



An unconditionally energy-stable scheme based on an implicit auxiliary energy variable for incompressible two-phase flows with different densities involving only precomputable coefficient matrices

Zhiguo Yang, Suchuan Dong*

Center for Computational and Applied Mathematics, Department of Mathematics, Purdue University, USA

ARTICLE INFO

Article history:

Received 18 November 2018
Received in revised form 25 April 2019
Accepted 11 May 2019
Available online 14 May 2019

Keywords:

Auxiliary variable
Implicit scalar auxiliary variable
Energy stability
Phase field
Multiphase flows
Two-phase flows

ABSTRACT

We present an energy-stable scheme for numerically approximating the governing equations for incompressible two-phase flows with different densities and dynamic viscosities for the two fluids. The proposed scheme employs a scalar-valued auxiliary energy variable in its formulation, and it satisfies a discrete energy stability property. More importantly, the scheme is computationally efficient. Within each time step, it computes two copies of the flow variables (velocity, pressure, phase field function) by solving individually a linear algebraic system involving a constant and time-independent coefficient matrix for each of these field variables. The coefficient matrices involved in these linear systems only need to be computed once and can be pre-computed. Additionally, within each time step the scheme requires the solution of a nonlinear algebraic equation about a *scalar-valued number* using the Newton's method. The cost for this nonlinear solver is very low, accounting for only a few percent of the total computation time per time step, because this nonlinear equation is about a scalar number, not a field function. Extensive numerical experiments have been presented for several two-phase flow problems involving large density ratios and large viscosity ratios. Comparisons with theory show that the proposed method produces physically accurate results. Simulations with large time step sizes demonstrate the stability of computations and verify the robustness of the proposed method. An implication of this work is that energy-stable schemes for two-phase problems can also become computationally efficient and competitive, eliminating the need for expensive re-computations of coefficient matrices, even at large density ratios and viscosity ratios.

© 2019 Elsevier Inc. All rights reserved.

1. Introduction

This work concerns the simulation of the dynamics of a mixture of two immiscible incompressible fluids with possibly very different densities and dynamic viscosities based on the phase field approach. The presence of fluid interfaces, the associated surface tension, the density contrast and viscosity contrast play an important role in the dynamics of such systems, and these factors also make such numerical simulations very challenging. Phase field (or diffuse interface) [33,44,5,29,23,27] is one of the main approaches for dealing with fluid interfaces in the modeling of two-phase systems, along

* Corresponding author.

E-mail address: sdong@purdue.edu (S. Dong).

with other related methods [30,36,43,37]. It has attracted an increasing interest from the community in the past years, in part because of its physics-based nature. With this approach the fluid interface is treated to be diffuse, as a thin smooth transition layer [5]. The state of the system is characterized by, apart from the hydrodynamic variables such as velocity and pressure, an order parameter (or phase field variable), which varies smoothly within the transition layer and is mostly uniform in the bulk phases. The evolution of the system is characterized by, apart from the kinetic energy, a free energy density function, which contains component terms that promote the mixing of the two fluids and also component terms that tend to separate these fluids. The interplay of these two opposing tendencies determines the dynamic profile of the interface. The adoption of the free energy in the formulation makes it possible to relate to other thermodynamic variables. Indeed, with the phase field approach the governing equations of the system can be rigorously derived based on the conservation laws and thermodynamic principles. Several thermodynamically consistent phase field models are already available in the literature for two-phase and multiphase flows, see e.g. [29,25,1,42,2,11,28,16,14,34], with various degrees of sophistication or the observance/violation of other physical principles such as Galilean invariance and reduction consistency. The mass conservation of the individual fluid components in the system, and the choice of an appropriate form for the free energy density function, naturally give rise to the Cahn-Hilliard equation in two phases or a system of coupled Cahn-Hilliard type equations (see e.g. [1,14], among others).

We focus on the numerical approximation and simulation of the governing equations for incompressible two-phase flows with different densities and viscosities in this work. A computational scientist/engineer interested in such problems confronts a compromise and needs to balance two seemingly incompatible aspects: the desire to be able to use a larger time step size (permissible by accuracy), and the computational cost. On the one hand, semi-implicit splitting type schemes (see e.g. [15,6,9,10,12], among others) induces a very low computational cost per time step, because among other things only de-coupled linear algebraic systems need to be solved after discretization and these linear systems only involve constant and time-independent coefficient matrices that can be pre-computed (see [15]), even with large density ratios and viscosity ratios. The downside of these schemes lies in that they are only conditionally stable and the time step size is restricted by CFL or related conditions. On the other hand, energy-stable schemes (see e.g. [40,35,21,18,41,20,48,34], among others) can potentially allow the use of much larger time step sizes in dynamic simulations. The downside lies in that, the computational cost per time step of these schemes can be very high. Energy-stable schemes often require the solution of coupled nonlinear algebraic field equations or coupled linear algebraic equations. The linear algebraic systems associated with these schemes involve time-dependent coefficient matrices, which require frequent re-computations (or at every time step).

Mindful of the strengths and weaknesses of both types of schemes, we would like to consider the following question. Can we devise an algorithm to combine the strengths of both types of schemes that can achieve unconditional energy stability and simultaneously require a relatively low computational cost? Summarized in this paper is our attempt to tackle this question and a numerical scheme that largely achieves this goal.

The current work has drawn inspirations from several previous studies in the literature. In the following we restrict our review of literature to the energy-stable schemes for the hydrodynamic interactions of two-phase flows with different densities and viscosities based on the phase field framework. This will leave out those algorithms that are devoted purely to the phase field such as the Cahn-Hilliard or Allen-Cahn equations [8,4] without hydrodynamic interactions, for which a large volume of literature exists. Those studies considering only matched densities for the two fluids (see e.g. [3,22,46], among others) will also be largely left out. In [40] a phase field model based on the combined Navier-Stokes/Allen-Cahn equations with different densities/viscosities for the two fluids is considered, and several discretely energy-stable schemes of first order in time are introduced based on projection, Gauge-Uzawa, and pressure stabilization formulations. What is interesting lies in that all these schemes are linear in nature. They only require the solution of linear, albeit coupled, algebraic equations for different flow variables after discretization. To accommodate different densities for the two fluids, the authors of [40] have adopted a reformulation of the inertial term [19] in the Navier-Stokes equation, which as pointed out by [18] might not be consistent with the phase field equation employed therein. Corresponding schemes for a Navier-Stokes/Cahn-Hilliard model are presented in [39]. A fractional-step scheme based on a pressure correction-type strategy is developed in [35] for a phase-field model in which the chemical potential contains a velocity term (see also [28]), together with a generalized Navier type boundary condition for contact lines [32]. This scheme is linear, and the discrete equations about the phase field function and the velocity are coupled together. Improved algorithms for the Navier-Stokes/Cahn-Hilliard model are later developed by [41], in which the discrete phase field equation and the momentum equations are de-coupled thanks to an extra stabilization term [7] employed for approximating the convection velocity in the Cahn-Hilliard equation. The scheme is first order in time, and it is unclear whether an analogous second-order stabilization term exists for the approximation of the convection velocity. In [18] a discretely energy stable scheme for the phase field model of [1] is developed, which gives rise to a system of nonlinear algebraic equations that couple together all the flow variables. It is interesting to note that the algorithm developed in [48] for the same phase field model only requires the solution of linear equations and that the phase-field and momentum equations are de-coupled owing to the same treatment of the discrete convection velocity of the Cahn-Hilliard equation as in [41]. Two discretely energy stable schemes are described in [21,20] for the quasi-incompressible Navier-Stokes/Cahn-Hilliard model of [29]. The schemes can preserve the mass conservation on the discrete level, and they lead to coupled highly-nonlinear algebraic systems after discretization. Numerical schemes for related quasi-incompressible hydrodynamic phase field models have also been proposed in [16,34,17]. In particular, in [17] the so-called invariant energy quadratization method has been used to reformulate the phase field equation, and the resultant numerical schemes are second-order in time, and involve the solution of linear algebraic systems that couple together the different flow variables.

Apart from the many contributions discussed above, an interesting strategy for formulating energy-stable schemes for gradient-type dynamical systems (gradient flows) based on certain auxiliary variables has emerged recently [45,38]. The invariant energy quadratization (IEQ) method [45] introduces an auxiliary field function related to the square root of the potential free energy density function together with a dynamic equation for this auxiliary variable, and allows one to reformulate the gradient-flow evolution equation to facilitate schemes for ensuring the energy stability relatively easily. The scalar auxiliary variable (SAV) method [38] introduces an auxiliary variable, which is a scalar-valued number rather than a field function, related to the square root of the total potential energy integral, and a dynamic equation about this scalar variable. Both types of auxiliary variables can simplify the formulation of schemes to achieve energy stability for gradient flows.

Another recent development that inspires the current work is [26], in which an energy-stable scheme for the incompressible Navier-Stokes equations based on a scalar auxiliary variable related to the total kinetic energy has been developed. Because the Navier-Stokes equation is not a gradient-type system, the scalar auxiliary variable formulation as developed in [38] for gradient flows cannot be directly used. Indeed, it is observed that if one reformulates the viscous term in the Navier-Stokes equation using the auxiliary variable, in a way analogous to the treatment of the dissipation term in the evolution equation for gradient flows, the simulation results turn out to be very poor. Instead, a viable strategy for Navier-Stokes equations seems to be to control the convection term with the auxiliary variable. Such a strategy is presented in [26], which hinges on reformulating the convection-term contribution into a boundary integral in the dynamic equation for the auxiliary total kinetic energy.

In this paper we build upon the scalar auxiliary variable idea and present an energy-stable scheme for the numerical approximation of the two-phase governing equations with different densities and viscosities for the two fluids based on the phase field model of [1]. By introducing a scalar-valued variable related to the total of the kinetic energy and the potential free energy of the two-phase system, we reformulate the two-phase governing equations into an equivalent form. By carefully treating the variable-density and variable-viscosity terms, we show that the proposed scheme honors a discrete energy stability relation. We present an efficient solution algorithm and a procedure for dealing with the integrals of the unknown field functions. Within each time step, our algorithm only requires the solution of several de-coupled individual linear algebraic systems, each for an individual field function, together with the solution of a nonlinear algebraic equation about a *scalar-valued number*. More importantly, those linear algebraic systems to be solved within a time step each involves a *constant and time-independent* coefficient matrix, which only needs to be computed once and can be pre-computed. The nonlinear algebraic equation involved therein requires Newton iterations in its solution. But its computational cost is very low, because this equation is about a single *scalar number*, not a field function. Numerical experiments show that the solution for the nonlinear equation accounts for around 8% of the total solver time within a time step, and essentially all this time is spent on computing the coefficients involved in the nonlinear equation in preparation for the Newton method, rather than in the actual Newton iterations. Because of these attractive properties, our algorithm is computationally very efficient. We will present a number of numerical experiments to demonstrate the stability of our method at large time step sizes for two-phase problems.

The new aspects of this work include the following: (i) the energy-stable scheme for the two-phase governing equations with different densities and viscosities for the two fluids, and (ii) the efficient solution algorithm for implementing the proposed scheme. The property of our solution algorithm that it involves only linear algebraic systems with constant and time-independent coefficient matrices, even at large density ratios and viscosity ratios, is particularly attractive, because this makes the cost low and the method computationally very efficient. To the best of the authors' knowledge, this is the first energy-stable scheme which involves only constant and time-independent coefficient matrices for incompressible two-phase flows with different densities and viscosities for the two fluids.

The rest of this paper is organized as follows. In Section 2, we introduce a scalar-valued auxiliary variable and reformulate the two-phase governing equations into an equivalent system employing this variable. Then we present a scheme for temporal discretization for the reformulated equivalent system, and prove that the scheme satisfies a discrete energy law. An efficient solution algorithm for implementing the scheme will be presented, and the spatial discretization based on C^0 spectral elements will be discussed. In Section 3 we present a few numerical examples of two-phase flows involving large density ratios and viscosity ratios to demonstrate the accuracy of our method and also its stability at large time step sizes. Section 4 then concludes the discussions with some closing remarks.

2. Energy-stable scheme for incompressible two-phase flows

2.1. Governing equations

Consider a mixture of two immiscible incompressible fluids contained in some domain Ω in two or three dimensions, whose boundary is denoted by $\partial\Omega$. Let ρ_1 and ρ_2 respectively denote the constant densities of the two fluids, and μ_1 and μ_2 denote their constant dynamic viscosities. The conservations of mass/momentum and the second law of thermodynamics for this two-phase system lead to the following coupled system of equations (see e.g. [1,11] for details):

$$\rho \frac{D\mathbf{u}}{Dt} + \tilde{\mathbf{J}} \cdot \nabla \mathbf{u} = -\nabla p + \nabla \cdot [\mu \mathbf{D}(\mathbf{u})] - \lambda \nabla \cdot (\nabla \phi \nabla \phi) + \mathbf{f}(\mathbf{x}, t), \quad (1a)$$

$$\nabla \cdot \mathbf{u} = 0, \quad (1b)$$

$$\frac{D\phi}{Dt} = m\nabla^2 C + g(\mathbf{x}, t), \quad (1c)$$

$$C = -\lambda \nabla^2 \phi + h(\phi), \quad (1d)$$

where $\frac{D}{Dt} = \frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla$ is the material derivative, and $\mathbf{D}(\mathbf{u}) = \nabla \mathbf{u} + \nabla \mathbf{u}^T$. In the above equations, $\mathbf{u}(\mathbf{x}, t)$ is the velocity, $p(\mathbf{x}, t)$ is the pressure, and $\mathbf{f}(\mathbf{x}, t)$ is an external body force, where t is time and \mathbf{x} is the spatial coordinate and $(\cdot)^T$ denotes the transpose of (\cdot) . $\phi(\mathbf{x}, t)$ denotes the phase field function, $-1 \leq \phi \leq 1$. The flow regions with $\phi = 1$ and $\phi = -1$ respectively represent the first and the second fluids. The iso-surface $\phi(\mathbf{x}, t) = 0$ marks the interface between the two fluids at time t . The function $h(\phi)$ in equation (1d) is given by

$$h(\phi) = F'(\phi) = \frac{\lambda}{\eta^2} \phi(\phi^2 - 1) \quad \text{with} \quad F(\phi) = \frac{\lambda}{4\eta^2} (1 - \phi^2)^2, \quad (2)$$

where η is a characteristic length scale of the interface thickness, and $F(\phi)$ is the potential free energy density (double-well function) of the system. C in the above equations denotes the chemical potential, satisfying the relation $C = \frac{\delta W}{\delta \phi}$, where W is the free energy of the system given by $W = \int_{\Omega} [\frac{\lambda}{2} \nabla \phi \cdot \nabla \phi + F(\phi)]$. λ is the mixing energy density coefficient and is related to the surface tension by [49]

$$\lambda = \frac{3}{2\sqrt{2}} \sigma \eta, \quad (3)$$

where σ is the interface surface tension and is assumed to be a constant. $m > 0$ is the mobility of the interface, and is assumed to be a constant in the current paper. The density ρ and the dynamic viscosity μ of the two-phase mixture are related to the phase field function by

$$\rho(\phi) = \frac{\rho_1 + \rho_2}{2} + \frac{\rho_1 - \rho_2}{2} \phi, \quad \mu(\phi) = \frac{\mu_1 + \mu_2}{2} + \frac{\mu_1 - \mu_2}{2} \phi. \quad (4)$$

In equation (1a), $\tilde{\mathbf{J}}$ is given by

$$\tilde{\mathbf{J}}(\phi) = J_0 \nabla C = J_0 \nabla (-\lambda \nabla^2 \phi + h(\phi)), \quad \text{with} \quad J_0 = -\frac{1}{2}(\rho_1 - \rho_2)m. \quad (5)$$

Note that $\tilde{\mathbf{J}}$ and ρ given above satisfy the following relation

$$\frac{D\rho}{Dt} = -\nabla \cdot \tilde{\mathbf{J}}. \quad (6)$$

$g(\mathbf{x}, t)$ in equation (1c) is a prescribed source term for the purpose of numerical testing only, and will be set to $g = 0$ in actual simulations.

To facilitate subsequent development of the unconditionally energy-stable scheme, we transform equation (1a) into an equivalent form:

$$\rho \frac{D\mathbf{u}}{Dt} + \tilde{\mathbf{J}} \cdot \nabla \mathbf{u} = -\nabla P + \nabla \cdot [\mu \mathbf{D}(\mathbf{u})] + C \nabla \phi + \mathbf{f}(\mathbf{x}, t), \quad (7)$$

where $P = p + \frac{\lambda}{2} \nabla \phi \cdot \nabla \phi + F(\phi)$ is an effective pressure and we have used the following identity

$$\nabla \cdot (\nabla \phi \nabla \phi) = \frac{1}{2} \nabla (\nabla \phi \cdot \nabla \phi) + (\nabla^2 \phi) \nabla \phi.$$

It can be shown that the system consisting of equations (1a)–(1d), with $\mathbf{f} = 0$ and $g = 0$, satisfies the following energy law (assuming that all boundary flux terms vanish),

$$\frac{d}{dt} \int_{\Omega} \left[\frac{1}{2} \rho |\mathbf{u}|^2 + \frac{\lambda}{2} \nabla \phi \cdot \nabla \phi + F(\phi) \right] = - \int_{\Omega} \frac{\mu}{2} \|\nabla \mathbf{u}\|^2 - \int_{\Omega} m \nabla C \cdot \nabla C. \quad (8)$$

Equations (7), (1b)–(1d) are to be supplemented by appropriate boundary and initial conditions for the velocity and the phase field function. We assume the following boundary conditions:

$$\mathbf{u}|_{\partial\Omega} = \mathbf{w}(\mathbf{x}, t), \quad (9)$$

$$\mathbf{n} \cdot \nabla \phi|_{\partial\Omega} = d_a(\mathbf{x}, t), \quad (10)$$

$$\mathbf{n} \cdot \nabla (\nabla^2 \phi)|_{\partial\Omega} = d_b(\mathbf{x}, t), \quad (11)$$

and the initial conditions:

$$\mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_{in}(\mathbf{x}), \quad (12)$$

$$\phi(\mathbf{x}, 0) = \phi_{in}(\mathbf{x}). \quad (13)$$

In the above equations d_a and d_b are prescribed source terms for the purpose of numerical testing only, and will be set to $d_a = 0$ and $d_b = 0$ in actual simulations. The boundary conditions (10) and (11) with $d_a = 0$ and $d_b = 0$ correspond to a solid-wall boundary of neutral wettability, i.e. contact angle is 90° when the fluid interface intersects the wall. $\mathbf{u}_{in}(\mathbf{x})$ and $\phi_{in}(\mathbf{x})$ are the initial velocity and phase field distributions.

