, $We = 250$, and $Fr = \infty$ are used. The density- and viscosity-matched conditions are also considered. Fig. 13(a) shows the profiles at $t = 0.2441$ with respect to different mesh sizes and (b) shows the CPU time consumed with temporal evolution. As we can observe from Fig. 13(a), the interfacial profile converges with the increase of mesh size in the full domain. By using the proposed method, a more accurate result can be obtained by using less mesh grids. Comparing with the full domain, the proposed adaptive domain significantly saves the computational time in the condition of using same mesh size.

3.7. Effect of the Weber number: We

In the simulation with the combination of surface tension and gravity, we briefly consider the effect of the Weber number: We on the dynamics of heavier droplet falling. The initial conditions are set to be:

$$\phi(x, y, 0) = \tanh\left(\frac{0.3 - \sqrt{(x-1)^2 + (y-1.5)^2}}{\sqrt{2}\epsilon}\right), \tag{34}$$

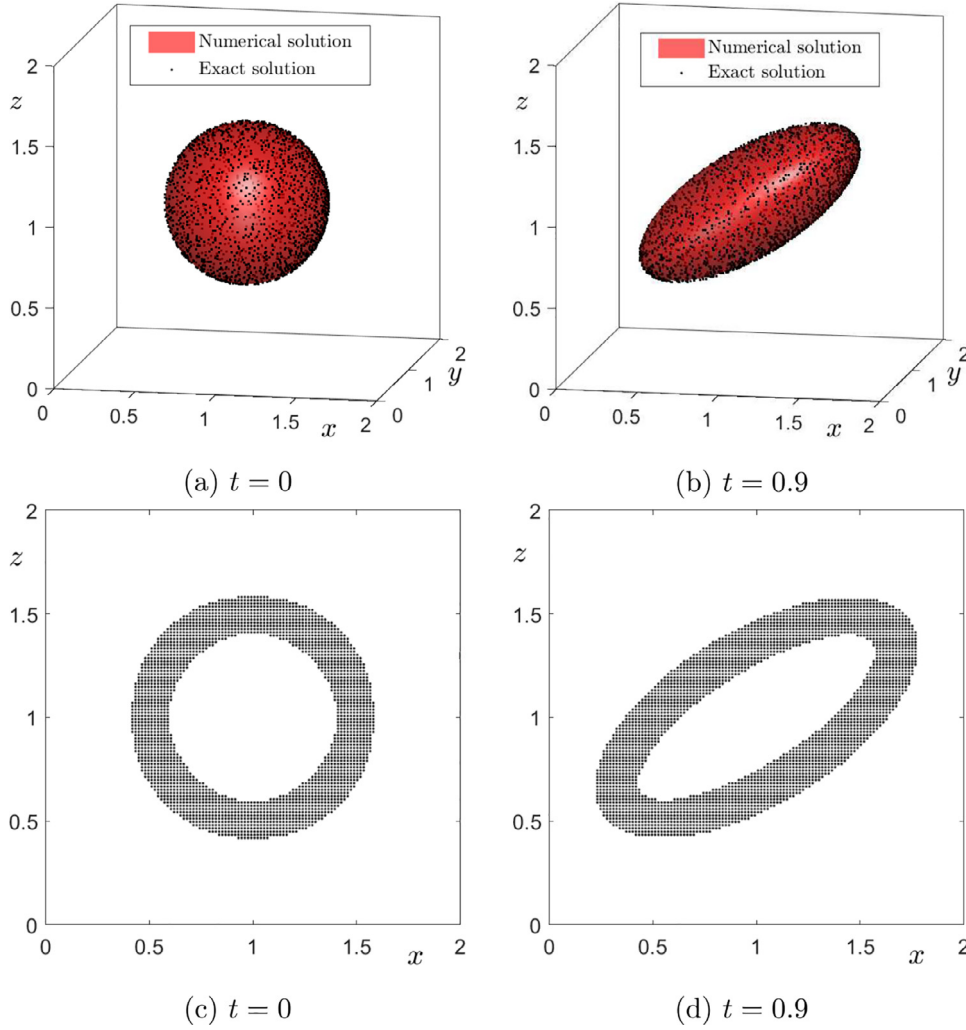$$u(x, y, 0) = v(x, y, 0) = 0 \tag{35}$$

Fig. 18. Evolutions at (a) $t = 0$ and (b) $t = 0.9$. The corresponding adaptive domain (cross profile along $y = 1$) of (a) and (b) are illustrated in (c) and (d), respectively.

in the domain $\Omega = (0, 2)^2$. We use the following numerical parameters: $h = 1/64$, $\Delta t = 0.2h^2$, $\epsilon = \epsilon_4$, $Pe = 0.03$, $Re = 1000$, and $Fr = 1$. The density ratio $\rho_1 : \rho_2 = 3 : 1$ and viscosity-matched condition are considered. Fig. 14(a) and (b) show the temporal evolutions of droplet falling with respect to different Weber numbers: $We = 20$ and $We = 200$. The numerical results indicate that the larger We causes the decrease of surface tension effect, and then leads to the larger deformation of the droplet.

