Journal of Computational Physics 551 (2026) 114680

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

A consistent and scalable framework suitable for boiling flows using the conservative diffuse interface method

Lorenz Weber [a,*], Aritra Mukherjee [b], Andreas G. Class [a], Luca Brandt [b,c]

[a] Institute for Thermal Energy Technology and Safety (ITES), Karlsruhe Institute of Technology (KIT), Hermann-Von-Helmholtz-Platz 1, Eggenstein-Leopoldshafen, 76344, Germany
[b] Department of Energy and Process Engineering, Norwegian University of Science and Technology (NTNU), 229 Strømningstekniske laboratorier Gløshaugen, Trondheim, 7034, Norway
[c] Department of Environment, Land and Infrastructure Engineering (DIATI), Politecnico di Torino, Corso Duca degli Abruzzi 24, Torino, 10129, Italy

ARTICLE INFO

Keywords:
Phase field
Boiling
Conservative diffuse interface
Phase change
FluTAS

ABSTRACT

Interface-resolved simulations are essential for predicting and understanding boiling heat transfer phenomena. Such simulations generally come at a high computational cost, which continues to motivate the development of efficient frameworks. In recent years, conservative second-order phase field methods have gained popularity due to their efficient representation of phase interfaces. However, their potential for simulating complex boiling phenomena has not yet been explored. To address this gap, we develop a consistent and highly efficient framework suitable for simulating large-scale boiling flows. We derive a set of mixture equations to describe the two-phase flow. The mixture equations are coupled with the *accurate conservative diffuse interface* method [1] to capture the interface. We present additional terms in the momentum balance equation and demonstrate that the proposed momentum balance modifications are mandatory for accurately capturing phase-change-induced pressure jumps. To solve the set of equations, an alternative Fast Fourier Transform (FFT)-based pressure solution scheme is proposed. Additionally, a modified kinetic phase change model is utilized that does not involve calculating temperature gradients and avoids problem-dependent parameters. The framework is tested against a variety of benchmark simulations, both with and without phase change. Moreover, we achieve improved accuracy when simulating bubble dynamics without phase change at high density ratios. We show that the proposed FFT-based pressure solution scheme exhibits superior performance in calculating interfacial pressure jumps compared with a commonly used FFT solver. Regardless of phase change, more accurate startup behaviour is observed. In the presence of phase change, we are successful in removing interfacial pressure oscillations. Across all phase-change benchmark simulations, the new phase change model consistently provides reliable results. Finally, we successfully simulate the dynamics of bubbles in superheated liquid subjected to gravity and validate the results with experimental data.

1. Introduction

Boiling flows enable highly efficient heat transfer, which is of ever-growing interest to meet the cooling requirements of numerous high-performance applications (e.g. computer electronics, vehicle electronics, fusion reactor design) [2]. To predict and improve

---

[*] Corresponding author.
E-mail address: lorenz.weber@kit.edu (L. Weber).

https://doi.org/10.1016/j.jcp.2026.114680
Received 14 July 2025; Received in revised form 5 December 2025; Accepted 9 January 2026

boiling heat transfer, researchers strive to delve deeper into the underlying physical phenomena through simulations and experiments. In this context, interface-resolved simulations are a crucial tool [3], as they provide full access to flow quantities and eliminate the need for potentially hazardous or expensive experiments. However, most interface-resolved simulations for boiling flows come at high computational costs, due to resolution requirements [4]. Consequently, complex flows become challenging, or even impossible, to resolve [3], which motivates the development of more efficient frameworks.

A wide range of methods for representing the gas-liquid interface have been applied to boiling flows. These can be categorized into sharp and diffuse interface methods. Sharp-interface methods, a class of methods in which the phases are fully separated by an interface, were first introduced in 1998, when Juric and Tryggvason [5] proposed a front-tracking method for simulating film boiling on a horizontal plate. In the same year, the level-set method was introduced by Son and Dhir [6]. Later, the volume of fluid (VOF) method for simulating film boiling was also successfully utilized by Welch and Wilson [7]. Computationally, sharp-interface methods may result in an increased cost due to interface reinitialization (level-set), interface reconstruction (VOF), or interpolation steps (front-tracking).

In contrast, diffuse interface methods, also known as phase field methods, are characterized by an artificially thickened interface region in which both phases coexist. In terms of computational costs, diffuse interface methods are advantageous, as they only require solving a single advection equation [8]. The phase field method was first used for one-dimensional phase change simulations in 2001 [9]. Around this time, Sun and Beckermann [10] derived a diffuse interface model, which was also used to simulate interfacial velocity and pressure jumps. Later, more complex two-dimensional simulations were conducted for boiling on a heated surface [11] and for studying bubble growth with contact angle dynamics [12]. For simulating three-dimensional flows, another phase-field-like approach was proposed in [13,14].