2.2. Reformulated equivalent system of equations

To derive the energy-stable scheme for the system (7), (1b)–(1d), we introduce a shifted energy consisting of the kinetic energy and the potential free energy component as follows,

$$E(t) = \int_{\Omega} \left[\frac{1}{2} \rho |\mathbf{u}|^2 + F(\phi) \right] d\Omega + C_0 \quad (14)$$

where C_0 is a chosen constant such that $E(t) > 0$ for all $t \geq 0$. Note that $E(t)$ is a scalar-valued number, not a field function. Define an auxiliary variable

$$R(t) = \sqrt{E(t)}. \quad (15)$$

With the help of the identities

$$\begin{cases} \rho \frac{D\mathbf{u}}{Dt} \cdot \mathbf{u} = \frac{D}{Dt} \left(\frac{1}{2} \rho |\mathbf{u}|^2 \right) - \frac{D\rho}{Dt} \left(\frac{1}{2} |\mathbf{u}|^2 \right) \\ (\tilde{\mathbf{J}} \cdot \nabla \mathbf{u}) \cdot \mathbf{u} = \nabla \cdot \left(\frac{1}{2} |\mathbf{u}|^2 \tilde{\mathbf{J}} \right) - \nabla \cdot \tilde{\mathbf{J}} \left(\frac{1}{2} |\mathbf{u}|^2 \right) \\ \mathbf{u} \cdot \nabla \left(\frac{1}{2} \rho |\mathbf{u}|^2 \right) = \nabla \cdot \left(\frac{1}{2} \rho |\mathbf{u}|^2 \mathbf{u} \right) - (\nabla \cdot \mathbf{u}) \frac{1}{2} \rho |\mathbf{u}|^2 \end{cases} \quad (16)$$

and the equations (1b) and (6), we can obtain that

$$\frac{\partial}{\partial t} \left(\frac{1}{2} \rho |\mathbf{u}|^2 \right) = \rho \frac{D\mathbf{u}}{Dt} \cdot \mathbf{u} + (\tilde{\mathbf{J}} \cdot \nabla \mathbf{u}) \cdot \mathbf{u} - \nabla \cdot \left(\frac{1}{2} \tilde{\mathbf{J}} |\mathbf{u}|^2 \right) - \nabla \cdot \left(\frac{1}{2} \rho |\mathbf{u}|^2 \mathbf{u} \right). \quad (17)$$

Thus, taking the derivative of $R(t)$ leads to

$$\begin{aligned} 2R \frac{dR}{dt} &= \frac{dE}{dt} = \int_{\Omega} \frac{\partial}{\partial t} \left[\frac{1}{2} \rho |\mathbf{u}|^2 + F(\phi) \right] \\ &= \int_{\Omega} \left[\rho \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) + \tilde{\mathbf{J}} \cdot \nabla \mathbf{u} \right] \cdot \mathbf{u} + \int_{\Omega} h(\phi) \frac{\partial \phi}{\partial t} - \int_{\partial\Omega} (\mathbf{n} \cdot \mathbf{u}) \frac{1}{2} \rho |\mathbf{u}|^2 - \int_{\partial\Omega} (\mathbf{n} \cdot \tilde{\mathbf{J}}) \frac{1}{2} |\mathbf{u}|^2, \end{aligned} \quad (18)$$

where \mathbf{n} is the outward-pointing unit vector normal to the boundary $\partial\Omega$, and we have used equation (17), integration by part, and the divergence theorem. It is crucial to note that both $E(t)$ and $R(t)$ are scalar-valued variables depending only on t , and the fact that

$$\frac{R(t)}{\sqrt{E(t)}} = 1. \quad (19)$$

In light of equation (19), we can re-write the system consisting of equations (7), (1b)–(1d) into the following equivalent form

$$\begin{aligned} \rho \frac{\partial \mathbf{u}}{\partial t} + \frac{R(t)}{\sqrt{E(t)}} \left\{ \rho \mathbf{u} \cdot \nabla \mathbf{u} + \tilde{\mathbf{J}} \cdot \nabla \mathbf{u} - \nabla \mu \cdot \mathbf{D}(\mathbf{u}) + \left(1 - \frac{\rho}{\rho_0} \right) \nabla P + (\underline{\mu} - \underline{\nu_m \rho}) \nabla \times \nabla \times \mathbf{u} \right\} \\ = -\frac{\rho}{\rho_0} \nabla P + \underline{\nu_m \rho} \nabla^2 \mathbf{u} + C \nabla \phi + \mathbf{f}, \end{aligned} \quad (20)$$

$$\nabla \cdot \mathbf{u} = 0, \quad (21)$$

$$\frac{\partial \phi}{\partial t} + \frac{R(t)}{\sqrt{E(t)}} \mathbf{u} \cdot \nabla \phi = m \nabla^2 C + g, \quad (22)$$

$$\mathcal{C} = -\lambda \nabla^2 \phi + S(\phi - \phi) + \frac{R(t)}{\sqrt{E(t)}} h(\phi), \quad (23)$$

$$\begin{aligned} 2R \frac{dR}{dt} &= \int_{\Omega} \rho \frac{\partial \mathbf{u}}{\partial t} \cdot \mathbf{u} - \int_{\Omega} \mathcal{C} \nabla \phi \cdot \mathbf{u} + \frac{R(t)}{\sqrt{E(t)}} \int_{\Omega} \mathcal{C} \nabla \phi \cdot \mathbf{u} + \frac{R(t)}{\sqrt{E(t)}} \int_{\Omega} h(\phi) \frac{\partial \phi}{\partial t} \\ &\quad + \frac{R(t)}{\sqrt{E(t)}} \int_{\Omega} \left\{ \rho \mathbf{u} \cdot \nabla \mathbf{u} + \tilde{\mathbf{J}} \cdot \nabla \mathbf{u} - \nabla \mu \cdot \mathbf{D}(\mathbf{u}) + \left(1 - \frac{\rho}{\rho_0}\right) \nabla P + (\mu - v_m \rho) \nabla \times \nabla \times \mathbf{u} \right\} \cdot \mathbf{u} \\ &\quad + \int_{\Omega} \left\{ \nabla \cdot [\mu \mathbf{D}(\mathbf{u})] + \left(\frac{\rho}{\rho_0} - 1\right) \nabla P - v_m \rho \nabla^2 \mathbf{u} \right\} \cdot \mathbf{u} \\ &\quad - \frac{1}{2} \int_{\partial \Omega} (\mathbf{n} \cdot \mathbf{u}) \rho |\mathbf{u}|^2 - \frac{1}{2} \int_{\partial \Omega} (\mathbf{n} \cdot \tilde{\mathbf{J}}) |\mathbf{u}|^2. \end{aligned} \quad (24)$$

To obtain equations (20) and (24), we have used equation (1b) and the identities

$$\begin{cases} \nabla \cdot [\mu \mathbf{D}(\mathbf{u})] = \mu [\nabla^2 \mathbf{u} + \nabla(\nabla \cdot \mathbf{u})] + \nabla \mu \cdot \mathbf{D}(\mathbf{u}), \\ \nabla^2 \mathbf{u} = \nabla(\nabla \cdot \mathbf{u}) - \nabla \times \nabla \times \mathbf{u}, \end{cases}$$

and have added/subtracted appropriate terms such as $\frac{\rho}{\rho_0} \nabla P$, $v_m \rho \nabla^2 \mathbf{u}$, $S\phi$, and $\mathcal{C} \nabla \phi \cdot \mathbf{u}$. In these equations, ρ_0 is a constant given by $\rho_0 = \min(\rho_1, \rho_2)$, v_m is a chosen constant satisfying $v_m \geq \frac{1}{2} \max\left(\frac{\mu_1}{\rho_1}, \frac{\mu_2}{\rho_2}\right)$, and S is a chosen non-negative constant satisfying a condition to be specified later in equation (57).

The original system consisting of equations (7), (1b)–(1d), (9)–(13) is equivalent to the reformulated system consisting of equations (20)–(24), together with the boundary conditions (9)–(11), and the initial conditions (12)–(13), supplemented by an extra initial condition for $R(t)$, i.e.

$$R(0) = \left(\int_{\Omega} \left[\frac{1}{2} \rho(\phi_{in}) |\mathbf{u}_{in}|^2 + F(\phi_{in}) \right] d\Omega + C_0 \right)^{1/2}. \quad (25)$$

In the reformulated system, the dynamic variables are $\mathbf{u}(\mathbf{x}, t)$, $P(\mathbf{x}, t)$, $\phi(\mathbf{x}, t)$ and $R(t)$. Note that $E(t)$ is computed by equation (14). We next focus on this reformulated equivalent system of equations, and present an unconditionally energy-stable scheme for approximating this system.

2.3. Formulation of numerical scheme

Let $n \geq 0$ denote the time step index, and $(\cdot)^n$ denote the variable (\cdot) at time step n . Let J ($J = 1$ or 2) denote the temporal order of accuracy of the scheme. We set

$$\phi^0 = \phi_{in}, \quad \mathbf{u}^0 = \mathbf{u}_{in}, \quad R^0 = R(0), \quad (26)$$

where $R(0)$ is given in equation (25). Define a scalar-valued variable ξ^{n+1}

$$\xi^{n+1} = \frac{R^{n+1}}{\sqrt{E^{n+1}}}. \quad (27)$$

Given (ϕ, \mathbf{u}, P, R) at time step n and previous time steps, we compute $(\phi^{n+1}, \mathbf{u}^{n+1}, P^{n+1}, R^{n+1})$ through the following scheme

$$\begin{aligned} \bar{\rho}^{n+1} \left(\frac{\gamma_0 \mathbf{u}^{n+1} - \hat{\mathbf{u}}}{\Delta t} \right) + \xi^{n+1} \bar{\rho}^{n+1} \mathcal{N} &= \\ &\quad - \frac{\bar{\rho}^{n+1}}{\rho_0} \nabla P^{n+1} + v_m \bar{\rho}^{n+1} \nabla^2 \mathbf{u}^{n+1} + \mathcal{C}^{n+1} \nabla \bar{\phi}^{n+1} + \mathbf{f}^{n+1}, \end{aligned} \quad (28)$$

$$\nabla \cdot \mathbf{u}^{n+1} = 0, \quad (29)$$

$$\frac{\gamma_0 \phi^{n+1} - \hat{\phi}}{\Delta t} + \xi^{n+1} (\mathbf{u}^{*,n+1} \cdot \nabla \phi^{*,n+1}) = m \nabla^2 \mathcal{C}^{n+1} + g^{n+1}, \quad (30)$$

$$\mathcal{C}^{n+1} = -\lambda \nabla^2 \phi^{n+1} + S(\phi^{n+1} - \phi^{*,n+1}) + \xi^{n+1} h(\phi^{*,n+1}), \quad (31)$$

$$\begin{aligned}
2R^{n+1} \frac{\gamma_0 R^{n+1} - \hat{R}}{\Delta t} &= \int_{\Omega} \bar{\rho}^{n+1} \left(\frac{\gamma_0 \mathbf{u}^{n+1} - \hat{\mathbf{u}}}{\Delta t} \right) \cdot \mathbf{u}^{n+1} - \int_{\Omega} C^{n+1} (\mathbf{u}^{n+1} \cdot \nabla \bar{\phi}^{n+1}) \\
&+ \xi^{n+1} \int_{\Omega} C^{n+1} (\mathbf{u}^{*,n+1} \cdot \nabla \phi^{*,n+1}) + \xi^{n+1} \int_{\Omega} h(\phi^{*,n+1}) \frac{\gamma_0 \phi^{n+1} - \hat{\phi}}{\Delta t} + \xi^{n+1} \int_{\Omega} \bar{\rho}^{n+1} \mathcal{N} \cdot \mathbf{u}^{n+1} \\
&+ \int_{\Omega} \left\{ \nabla \bar{\mu}^{n+1} \cdot \mathbf{D}(\mathbf{u}^{n+1}) + \left(\frac{\bar{\rho}^{n+1}}{\rho_0} - 1 \right) \nabla P^{n+1} + (\bar{\mu}^{n+1} - \nu_m \bar{\rho}^{n+1}) \nabla^2 \mathbf{u}^{n+1} \right\} \cdot \mathbf{u}^{n+1} \\
&- \frac{1}{2} \int_{\partial\Omega} (\mathbf{n} \cdot \mathbf{u}^{n+1}) \bar{\rho}^{n+1} |\mathbf{u}^{n+1}|^2 - \frac{1}{2} \int_{\partial\Omega} (\mathbf{n} \cdot \tilde{\mathbf{J}}^{n+1}) |\mathbf{u}^{n+1}|^2,
\end{aligned} \tag{32}$$

$$\mathbf{u}^{n+1} = \mathbf{w}^{n+1} \quad \text{on } \partial\Omega, \tag{33}$$

$$\mathbf{n} \cdot \nabla \phi^{n+1} = d_a^{n+1} \quad \text{on } \partial\Omega \tag{34}$$

$$\mathbf{n} \cdot \nabla (\nabla^2 \phi^{n+1}) = d_b^{n+1} \quad \text{on } \partial\Omega. \tag{35}$$

The symbols in the above equations are defined as follows. In equations (28) and (32)

$$\mathcal{N} = \mathbf{Q} + \left(\frac{\bar{\mu}^{n+1}}{\bar{\rho}^{n+1}} - \nu_m \right) \nabla \times \nabla \times \mathbf{u}^{*,n+1}, \tag{36}$$

and

$$\begin{aligned}
\mathbf{Q} = \mathbf{Q}(\bar{\phi}^{n+1}, \mathbf{u}^{*,n+1}) &= \mathbf{u}^{*,n+1} \cdot \nabla \mathbf{u}^{*,n+1} + \frac{1}{\bar{\rho}^{n+1}} \tilde{\mathbf{J}}^{n+1} \cdot \nabla \mathbf{u}^{*,n+1} - \frac{1}{\bar{\rho}^{n+1}} \nabla \bar{\mu}^{n+1} \cdot \mathbf{D}(\mathbf{u}^{*,n+1}) \\
&+ \left(\frac{1}{\bar{\rho}^{n+1}} - \frac{1}{\rho_0} \right) \nabla P^{*,n+1}.
\end{aligned} \tag{37}$$

In equation (27), E^{n+1} is given by (see equation (14))

$$E^{n+1} = \int_{\Omega} \left[\frac{1}{2} \bar{\rho}^{n+1} |\mathbf{u}^{n+1}|^2 + F(\phi^{n+1}) \right] d\Omega + C_0. \tag{38}$$

In these equations, $\bar{\phi}^{n+1}$ is a J -order approximation of ϕ to be specified later in equation (66), and $\bar{\rho}^{n+1}$, $\bar{\mu}^{n+1}$ and $\tilde{\mathbf{J}}^{n+1}$ are given by

$$\bar{\rho}^{n+1} = \rho(\bar{\phi}^{n+1}), \quad \bar{\mu}^{n+1} = \mu(\bar{\phi}^{n+1}), \quad \tilde{\mathbf{J}}^{n+1} = \tilde{\mathbf{J}}(\bar{\phi}^{n+1}); \tag{39}$$

see equations (4)–(5). Let χ denote a generic variable. Then in equations (28)–(32), $\frac{1}{\Delta t}(\gamma_0 \chi^{n+1} - \hat{\chi})$ represents an approximation of $\left. \frac{\partial \chi}{\partial t} \right|^{n+1}$ with the J -th order backward differentiation formula (BDF), with γ_0 and $\hat{\chi}$ given by

$$\hat{\chi} = \begin{cases} \chi^n, & J=1, \\ 2\chi^n - \frac{1}{2}\chi^{n-1}, & J=2; \end{cases} \quad \gamma_0 = \begin{cases} 1, & J=1, \\ 3/2, & J=2. \end{cases} \tag{40}$$

$\chi^{*,n+1}$ denotes a J -th order explicit approximation of χ^{n+1} given by

$$\chi^{*,n+1} = \begin{cases} \chi^n, & J=1, \\ 2\chi^n - \chi^{n-1}, & J=2. \end{cases} \tag{41}$$

It should be noted that the different approximations for various terms in equations (32), (28) and (30) are crucial. They are to ensure a discrete energy stability property (see Section 2.4), and simultaneously allow an implementation in which the resultant linear algebraic systems involve only constant and time-independent coefficient matrices after discretization.

2.4. Discrete energy law

Our numerical scheme satisfies a discrete energy stability property, and thus can be potentially favorable for long-time simulations. More specifically, the following discrete energy law holds:

Theorem 2.1. *In the absence of the external force \mathbf{f}^{n+1} and source term g^{n+1} , and with zero boundary conditions $\mathbf{w}^{n+1} = \mathbf{0}$, $d_a^{n+1} = d_b^{n+1} = 0$, the scheme consisting of equations (27)–(35) satisfies the following property:*

$$\mathcal{E}^{n+1} - \mathcal{E}^n = -D^{n+1} - \frac{1}{2} \int_{\Omega} \bar{\mu}^{n+1} \|\mathcal{D}(\mathbf{u}^{n+1})\|^2 - \int_{\Omega} m |\nabla \mathcal{C}^{n+1}|^2, \quad (42)$$

where \mathcal{E}^n is a discrete energy

$$\mathcal{E}^n = \begin{cases} \frac{\lambda}{2\Delta t} \|\nabla \phi^n\|^2 + \frac{1}{\Delta t} |R^n|^2, & J = 1, \\ \frac{S}{2\Delta t} \|\phi^n - \phi^{n-1}\|^2 + \frac{\lambda}{4\Delta t} (\|\nabla \phi^n\|^2 + \|\nabla \phi^{*,n}\|^2) + \frac{1}{2\Delta t} (|R^n|^2 + |R^{*,n}|^2), & J = 2, \end{cases} \quad (43)$$

and D^{n+1} is the discrete dissipation

$$D^{n+1} = \begin{cases} \frac{S}{\Delta t} \|\phi^{n+1} - \phi^n\|^2 + \frac{\lambda}{2\Delta t} \|\nabla \phi^{n+1} - \nabla \phi^n\|^2 + \frac{1}{\Delta t} |R^{n+1} - R^n|^2, & J = 1, \\ \frac{S}{\Delta t} \|\phi^{n+1} - \phi^{*,n+1}\|^2 + \frac{\lambda}{4\Delta t} \|\nabla \phi^{n+1} - \nabla \phi^{*,n+1}\|^2 + \frac{1}{2\Delta t} |R^{n+1} - R^{*,n+1}|^2, & J = 2. \end{cases} \quad (44)$$

Proof. Multiplying equation (28) by \mathbf{u}^{n+1} , equation (30) by \mathcal{C}^{n+1} , and equation (31) by $-\frac{\gamma_0 \phi^{n+1} - \hat{\phi}}{\Delta t}$, and taking the L^2 inner product lead to

$$\begin{aligned} & \int_{\Omega} \bar{\rho}^{n+1} \left(\frac{\gamma_0 \mathbf{u}^{n+1} - \hat{\mathbf{u}}}{\Delta t} \right) \cdot \mathbf{u}^{n+1} + \xi^{n+1} \int_{\Omega} \bar{\rho}^{n+1} \mathcal{N} \cdot \mathbf{u}^{n+1} \\ &= \int_{\Omega} \left\{ -\frac{\bar{\rho}^{n+1}}{\rho_0} \nabla P^{n+1} + \nu_m \bar{\rho}^{n+1} \nabla^2 \mathbf{u}^{n+1} \right\} \cdot \mathbf{u}^{n+1} + \int_{\Omega} \mathcal{C}^{n+1} (\mathbf{u}^{n+1} \cdot \nabla \bar{\phi}^{n+1}) + \int_{\Omega} \mathbf{f}^{n+1} \cdot \mathbf{u}^{n+1}, \\ & \int_{\Omega} \frac{\gamma_0 \phi^{n+1} - \hat{\phi}}{\Delta t} \mathcal{C}^{n+1} + \xi^{n+1} \int_{\Omega} \mathcal{C}^{n+1} (\mathbf{u}^{*,n+1} \cdot \nabla \phi^{*,n+1}) = m \int_{\Omega} \mathcal{C}^{n+1} \nabla^2 \mathcal{C}^{n+1} + \int_{\Omega} \mathbf{g}^{n+1} \mathcal{C}^{n+1}, \\ & \int_{\Omega} \frac{\hat{\phi} - \gamma_0 \phi^{n+1}}{\Delta t} \mathcal{C}^{n+1} = \frac{\lambda}{\Delta t} \int_{\Omega} \nabla^2 \phi^{n+1} (\gamma_0 \phi^{n+1} - \hat{\phi}) - \frac{S}{\Delta t} \int_{\Omega} (\phi^{n+1} - \phi^{*,n+1}) (\gamma_0 \phi^{n+1} - \hat{\phi}) \\ & \quad - \xi^{n+1} \int_{\Omega} h(\phi^{*,n+1}) \frac{\gamma_0 \phi^{n+1} - \hat{\phi}}{\Delta t}. \end{aligned} \quad (45)$$

Summing up the three equations in (45) and the equation (32), we arrive at

$$\begin{aligned} & \frac{2}{\Delta t} R^{n+1} (\gamma_0 R^{n+1} - \hat{R}) = - \int_{\Omega} \nabla P^{n+1} \cdot \mathbf{u}^{n+1} + \int_{\Omega} \nabla \cdot [\bar{\mu}^{n+1} \mathcal{D}(\mathbf{u}^{n+1})] \cdot \mathbf{u}^{n+1} + m \int_{\Omega} \mathcal{C}^{n+1} \nabla^2 \mathcal{C}^{n+1} \\ & + \frac{\lambda}{\Delta t} \int_{\Omega} \nabla^2 \phi^{n+1} (\gamma_0 \phi^{n+1} - \hat{\phi}) - \frac{S}{\Delta t} \int_{\Omega} (\phi^{n+1} - \phi^{*,n+1}) (\gamma_0 \phi^{n+1} - \hat{\phi}) + \int_{\Omega} \mathbf{f}^{n+1} \cdot \mathbf{u}^{n+1} + \int_{\Omega} \mathbf{g}^{n+1} \mathcal{C}^{n+1} \\ & - \frac{1}{2} \int_{\partial\Omega} (\mathbf{n} \cdot \mathbf{u}^{n+1}) \bar{\rho}^{n+1} |\mathbf{u}^{n+1}|^2 - \frac{1}{2} \int_{\partial\Omega} (\mathbf{n} \cdot \tilde{\mathbf{J}}^{n+1}) |\mathbf{u}^{n+1}|^2. \end{aligned} \quad (46)$$

By equation (29), integration by part and the divergence theorem, we obtain that

$$\int_{\Omega} \nabla P^{n+1} \cdot \mathbf{u}^{n+1} = \int_{\partial\Omega} P^{n+1} (\mathbf{n} \cdot \mathbf{u}^{n+1}). \quad (47)$$