3.8. Rising bubble with large density and viscosity ratios

In this part, we simulate the rising bubble in water with a density ratio $\rho_1 : \rho_2 = 1 : 1000$, and a viscosity ratio $\eta_1 : \eta_2 = 1 : 100$ (Aland and Voigt, 2012) by the proposed method. The circle bubble with radius $R = 0.5$ is set in the domain $\Omega = (0, 2) \times (0, 4)$. The initial conditions are defined as:

$$\phi(x, y, 0) = \tanh\left(\frac{0.5 - \sqrt{(x-1)^2 + (y-1)^2}}{\sqrt{2}\epsilon}\right), \quad (36)$$

$$u(x, y, 0) = v(x, y, 0) = 0. \quad (37)$$

The numerical parameters: $h = 1/64$, $\Delta t = 0.00001$, ϵ_4, $Pe = 0.03$, $Re = 35$, $We = 125$, and $Fr = 1$ are used. Fig. 15(a) shows

the temporal evolution of bubble at $t = 0$, 1, 2, 3, and 4; and (b) shows the temporal evolution of adaptive domain. We can see that the result is similar with the numerical results in Ref. (Jeong and Kim, 2017).

3.9. Falling droplets of various sizes

In this subsection, we consider the movement of several droplets of different sizes in the domain $\Omega = (0, 2) \times (0, 4)$. The following initial conditions are used:

$$\phi(x, y, 0) = \tanh\left(\frac{0.5 - \sqrt{(x-1)^2 + (y-0.9)^2}}{\sqrt{2}\epsilon}\right)$$

$$+ \tanh\left(\frac{0.4 - \sqrt{(x-0.7)^2 + (y-3.3)^2}}{\sqrt{2}\epsilon}\right)$$

$$+ \tanh\left(\frac{0.35 - \sqrt{(x-1.3)^2 + (y-2.2)^2}}{\sqrt{2}\epsilon}\right) + 2, \quad (38)$$

$$u(x, y, 0) = v(x, y, 0) = 0, \quad (39)$$

which indicates the initial radii of three droplets are $R = 0.5$, 0.4, and 0.35. The following parameters: $h = 1/64$, $\Delta t = 0.2h^2$, $Re =$
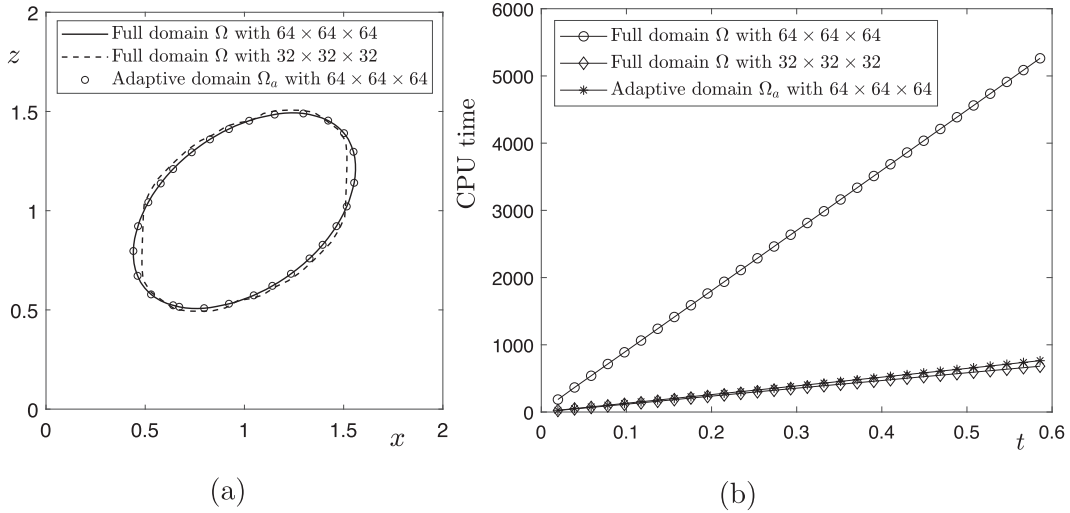
Fig. 19. (a) Cross profile (along $y = 1$) of contour line $\phi = 0$ at $t = 0.5859$ and (b) evolution of CPU time with respect to different domain.

1000, $We = 5$, ϵ_4, $Pe = 0.06$, $Fr = 1$, $\rho_1 : \rho_2 = 10 : 1$, and $\eta_1 : \eta_2 = 1 : 1$ are used. We plot the snapshots in Fig. 16 and the temporal evolution of area of each droplet is shown in Fig. 17. We can find that three droplets approximately retain their initial mass. Although the slight increase or decrease of the mass of droplet appears due to the effect of volume-preserving mean curvature dynamics of the CAC model, we can use proper value of Pe to suppress this nonphysical phenomenon. To fix this problem in another way, a recently developed CAC model with anti-curvature term and nonstandard mobility (Yang et al., 2020) will be a good choice, we will consider this in the future.

3.10. 3D droplet deformation in a background flow

In the three-dimensional space, we first consider the droplet deformation in the absence of the NS equation. The initial condition of a spherical droplet is defined as

$$\phi(x, y, z, 0) = \tanh\left(\frac{0.5 - \sqrt{(x-1)^2 + (y-1)^2 + (z-1)^2}}{\sqrt{2}\epsilon}\right) \tag{40}$$

in the domain $\Omega = (0, 2)^3$. The background flow field is set to be $u(x, y, z) = z - 1$, and $v(x, y, z) = w(x, y, z) = 0$. We use the following numerical parameters: $h = 1/64$, $\Delta t = 0.1h^2$, ϵ_4, and $Pe = 0.03$. Fig. 18(a) and (b) illustrate the evolutions at $t = 0$ and $t = 0.9$, respectively. Their corresponding adaptive domains are represented in Fig. 18(c) and (d), respectively. We can observe that the numerical result and exact solution are in good agreement.

3.11. Efficiency of the proposed method in the 3D space

Without the loss of generality, we consider the droplet deformation in shear flow to investigate the efficiency of the proposed method. The initial conditions are set to be

$$\phi(x, y, z, 0) = \tanh\left(\frac{0.5 - \sqrt{(x-1)^2 + (y-1)^2 + (z-1)^2}}{\sqrt{2}\epsilon}\right), \tag{41}$$

$$u(x, y, z, 0) = z - 1, \tag{42}$$

$$v(x, y, z, 0) = w(x, y, z, 0) = 0 \tag{43}$$

in the domain $\Omega = (0, 2)^3$. The following numerical parameters are used by $\Delta t = 9.766 \times 10^{-5}$, $\epsilon = 0.03$, $Pe = 0.03$, $Re = 10$, $We = 40$, and $Fr = \infty$. The computation is performed until $t = 0.5859$. Fig. 19(a) shows the evolutions at $t = 0.5859$ with respect to the full domain with mesh size: $64 \times 64 \times 64$, adaptive domain with mesh size: $64 \times 64 \times 64$, and the full domain with mesh size: $32 \times 32 \times 32$. The temporal evolutions of the consumed CPU time are illustrated in Fig. 19(b). Although the CPU time is less consumed in the full domain with mesh size: $32 \times 32 \times 32$, the result can not guarantee the accuracy. For the same mesh size, the proposed method can obviously save the computational time.

3.12. Rising bubble in the 3D space

Finally, we show that the proposed method can also be applied to simulate the rising bubble with large density and viscosity ratios. In this test, the initial conditions are defined as:

$$\phi(x, y, z, 0)$$
$$= \tanh\left(\frac{0.35 - \sqrt{(x-1)^2 + (y-1)^2 + (z-0.6)^2}}{\sqrt{2}\epsilon}\right), \tag{44}$$

$$u(x, y, z, 0) = v(x, y, z, 0) = w(x, y, z, 0) = 0 \tag{45}$$

in the domain $\Omega = (0, 2)^3$. The numerical parameters: $h = 1/32$, $\Delta t = 0.1h^2$, ϵ_4, $Pe = 0.03$, $Re = 100$, $We = 250$, and $Fr = 1$ are used. The density and viscosity ratios are sent to be: $\rho_1 : \rho_2 = 1 : 1000$, $\eta_1 : \eta_2 = 1 : 100$, respectively. From the temporal evolution shown in Fig. 20, we can see that the bubble raises and deforms due to the buoyancy-driven force. The mass conservation is an important property for the two-phase incompressible fluid system, Fig. 21 illustrates the temporal evolution of discrete polyhedron volume (Li et al., 2013) of the droplet, we can see that the mass is conserved by using our method.