Using equation (31) and the boundary conditions (34) and (35), we have

$$\mathbf{n} \cdot \nabla \mathcal{C}^{n+1} = -\lambda d_b^{n+1} + S(d_a^{n+1} - d_a^{*,n+1}) + \xi^{n+1} h'(\phi^{*,n+1}) d_a^{*,n+1} \text{ on } \partial\Omega. \quad (48)$$

With the help of equation (47) and the relations

$$\begin{cases} \int_{\Omega} \nabla \cdot [\bar{\mu}^{n+1} \mathcal{D}(\mathbf{u}^{n+1})] \cdot \mathbf{u}^{n+1} = \int_{\partial\Omega} \bar{\mu}^{n+1} \mathbf{n} \cdot \mathcal{D}(\mathbf{u}^{n+1}) \cdot \mathbf{u}^{n+1} - \frac{1}{2} \int_{\Omega} \bar{\mu}^{n+1} \|\mathcal{D}(\mathbf{u}^{n+1})\|^2, \\ \int_{\Omega} \mathcal{C}^{n+1} \nabla^2 \mathcal{C}^{n+1} = \int_{\partial\Omega} (\mathbf{n} \cdot \nabla \mathcal{C}^{n+1}) \mathcal{C}^{n+1} - \int_{\Omega} |\nabla \mathcal{C}^{n+1}|^2, \\ \int_{\Omega} \nabla^2 \phi^{n+1} (\gamma_0 \phi^{n+1} - \hat{\phi}) = \int_{\partial\Omega} (\mathbf{n} \cdot \nabla \phi^{n+1}) (\gamma_0 \phi^{n+1} - \hat{\phi}) - \int_{\Omega} \nabla \phi^{n+1} \cdot (\gamma_0 \nabla \phi^{n+1} - \nabla \hat{\phi}), \end{cases} \quad (49)$$

equation (46) can be transformed into

$$\begin{aligned} \frac{2}{\Delta t} R^{n+1} (\gamma_0 R^{n+1} - \hat{R}) &= -\frac{1}{2} \int_{\Omega} \bar{\mu}^{n+1} \|\mathcal{D}(\mathbf{u}^{n+1})\|^2 - \int_{\Omega} m |\nabla C^{n+1}|^2 + \int_{\Omega} \mathbf{f}^{n+1} \cdot \mathbf{u}^{n+1} + \int_{\Omega} g^{n+1} C^{n+1} \\ &\quad - \frac{\lambda}{\Delta t} \int_{\Omega} \nabla \phi^{n+1} \cdot (\gamma_0 \nabla \phi^{n+1} - \nabla \hat{\phi}) - \frac{S}{\Delta t} \int_{\Omega} (\phi^{n+1} - \phi^{*,n+1}) (\gamma_0 \phi^{n+1} - \hat{\phi}) \\ &\quad + \int_{\partial\Omega} m (-\lambda d_b^{n+1} + S(d_a^{n+1} - d_a^{*,n+1}) + \xi^{n+1} h'(\phi^{*,n+1}) d_a^{*,n+1}) C^{n+1} + \frac{\lambda}{\Delta t} \int_{\partial\Omega} d_a^{n+1} (\gamma_0 \phi^{n+1} - \hat{\phi}) \\ &\quad + \int_{\partial\Omega} \left[-P^{n+1} \mathbf{n} + \bar{\mu}^{n+1} \mathbf{n} \cdot \mathcal{D}(\mathbf{u}^{n+1}) - \frac{1}{2} \bar{\rho}^{n+1} (\mathbf{n} \cdot \mathbf{w}^{n+1}) \mathbf{w}^{n+1} - \frac{1}{2} (\mathbf{n} \cdot \tilde{\mathbf{J}}^{n+1}) \mathbf{w}^{n+1} \right] \cdot \mathbf{w}^{n+1}, \end{aligned} \quad (50)$$

where we have used the boundary conditions (33)–(35) and (47)–(48).

Note the following relations for a generic variable χ ,

$$\begin{cases} \chi^{n+1} (\gamma_0 \chi^{n+1} - \hat{\chi}) = \frac{1}{2} (|\chi^{n+1}|^2 - |\chi^n|^2 + |\chi^{n+1} - \chi^n|^2) \\ (\chi^{n+1} - \chi^{*,n+1}) (\gamma_0 \chi^{n+1} - \hat{\chi}) = |\chi^{n+1} - \chi^n|^2, \text{ for } J = 1, \end{cases}$$

and

$$\begin{cases} \chi^{n+1} (\gamma_0 \chi^{n+1} - \hat{\chi}) = \frac{1}{4} (|\chi^{n+1}|^2 - |\chi^n|^2) + \frac{1}{4} |\chi^{n+1} - 2\chi^n + \chi^{n-1}|^2 + \frac{1}{4} (|2\chi^{n+1} - \chi^n|^2 - |2\chi^n - \chi^{n-1}|^2) \\ (\chi^{n+1} - \chi^{*,n+1}) (\gamma_0 \chi^{n+1} - \hat{\chi}) = \frac{1}{2} (|\chi^{n+1} - \chi^n|^2 - |\chi^n - \chi^{n-1}|^2) + |\chi^{n+1} - 2\chi^n + \chi^{n-1}|^2, \text{ for } J = 2. \end{cases}$$

Combining the above relations with equation (50) and letting $\mathbf{f}^{n+1} = \mathbf{0}$, $g^{n+1} = 0$ in Ω , and $\mathbf{w}^{n+1} = \mathbf{0}$, $d_a^{n+1} = d_b^{n+1} = 0$ on $\partial\Omega$, all the boundary terms vanish, and equation (50) is transformed into

$$\begin{aligned} \frac{S}{\Delta t} \|\phi^{n+1} - \phi^n\|^2 + \frac{\lambda}{2\Delta t} (\|\nabla \phi^{n+1}\|^2 + \|\nabla \phi^{n+1} - \nabla \phi^n\|^2) + \frac{1}{\Delta t} (|R^{n+1}|^2 + |R^{n+1} - R^n|^2) \\ = \frac{\lambda}{2\Delta t} \|\nabla \phi^n\|^2 + \frac{1}{\Delta t} |R^n|^2 - \frac{1}{2} \int_{\Omega} \bar{\mu}^{n+1} \|\mathcal{D}(\mathbf{u}^{n+1})\|^2 - \int_{\Omega} m |\nabla C^{n+1}|^2 \text{ for } J = 1, \end{aligned} \quad (51)$$

and

$$\begin{aligned} \frac{S}{\Delta t} \left(\frac{1}{2} \|\phi^{n+1} - \phi^n\|^2 + \|\phi^{n+1} - 2\phi^n + \phi^{n-1}\|^2 \right) \\ + \frac{\lambda}{4\Delta t} (\|\nabla \phi^{n+1}\|^2 + \|\nabla \phi^{n+1} - 2\nabla \phi^n + \nabla \phi^{n-1}\|^2 + \|2\nabla \phi^{n+1} - \nabla \phi^n\|^2) \\ + \frac{1}{2\Delta t} (|R^{n+1}|^2 + |R^{n+1} - 2R^n + R^{n-1}|^2 + |2R^{n+1} - R^n|^2) \\ = \frac{\lambda}{4\Delta t} (\|\nabla \phi^n\|^2 + \|2\nabla \phi^n - \nabla \phi^{n-1}\|^2) + \frac{S}{2\Delta t} \|\phi^n - \phi^{n-1}\|^2 \\ + \frac{1}{2\Delta t} (|R^n|^2 + |2R^n - R^{n-1}|^2) - \frac{1}{2} \int_{\Omega} \bar{\mu}^{n+1} \|\mathcal{D}(\mathbf{u}^{n+1})\|^2 - \int_{\Omega} m |\nabla C^{n+1}|^2, \text{ for } J = 2. \end{aligned} \quad (52)$$

The energy stability result in Theorem 2.1 can be obtained directly from the above equations (51)–(52) by defining the discrete energy and dissipation in equations (43)–(44). \square

Remark 1. Note that in equation (31) an extra zero term, $S(\phi^{n+1} - 2\phi^n + \phi^{n-1})$, has been included in the scheme, with S being a constant (non-negative) satisfying a condition given later in (57). Similar terms can be found in other works from the literature (see e.g. [40,41], among others) for approximating the Cahn-Hilliard or Allen-Cahn equations, in which such a term is often referred to as a stabilization term because it is crucial to the proof of stability of those schemes. From the discussions of this subsection it is evident that this term is not essential to the stability of the current scheme. Theorem 2.1 still holds even when one sets $S = 0$. The goal for the S term in the current work is to simplify the implementation of the scheme herein. Inclusion of this term will allow the fourth-order Cahn-Hilliard equation to be reformulated into two de-coupled Helmholtz-type equations, making its solution simpler. This point will become clearer when discussing the solution algorithm in the subsequent subsection.

2.5. Efficient solution algorithm

We next consider how to implement the algorithm represented by the equations (27)–(35). Although E^{n+1} , R^{n+1} as well as ξ^{n+1} are implicit and E^{n+1} involves the integral of the unknown field functions \mathbf{u}^{n+1} and ϕ^{n+1} over the domain, the scheme can be implemented in an efficient way.

Combining equations (31) and (30) leads to

$$\frac{\gamma_0 \phi^{n+1} - \hat{\phi}}{\Delta t} + \xi^{n+1} (\mathbf{u}^{*,n+1} \cdot \nabla \phi^{*,n+1}) = m \nabla^2 \left[-\lambda \nabla^2 \phi^{n+1} + S(\phi^{n+1} - \phi^{*,n+1}) + \xi^{n+1} h(\phi^{*,n+1}) \right] + g^{n+1}. \quad (53)$$

This equation can be re-written as

$$\frac{\gamma_0}{\lambda m \Delta t} \phi^{n+1} + \nabla^2 (\nabla^2 \phi^{n+1}) - \frac{S}{\lambda} \nabla^2 \phi^{n+1} = Z_1^n + \xi^{n+1} Z_2^n, \quad (54)$$

where

$$Z_1^n = \frac{1}{\lambda m} \left(\frac{\hat{\phi}}{\Delta t} - m S \nabla^2 \phi^{*,n+1} + g^{n+1} \right), \quad Z_2^n = \frac{1}{\lambda m} \left(m \nabla^2 h(\phi^{*,n+1}) - \mathbf{u}^{*,n+1} \cdot \nabla \phi^{*,n+1} \right). \quad (55)$$

Barring the unknown scalar-valued variable ξ^{n+1} , equation (54) is a fourth-order equation about ϕ^{n+1} , which can be transformed into two decoupled Helmholtz-type equations (see e.g. [49,15,47]). Basically, by adding/subtracting a term $\alpha \nabla^2 \phi^{n+1}$ (α denoting a constant to be determined) on the left hand side (LHS), we can transform equation (54) into

$$\nabla^2 \left(\nabla^2 \phi^{n+1} + \alpha \phi^{n+1} \right) - \left(\alpha + \frac{S}{\lambda} \right) \left[\nabla^2 \phi^{n+1} - \frac{\gamma_0}{m \Delta t (S + \alpha \lambda)} \phi^{n+1} \right] = Z_1^n + \xi^{n+1} Z_2^n. \quad (56)$$

By requiring that $\alpha = -\frac{\gamma_0}{m \Delta t (S + \alpha \lambda)}$, we obtain

$$\alpha = \frac{1}{2\lambda} \left[-S + \sqrt{S^2 - 4 \frac{\gamma_0 \lambda}{m \Delta t}} \right], \quad \text{and the condition } S \geq \sqrt{4 \frac{\gamma_0 \lambda}{m \Delta t}}. \quad (57)$$

The chosen constant S must satisfy the above condition. Therefore, equation (56) can be written equivalently as

$$\nabla^2 \psi^{n+1} - \left(\alpha + \frac{S}{\lambda} \right) \psi^{n+1} = Z_1^n + \xi^{n+1} Z_2^n, \quad (58a)$$

$$\nabla^2 \phi^{n+1} + \alpha \phi^{n+1} = \psi^{n+1}, \quad (58b)$$

where ψ^{n+1} is an auxiliary variable defined by equation (58b). Note that if the scalar variable ξ^{n+1} is given, equations (58a) and (58b) are decoupled. One can solve equation (58a) for ψ^{n+1} , and then solve equation (58b) for ϕ^{n+1} .

Correspondingly, the boundary condition (35) can be transformed into

$$\mathbf{n} \cdot \nabla \psi^{n+1} = \alpha d_a^{n+1} + d_b^{n+1}, \quad \text{on } \partial \Omega, \quad (59)$$

where we have used equation (58b) and (34).

Taking advantage of the fact that ξ^{n+1} is a scalar number, instead of a field function, we introduce two sets of phase field functions $(\psi_i^{n+1}, \phi_i^{n+1})$, $i = 1, 2$ as solutions to the following two Helmholtz-type problems:

$$\nabla^2 \psi_i^{n+1} - \left(\alpha + \frac{S}{\lambda} \right) \psi_i^{n+1} = Z_i^n, \quad i = 1, 2, \quad \mathbf{n} \cdot \nabla \psi_1^{n+1} = \alpha d_a^{n+1} + d_b^{n+1}, \quad \mathbf{n} \cdot \nabla \psi_2^{n+1} = 0, \quad \text{on } \partial \Omega; \quad (60a)$$

$$\nabla^2 \phi_i^{n+1} + \alpha \phi_i^{n+1} = \psi_i^{n+1}, \quad i = 1, 2, \quad \mathbf{n} \cdot \nabla \phi_1^{n+1} = d_a^{n+1}, \quad \mathbf{n} \cdot \nabla \phi_2^{n+1} = 0, \quad \text{on } \partial \Omega. \quad (60b)$$

Then, given scalar value ξ^{n+1} the following field functions solve the system consisting of equations (58a)–(59) and (34):

$$\psi^{n+1} = \psi_1^{n+1} + \xi^{n+1} \psi_2^{n+1}, \quad (61a)$$

$$\phi^{n+1} = \phi_1^{n+1} + \xi^{n+1} \phi_2^{n+1}, \quad (61b)$$

where $(\psi_i^{n+1}, \phi_i^{n+1})$ ($i = 1, 2$) are given by equations (60a)–(60b).

It should be noted that ψ_1^{n+1} and ψ_2^{n+1} defined in (60a) are not coupled, and ϕ_1^{n+1} and ϕ_2^{n+1} defined in (60b) are not coupled either. One can first compute ψ_1^{n+1} and ψ_2^{n+1} from (60a), and then compute ϕ_1^{n+1} and ϕ_2^{n+1} from (60b).

The corresponding weak formulations for $(\psi_i^{n+1}, \phi_i^{n+1})$ ($i = 1, 2$) can be obtained by taking the L^2 inner product between equations (60a)–(60b) and a test function, and they are as follows:

Find $\psi_1^{n+1} \in H^1(\Omega)$, such that

$$\begin{aligned} \int_{\Omega} \nabla \psi_1^{n+1} \cdot \nabla \varphi + \left(\alpha + \frac{S}{\lambda} \right) \int_{\Omega} \psi_1^{n+1} \varphi = & -\frac{S}{\lambda} \int_{\Omega} \nabla \phi^{*,n+1} \cdot \nabla \varphi - \frac{1}{\lambda m} \int_{\Omega} \left(\frac{\hat{\phi}}{\Delta t} + g^{n+1} \right) \varphi \\ & + \int_{\partial\Omega} \left[\frac{S}{\lambda} d_a^{*,n+1} + \alpha d_a^{n+1} + d_b^{n+1} \right] \varphi, \quad \forall \varphi \in H^1(\Omega). \end{aligned} \quad (62)$$

Find $\phi_1^{n+1} \in H^1(\Omega)$, such that

$$\int_{\Omega} \nabla \phi_1^{n+1} \cdot \nabla \varphi - \alpha \int_{\Omega} \phi_1^{n+1} \varphi = - \int_{\Omega} \psi_1^{n+1} \varphi + \int_{\partial\Omega} d_a^{n+1} \varphi, \quad \forall \varphi \in H^1(\Omega). \quad (63)$$

Find $\psi_2^{n+1} \in H^1(\Omega)$, such that

$$\begin{aligned} \int_{\Omega} \nabla \psi_2^{n+1} \cdot \nabla \varphi + \left(\alpha + \frac{S}{\lambda} \right) \int_{\Omega} \psi_2^{n+1} \varphi = & \frac{1}{\lambda} \int_{\Omega} \nabla h(\phi^{*,n+1}) \cdot \nabla \varphi \\ & + \frac{1}{\lambda m} \int_{\Omega} (\mathbf{u}^{*,n+1} \cdot \nabla \phi^{*,n+1}) \varphi - \frac{1}{\lambda} \int_{\partial\Omega} h'(\phi^{*,n+1}) d_a^{*,n+1}, \quad \forall \varphi \in H^1(\Omega). \end{aligned} \quad (64)$$

Find $\phi_2^{n+1} \in H^1(\Omega)$, such that

$$\int_{\Omega} \nabla \phi_2^{n+1} \cdot \nabla \varphi - \alpha \int_{\Omega} \phi_2^{n+1} \varphi = - \int_{\Omega} \psi_2^{n+1} \varphi, \quad \forall \varphi \in H^1(\Omega). \quad (65)$$

Remark 2. In the current implementation we compute the fourth-order phase field equation (barring the scalar number ξ^{n+1} still to be determined) by solving two de-coupled Helmholtz-type equations based on the decompositions in (60a) and (60b). This decomposition requires the chosen constant S to be sufficiently large (satisfying the condition given in (57)). If this condition is not satisfied the solution method will break down because the decompositions in (60a) and (60b) will no longer be valid. This is a restriction of the current implementation. Given a range of Δt values one wants to employ in simulations, it is straightforward to estimate an appropriate constant S to satisfy (57). For a given constant S , if one reduces Δt successively, the current solution method for the phase field equation will no longer work when Δt reaches a small enough value (determined by the lower bound in (57)). In this case, one can employ a larger constant S in order to perform simulations for smaller time step sizes. So this restriction on S is not an issue in practical simulations. Alternatively, one can allow S to vary with respect to Δt instead of being a constant in order to use this method, e.g. by always choosing S as the lower bound in (57). It should be noted, however, that with such a choice for S the temporal convergence rate will not be of second order because S is not a constant. It will be of order 3/2 if the lower bound in (57) is always used for S . This issue has been discussed in some detail in [47]. The restriction on S (condition (57)) can be eliminated if one chooses not to use the decompositions of (60a) and (60b) when solving the phase field equation. However, in this case one needs to solve two second-order equations in a coupled fashion, involving a non-symmetric coefficient matrix, which makes the implementation more involved and the resultant linear algebraic system more challenging to solve. Such a method has been presented in [47], in which the constant S is only required to be $S \geq 0$, but this is not implemented in the current work.

We define $\bar{\phi}^{n+1}$ as a J -th order approximation of ϕ^{n+1} given by

$$\bar{\phi}^{n+1} = \phi_1 + \phi_2. \quad (66)$$

Then $\bar{\rho}^{n+1}$, $\bar{\mu}^{n+1}$ and $\bar{\mathbf{j}}^{n+1}$ can be evaluated accordingly by equation (39).