3.13. Two-phase fluid flows with topological change

In this subsection, we investigate the two-phase fluid flows with topological change of the interface, such as the coalescence of two droplets (Shah et al., 2018), and the Rayleigh instability (Popinet, 2009). First of all, we consider two droplets in the two-
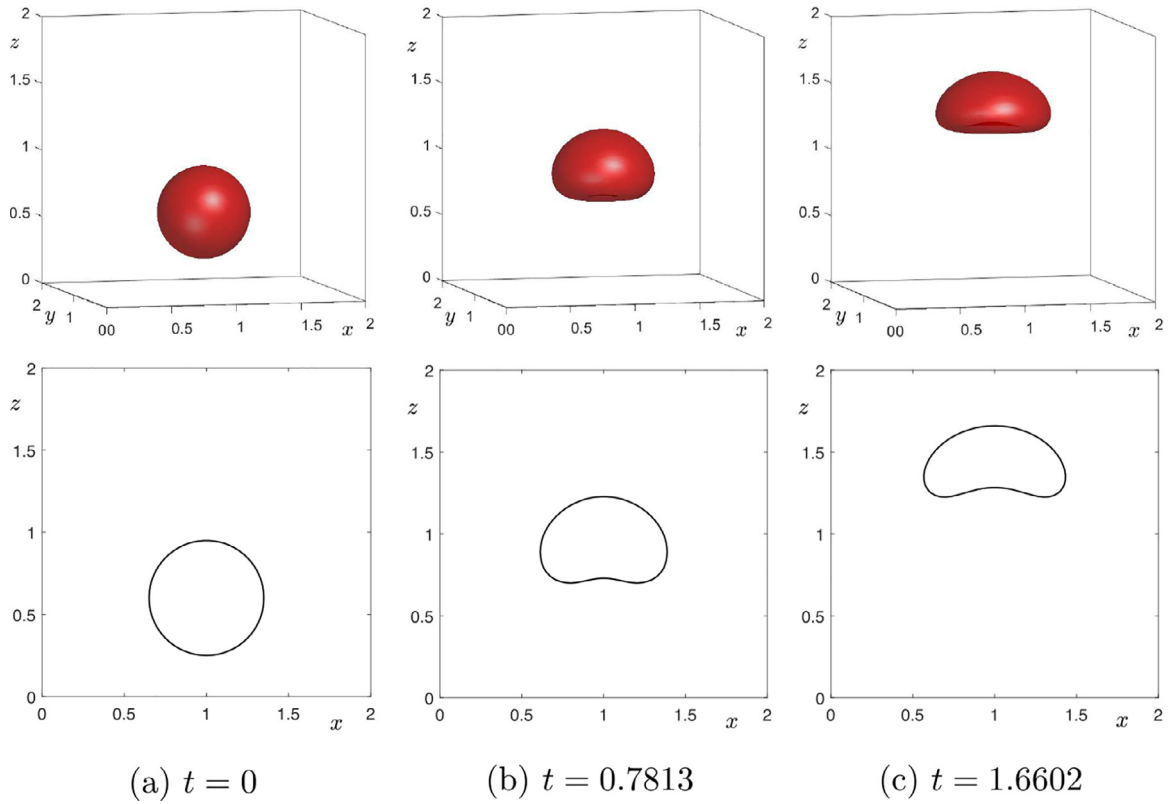
Fig. 20. Temporal evolution of rising bubble with large density and viscosity ratios. The second row illustrates the cross profiles at $y = 1$. The computational time is illustrated below each figure.
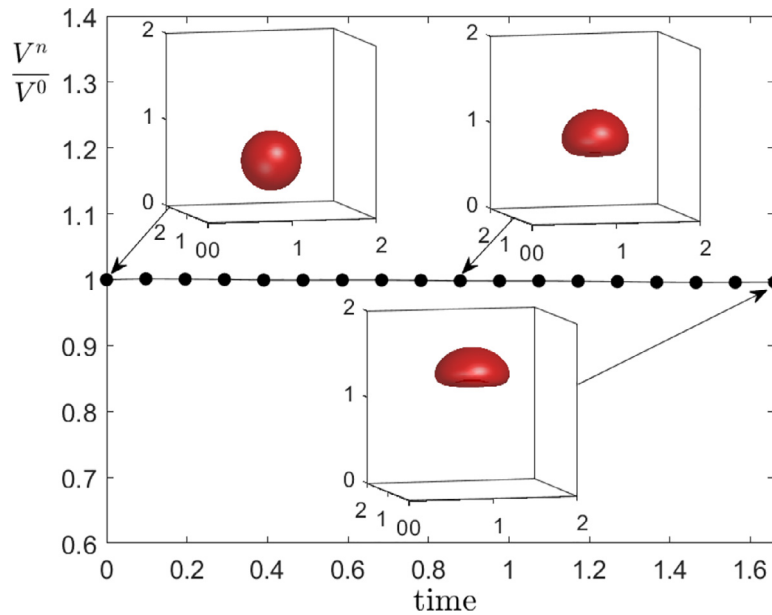
(a) $t = 0$ (b) $t = 0.7813$ (c) $t = 1.6602$

Fig. 21. Temporal evolution of discrete polyhedron volume of the droplet. The embedded figures are the evolutions at specific time.

dimensional shear flow with the following initial conditions:

$$\phi(x, y, 0) = \tanh\left(\frac{0.39 - \sqrt{(x - 1.6)^2 + (y - 1.3)^2}}{\sqrt{2}\epsilon}\right)$$

$$+ \tanh\left(\frac{0.39 - \sqrt{(x - 2.4)^2 + (y - 0.7)^2}}{\sqrt{2}\epsilon}\right) + 1, \quad (46)$$