Given $\{\psi_i^{n+1}, \phi_i^{n+1}\}$ ($i = 1, 2$), \mathcal{C}^{n+1} in (31) can be decomposed into

$$\mathcal{C}^{n+1} = \mathcal{C}_1^{n+1} + \xi^{n+1} \mathcal{C}_2^{n+1}, \quad (67)$$

where by equations (60b) and (61a)-(61b) we have

$$\mathcal{C}_1^{n+1} = -\lambda \psi_1^{n+1} + (\alpha \lambda + S) \phi_1^{n+1} - S \phi^{*,n+1}, \quad (68a)$$

$$\mathcal{C}_2^{n+1} = -\lambda \psi_2^{n+1} + (\alpha \lambda + S) \phi_2^{n+1} + h(\phi^{*,n+1}), \quad (68b)$$

Substituting (67) into (28) leads to

$$\begin{aligned} \nabla P^{n+1} = & -\rho_0 \nu_m \nabla \times \nabla \times \mathbf{u}^{n+1} - \frac{\gamma_0 \rho_0}{\Delta t} \mathbf{u}^{n+1} + \rho_0 \left\{ \frac{\hat{\mathbf{u}}}{\Delta t} + \frac{C_1^{n+1}}{\bar{\rho}^{n+1}} \nabla \bar{\phi}^{n+1} + \frac{\mathbf{f}^{n+1}}{\bar{\rho}^{n+1}} \right\} \\ & + \xi^{n+1} \rho_0 \left\{ \frac{C_2^{n+1}}{\bar{\rho}^{n+1}} \nabla \bar{\phi}^{n+1} - \mathbf{Q} - \left(\frac{\bar{\mu}^{n+1}}{\bar{\rho}^{n+1}} - \nu_m \right) \nabla \times \nabla \times \mathbf{u}^{*,n+1} \right\}, \end{aligned} \quad (69)$$

where we have used equation (29). Let $q(\mathbf{x})$ denote an arbitrary test function in the continuous space. Take the L^2 -inner product between ∇q and equation (69), and we obtain

$$\begin{aligned} \int_{\Omega} \nabla P^{n+1} \cdot \nabla q = & -\frac{\rho_0 \gamma_0}{\Delta t} \int_{\partial \Omega} \mathbf{n} \cdot \mathbf{w}^{n+1} q - \rho_0 \nu_m \int_{\partial \Omega} \mathbf{n} \times \boldsymbol{\omega}^{n+1} \cdot \nabla q \\ & + \rho_0 \int_{\Omega} \left(\frac{\hat{\mathbf{u}}}{\Delta t} + \frac{C_1^{n+1}}{\bar{\rho}^{n+1}} \nabla \bar{\phi}^{n+1} + \frac{\mathbf{f}^{n+1}}{\bar{\rho}^{n+1}} \right) \cdot \nabla q - \xi^{n+1} \rho_0 \int_{\partial \Omega} \left(\frac{\bar{\mu}^{n+1}}{\bar{\rho}^{n+1}} - \nu_m \right) \mathbf{n} \times \boldsymbol{\omega}^{*,n+1} \cdot \nabla q \\ & + \xi^{n+1} \rho_0 \int_{\Omega} \left(\frac{C_2^{n+1}}{\bar{\rho}^{n+1}} \nabla \bar{\phi}^{n+1} - \mathbf{Q} + \nabla \left(\frac{\bar{\mu}^{n+1}}{\bar{\rho}^{n+1}} \right) \times \boldsymbol{\omega}^{*,n+1} \right) \cdot \nabla q, \quad \forall q \end{aligned} \quad (70)$$

where $\boldsymbol{\omega} = \nabla \times \mathbf{u}$ is the vorticity, and we have used integration by part, the divergence theorem, equations (29) and (33), and the identity

$$\frac{\mu}{\rho} \nabla \times \boldsymbol{\omega} \cdot \nabla q = \nabla \cdot \left(\frac{\mu}{\rho} \boldsymbol{\omega} \times \nabla q \right) - \nabla \left(\frac{\mu}{\rho} \right) \times \boldsymbol{\omega} \cdot \nabla q. \quad (71)$$

Note that in equation (70), P^{n+1} is weakly coupled with \mathbf{u}^{n+1} as the boundary integral involves $\boldsymbol{\omega}^{n+1} = \nabla \times \mathbf{u}^{n+1}$. We have implemented two methods to solve equation (70):

- We solve equation (70), and the velocity equation (78) below, via a sub-iteration procedure (fixed-point iteration). We start by setting $\mathbf{n} \times \boldsymbol{\omega}^{n+1}|_{\partial \Omega} = \mathbf{n} \times \boldsymbol{\omega}^{*,n+1}|_{\partial \Omega}$ on the right hand side (RHS) of (70), and solve (70) and (78) alternately until convergence. See Remark 3 below for details.
- We make the following approximation on the boundary,

$$\mathbf{n} \times \boldsymbol{\omega}^{n+1}|_{\partial \Omega} \approx \frac{R^{n+1}}{\sqrt{E^{n+1}}} \mathbf{n} \times \boldsymbol{\omega}^{*,n+1}|_{\partial \Omega} = \xi^{n+1} \mathbf{n} \times \boldsymbol{\omega}^{*,n+1}|_{\partial \Omega}. \quad (72)$$

Then, except for the scalar number ξ^{n+1} , the computation for P^{n+1} from equation (70) is not coupled with the computation for the velocity \mathbf{u}^{n+1} . This implementation is similar to the first step in the sub-iteration procedure (but without sub-iteration). How then to deal with ξ^{n+1} in equations (70) and (78) with the approximation (72) is described below.

We observe that there is essentially no difference in the simulation results obtained with these two implementations, and that the approximation (72) on the boundary does not weaken the stability of the method, e.g. the ability to use large Δt in simulations. In Section 3 we will provide numerical results comparing these two methods, and show that there is very little and basically no difference in their simulation results. Because the method with (72) is much simpler and produces the same results as the sub-iteration procedure, the majority of simulations in Section 3 and the simulations there by default (unless otherwise specified) are performed using this method.

Let us now consider how to deal with ξ^{n+1} in (70), with the approximation (72). The following discussions also pertain to the implementation with the sub-iteration procedure. With the approximation (72), we can transform (70) into

$$\begin{aligned} \int_{\Omega} \nabla P^{n+1} \cdot \nabla q = & \rho_0 \int_{\Omega} \left(\frac{\hat{\mathbf{u}}}{\Delta t} + \frac{C_1^{n+1}}{\bar{\rho}^{n+1}} \nabla \bar{\phi}^{n+1} + \frac{\mathbf{f}^{n+1}}{\bar{\rho}^{n+1}} \right) \cdot \nabla q - \frac{\rho_0 \gamma_0}{\Delta t} \int_{\partial \Omega} \mathbf{n} \cdot \mathbf{w}^{n+1} q \\ & + \xi^{n+1} \left\{ \rho_0 \int_{\Omega} \left(\frac{C_2^{n+1}}{\bar{\rho}^{n+1}} \nabla \bar{\phi}^{n+1} - \mathbf{Q} + \nabla \left(\frac{\bar{\mu}^{n+1}}{\bar{\rho}^{n+1}} \right) \times \boldsymbol{\omega}^{*,n+1} \right) \cdot \nabla q - \rho_0 \int_{\partial \Omega} \frac{\bar{\mu}^{n+1}}{\bar{\rho}^{n+1}} \mathbf{n} \times \boldsymbol{\omega}^{*,n+1} \cdot \nabla q \right\}, \quad \forall q. \end{aligned} \quad (73)$$

Thanks to the fact that ξ^{n+1} is a scalar number (not a field function), we can solve equation (73) as follows. Define (P_1^{n+1}, P_2^{n+1}) as solutions to the following two Poisson-type problems:

Find $P_1^{n+1} \in \mathbb{V}(\Omega) = \{v \in H^1(\Omega), \int_{\Omega} v = 0\}$ such that

$$\int_{\Omega} \nabla P_1^{n+1} \cdot \nabla q = \rho_0 \int_{\Omega} \left(\frac{\hat{\mathbf{u}}}{\Delta t} + \frac{C_1^{n+1}}{\bar{\rho}^{n+1}} \nabla \bar{\phi}^{n+1} + \frac{\mathbf{f}^{n+1}}{\bar{\rho}^{n+1}} \right) \cdot \nabla q - \frac{\rho_0 \gamma_0}{\Delta t} \int_{\partial \Omega} \mathbf{n} \cdot \mathbf{w}^{n+1} q, \quad \forall q \in H^1(\Omega) \quad (74)$$

Find $P_2^{n+1} \in \mathbb{V}(\Omega)$ such that

$$\begin{aligned} \int_{\Omega} \nabla P_2^{n+1} \cdot \nabla q &= \rho_0 \int_{\Omega} \left(\frac{C_2^{n+1}}{\bar{\rho}^{n+1}} \nabla \bar{\phi}^{n+1} - \mathbf{Q} + \nabla \left(\frac{\bar{\mu}^{n+1}}{\bar{\rho}^{n+1}} \right) \times \boldsymbol{\omega}^{*,n+1} \right) \cdot \nabla q \\ &\quad - \rho_0 \int_{\partial\Omega} \frac{\bar{\mu}^{n+1}}{\bar{\rho}^{n+1}} \mathbf{n} \times \boldsymbol{\omega}^{*,n+1} \cdot \nabla q, \quad \forall q \in H^1(\Omega). \end{aligned} \quad (75)$$

Then it is straightforward to verify that the solution to equation (73) is given by

$$P^{n+1} = P_1^{n+1} + \xi^{n+1} P_2^{n+1}, \quad (76)$$

where ξ^{n+1} is to be determined.

In light of the equations (76) and (67), we can transform equation (28) into

$$\begin{aligned} \frac{\gamma_0}{\nu_m \Delta t} \mathbf{u}^{n+1} - \nabla^2 \mathbf{u}^{n+1} &= \frac{1}{\nu_m} \left[-\frac{1}{\rho_0} \nabla P_1^{n+1} + \frac{\hat{\mathbf{u}}}{\Delta t} + \frac{C_1^{n+1}}{\bar{\rho}^{n+1}} \nabla \bar{\phi}^{n+1} + \frac{\mathbf{f}^{n+1}}{\bar{\rho}^{n+1}} \right] \\ &\quad + \xi^{n+1} \frac{1}{\nu_m} \left[-\frac{1}{\rho_0} \nabla P_2^{n+1} + \frac{C_2^{n+1}}{\bar{\rho}^{n+1}} \nabla \bar{\phi}^{n+1} - \mathbf{Q} - \left(\frac{\bar{\mu}^{n+1}}{\bar{\rho}^{n+1}} - \nu_m \right) \nabla \times \boldsymbol{\omega}^{*,n+1} \right]. \end{aligned} \quad (77)$$

Let $\varphi(\mathbf{x})$ denote an arbitrary test function that vanishes on $\partial\Omega$, i.e. $\varphi|_{\partial\Omega} = 0$. Taking the L^2 inner product between φ and the equation (77), we obtain the weak formulation for \mathbf{u}^{n+1} :

$$\begin{aligned} \frac{\gamma_0}{\nu_m \Delta t} \int_{\Omega} \mathbf{u}^{n+1} \varphi + \int_{\Omega} \nabla \mathbf{u}^{n+1} \cdot \nabla \varphi &= \frac{1}{\nu_m} \int_{\Omega} \left(-\frac{1}{\rho_0} \nabla P_1^{n+1} + \frac{\hat{\mathbf{u}}}{\Delta t} + \frac{C_1^{n+1}}{\bar{\rho}^{n+1}} \nabla \bar{\phi}^{n+1} + \frac{\mathbf{f}^{n+1}}{\bar{\rho}^{n+1}} \right) \varphi \\ &\quad + \xi^{n+1} \frac{1}{\nu_m} \left\{ \int_{\Omega} \left(-\frac{1}{\rho_0} \nabla P_2^{n+1} + \frac{C_2^{n+1}}{\bar{\rho}^{n+1}} \nabla \bar{\phi}^{n+1} - \mathbf{Q} + \nabla \left(\frac{\bar{\mu}^{n+1}}{\bar{\rho}^{n+1}} \right) \times \boldsymbol{\omega}^{*,n+1} \right) \varphi \right. \\ &\quad \left. - \int_{\Omega} \left(\frac{\bar{\mu}^{n+1}}{\bar{\rho}^{n+1}} - \nu_m \right) \boldsymbol{\omega}^{*,n+1} \times \nabla \varphi \right\}, \quad \forall \varphi \in H_0^1(\Omega), \end{aligned} \quad (78)$$

where we have used equation (71).

Again we exploit the fact that ξ^{n+1} is a scalar number, and equation (78) can be solved as follows. Introducing two auxiliary velocity field functions $(\mathbf{u}_1^{n+1}, \mathbf{u}_2^{n+1})$ as the solutions of the following Helmholtz-type problems:

Find $\mathbf{u}_1^{n+1} \in [H^1(\Omega)]^d$ such that

$$\frac{\gamma_0}{\nu_m \Delta t} \int_{\Omega} \mathbf{u}_1^{n+1} \varphi + \int_{\Omega} \nabla \mathbf{u}_1^{n+1} \cdot \nabla \varphi = \frac{1}{\nu_m} \int_{\Omega} \left(-\frac{1}{\rho_0} \nabla P_1^{n+1} + \frac{\hat{\mathbf{u}}}{\Delta t} + \frac{C_1^{n+1}}{\bar{\rho}^{n+1}} \nabla \bar{\phi}^{n+1} + \frac{\mathbf{f}^{n+1}}{\bar{\rho}^{n+1}} \right) \varphi, \quad \forall \varphi \in H_0^1(\Omega), \quad (79a)$$

$$\mathbf{u}_1^{n+1} = \mathbf{w}^{n+1}, \quad \text{on } \partial\Omega. \quad (79b)$$

Find $\mathbf{u}_2^{n+1} \in [H_0^1(\Omega)]^d$ such that

$$\begin{aligned} \frac{\gamma_0}{\nu_m \Delta t} \int_{\Omega} \mathbf{u}_2^{n+1} \varphi + \int_{\Omega} \nabla \mathbf{u}_2^{n+1} \cdot \nabla \varphi &= -\frac{1}{\nu_m} \int_{\Omega} \left(\frac{\bar{\mu}^{n+1}}{\bar{\rho}^{n+1}} - \nu_m \right) \boldsymbol{\omega}^{*,n+1} \times \nabla \varphi \\ &\quad + \frac{1}{\nu_m} \int_{\Omega} \left(-\frac{1}{\rho_0} \nabla P_2^{n+1} + \frac{C_2^{n+1}}{\bar{\rho}^{n+1}} \nabla \bar{\phi}^{n+1} - \mathbf{Q} + \nabla \left(\frac{\bar{\mu}^{n+1}}{\bar{\rho}^{n+1}} \right) \times \boldsymbol{\omega}^{*,n+1} \right) \varphi, \quad \forall \varphi \in H_0^1(\Omega), \end{aligned} \quad (80)$$

where $H_0^1(\Omega) = \{v \in H^1(\Omega), v|_{\partial\Omega} = 0\}$. It is straightforward to verify that the solution \mathbf{u}^{n+1} to equations (78) and (33) is computed by

$$\mathbf{u}^{n+1} = \mathbf{u}_1^{n+1} + \xi^{n+1} \mathbf{u}_2^{n+1}, \quad (81)$$

provided that the scalar variable ξ^{n+1} is given.

The scalar value ξ^{n+1} still needs to be determined. Note that equation (32) can be transformed into equation (50), and this equation can be written as

$$\begin{aligned}
& \frac{2\xi^{n+1}}{\Delta t} R^{n+1} (\gamma_0 R^{n+1} - \hat{R}) + \frac{\xi^{n+1}}{2} \int_{\Omega} \bar{\mu}^{n+1} \|\mathcal{D}(\mathbf{u}^{n+1})\|^2 + m \xi^{n+1} \int_{\Omega} |\nabla \mathcal{C}^{n+1}|^2 - \xi^{n+1} \int_{\Omega} \mathbf{f}^{n+1} \cdot \mathbf{u}^{n+1} \\
& - \xi^{n+1} \int_{\Omega} \mathbf{g}^{n+1} \mathcal{C}^{n+1} + \frac{\lambda \xi^{n+1}}{\Delta t} \int_{\Omega} \nabla \phi^{n+1} \cdot \nabla (\gamma_0 \phi^{n+1} - \hat{\phi}) + \frac{S \xi^{n+1}}{\Delta t} \int_{\Omega} (\phi^{n+1} - \phi^{*,n+1}) (\gamma_0 \phi^{n+1} - \hat{\phi}) \\
& + \xi^{n+1} \int_{\partial\Omega} m (\lambda d_b^{n+1} - S(d_a^{n+1} - d_a^{*,n+1}) - \xi^{n+1} h'(\phi^{*,n+1}) d_a^{*,n+1}) \mathcal{C}^{n+1} - \frac{\lambda \xi^{n+1}}{\Delta t} \int_{\partial\Omega} d_a^{n+1} (\gamma_0 \phi^{n+1} - \hat{\phi}) \\
& - \xi^{n+1} \int_{\partial\Omega} \left[-P^{n+1} \mathbf{n} + \bar{\mu}^{n+1} \mathbf{n} \cdot \mathcal{D}(\mathbf{u}^{n+1}) - \frac{1}{2} \bar{\rho}^{n+1} (\mathbf{n} \cdot \mathbf{w}^{n+1}) \mathbf{w}^{n+1} - \frac{1}{2} (\mathbf{n} \cdot \tilde{\mathbf{j}}^{n+1}) \mathbf{w}^{n+1} \right] \cdot \mathbf{w}^{n+1} = 0,
\end{aligned} \tag{82}$$

where we have multiplied both sides of the equation by $\frac{R^{n+1}}{\sqrt{E^{n+1}}}$, which we observe can improve the robustness of the scheme. This is a nonlinear equation with the unknown scalar variable ξ^{n+1} , and it will be solved for ξ^{n+1} . In light of equation (38), we have

$$\begin{aligned}
E^{n+1} &= E(\xi^{n+1}) = \int_{\Omega} \left[\frac{\bar{\rho}^{n+1}}{2} |\mathbf{u}^{n+1}|^2 + F(\phi^{n+1}) \right] d\Omega + C_0 \\
&= A_0 + A_1 \xi^{n+1} + A_2 (\xi^{n+1})^2 + A_3 (\xi^{n+1})^3 + A_4 (\xi^{n+1})^4,
\end{aligned} \tag{83}$$

where

$$\begin{aligned}
A_0 &= \int_{\Omega} \left\{ \frac{1}{2} \bar{\rho}^{n+1} |\mathbf{u}_1^{n+1}|^2 + \frac{\lambda}{4\eta^2} (1 - (\phi_1^{n+1})^2)^2 \right\} d\Omega + C_0, \\
A_1 &= \int_{\Omega} \left\{ \bar{\rho}^{n+1} \mathbf{u}_1^{n+1} \cdot \mathbf{u}_2^{n+1} - \frac{\lambda}{\eta^2} (1 - (\phi_1^{n+1})^2) \phi_1^{n+1} \phi_2^{n+1} \right\} d\Omega, \\
A_2 &= \int_{\Omega} \left\{ \frac{1}{2} \bar{\rho}^{n+1} |\mathbf{u}_2^{n+1}|^2 + \frac{\lambda}{4\eta^2} (6(\phi_1^{n+1})^2 - 2) (\phi_2^{n+1})^2 \right\} d\Omega, \\
A_3 &= \frac{\lambda}{\eta^2} \int_{\Omega} \phi_1^{n+1} (\phi_2^{n+1})^3 d\Omega, \\
A_4 &= \frac{\lambda}{4\eta^2} \int_{\Omega} (\phi_2^{n+1})^4 d\Omega.
\end{aligned} \tag{84}$$

In light of equations (27), (34), (67), (76) and (81), we can transform equation (82) into

$$F(\xi^{n+1}) = \frac{2\gamma_0}{\Delta t} (\xi^{n+1})^3 E(\xi^{n+1}) - \frac{2\hat{R}}{\Delta t} (\xi^{n+1})^2 \sqrt{E(\xi^{n+1})} + B_0 \xi^{n+1} + B_1 (\xi^{n+1})^2 + B_2 (\xi^{n+1})^3 = 0, \tag{85}$$

where \hat{R} is defined by (40) and

$$\begin{aligned}
B_0 &= \frac{1}{2} \int_{\Omega} \bar{\mu}^{n+1} \|\mathcal{D}(\mathbf{u}_1^{n+1})\|^2 + m \int_{\Omega} |\nabla \mathcal{C}_1^{n+1}|^2 + \frac{\lambda}{\Delta t} \int_{\Omega} \nabla \phi_1^{n+1} \cdot (\gamma_0 \nabla \phi_1^{n+1} - \nabla \hat{\phi}) \\
& + \frac{S}{\Delta t} \int_{\Omega} (\phi_1^{n+1} - \phi^{*,n+1}) (\gamma_0 \phi_1^{n+1} - \hat{\phi}) - \int_{\Omega} \mathbf{f}^{n+1} \cdot \mathbf{u}_1^{n+1} - \int_{\Omega} \mathbf{g}^{n+1} \mathcal{C}_1^{n+1} \\
& - m \int_{\partial\Omega} (S d_a^{n+1} - \lambda d_b^{n+1} - S d_a^{*,n+1}) \mathcal{C}_1^{n+1} - \frac{\lambda}{\Delta t} \int_{\partial\Omega} d_a^{n+1} (\gamma_0 \phi_1^{n+1} - \hat{\phi}) \\
& - \int_{\partial\Omega} \left[-P_1^{n+1} \mathbf{n} + \bar{\mu}^{n+1} \mathbf{n} \cdot \mathcal{D}(\mathbf{u}_1^{n+1}) - \frac{1}{2} \bar{\rho}^{n+1} |\mathbf{w}^{n+1}|^2 \mathbf{n} \right] \cdot \mathbf{w}^{n+1} \\
& + \frac{J_0}{2} \int_{\partial\Omega} \left\{ \frac{\lambda}{\eta^2} d_a^{n+1} (3(\phi_1^{n+1})^2 - 1) - \lambda d_b^{n+1} \right\} |\mathbf{w}^{n+1}|^2,
\end{aligned} \tag{86}$$

$$\begin{aligned}
B_1 = & \int_{\Omega} \bar{\mu}^{n+1} \mathcal{D}(\mathbf{u}_1^{n+1}) : \mathcal{D}(\mathbf{u}_2^{n+1}) + 2m \int_{\Omega} \nabla C_1^{n+1} \cdot \nabla C_2^{n+1} - \int_{\Omega} \mathbf{f}^{n+1} \cdot \mathbf{u}_2^{n+1} - \int_{\Omega} g^{n+1} C_2^{n+1} \\
& + \frac{\lambda}{\Delta t} \int_{\Omega} (2\gamma_0 \nabla \phi_1^{n+1} - \nabla \hat{\phi}) \cdot \nabla \phi_2^{n+1} + \frac{S}{\Delta t} \int_{\Omega} (2\gamma_0 \phi_1^{n+1} - \gamma_0 \phi^{*,n+1} - \hat{\phi}) \phi_2^{n+1} \\
& - m \int_{\partial\Omega} \left\{ (Sd_a^{n+1} - \lambda d_b^{n+1} - Sd_a^{*,n+1}) C_2^{n+1} + h'(\phi^{*,n+1}) d_a^{*,n+1} C_1^{n+1} \right\} - \frac{\lambda\gamma_0}{\Delta t} \int_{\partial\Omega} d_a^{n+1} \phi_2^{n+1} \\
& - \int_{\partial\Omega} \left[-P_2^{n+1} \mathbf{n} + \bar{\mu}^{n+1} \mathbf{n} \cdot \mathcal{D}(\mathbf{u}_2^{n+1}) \right] \cdot \mathbf{w}^{n+1} + \frac{6\lambda J_0}{2\eta^2} \int_{\partial\Omega} d_a^{n+1} \phi_1^{n+1} \phi_2^{n+1} |\mathbf{w}^{n+1}|^2, \\
B_2 = & \frac{1}{2} \int_{\Omega} \bar{\mu}^{n+1} \|\mathcal{D}(\mathbf{u}_2^{n+1})\|^2 + m \int_{\Omega} |\nabla C_2^{n+1}|^2 + \frac{\lambda\gamma_0}{\Delta t} \int_{\Omega} |\nabla \phi_2^{n+1}|^2 + \frac{S\gamma_0}{\Delta t} \int_{\Omega} |\phi_2^{n+1}|^2 \\
& - m \int_{\partial\Omega} h'(\phi^{*,n+1}) d_a^{*,n+1} C_2^{n+1} + \frac{3\lambda J_0}{2\eta^2} \int_{\partial\Omega} d_a^{n+1} (\phi_2^{n+1})^2 |\mathbf{w}^{n+1}|^2.
\end{aligned} \tag{87}$$

$$\begin{aligned}
B_2 = & \frac{1}{2} \int_{\Omega} \bar{\mu}^{n+1} \|\mathcal{D}(\mathbf{u}_2^{n+1})\|^2 + m \int_{\Omega} |\nabla C_2^{n+1}|^2 + \frac{\lambda\gamma_0}{\Delta t} \int_{\Omega} |\nabla \phi_2^{n+1}|^2 + \frac{S\gamma_0}{\Delta t} \int_{\Omega} |\phi_2^{n+1}|^2 \\
& - m \int_{\partial\Omega} h'(\phi^{*,n+1}) d_a^{*,n+1} C_2^{n+1} + \frac{3\lambda J_0}{2\eta^2} \int_{\partial\Omega} d_a^{n+1} (\phi_2^{n+1})^2 |\mathbf{w}^{n+1}|^2.
\end{aligned} \tag{88}$$

To determine ξ^{n+1} , we solve the *scalar* nonlinear equation (85) about ξ^{n+1} using the Newton's method with an initial guess $\xi^{n+1} = 1$. The cost of this computation is very small compared to the total cost within a time step, because this equation is about a scalar number (not a field function). Once ξ^{n+1} is known, E^{n+1} can be computed based on equation (83), and R^{n+1} can be computed using equation (27). The phase field ϕ^{n+1} , pressure P^{n+1} and velocity \mathbf{u}^{n+1} are then given by equations (61b), (76) and (81), respectively.

Remark 3. The weakly coupled equation (70) can be solved by a sub-iteration method. In the first iteration step, we use $\omega_{(0)}^{n+1} = \omega^{*,n+1}$ to replace \mathbf{w}^{n+1} . As a result, we can solve $P_{(1)}^{n+1}$ and $\mathbf{u}_{(1)}^{n+1}$, as well as the vorticity $\omega_{(1)}^{n+1} = \nabla \times \mathbf{u}_{(1)}^{n+1}$ by the algorithm discussed above. Then we replace ω^{n+1} with $\omega_{(1)}^{n+1}$ in equation (70) and repeat the iterations until $\|\omega_{(k+1)}^{n+1} - \omega_{(k)}^{n+1}\|$ becomes sufficiently small. See Section 3 (e.g. Fig. 2) for numerical results with this method.

In the current work we employ high-order C^0 spectral elements for the spatial discretizations of the equations (62)–(65), (74)–(75) and (79)–(80). Since these equations are in weak forms, this can be carried out in a quite standard way. For the sake of completeness we have provided a summary of the fully discretized forms of these equations in the Appendix A.

input : \mathbf{u} , P , ϕ and R of time steps n and $(n-1)$
output: \mathbf{u}^{n+1} , P^{n+1} , ϕ^{n+1} , ψ^{n+1} , and R^{n+1}

begin

Solve equations (62)–(65) for $(\psi_1^{n+1}, \psi_2^{n+1})$ and $(\phi_1^{n+1}, \phi_2^{n+1})$ successively;
 Compute $\bar{\phi}^{n+1}$, $\bar{\rho}^{n+1}$, $\bar{\mu}^{n+1}$ and $\bar{\mathbf{j}}^{n+1}$ based on equations (66) and (39);
 Compute C_1^{n+1} and C_2^{n+1} based on equation (68);

Solve equations (74)–(75) for (P_1^{n+1}, P_2^{n+1}) ;
 Solve equations (79)–(80) for $(\mathbf{u}_1^{n+1}, \mathbf{u}_2^{n+1})$;

Compute the coefficients A_0 – A_4 using equation (84);
 Compute the coefficients B_0 – B_2 using equations (86)–(88);
 Solve equation (85) for ξ^{n+1} using the Newton's method with an initial guess $\xi^{n+1} = 1$;

Compute ϕ^{n+1} based on equation (61b);
 Compute P^{n+1} based on equation (76);
 Compute \mathbf{u}^{n+1} based on equation (81);
 Compute E^{n+1} based on equation (83);
 Compute R^{n+1} based on equation (27);

end

Algorithm 1: Solution algorithm within a time step.

Algorithm 1 is a summary of the final solution algorithm. It can be noted that with this algorithm two sets of field variables for each of the velocity, pressure, and phase-field functions are computed, together with a nonlinear algebraic equation about a *scalar-valued* number. The computations for different field variables are de-coupled. When computing the field variables, all the resultant linear algebraic systems after discretization only involve constant and time-independent coefficient matrices, which only need to be computed once and can be pre-computed.

Table 1
Normalization constants for various physical variables and parameters.

Variable	Normalization constant	Variable	Normalization constant
\mathbf{x}, η	L	$\mathbf{u}, \mathbf{w}, \mathbf{u}_{in}$	U_0
$t, \Delta t, J_0$	L/U_0	\mathbf{g}_r (gravity)	U_0^2/L
$p, P, P_1, P_2, h(\phi), F(\phi), C, C_1, C_2, S$	$\varrho_d U_0^2$	$\phi, \phi_1, \phi_2, \phi_{in}, \xi$	1
$\rho_1, \rho_2, \rho, \rho_0$	ϱ_d	μ_1, μ_2, μ	$\varrho_d U_0 L$
m	$L/(\varrho_d U_0)$	σ	$\varrho_d U_0^2 L$
λ	$\varrho_d U_0^2 L^2$	v_m	$U_0 L$
\mathbf{f}	$\varrho_d U_0^2/L$	$\tilde{\mathbf{J}}$	$\varrho_d U_0$
$\mathbf{g}, \boldsymbol{\omega}$	U_0/L	d_a	$1/L$
$E, C_0, A_0, A_1, A_2, A_3, A_4$	$\varrho_d U_0^2 L^d$ ($d = 2$ or 3)	d_b	$1/L^3$
R	$\sqrt{\varrho_d U_0^2 L^d}$	$\psi, \psi_1, \psi_2, \alpha$	$1/L^2$
B_0, B_1, B_2	$\varrho_d U_0^3 L^{d-1}$		

3. Representative numerical examples

In this section, we provide numerical results in two dimensions to illustrate the accuracy and robustness of the energy-stable scheme described in the previous section. Both spatial and temporal convergence rates are presented via a manufactured analytic solution. We then use the capillary wave problem as a benchmark to demonstrate that the proposed algorithm produces physically accurate results. We also show numerical simulations of a rising bubble problem with large density contrast and viscosity contrast. It should be noted that when the physical variables and parameters are normalized consistently, the dimensional and non-dimensionalized governing equations, boundary conditions and initial conditions will have the same form [11,13]. Let L , U_0 and ϱ_d denote the characteristic length, velocity and density scales, respectively. Table 1 lists the constants for consistently normalizing different physical variables and parameters. In subsequent discussions all the physical variables and parameters are assumed to have been normalized according to Table 1. The majority of simulation results reported in this section, and unless otherwise specified, the simulations of this section are computed using the method with the approximation (72) when solving equation (70).

3.1. Convergence rates

To begin with, we employ a manufactured analytic solution to demonstrate the convergence rates of the scheme developed herein against both the element order and the time step size. Consider the computational domain $\Omega = \{(x, y) | 0 \leq x \leq 2, -1 \leq y \leq 1\}$ and a mixture of two immiscible fluids contained in this domain. We assume the following analytic solution to the Navier-Stokes/Cahn-Hilliard coupled system of equations (7), (1b)–(1d):

$$\begin{cases} u = \sin(\pi x) \cos(\pi y) \sin(t), \\ v = -\cos(\pi x) \sin(\pi y) \sin(t), \\ P = \sin(\pi x) \sin(\pi y) \cos(t), \\ \phi = \cos(\pi x) \cos(\pi y) \sin(t), \end{cases} \quad (89)$$

where (u, v) are the two components of the velocity \mathbf{u} . Accordingly, the source terms \mathbf{f} and \mathbf{g} in equations (7) and (1c) are chosen such that the analytic expressions (89) satisfy equations (7) and (1c), respectively. The source terms \mathbf{w} , d_a and d_b in equations (9)–(11) are chosen such that the contrived solution in (89) satisfies the boundary conditions. The initial conditions (12)–(13) are imposed for the velocity and phase field function, respectively, where \mathbf{u}_{in} and ϕ_{in} are chosen by letting $t = 0$ in the contrived solution (89).

We discretize the domain using two equal-sized quadrilateral elements, with the element order and the time step size Δt varied systematically in the spatial and temporal convergence tests. The numerical algorithm from Section 2 is employed to numerically integrate the governing equations in time from $t = 0$ to $t = t_f$. Then the numerical solution and the exact solution as given by equation (89) at $t = t_f$ are compared, and the errors in the L^2 and L^∞ norm for various flow variables are computed. All the physical and numerical parameters involved in the simulation of this problem are tabulated in Table 2.

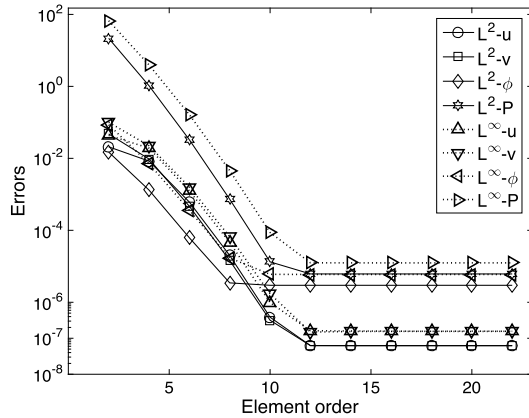
In the spatial convergence test, we fix the integration time at $t_f = 0.1$, the time step size at $\Delta t = 0.001$ (100 time steps), and the constant in the algorithm at $S = \sqrt{\frac{4\gamma_0\lambda}{m\Delta t}}$, and vary the element order systematically between 2 and 22. The same element order has been used for these two spectral elements. Fig. 1(a) depicts the numerical errors at $t = t_f$ in L^2 and L^∞ norms for different flow variables as a function of the element order. It is evident that within a specific range of the element order (below around 12), the errors decrease exponentially with increasing element order, displaying an exponential convergence rate in space. Beyond the element order of about 12, the error curves level off as the element order further increases, showing a saturation caused by the temporal truncation error.

In the temporal convergence test, we fix the integration time at $t_f = 1$ and the element order at a large value 22, and vary the time step size systematically between $\Delta t = 2.44141 \times 10^{-4}$ and $\Delta t = 0.25$. The constant S in the algorithm is fixed

Table 2

Simulation parameter values for convergence tests.

Parameter	Value	Parameter	Value
C_0	0	λ	0.001
m	0.001	η	0.1
(ρ_1, ρ_2)	(1.0, 3.0)	(μ_1, μ_2)	(0.01, 0.02)
ρ_0	$\min(\rho_1, \rho_2)$	ν_m	$2 \times \frac{\max(\mu_1, \mu_2)}{\min(\rho_1, \rho_2)}$
Δt_{\min}	$1e-4$	Δt	(varied)
t_f	0.1 (spatial tests) or 1.0 (temporal tests)	S	$\sqrt{\frac{4\gamma_0\lambda}{m\Delta t}}$ or $\sqrt{\frac{4\gamma_0\lambda}{m\Delta t_{\min}}}$
Element order	(varied)	Number of elements	2



(a) Errors vs Element order

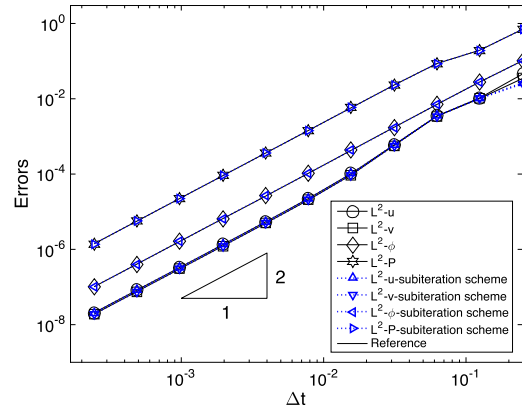
(b) Errors vs Δt

Fig. 1. Spatial/temporal convergence tests: (a) L^2 and L^∞ errors of flow variables versus element order (fixed $\Delta t = 0.001$ and $t_f = 0.1$); (b) L^2 errors of flow variables versus Δt (fixed element order 22 and $t_f = 1$) and comparison with the sub-iteration method discussed in Remark 3.

at $S = \sqrt{\frac{4\gamma_0\lambda}{m\Delta t_{\min}}}$ (where $\Delta t_{\min} = 10^{-4}$). Fig. 1(b) shows the numerical errors at $t = t_f$ in L^2 norm for different variables as a function of Δt in logarithmic scales. It can be observed that the numerical errors exhibit a second order convergence rate in time. Note that here we have used a sufficiently large value for S and it is kept as a constant in these tests. As discussed in Remark 2, if S is allowed to vary with respect to Δt then the temporal rate of convergence will not be second order in the tests.

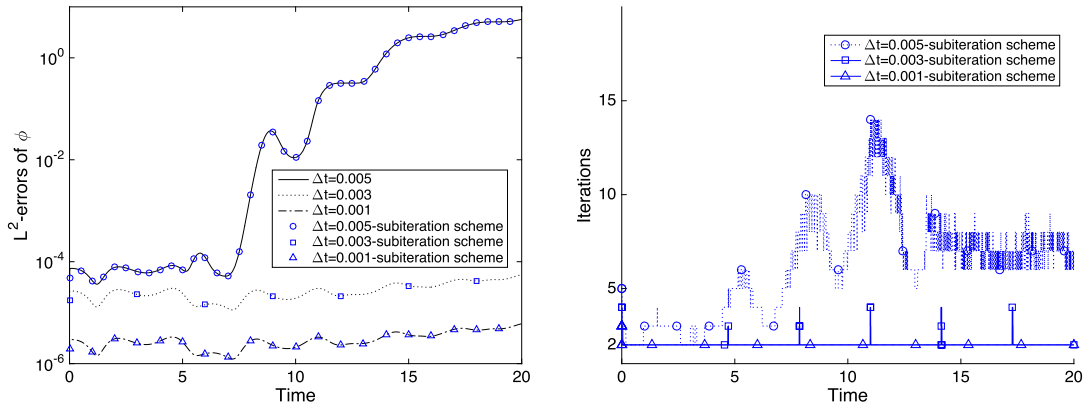
As discussed in Section 2, two methods have been implemented for solving equation (70), based on a sub-iteration procedure as discussed in Remark 3 and based on the approximation (72) on $\partial\Omega$. The simulation results presented so far are obtained based on the method with the approximation (72) by default. To compare these two methods, in Fig. 1(b) we have also included the numerical errors obtained with the sub-iteration method (marked by “subiteration scheme” in the legend). It is observed that the difference in the numerical errors between these two methods is negligible. In Fig. 2(a) we compare the time histories of the L^2 -errors for ϕ computed with several Δt values for longer-time simulations ($t_f = 20$) with both methods. We observe that the error histories of both methods essentially overlap with each other. In Fig. 2(b), we plot the time histories of the number of sub-iterations needed for convergence within a time step for the sub-iteration method. The results indicate that for the smaller Δt values around 2–4 sub-iterations are needed for convergence within a time step, while for the larger $\Delta t = 0.005$ it can take 8 or more sub-iterations within a time step. These results show that the method with the approximation (72) is clearly advantageous. It is cheaper computationally, and produces essentially the same simulation results.

3.2. Capillary wave problem

In this subsection, we use the capillary wave problem, for which an exact solution is available, as a benchmark to test the accuracy of the proposed algorithm.

The problem setting is as follows. We consider two immiscible incompressible fluids contained in an infinite domain. The upper half of the domain is occupied by the lighter fluid (with density ρ_1), and the lower half of the domain is occupied by the heavier fluid (with density ρ_2). The gravity is assumed to be in the downward direction. The interface formed between these two fluids is perturbed from its horizontal equilibrium position by a small-amplitude sinusoidal wave form, and starts to oscillate at $t = 0$. The objective here is to study the motion of the interface over time.

In [31] an exact time-dependent standing-wave solution to this capillary wave problem was derived, under the condition that the two fluids must have matched kinematic viscosities (but their densities and dynamic viscosities can differ). To



(a) Comparison of methods with and without sub-iterations

(b) Number of sub-iterations within a time step

Fig. 2. Computations with sub-iterations: (a) Comparison of L^2 errors of ϕ between the method using approximation (72) and the sub-iteration method with several Δt values; (b) Evolution of the number of sub-iterations needed within a time step over time for the sub-iteration method. Results are obtained with a fixed element order 22 and $t_f = 20$. Those curves in (a) with legends not marked by “subiteration scheme” are obtained with the method using the approximation (72).

Table 3

Simulation parameter values for the capillary wave problem.

Parameter	Value	Parameter	Value
H_0	0.01	k_w (wave number)	2π
σ	1.0	$ \mathbf{g}_r $ (gravity)	1.0
ρ_1	1.0	ρ_2	(varied)
μ_1	0.01	μ_2	$\mu_1 \frac{\rho_2}{\rho_1}$
ρ_0	$\min(\rho_1, \rho_2)$	ν_m	$2 \times \frac{\max(\mu_1, \mu_2)}{\min(\rho_1, \rho_2)}$
C_0	(varied)	S	$\sqrt{\frac{4\gamma_0 \lambda}{m \Delta t}}$
J (temporal order)	2	Number of elements	240
m	5×10^{-4} or 10^{-5}	η	(varied)
λ	$\frac{3}{2\sqrt{2}} \sigma \eta$	Δt	(varied)

compare with Prosperetti’s exact physical solution (see [31]), we adopt the following settings to simulate the capillary wave problem: (i) the capillary amplitude is small compared with the vertical dimension of the domain; and (ii) the kinematic viscosity ν satisfies $\nu = \frac{\mu_1}{\rho_1} = \frac{\mu_2}{\rho_2}$.