$$u(x, y, 0) = y - 1, \quad v(x, y, 0) = 0. \quad (47)$$

The computation is performed in the domain $\Omega = (0, 4) \times (0, 2)$ with the following parameters: $h = 1/64$, $\Delta t = 0.4h^2$, $\epsilon = \epsilon_4$, $Pe = 0.03$, $Re = 30$, $We = 1$, $\rho_1 : \rho_2 = 1 : 1$, and $\eta_1 : \eta_2 = 1 : 1$. From Fig. 22, we observe that two droplets deform and then coalesce with each other.
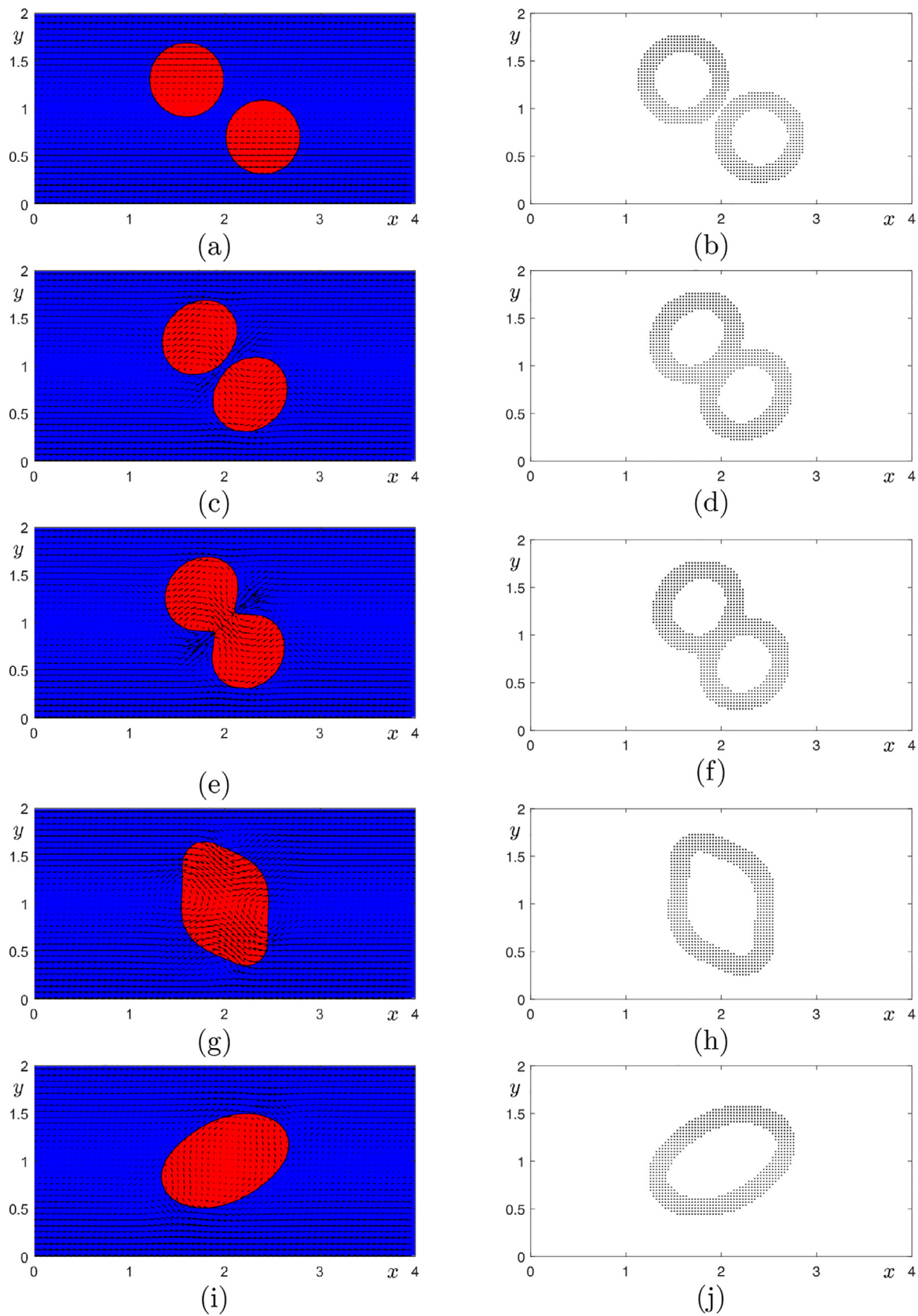
Fig. 22. Profiles of the two droplets in shear flow. The right column shows the corresponding adaptive domain. The figures from the top to bottom in each column are at $t = 0, 0.4883, 0.5859, 0.8789, 1.4648$.
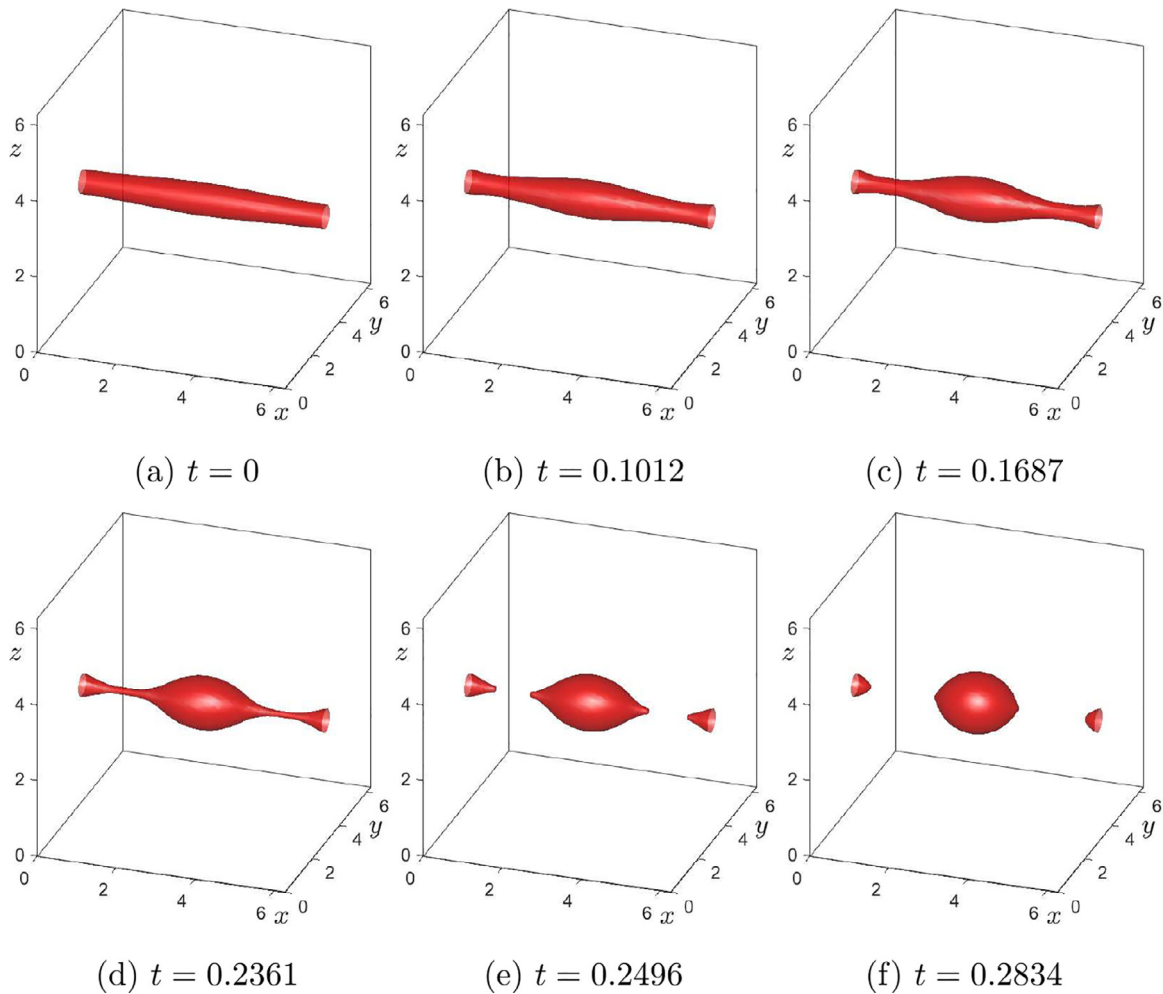
Fig. 23. The breakup of liquid column driven by the Rayleigh instability. The computational time is illustrated below each figure.