To simulate this problem, we consider a rectangular computational domain $\Omega = \{(x, y) | 0 \leq x \leq 1, -1 \leq y \leq 1\}$. The upper and bottom sides of the domain are assumed to be solid walls of neutral wettability. On the left and right sides, all the variables are assumed to be periodic at $x = 0$ and $x = 1$. The equilibrium position of the fluid interface is assumed to coincide with $y = 0$. The initial perturbed profile of the fluid interface is given by $y = H_0 \cos(k_w x)$, where $H_0 = 0.01$ is the initial amplitude, $\lambda_w = 1$ is the wavelength of the perturbation profiles, and $k_w = 2\pi/\lambda_w$ is the wave number. Note that the initial capillary amplitude H_0 is small compared with the dimension of the domain in the vertical direction. Therefore, the effect of the walls at the domain top/bottom will be small.

The computational domain is partitioned with 240 quadrilateral elements, with 10 and 12 elements respectively in x and y directions. The elements are uniform in the x direction, and are non-uniform and clustered around the regions $-0.015 \leq y \leq 0.015$ in the y direction. In the simulations, the external body force \mathbf{f} in equation (7) is set to $\mathbf{f} = \rho \mathbf{g}_r$, where \mathbf{g}_r is the gravitational acceleration, and the source term in equation (1c) is set to $g = 0$. On the upper and bottom wall, the boundary condition (9) with $\mathbf{w} = \mathbf{0}$ is imposed for the velocity, and the boundary conditions (10) and (11) with $d_a = d_b = 0$ are imposed for the phase field functions. The initial velocity is set to zero, and the initial phase field function is prescribed as follows:

$$\phi(\mathbf{x}, 0) = \tanh \frac{y - H_0 \cos(k_w x)}{\sqrt{2}\eta}. \quad (90)$$

We list in Table 3 the values for the physical and numerical parameters involved in this problem.

Let us first focus on the case with a matched density for the two fluids, i.e., $\rho_1 = \rho_2 = 1$, and study the effects of several parameters on the simulation results with the data from Figs. 3 and 4. Fig. 3(a) shows a spatial resolution test. Here we compare the time histories of the capillary wave amplitude obtained with several element orders ranging from 6 to 14 in the simulations. These results are obtained with a fixed mobility $m = 5 \times 10^{-4}$, interfacial thickness $\eta = 0.005$, $\Delta t = 10^{-5}$

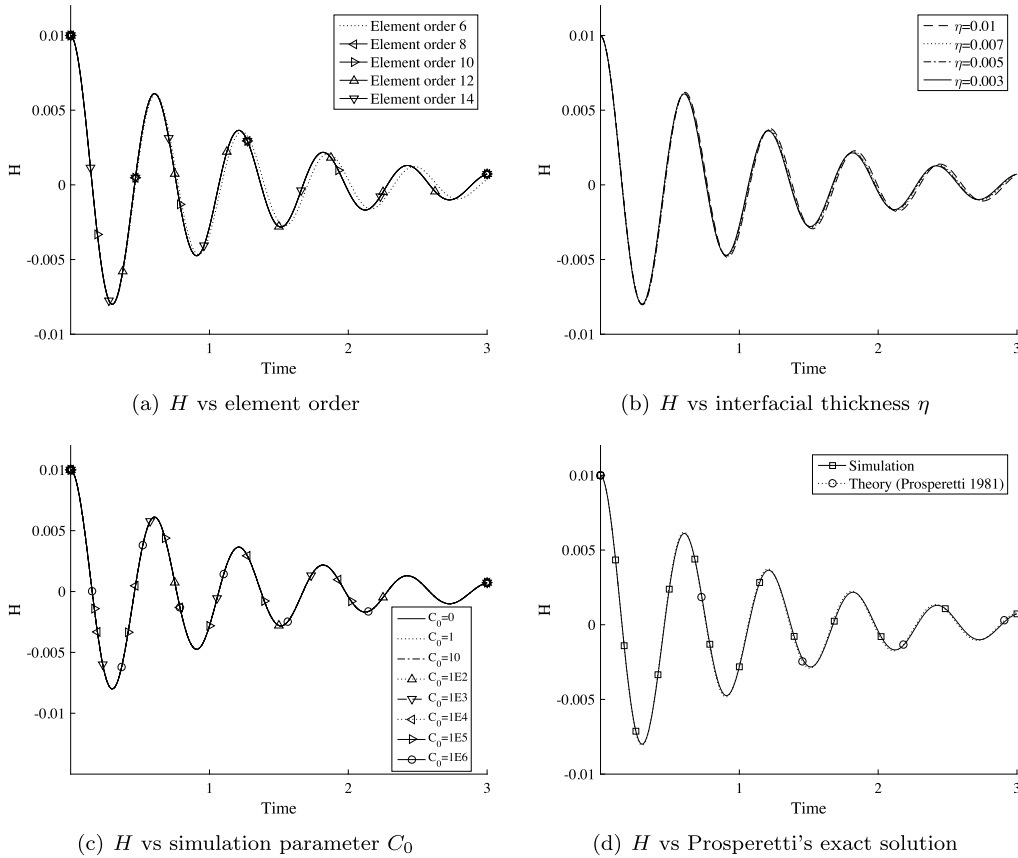


Fig. 3. Capillary wave problem (matched density $\rho_1 = \rho_2 = 1$): Comparison of capillary amplitude histories with different element orders (a), with different interfacial thickness η values (b), with different C_0 (c) and between simulation result and Prosperetti's exact solution. In (a), $\eta = 0.005$, $C_0 = 0$; In (b), element order 10, $C_0 = 0$; In (c), $\eta = 0.005$, element order = 10; In (d), element order 10, $\eta = 0.005$, $C_0 = 0$.

and $C_0 = 0$. The history curves corresponding to different element orders overlap with one another (with elements orders 8 and above), suggesting independence of the results with respect to the grid resolution. In the forthcoming simulations, we will fix element order equal to 10.

Fig. 3(b) shows the effect of the interfacial thickness scale η on the simulation results. In this figure we compare time histories of the capillary amplitude obtained with the interfacial thickness scale parameter η ranging from 0.01 to 0.003. These results correspond to a fixed mobility 5×10^{-4} , $\Delta t = 10^{-5}$ and $C_0 = 0$. Some influence on the amplitude and the phase of the history curves can be observed as η decreases from 0.01 to 0.007. From $\eta = 0.007$ and below, on the other hand, the history curves essentially overlap with one another and little difference can be discerned among them, suggesting a convergence of the results with respect to η .

Fig. 3(c) demonstrates the effect of the parameter C_0 on the simulation results with a fixed $m = 5 \times 10^{-4}$ and $\eta = 0.005$. No difference can be discerned among the numerical results obtained with different C_0 values. In the following simulations, we will employ $C_0 = 0$.

In Fig. 3(d), we compare the time history of the capillary amplitude obtained using $m = 5 \times 10^{-4}$ and $\eta = 0.005$ with the exact solution by [31]. The difference between the numerical result and the theoretic solution is negligible, showing that our method produces physically accurate numerical results.

Next, we demonstrate the robustness of our method with various time step sizes, ranging from very small to very large values. In Fig. 4(a), we show time histories of the capillary amplitude obtained with time step size in a range of smaller values, $\Delta t = 10^{-2} \sim 5 \times 10^{-6}$. The results correspond to a fixed mobility $m = 5 \times 10^{-4}$ and interfacial thickness $\eta = 0.005$ in the simulations. For comparison, this figure also includes the results obtained using the semi-implicit method from [15] with time step sizes $\Delta t = 0.001$ and 0.002. All the simulations using the current method are stable, and the time history curves collapse to a single one when the time step size is reduced to $\Delta t = 10^{-4}$ and below, suggesting convergence with respect to Δt . In contrast, the simulation using the semi-implicit scheme of [15] blows up with $\Delta t = 0.002$. With $\Delta t = 0.001$, the semi-implicit simulation is stable. But the resultant history curve contains considerably larger errors in both amplitude and phase, with respect to the theoretical solution, than the result obtained using the current method with the same time step size. In Fig. 4(b), we plot the corresponding time histories of $\xi^{n+1} = \frac{R^{n+1}}{\sqrt{E^{n+1}}}$ computed from equation (85). Note that ξ^{n+1}

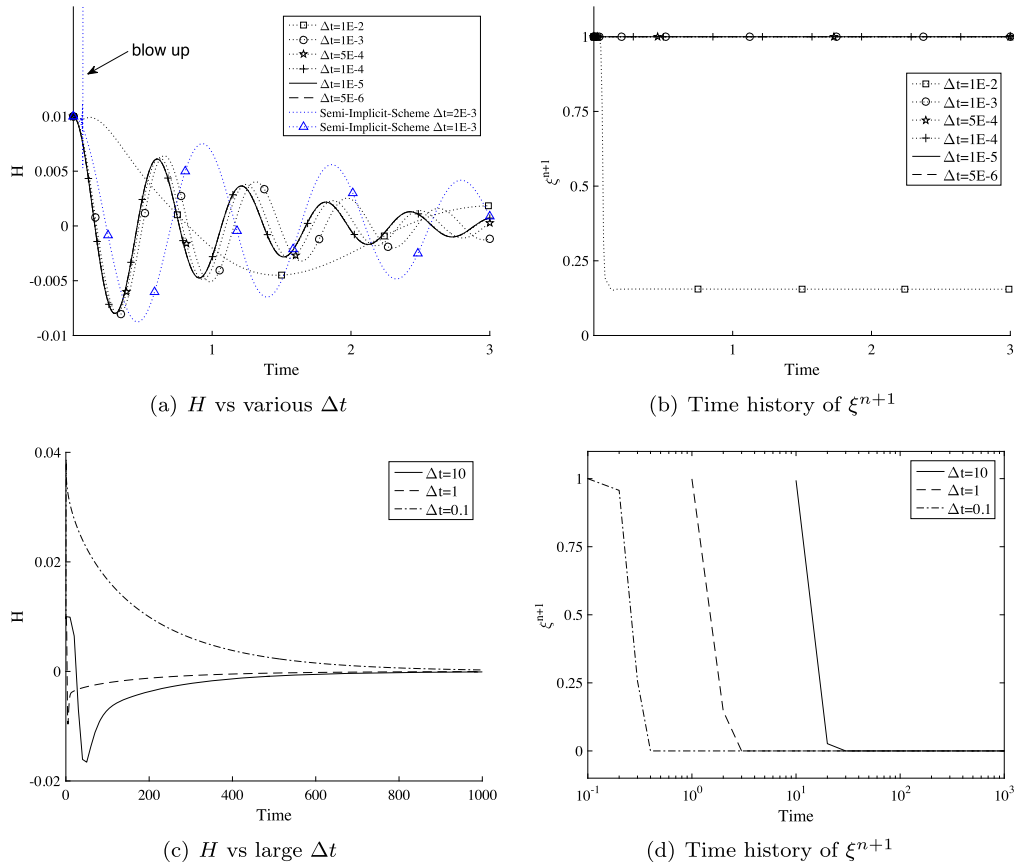


Fig. 4. Capillary wave problem (matched density $\rho_1 = \rho_2 = 1$): (a) Comparison of capillary amplitude histories obtained with a range of smaller Δt , and also comparison with results obtained by the semi-implicit method from [15], and (b) corresponding time histories of ξ^{n+1} . (c) Comparison of capillary amplitude histories obtained with a range of large time step sizes Δt , and (d) corresponding time histories of ξ^{n+1} in (d).

should physically take the unit value, because $R(t) = \sqrt{E(t)}$ by definition when deriving the dynamic equation about $R(t)$. A substantial deviation of ξ^{n+1} from the unit value indicates that $R(t)$ is no longer an accurate approximation of $\sqrt{E(t)}$, and therefore the simulation will contain significant errors. So the value of ξ^{n+1} is a good indicator of the accuracy of the simulation results. We observe that when Δt is small (less than 10^{-3}), ξ^{n+1} is identical to 1. But when $\Delta t = 10^{-2}$, ξ^{n+1} decreases sharply to 0.2 and then remains at that level over time. This suggests that the simulation is no longer accurate and the result contains significant errors with this Δt .

Thanks to its energy-stable nature, our algorithm can produce stable simulation results even with very large time step sizes. This is demonstrated by Figs. 4(c) and (d) for a range of large time step sizes. Here we increase the time step sizes to $\Delta t = 0.1, 1, 10$ and depict the time histories of the capillary amplitude in Fig. 4 (c) for a much longer simulation (up to $t = 1000$). The long time histories demonstrate that the computations with these large Δt values are indeed stable using the current algorithm. On the other hand, because these time step sizes are very large, we do not expect that these results will be accurate. It is observed that the characteristics of these history curves are quite different from those obtained with smaller time step sizes (comparing with e.g. Fig. 4(a)). Fig. 4(d) shows the corresponding time histories of ξ^{n+1} with this range of large Δt values. The behaviors of ξ^{n+1} are quite different from those obtained with small Δt (see Fig. 4(b)). The general characteristics seem to be that, when Δt increases to a fairly large value the computed ξ^{n+1} would decrease over time and its history curve would level off at a certain value for a given Δt . With a larger Δt , the ξ^{n+1} history tends to level off at a smaller value. With very large Δt values, ξ^{n+1} will essentially tend to zero. Note that these simulations are performed using the method with the approximation (72) when solving equation (70). The results with the large Δt values demonstrate that the approximation (72) on the boundary does not weaken the stability of our method.

Let us next investigate the effect of the density ratio on the dynamics of the fluid interface. In these tests we fix $\rho_1 = 1$ and $\mu_1 = 0.01$ for the first fluid and vary the density ρ_2 and the dynamic viscosity μ_2 of the second fluid systematically while the relation $\nu = \frac{\mu_1}{\rho_1} = \frac{\mu_2}{\rho_2}$ is maintained as required by the theoretic solution in [31]. In Fig. 5, we show the time histories of the capillary amplitudes corresponding to density ratios $\rho_2/\rho_1 = 10, 100, 1000$, and compare them with the theoretic solutions from [31]. The simulation results are obtained with a fixed interfacial thickness $\eta = 0.005$, mobility $m = 10^{-5}$, and $\Delta t = 10^{-5}$ for Figs. 5(a)-(b) and $\Delta t = 2.5 \times 10^{-6}$ for Fig. 5(c). We observe that the density contrasts have

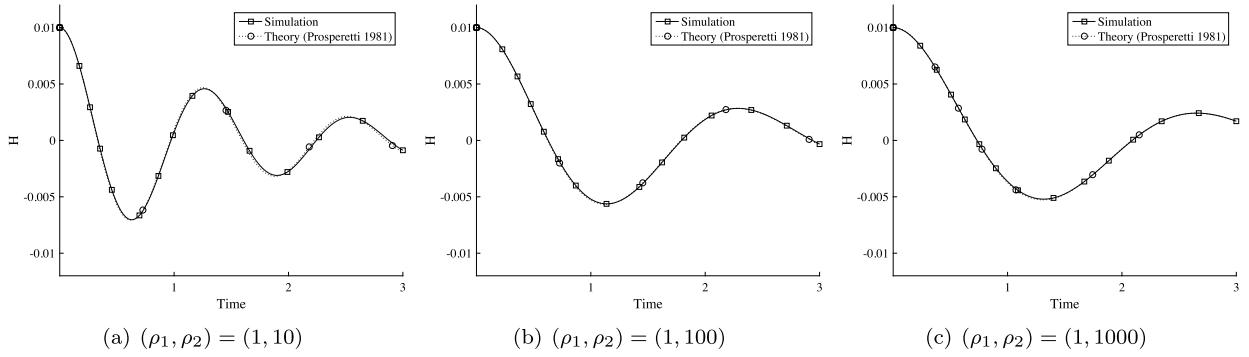


Fig. 5. Capillary wave problem (different density ratios): Comparison of capillary amplitude histories with Prosperetti's exact solution for density ratios (a) $\frac{\rho_2}{\rho_1} = 10$ (b) $\frac{\rho_2}{\rho_1} = 100$ (c) $\frac{\rho_2}{\rho_1} = 1000$. The simulation results are obtained with $\Delta t = 10^{-5}$ for (a)–(b) and $\Delta t = 2.5e-6$ for (c).

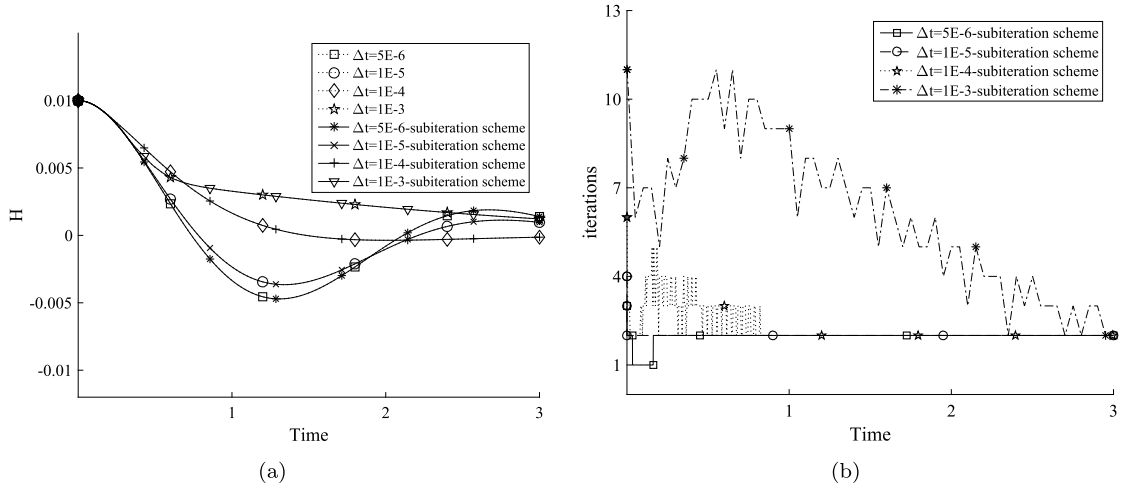


Fig. 6. Capillary wave (density ratio $\rho_2/\rho_1 = 1000$): (a) Comparison of time histories of capillary amplitudes obtained using methods with and without subiterations corresponding to various Δt . (b) Evolution of the number of subiterations needed within a time step over time for the sub-iteration method. Those curves in (a) with legends not marked by “subiteration scheme” are obtained using the method without subiteration.

a dramatic effect on the motions of the interfaces. Increase in the density ratio increases the oscillation period and the attenuation of the amplitude becomes more pronounced. It can also be observed that the history curves from the simulations agree well with those of the exact solutions for all the density ratios, indicating that our method has captured the dynamics of the fluid interfaces correctly.

As mentioned before, the results reported here by default are obtained using the approximation (72) (i.e. without subiteration) when solving equation (70). In Fig. 6 we compare the results obtained using the methods with and without subiterations when solving equation (70). Fig. 6(a) shows time histories of the capillary wave amplitude obtained using the two methods corresponding to several Δt values ranging from $\Delta t = 5 \times 10^{-6}$ to $\Delta t = 10^{-3}$ for the density ratio $\rho_2/\rho_1 = 1000$. We observe that the time histories with and without subiterations overlap with each other corresponding to different Δt values, suggesting that both methods have produced the same simulation results. Fig. 6(b) shows the evolution of the number of subiterations within a time step over time. The number of subiterations is observed to vary over time (around 2 to 10), and it tends to increase with a larger Δt .

The capillary wave problem and in particular the comparisons with Prosperetti's exact solution for this problem demonstrate that our algorithm produces physically accurate results for a wide range of density ratios (up to density ratio 1000 tested here), and that it is stable for large time step sizes in long-time simulations.

3.3. Rising air bubble in water

In this section, we test the proposed method using a two-phase flow problem in an irregular domain, involving realistically large density ratios and viscosity ratios encountered in the physical world. The density ratio and viscosity ratio of the two fluids correspond to those of the air and water in the majority of cases considered here. But for some cases the density ratio of the fluids will also be artificially varied, and for those cases we will still refer to these fluids as air and water.

Table 4
Simulation parameter values for the rising bubble problem.

Parameter	Value	Parameter	Value
(x_0, y_0)	(0.08, -0.3)	R_0	0.1
σ	10	$ \mathbf{g}_r $ (gravity)	1.0
ρ_1	1.0	ρ_2	100 or 1000
μ_1	0.01	μ_2	0.5
ρ_0	$\min(\rho_1, \rho_2)$	ν_m	$2 \times \frac{\max(\mu_1, \mu_2)}{\min(\rho_1, \rho_2)}$
C_0	0	S	$\sqrt{\frac{4\gamma_0\lambda}{m\Delta t}}$
m	10^{-7}	η	0.01
λ	$\frac{3}{2\sqrt{2}}\sigma\eta$	Δt	5×10^{-5} (or varied)
J (temporal order)	2	Number of elements	146
Element order	10		

To be more specific, we consider a rectangular domain, defined by $\Omega = \{(x, y) | -0.25 \leq x \leq 0.25, -0.5 \leq y \leq 0.25\}$. The top and bottom of the domain are solid walls, and in the horizontal direction it is periodic. A square region inside this domain, centered at (0, 0) with a dimension $l = 0.1$ on each side, is occupied by a solid object; see Fig. 7(a) for a sketch of the configuration. The flow (and computational) domain consists of the space between the interior solid-square object and the top/bottom walls on the outside. This domain is filled with water, which traps an air bubble inside; see Fig. 7(a). The gravity is assumed to point downward. At $t = 0$, the air bubble is circular and centered at $(x_0, y_0) = (0.08, -0.3)$ with a radius $R_0 = 0.1$, and it is at rest. Then the system is released, and the air bubble rises through the water and interacts with the walls of the interior object and the outside domain wall. We would like to simulate this dynamic process using the current method.

The computational domain is partitioned with 146 quadrilateral elements, with 10 and 15 uniform elements respectively in the x and y directions (with 4 elements excluded from the region of the solid square). In the simulations, the external body force \mathbf{f} in equation (7) is set to $\mathbf{f} = \rho \mathbf{g}_r$, where \mathbf{g}_r is the gravitational acceleration and ρ is the mixture density. The source term in equation (1c) is set to $g = 0$. We further assume that all the wall surfaces have a neutral wettability. In other words, if the air-water interface intersects the outside domain walls or the walls of the interior solid square, the contact angle on the wall would be 90° . On the outside top/bottom walls and the surfaces of the interior solid square, the boundary condition (9) with $\mathbf{w} = \mathbf{0}$ is imposed for the velocity, and the boundary conditions (10) and (11) with $d_a = d_b = 0$ are imposed for the phase field function. On the two sides of the domain in the horizontal direction ($x = \pm 0.25$) periodic boundary conditions are imposed for all flow variables. The initial velocity is set to zero, and the initial phase field function is prescribed as follows:

$$\phi(\mathbf{x}, 0) = -\tanh \frac{\sqrt{(x-x_0)^2 + (y-y_0)^2} - R_0}{\sqrt{2}\eta}. \quad (91)$$

In the simulations, we treat air as the first fluid and water as the second fluid. We employ a viscosity ratio $\frac{\mu_2}{\mu_1} = 50$ and a density ratio $\frac{\rho_2}{\rho_1} = 1000$ for the air and water. Another artificial density ratio $\frac{\rho_2}{\rho_1} = 100$ is also considered in the tests. The normalized physical and numerical parameters involved in this problem are listed in Table 4.

We first consider the case with an artificially smaller density contrast for air and water, $(\rho_1, \rho_2) = (1, 100)$. Fig. 7 is a sequence of snapshots in time of the fluid interfaces for this case, visualized by the contour level $\phi(\mathbf{x}, t) = 0$. These results are computed with a time step size $\Delta t = 5 \times 10^{-5}$. From $t = 0$ to about $t = 1.6$ (Fig. 7(a)-(c)), the air bubble rises through the water inside the container, and gradually approaches the bottom right corner of the interior square object. A slight deformation of the bubble can be noticed. The air bubble collides onto the square object at approximately $t = 1.7$ (Fig. 7(d)), and is drawn rapidly towards the cylinder due to the strong effect of surface tension (Fig. 7(e)-(g)). From $t = 2.25$ to $t = 6$ (Fig. 7(h)-(l)), the air bubble slides on the bottom surface of the interior square object, and moves toward the right. It eventually settles down at an equilibrium position, still attached to the right and top sides of the solid square.

We have also simulated this problem using the subiteration method, in which a subiteration procedure is employed when solving the equation (70). Fig. 8 shows a temporal sequence of snapshots of the air bubble, visualized by the contour levels $\phi(\mathbf{x}, t) = 0$, obtained using the subiteration method. The computed configurations of the air bubble by the subiteration method, instant by instant, are the same as those shown in Fig. 7, which are computed based on the method using the approximation (72). Therefore, only the snapshots on a subset of time instants corresponding to Fig. 7 have been shown here in Fig. 8 for the subiteration method. Fig. 9 shows the evolution of the number of subiterations needed within a time step over time. It can be observed that the number of subiterations varies between 2 and 14 for this problem and it changes over time. The above results show that the method using the approximation (72) and the subiteration method lead to the same results even for problems with a fairly complicated flow domain.

Let us now look into the case with a normal density contrast for air and water, $(\rho_1, \rho_2) = (1, 1000)$. In Fig. 10, we plot a temporal sequence of snapshots of the air-water interface for this case, computed with $\Delta t = 5 \times 10^{-5}$. The dynamical characteristics observed here are notably different and much more complicated than those in the previous case with a

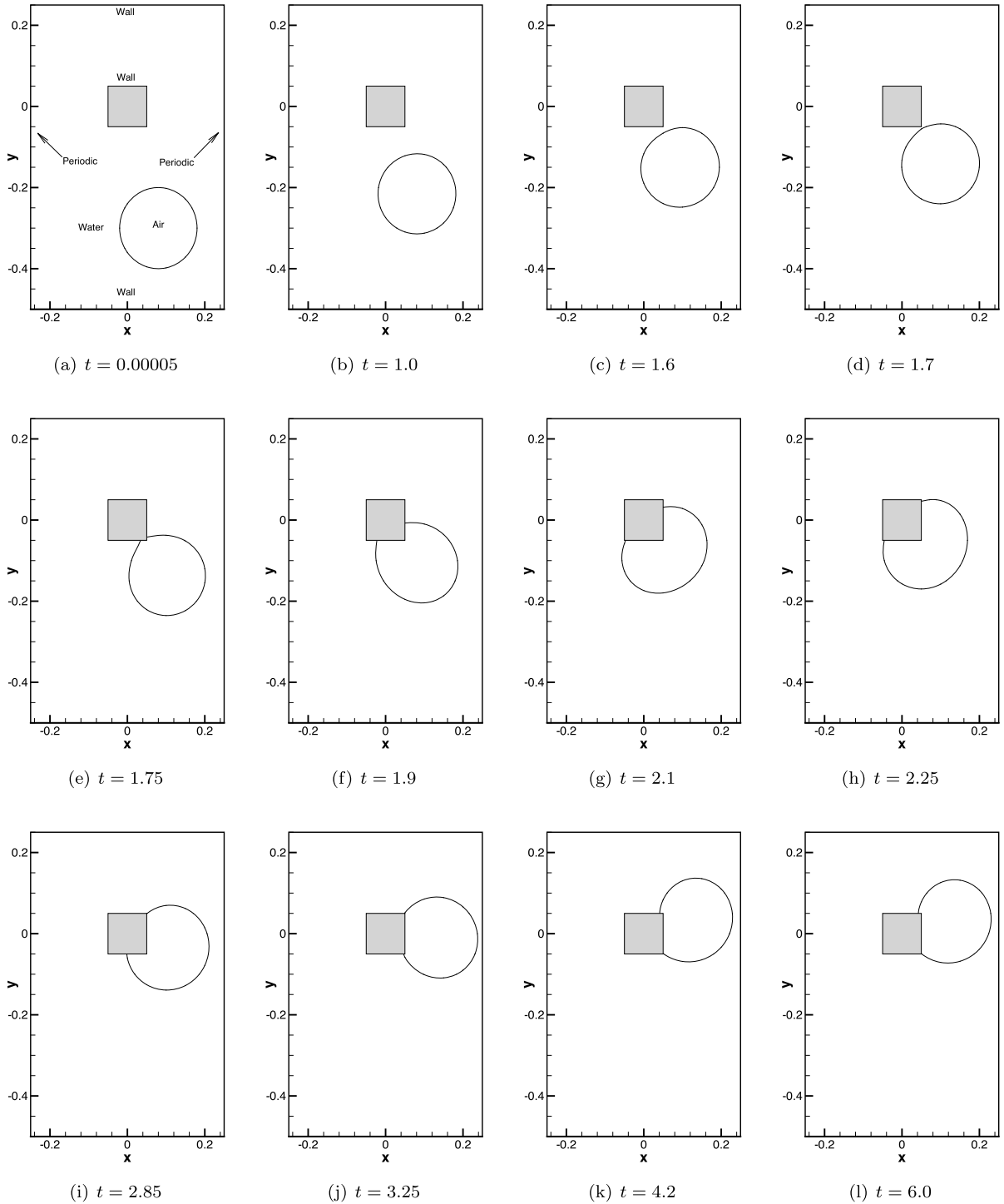


Fig. 7. Time sequence of snapshots of an air bubble rising in water with a smaller density contrast $(\rho_1, \rho_2) = (1, 100)$. Fluid interface is visualized by the contour levels $\phi(\mathbf{x}, t) = 0$.

smaller density contrast. As the air bubble rises through the water from $t = 0$ to $t = 0.85$ (Fig. 10(a)–(c)), the bubble experiences more considerable deformation and one can observe a flat bottom side in the shape of the bubble. Notice also that the rise of the bubble is more rapid than in the previous case. From $t = 1.0$ to $t = 1.1$ (Fig. 10(d)–(e)), the air bubble approaches the bottom right corner of the solid square, and an indentation appears on the shoulder of the bubble. The air bubble collides onto the square and is attached to its surface near the bottom right corner from $t = 1.2$ to $t = 1.55$

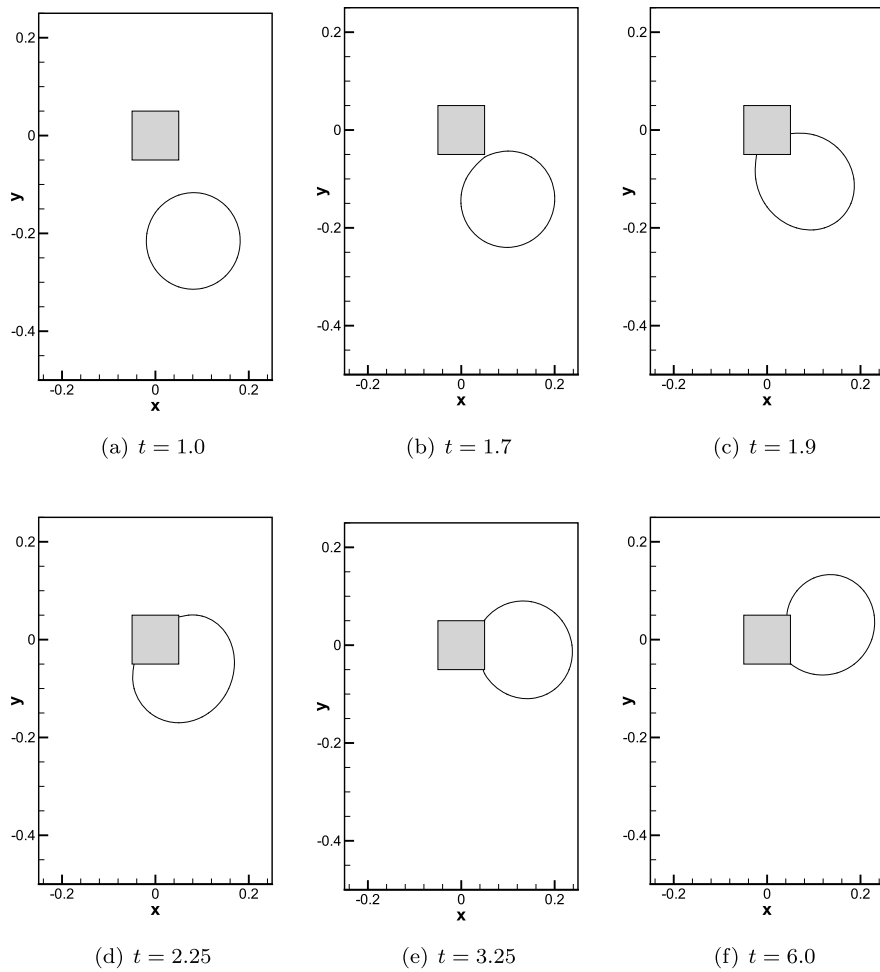


Fig. 8. Comparison with the sub-iteration method: Time sequence of snapshots of the air bubble rising in water with a density contrast $(\rho_1, \rho_2) = (1, 100)$, obtained using the sub-iteration method. Shown are the contour levels $\phi(\mathbf{x}, t) = 0$. Simulation parameters are identical to those of Fig. 7.

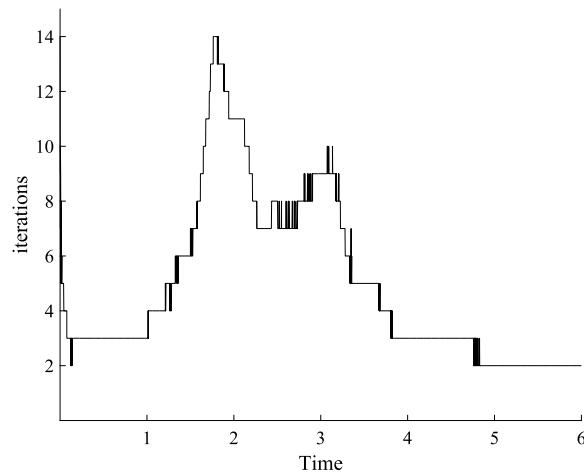


Fig. 9. Rising air bubble in water (density contrast $(\rho_1, \rho_2) = (1, 100)$) computed using the subiteration method: Evolution of the number of subiterations needed within a time step over time.

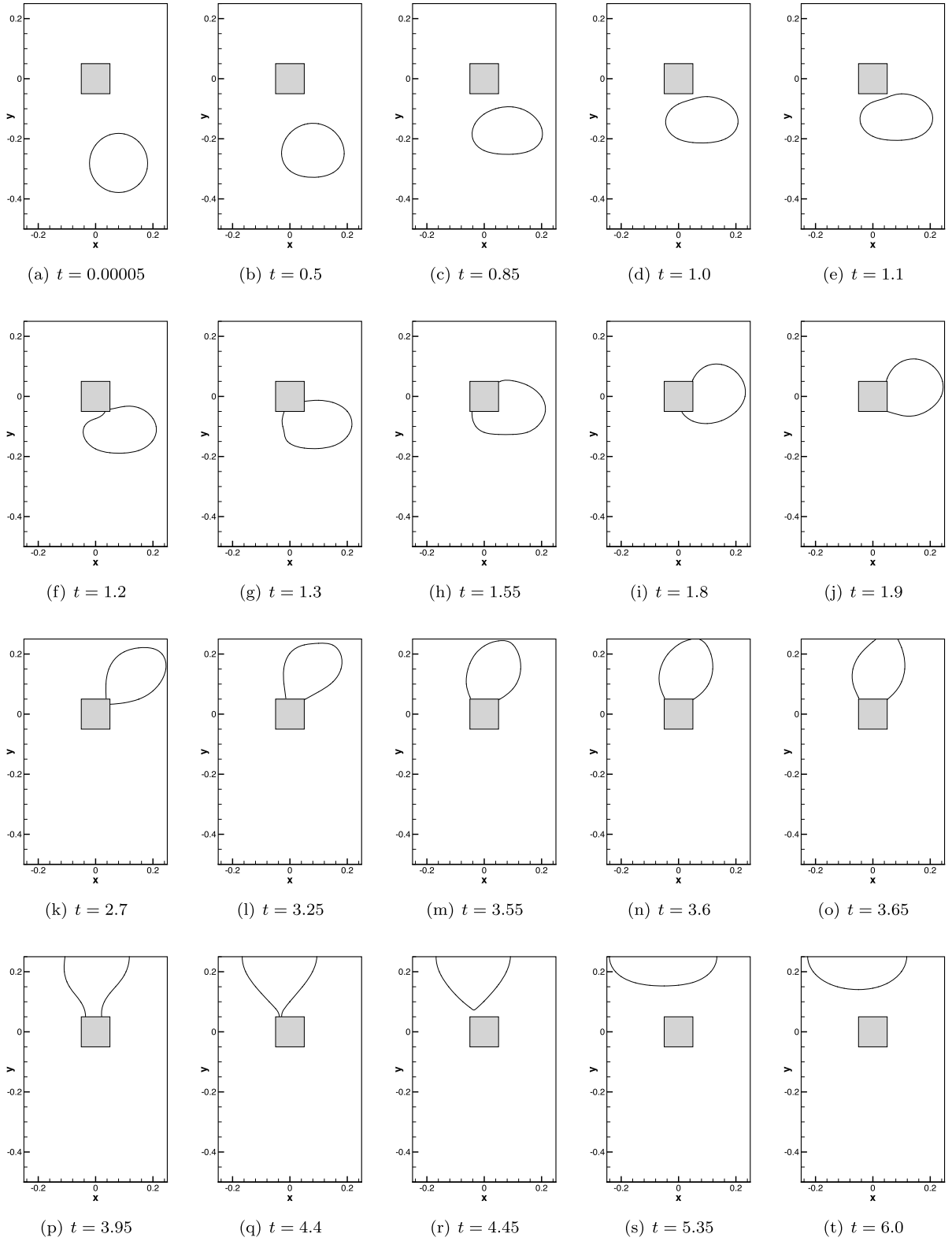


Fig. 10. Time sequence of snapshots of an air bubble rising in water with a normal density contrast $(\rho_1, \rho_2) = (1, 1000)$. Shown are the contour levels $\phi(\mathbf{x}, t) = 0$.

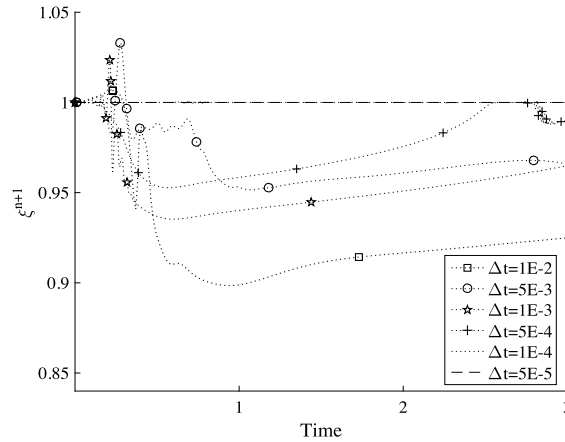


Fig. 11. Comparison of time histories of ξ^{n+1} obtained with various time step sizes $\Delta t = 10^{-2} \sim 5 \times 10^{-5}$ for the rising air bubble problem with density contrast $(\rho_1, \rho_2) = (1, 1000)$.

(Fig. 10(f)–(h)). Afterwards, the bubble slides on the surface of the solid square, from the bottom to the right and then to the top of the interior square (Fig. 10(g)–(m)). The bubble deformation is very pronounced during this process. Due to strong buoyancy, the bubble is stretched upward while still attached to the top surface of the solid square, and it approaches the top wall of the domain (Fig. 10(m)). From $t = 3.6$ to $t = 4.4$ (Fig. 10(m)–(q)), the air bubble touches the upper domain wall and forms contact lines on that wall. Simultaneously, the area of contact between the bubble and the top surface of the interior square shrinks, and the bubble deforms into a funnel (Fig. 10(p)–(q)). From $t = 4.45$ to $t = 6.0$ (Fig. 10(r)–(t)), the air bubble detaches from the interior solid square, and forms a dome attached to the upper domain wall over time.

In order to demonstrate the robustness of our method, we have performed tests using the second case $(\rho_1, \rho_2) = (1, 1000)$ for various time step sizes ranging from $\Delta t = 10^{-2}$ to 5×10^{-5} . Fig. 11 is a comparison of the time histories of ξ^{n+1} for each time step size. We observe that even with large time step sizes (e.g. $\Delta t = 10^{-2}$), the computation is stable. We can observe that when Δt is relatively small ($\Delta t \leq 10^{-4}$), the computed ξ^{n+1} is identical to 1. For larger Δt , the computed values for ξ^{n+1} initially oscillate near 1 (from $t = 0.2$ to $t = 0.4$), and then decreases and settles to some levels less than 1 over a long time. With a larger Δt we generally attain a ξ^{n+1} level over time that tends to be smaller.

Remark 4. The computational cost for solving the nonlinear equation (85) using the Newton's method, including the computations for the coefficients involved therein for preparation of the Newton solver, is small compared with the total cost within each time step. For example, with the same computational setting for producing the results of Fig. (10) for the rising air bubble problem, the time spent in the Newton solver (including the time spent on computing the coefficients in equation (85)) per time step accounts for about 8% of the total time per time step with our current method. The total wall time per time step is 0.3 seconds on the average, in which around 0.024 seconds is spent in the Newton solution for equation (85). Of the time spent in the Newton method, essentially all is spent on computing the coefficients involved in the nonlinear equation (85), and the actual Newton iteration time is negligible.