Next, we consider the breakup of fluid column driven by the Rayleigh instability in three-dimensional space $\Omega = (0, 2\pi)^2$. The initial conditions are as follows:

$$\phi(x, y, z, 0)$$
$$= \tanh\left(\frac{0.35 - \sqrt{(y - \pi)^2 + (z - \pi)^2} - 0.05\cos(x)}{\sqrt{2}\epsilon}\right), \quad (48)$$

$$u(x, y, z, 0) = v(x, y, z, 0) = w(x, y, z, 0) = 0. \quad (49)$$

The parameters used in this simulation are: $h = \pi/32$, $\Delta t = 0.1h^2$, $\epsilon = \epsilon_4$, $Pe = 0.03$, $Re = 0.16$, $We = 0.002$, $\rho_1 : \rho_2 = 1 : 1$, and $\eta_1 : \eta_2 = 1 : 1$. Fig. 23(a)–(f) show the snapshots of liquid column at different moments. Due to the effect of surface tension, the liquid column undergoes pinch-off and satellite droplets form. The results in this subsection indicate that our method still works well for two-phase flows with topological change of the interface.

Remarks. For comparison studies, we took proper values of ϵ and Pe to fit the theoretical or previous results. In a recent work (Jeong and Kim, 2017), authors investigated the effect of Pe on the CACNS system and proper value of Pe was $0.01 \sim 1.0$. Therefore, we used Pe in this range for the simulations of two-phase fluid flows.

4. Conclusions

In this work, we presented an efficient and practical adaptive finite difference method for the CACNS system to simulate the two-phase fluid flow. For the CAC equation, we used a temporally adaptive narrow band domain embedded in the uniform discrete rectangular domain. The narrow band domain was defined as a neighboring region of the interface. The NS equation was solved in a full discrete domain with a coarser mesh than that for the CAC equation. In the simulation with a background flow, the numerical results showed a good agreement with the exact solution. In the simulations of CACNS system, some comparison studies indicated that the proposed method is practical and efficient. In particular, the proposed method could also be applied for the two-phase fluid flow with large density and viscosity ratios. As the future work, the proposed method can be extended to efficiently simulate various multi-component (more than two) fluid flow phenomena.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Junxiang Yang: Methodology, Software, Validation, Investigation, Visualization, Writing - original draft, Writing - review & editing. **Darae Jeong:** Software, Validation, Writing - original draft, Visualization, Project administration. **Junseok Kim:** Conceptualization, Methodology, Investigation, Writing - original draft, Writing -

review & editing, Supervision, Project administration, Funding acquisition.

Acknowledgments

References

Aguilar, R.L., Vella, D., Yeomans, J.M., 2014. Lattice–boltzmann simulations of droplet evaporation. Soft Matter 10, 8267–8275.

Aihara, S., Takaki, T., Takada, N., 2019. Multi-phase-field modeling using a conservative Allen–Cahn equation for multiphase flow. Comput. Fluids 178 (15), 141–151.

Aland, S., Voigt, A., 2012. Benchmark computations of diffuse interface models for two-dimensional bubble dynamics. Int. J. Numer. Meth. Fluid 69 (3), 747–761.

Bai, F., He, X., Yang, X., Zhou, R., Wang, C., 2017. Three dimensional phase-field investigation of droplet formation in micro fluidic flow focusing devices with experimental validation. Int. J. Multiphase Flow 93, 130–141.

Balcazar, N., Lehmkuhl, O., Jofre, L., Oliva, A., 2015. Level-set simulations of buoyancy-driven motion of single and multiple bubbles. Int. J. Heat Fluid Flow 56, 91–107.

Burzynski, D.A., Bansmer, S.E., 2018. Droplet splashing on thin moving films at high weber numbers. Int. J. Multiphase Flow 101, 202–211.

Chen, N., Chen, H., Amirfazli, A., 2017. Drop impact on a thin film: miscibility effect. Phys. Fluids 29, 092106.

Chen, Y., Müller, C.R., 2020. A Dirichlet boundary condition for the thermal lattice Boltzmann method. Int. J. Multiphase Flow 123, 103184.

Gronskis, A., Artana, G., 2016. A simple and efficient direct forcing immersed boundary method combined with a high order compact scheme for simulating flows with moving rigid boundaries. Comput. Fluids 124 (2), 86–104.

Harlow, F.H., Welch, J.E., 1966. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. Phys. Fluids 8 (12), 2182–2189.

Hong, D.A., Steijn, V.V., Portela, L.M., Kreutzer, M.T., Kleijn, C.R., 2013. Benchmark numerical simulations of segmented two-phase flows in microchannels using the volume of fluid method. Comput. Fluids 86, 28–36.

Horng, T.L., Hsieh, P.W., Yang, S.Y., You, C.S., 2018. A simple direct-forcing immersed boundary projection method with prediction-correction for fluid-solid interaction problems. Comput. Fluids 176, 135–152.

Hua, J., Mortensen, D., 2019. A front tracking method for simulation of two-phase interfacial flows on adaptive unstructured meshes for complex geometries. Int. J. Multiphase Flow 119, 166–179.

Jeong, D., Kim, J., 2017. Conservative Allen–Cahn–Navier–Stokes system for incompressible two-phase fluid flows. Comput. Fluids 156, 239–246.

Jeong, D., Kim, J., 2019. Fast and accurate adaptive finite difference method for dendritic growth. Comput. Phys. Commun. 236, 95–103.

Joshi, V., Jaiman, R.K., 2018. An adaptive variational procedure for the conservative and positivity preserving Allen–Cahn phase-filed model. J. Comput. Phys. 336 (1), 478–504.

Karami, K., Rahimian, M.H., Farhadzadeh, M., 2017. Numerical simulation of droplet evaporation on a hot surface near Leidenfrost regime using multiphase lattice Boltzmann method. Appl. Math. Comput. 312 (1), 91–108.

Khan, S.A., Shah, A., 2019. Simulation of the two-dimensional Rayleigh–Taylor instability problem by using diffuse-interface model. AIP Adv. 9, 085312.

Kim, J., 2012. Phase-field models for multi-component fluid flow. Commun. Comput. Phys. 12 (3), 613–661.

Kim, J., Lee, S., Choi, Y., 2014. A conservative Allen–Cahn equation with a space-time dependent lagrange multiplier. Int. J. Eng. Sci. 84, 11–17.

Lee, H.G., Kim, J., 2012. Regularized dirac delta functions for phase field models. Int. J. Numer. Meth. Eng. 91, 269–288.