4. Concluding remarks

In this paper we have developed an energy-stable scheme for approximating the coupled system of Navier-Stokes/Cahn-Hilliard equations for incompressible two-phase flows with different densities and viscosities for the two fluids. The scheme employs a scalar-valued auxiliary variable in the formulation, and it satisfies a discrete energy stability property regardless of the time step sizes. The scheme allows an efficient solution procedure and is computationally attractive. Within a time step, it computes two copies of the flow variables (velocity, pressure, phase field function), by solving individually a linear algebraic system involving a constant and time-independent coefficient matrix for each of these field functions. The coefficient matrices involved in these linear systems only need to be computed once and they can be pre-computed, even with large density ratios and large viscosity ratios. Additionally, the algorithm requires the solution of a nonlinear algebraic equation about a *scalar-valued number* using the Newton's method within each time step. The computational cost for this nonlinear solver is very low, accounting for around 8% of the total solver time per time step, because this nonlinear equation is about a scalar number, not a field function. Because of these properties, the presented method is robust and computationally efficient.

We have tested the proposed method with extensive numerical experiments for several two-phase flow problems involving large density ratios and viscosity ratios. In particular, by comparing with Prosperetti's exact solution for the capillary wave problem, we have shown that our method produces physically accurate results for various density ratios and viscosity ratios. Large time step sizes have been tested with the proposed method in two-phase simulations. While the simulation

results cannot be expected to be accurate at those time step sizes, the computations have been shown to be stable nonetheless, verifying the robustness of the method developed herein.

Some comments concerning the auxiliary variable strategy are in order. It should be noted that the coupled Navier-Stokes/Cahn-Hilliard equations are not a gradient-type system. As such, we find that the auxiliary variable formulation as described in e.g. [38] for gradient flows does not carry over to Navier-Stokes equations or two-phase flows. Two lessons are learned:

- Attempts to reformulate the viscous terms in these equations using the auxiliary variable, which is analogous to the auxiliary-variable treatments of the dissipation terms in the evolution equation for gradient flows, lead to very poor simulation results. A viable strategy for the two-phase governing equations is to employ the auxiliary variable to reformulate the convection terms in the momentum equation and the phase field equation. The integral form of the dynamic equation for the scalar-valued auxiliary energy variable provides a great deal of flexibility to be commensurate with the corresponding treatments of various terms in the momentum and phase-field equations to ensure a discrete energy stability property.
- When solving for the scalar-valued number, $\xi^{n+1} = \frac{R^{n+1}}{\sqrt{E^{n+1}}}$, it should be noted that in the implementation we have employed equation (82), which is based on the transformed dynamic equation (50) instead of the original equation (32). While equations (32) and (50) are mathematically equivalent, we find that in practice the method is more robust if the implementation is based on the transformed equation (50).

An important implication of the current method lies in the following. It demonstrates that the energy-stable schemes for two-phase flows can also become computationally efficient, eliminating the expensive re-computations for the coefficient matrices involved in the associated linear algebraic systems. In terms of the amount of operations per time step, the energy-stable schemes can be competitive. For example, within each time step, the amount of operations involved in the current method is approximately twice that of the semi-implicit method of [15], which is only conditionally stable, or slightly larger than that due to the extra nonlinear equation about a scalar-valued number.

Acknowledgement

This work was partially supported by NSF (DMS-1318820, DMS-1522537). Useful discussions with Professor J. Shen (Purdue University) are gratefully acknowledged.

Appendix A. Spectral element implementation of numerical scheme

In this appendix we provide a summary of the spatial discretization of the equations (62)–(65), (74)–(75) and (79)–(80) using high-order spectral elements. We discretize the domain Ω with a mesh consisting of N_{el} conforming C^0 spectral elements [24]. Let the positive integer K denote the element order, which represents a measure of the highest polynomial degree in the polynomial expansions of the field variables within an element. Let Ω_h denote the discretized domain, $\partial\Omega_h$ denote the boundary of Ω_h , and Ω_h^e ($1 \leq e \leq N_{el}$) denote the element e . Define three function spaces

$$\begin{aligned}\mathbb{X} &= \{v \in H^1(\Omega_h) : v \text{ is a polynomial characterized by } K \text{ on } \Omega_h^e, \text{ for } 1 \leq e \leq N_{el}\}, \\ \mathbb{V}_0 &= \{v \in \mathbb{X} : \int_{\Omega_h} v d\Omega_h = 0\}, \\ \mathbb{V}_1 &= \{v \in \mathbb{X} : v|_{\partial\Omega_h} = 0\}.\end{aligned}\tag{92}$$

In the following, let d ($d = 2$ or 3) denote the spatial dimension, and the subscript in $(\cdot)_h$ denote the discretized version of the variable (\cdot) . The fully discretized equations corresponding to equations (62)–(65) are:

Find $\psi_{1h}^{n+1} \in \mathbb{X}$, such that

$$\begin{aligned}\int_{\Omega_h} \nabla \psi_{1h}^{n+1} \cdot \nabla \varphi_h + \left(\alpha + \frac{S}{\lambda}\right) \int_{\Omega_h} \psi_{1h}^{n+1} \varphi_h &= -\frac{S}{\lambda} \int_{\Omega_h} \nabla \phi_h^{*,n+1} \cdot \nabla \varphi_h - \frac{1}{\lambda m} \int_{\Omega_h} \left(\frac{\hat{\phi}_h}{\Delta t} + g_h^{n+1}\right) \varphi_h \\ &+ \int_{\partial\Omega_h} \left[\frac{S}{\lambda} d_{ah}^{*,n+1} + \alpha d_{ah}^{n+1} + d_{bh}^{n+1}\right] \varphi_h, \quad \forall \varphi_h \in \mathbb{X}.\end{aligned}\tag{93}$$

Find $\phi_{1h}^{n+1} \in \mathbb{X}$, such that

$$\int_{\Omega_h} \nabla \phi_{1h}^{n+1} \cdot \nabla \varphi_h - \alpha \int_{\Omega_h} \phi_{1h}^{n+1} \varphi_h = - \int_{\Omega_h} \psi_{1h}^{n+1} \varphi_h + \int_{\partial \Omega_h} d_{ah}^{n+1} \varphi_h, \quad \forall \varphi_h \in \mathbb{X}, \quad (94)$$

Find $\psi_{2h}^{n+1} \in \mathbb{X}$, such that

$$\begin{aligned} \int_{\Omega_h} \nabla \psi_{2h}^{n+1} \cdot \nabla \varphi_h + \left(\alpha + \frac{S}{\lambda} \right) \int_{\Omega_h} \psi_{2h}^{n+1} \varphi_h &= \frac{1}{\lambda} \int_{\Omega_h} \nabla h(\phi_h^{*,n+1}) \cdot \nabla \varphi_h \\ &+ \frac{1}{\lambda m} \int_{\Omega_h} (\mathbf{u}_h^{*,n+1} \cdot \nabla \phi_h^{*,n+1}) \varphi_h - \frac{1}{\lambda} \int_{\partial \Omega_h} h'(\phi_h^{*,n+1}) d_{ah}^{*,n+1}, \quad \forall \varphi_h \in \mathbb{X}, \end{aligned} \quad (95)$$

Find $\phi_{2h}^{n+1} \in \mathbb{X}$, such that

$$\int_{\Omega_h} \nabla \phi_{2h}^{n+1} \cdot \nabla \varphi_h - \alpha \int_{\Omega_h} \phi_{2h}^{n+1} \varphi_h = - \int_{\Omega_h} \psi_{2h}^{n+1} \varphi_h, \quad \forall \varphi_h \in \mathbb{X}, \quad (96)$$

The fully discretized equations corresponding to (74)–(75) are:

Find $P_{1h}^{n+1} \in \mathbb{V}_0$ such that

$$\int_{\Omega_h} \nabla P_{1h}^{n+1} \cdot \nabla q_h = \rho_0 \int_{\Omega_h} \left(\frac{\hat{\mathbf{u}}_h}{\Delta t} + \frac{C_{2h}^{n+1}}{\bar{\rho}_h^{n+1}} \nabla \bar{\phi}_h^{n+1} + \frac{\mathbf{f}_h^{n+1}}{\bar{\rho}_h^{n+1}} \right) \cdot \nabla q_h - \frac{\rho_0 \gamma_0}{\Delta t} \int_{\partial \Omega_h} \mathbf{n}_h \cdot \mathbf{w}_h^{n+1} q_h, \quad \forall q_h \in \mathbb{X} \quad (97)$$

Find $P_{2h}^{n+1} \in \mathbb{V}_0$ such that

$$\begin{aligned} \int_{\Omega_h} \nabla P_{2h}^{n+1} \cdot \nabla q_h &= \rho_0 \int_{\Omega_h} \left(\frac{C_{2h}^{n+1}}{\bar{\rho}_h^{n+1}} \nabla \bar{\phi}_h^{n+1} - \mathbf{Q}_h + \nabla \left(\frac{\bar{\mu}_h^{n+1}}{\bar{\rho}_h^{n+1}} \right) \times \boldsymbol{\omega}_h^{*,n+1} \right) \cdot \nabla q_h \\ &- \rho_0 \int_{\partial \Omega_h} \frac{\bar{\mu}_h^{n+1}}{\bar{\rho}_h^{n+1}} \mathbf{n}_h \times \boldsymbol{\omega}_h^{*,n+1} \cdot \nabla q_h, \quad \forall q_h \in \mathbb{X}. \end{aligned} \quad (98)$$

The fully discretized equations corresponding to (79)–(80) are:

Find $\mathbf{u}_1^{n+1} \in [\mathbb{X}]^d$ such that

$$\frac{\gamma_0}{v_m \Delta t} \int_{\Omega_h} \mathbf{u}_{1h}^{n+1} \varphi_h + \int_{\Omega_h} \nabla \mathbf{u}_{1h}^{n+1} \cdot \nabla \varphi_h = \frac{1}{v_m} \int_{\Omega_h} \left(- \frac{1}{\rho_0} \nabla P_{1h}^{n+1} + \frac{\hat{\mathbf{u}}_h}{\Delta t} + \frac{C_{2h}^{n+1}}{\bar{\rho}_h^{n+1}} \nabla \bar{\phi}_h^{n+1} + \frac{\mathbf{f}_h^{n+1}}{\bar{\rho}_h^{n+1}} \right) \varphi_h, \quad \forall \varphi_h \in \mathbb{V}_1, \quad (99a)$$

$$\mathbf{u}_{1h}^{n+1} = \mathbf{w}_h^{n+1}, \quad \text{on } \partial \Omega. \quad (99b)$$

Find $\mathbf{u}_2^{n+1} \in [\mathbb{V}_1]^d$ such that

$$\begin{aligned} \frac{\gamma_0}{v_m \Delta t} \int_{\Omega_h} \mathbf{u}_{2h}^{n+1} \varphi_h + \int_{\Omega_h} \nabla \mathbf{u}_{2h}^{n+1} \cdot \nabla \varphi_h &= - \frac{1}{v_m} \int_{\Omega_h} \left(\frac{\bar{\mu}_h^{n+1}}{\bar{\rho}_h^{n+1}} - v_m \right) \boldsymbol{\omega}_h^{*,n+1} \times \nabla \varphi_h \\ &+ \frac{1}{v_m} \int_{\Omega_h} \left(- \frac{1}{\rho_0} \nabla P_{2h}^{n+1} + \frac{C_{2h}^{n+1}}{\bar{\rho}_h^{n+1}} \nabla \bar{\phi}_h^{n+1} - \mathbf{Q}_h + \nabla \left(\frac{\bar{\mu}_h^{n+1}}{\bar{\rho}_h^{n+1}} \right) \times \boldsymbol{\omega}_h^{*,n+1} \right) \varphi_h, \quad \forall \varphi_h \in \mathbb{V}_1, \end{aligned} \quad (100)$$

References

- [1] H. Abels, H. Garcke, G. Grün, Thermodynamically consistent, frame indifferent diffuse interface models for incompressible two-phase flows with different densities, *Math. Models Methods Appl. Sci.* 22 (2012) 1150013.
- [2] G.L. Aki, W. Dreyer, J. Giesselmann, A quasi-incompressible diffuse interface model with phase transition, *Math. Models Methods Appl. Sci.* 24 (2014) 827–861.

- [3] S. Aland, F. Chen, An efficient and energy stable scheme for a phase-field model for the moving contact line problem, *Int. J. Numer. Methods Fluids* 81 (2016) 657–671.
- [4] S.M. Allen, J.W. Cahn, A microscopic theory for antiphase boundary motion and its application to antiphase domain coarsening, *Acta Metall.* 27 (1979) 1085–1095.
- [5] D.M. Anderson, G.B. McFadden, A.A. Wheeler, Diffuse-interface methods in fluid mechanics, *Annu. Rev. Fluid Mech.* 30 (1998) 139–165.
- [6] V.E. Badalassi, H.D. Ceniceros, S. Banerjee, Computation of multiphase systems with phase field models, *J. Comput. Phys.* 190 (2003) 371–397.
- [7] F. Boyer, S. Minjeaud, Numerical schemes for a three component Cahn-Hilliard model, *ESAIM: Modél. Math. Anal. Numér.* 45 (2011) 697–738.
- [8] J.W. Cahn, J.E. Hilliard, Free energy of a nonuniform system. I interfacial free energy, *J. Chem. Phys.* 28 (1958) 258–267.
- [9] H. Ding, P.D.M. Spelt, C. Shu, Diffuse interface model for incompressible two-phase flows with large density ratios, *J. Comput. Phys.* 226 (2007) 2078–2095.
- [10] S. Dong, On imposing dynamic contact-angle boundary conditions for wall-bounded liquid-gas flows, *Comput. Methods Appl. Mech. Eng.* 247–248 (2012) 179–200.
- [11] S. Dong, An efficient algorithm for incompressible N-phase flows, *J. Comput. Phys.* 276 (2014) 691–728.
- [12] S. Dong, An outflow boundary condition and algorithm for incompressible two-phase flows with phase field approach, *J. Comput. Phys.* 266 (2014) 47–73.
- [13] S. Dong, Physical formulation and numerical algorithm for simulating N immiscible incompressible fluids involving general order parameters, *J. Comput. Phys.* 283 (2015) 98–128.
- [14] S. Dong, Multiphase flows of N immiscible incompressible fluids: a reduction-consistent and thermodynamically-consistent formulation and associated algorithm, *J. Comput. Phys.* 361 (2018) 1–49.
- [15] S. Dong, J. Shen, A time-stepping scheme involving constant coefficient matrices for phase field simulations of two-phase incompressible flows with large density ratios, *J. Comput. Phys.* 231 (2012) 5788–5804.
- [16] Y. Gong, J. Zhao, Q. Wang, An energy stable algorithm for a quasi-incompressible hydrodynamic phase-field model of viscous fluid mixtures with variable densities and viscosities, *Comput. Phys. Commun.* 219 (2017) 20–34.
- [17] Y. Gong, J. Zhao, X. Yang, Q. Wang, Fully discrete second-order linear schemes for hydrodynamic phase field models of binary viscous fluid flows with variable densities, *SIAM J. Sci. Comput.* 40 (2018) B138–B167.
- [18] G. Grun, F. Klingbeil, Two-phase flow with mass density contrast: stable schemes for a thermodynamically consistent and frame-indifferent diffuse-interface model, *J. Comput. Phys.* 257 (2014) 708–725.
- [19] J.L. Guermond, L. Quartapelle, A projection FEM for variable density incompressible flows, *J. Comput. Phys.* 165 (2000) 167–188.
- [20] Z. Guo, P. Lin, J. Lowengrub, S.M. Wise, Mass conservative and energy stable finite difference methods for the quasi-incompressible Navier-Stokes-Cahn-Hilliard system: primitive and projection-type schemes, *Comput. Methods Appl. Mech. Eng.* 326 (2017) 144–174.
- [21] Z. Guo, P. Lin, J.S. Lowengrub, A numerical method for the quasi-incompressible Cahn-Hilliard-Navier-Stokes equations for variable density flows with a discrete energy law, *J. Comput. Phys.* 276 (2014) 486–507.
- [22] D. Han, A. Beylev, X. Yang, Z. Tan, Numerical analysis of second order, fully discrete energy stable schemes for phase field models of two-phase incompressible flows, *J. Sci. Comput.* 70 (2017) 965–989.
- [23] D. Jacqmin, Calculation of two-phase Navier-Stokes flows using phase-field modeling, *J. Comput. Phys.* 155 (1999) 96–127.
- [24] G.E. Karniadakis, S.J. Sherwin, *Spectral/hp Element Methods for Computational Fluid Dynamics*, 2nd edition, Oxford University Press, 2005.
- [25] J. Kim, J. Lowengrub, Phase field modeling and simulation of three-phase flows, *Interfaces Free Bound.* 7 (2005) 435–466.
- [26] L. Lin, Z. Yang, S. Dong, Numerical approximation of incompressible Navier-Stokes equations based on an auxiliary energy variable, *J. Comput. Phys.* 388 (2019) 1–22.
- [27] C. Liu, J. Shen, A phase field model for the mixture of two incompressible fluids and its approximation by a Fourier-spectral method, *Physica D* 179 (2003) 211–228.
- [28] C. Liu, J. Shen, X. Yang, Decoupled energy stable schemes for a phase-field model of two-phase incompressible flows with variable density, *J. Sci. Comput.* 62 (2015) 601–622.
- [29] J. Lowengrub, L. Truskinovsky, Quasi-incompressible Cahn-Hilliard fluids and topological transitions, *Proc. R. Soc. Lond. A* 454 (1998) 2617–2654.
- [30] S.J. Osher, J.A. Sethian, Fronts propagating with curvature dependent speed: algorithms based on Hamilton-Jacobi formulations, *J. Comput. Phys.* 79 (1988) 12–49.
- [31] A. Prosperetti, Motion of two superposed viscous fluids, *Phys. Fluids* 24 (1981) 1217–1223.
- [32] T. Qian, X.-P. Wang, P. Sheng, A variational approach to moving contact line hydrodynamics, *J. Fluid Mech.* 564 (2006) 333–360.
- [33] L. Rayleigh, On the theory of surface forces II, *Philos. Mag.* 33 (1892) 209.
- [34] M. Schorpour Roudbari, G. Simsek, E.H. van Brummelen, K.G. van der Zee, Diffuse-interface two-phase flow models with different densities: a new quasi-incompressible form and a linear energy-stable method, *Math. Models Methods Appl. Sci.* 28 (2018) 733–770.
- [35] A.J. Salgado, A diffuse interface fractional time-stepping technique for incompressible two-phase flows with moving contact lines, *ESAIM: Modél. Math. Anal. Numér.* 47 (2013) 743–769.
- [36] R. Scardovelli, S. Zaleski, Direct numerical simulation of free-surface and interfacial flow, *Annu. Rev. Fluid Mech.* 31 (1999) 567–603.
- [37] J.A. Sethian, P. Semerka, Level set methods for fluid interfaces, *Annu. Rev. Fluid Mech.* 35 (2003) 341–372.
- [38] J. Shen, J. Xu, J. Yang, The scalar auxiliary variable (SAV) approach for gradient flows, *J. Comput. Phys.* 353 (2018) 407–416.
- [39] J. Shen, X. Yang, Energy stable schemes for Cahn-Hilliard phase-field model of two-phase incompressible flows, *Chin. Ann. Math., Ser. B* 31 (2010) 743–758.
- [40] J. Shen, X. Yang, A phase-field model and its numerical approximation for two-phase incompressible flows with different densities and viscosities, *SIAM J. Sci. Comput.* 32 (2010) 1159–1179.
- [41] J. Shen, X. Yang, Decoupled, energy stable schemes for phase-field models of two-phase incompressible flows, *SIAM J. Numer. Anal.* 53 (2015) 279–296.
- [42] J. Shen, X. Yang, Q. Wang, On mass conservation in phase field models for binary fluids, *Commun. Comput. Phys.* 13 (2013) 1045–1065.
- [43] G. Tryggvason, B. Bunner, A. Esmaeili, et al., A front-tracking method for computations of multiphase flow, *J. Comput. Phys.* 169 (2001) 708–759.
- [44] J. van der Waals, The thermodynamic theory of capillarity under the hypothesis of a continuous density variation, *J. Stat. Phys.* 20 (1893) 197–244.
- [45] X. Yang, Linear first and second-order, unconditionally energy stable numerical schemes for the phase field model of homopolymer blends, *J. Comput. Phys.* 327 (2016) 294–316.
- [46] X. Yang, H. Yu, Efficient second order unconditionally stable schemes for a phase field moving contact line model using an invariant energy quadratization approach, *SIAM J. Sci. Comput.* 40 (2018) B889–B914.
- [47] Z. Yang, L. Lin, S. Dong, A family of second-order energy-stable schemes for Cahn-Hilliard type equations, *J. Comput. Phys.* 383 (2019) 24–54.
- [48] H. Yu, X. Yang, Numerical approximations for a phase-field moving contact line model with variable densities and viscosities, *J. Comput. Phys.* 334 (2017) 665–686.
- [49] 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.