Li, Y., Jeong, D., Kim, J., 2014. Adaptive mesh refinement for simulation of thin film flows. Meccanica 49, 239–252.

Li, Y., Jung, E., Lee, W., Lee, H.G., Kim, J., 2012. Volume preserving immersed boundary methods for two-phase fluid flows. Int. J. Numer. Meth. Fluids 69, 842–858.

Li, Y., Kim, J., 2012. Phase-field simulations of crystal growth with adaptive mesh refinement. Int. J. Heat Mass Trans. 55, 7926–7932.

Li, Y., Lee, H.G., Xia, B., Kim, J., 2016. A compact fourth-order finite difference scheme for the three-dimensional Cahn–Hilliard equation. Comput. Phys. Commun. 200, 108–116.

Li, Y., Yun, A., Lee, D., Shin, J., Jeong, D., Kim, J., 2013. Three-dimensional volume-conserving immersed boundary model for two-phase fluid flows. Comput. Methods Appl. Mech. Eng. 257, 36–46.

Luo, K., Shao, C., Yang, Y., Fan, J., 2015. A mass conserving level set method for detailed numerical simulation of liquid atomization. J. Comput. Phys. 298, 495–519.

Majidi, M., Haghani-Hassan-Abadi, R., Rahimian, M.-H., 2020. Single recalcitrant bubble simulation using a hybrid lattice Boltzmann finite difference model. Int. J. Multiphase Flow. In press

Mitchell, T., Leonardi, C., Fakhari, A., 2018. Development of a three-dimensional phase-field lattice Boltzmann method for the study of immiscible fluids at high density ratios. Int. J. Multiphase Flow 107, 1–15.

Mu, K., Si, T., Li, E., Xu, R.X., Ding, H., 2018. Numerical study on droplet generation in axisymmetric flow focusing upon actuation. Phys. Fluids 30, 012111.

Pivello, M.R., Villar, M.M., Serfaty, R., Roma, A.M., Silveira-Neto, A., 2014. A fully adaptive front tracking method for the simulation of two phase flows. Int. J. Multiphase Flow 58, 72–82.

Popinet, S., 2009. An accurate adaptive solver for surface-tension-driven interfacial flows. J. Comput. Phys. 228, 5838–5866.

Rodríguez, D., 2017. A combination of parabolized Navier–Stokes equations and level-set method for stratified two-phase internal flow. Int. J. Multiphase Flow 88, 50–62.

Sahu, K.C., Vanka, S.P., 2011. A multiphase lattice Boltzmann study of buoyancy-induced mixing in a tilted channel. Comput. Fluids 50, 199–215.

Scheid, B., Dorbolo, S., Arriaga, L.R., Rio, E., 2012. Antibubble dynamics: the drainage of an air film with viscous interfaces. Phys. Rev. Lett. 109, 264502.

Shah, A., Saeed, S., Khan, S.A., 2018. Numerical investigation of bubbles coalescence in a shear flow with diffuse-interface model. Heliyon 4, e01024.

Shen, M., Li, B.Q., Yang, Q., Bai, Y., Wang, Y., Zhou, S., Zhou, B., Li, T., Hu, Y., 2019. A modified phase-field three-dimensional model for droplet impact with solidification. Int. J. Multiphase Flow 116, 51–66.

Stam, J., 1999. Stable fluids. In: Waggenspack (Ed.), Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques. ACM Press, New York, pp. 121–128.

Sui, Y., Ding, H., Spelt, P.D.M., 2014. Numerical simulations of flows with moving contact lines. Annu. Rev. Fluid Mech. 46, 97–119.

Tanguy, S., Berlemont, A., 2005. Application of a level-set method for simulation of droplet collisions. Int. J. Multiphase Flow 31 (9), 1015–1035.

Theodorakakos, A., Bergeles, G., 2004. Simulation of sharp gas-liquid interface using VOF method and adaptive grid local refinement around the interface. Int. J. Numer. Meth. Fluids 45, 421–439.

Vian, A., Reuse, B., Amstad, E., 2018. Scalable production of double emulsion drops with thin shells. Lab Chip 18, 1936–1942.

Vu, T.-V., Vu, T.V., Bui, D.T., 2019. Numerical study of deformation and breakup of a multi-core compound droplet in simple shear flow. Int. J. Heat Mass Tran. 131, 1083–1094.

Webster, D.R., Longmire, E.K., 2001. Jet pinch-off and drop formation in immiscible liquid-liquid systems. Exp. Fluids 30, 47–56.

Yan, X., Chen, G., 2020. An approximation approach for the simulation of vapor-liquid phase change by the volume-of-fluid method. Int. J. Multiphase Flow. In press

Yang, J., Kim, J., 2018. Phase-field simulation of rayleigh instability on a fibre. Int. J. Multiphase Flow 105, 84–90.

Yang, J., Li, Y., Lee, C., Kim, J., 2020. Conservative Allen–Cahn equation with a non-standard variable mobility. Acta Mech. 231, 561–576.