Within the class of diffuse interface methods, most approaches are based on either the Allen-Chan (AC) equation [15] or the Cahn-Hillard (CH) equation [16]. The CH-based approach, often chosen for its mass-conserving property, has seen several applications for boiling flows [17–19]. However, the CH approach can be numerically problematic as it entails solving a fourth-order partial differential equation (PDE). In this regard, the AC-based approach is preferable as it only contains second derivatives in space, but it does not conserve mass. For that reason, the AC equation was modified [20,21] to derive a mass-conservative second-order phase field equation, which has been successfully used for simulating a variety of boiling flow problems [22,23].

The conservative second-order phase field equation was further investigated by Mirjalili et al. [24] and Jain et al. [25] for application to incompressible and compressible flows, respectively. It was proven that specific parameter choices allow for discretizing spatial derivatives using exclusively central differences. These advancements have significantly improved the simulation of complex flows, as scalability is enhanced, and resolving turbulence benefits from the non-dissipative nature of the discretization [26]. In the following, we refer to this subclass as the *conservative diffuse interface* (CDI) method [24]. Further improvements [1] led to (i) relaxing the parameter constraints and (ii) enhancing the model's accuracy, which is referred to as the *accurate conservative diffuse interface* (ACDI) method [1]. This method has been used for a variety of phase change phenomena, including icing [27] and concentration-driven evaporation [28,29]. Few researchers have explored the capabilities of these recent advancements in simulating boiling phenomena. To the best of our knowledge, only nucleate boiling [30], and bubble growth [31], in two dimensions have been studied. Therefore, a versatile framework for complex boiling flow using the ACDI equation is needed. Such a framework should consider the aspects: (i) the coupling with the Navier-Stokes equation, (ii) the modelling of the surface tension, (iii) the efficient solution of the pressure, and (iv) the use of a suitable phase change model. The reasons these aspects should be discussed are presented below.

Coupling a diffuse interface method to the Navier-Stokes equation by deriving consistent equations to describe a multiphase mixture is still an open topic [32]. In the CDI community, a mixture formulation, as in [33], is commonly selected, however, the rationale behind this choice is rarely discussed. In contrast, many [32,34–40] have discussed mathematical variations for consistently describing CH-type mixtures. The selection of a mixture formulation is generally based on the model assumptions and numerical design choices. For example, [38] indicate that some mixture descriptions might not be consistent for arbitrary *differential fluxes* (or *regularization fluxes* in the CDI community). However, a comprehensive discussion in the context of boiling flow simulations is missing.

Modelling of the surface tension in the momentum balance equation is an essential aspect when deriving mixture equations. Surface tension models are commonly associated with spurious flow structures [41], which can negatively impact the results or even lead to instabilities. For CDI equations, two classes of surface tension models are commonly used [42]: (i) the continuum surface force (CSF) [43], and (ii) the energy-based (or chemical potential-based) model [44,45]. In the absence of phase change, the energy-based model was found to have better accuracy and convergence properties in terms of spurious currents [42,46,47]. With phase change, an amplification of spurious currents is generally observed [48]. Researchers [49] have also reported interactions with the thermal boundary layer and the phase change model. To reduce these currents, several variations of both the CSF model, e.g. [50,51], and the energy-based model, e.g. [52,53], exist. Regarding boiling flow simulations, a comprehensive suitability assessment for the CDI approach has not been conducted yet.

As concerns the numerical solution of the CDI mixture equations, the pressure Poisson equation (PPE) is commonly associated with the highest computational cost [8] at each time step. For high-density ratios, iterative solvers exhibit a reduced convergence rate [54], which motivates the search for fast alternatives. A common method is to reformulate the PPE with constant coefficients as in [55–57], allowing for the use of Fast Fourier transformations (FFTs) [58]. In recent years, many studies have successfully utilized this approach for simulating complex multiphase flows, e.g. [29,59–61]. Despite its advantages, this approach was shown to cause pressure oscillations at interfaces subjected to pressure jumps due to surface tension [62] or due to the recoil pressure jump in phase change scenarios [61]. Although the latter two studies present significant improvements, complex flow cases can still suffer from oscillations in boiling flow simulations [61].

The selection of the phase change model can also influence the framework's scalability potential for parallel computing. Generally, two types of models exist: (i) *heat conduction models* [49,63,64] that use the interfacial jump conditions to calculate the mass transfer rate, and (ii) *kinetic models* [65–67] that use the local superheat. The first model class is computationally more expensive, as it requires temperature gradient calculations. Nevertheless, *heat conduction models* are widely used in phase field methods [19,23,31], as they are free from empirical coefficients [3]. Although traditional *kinetic models* come with problem-dependent constants [65–67], simpler formulations exist [68,69] that have not been adopted by the CDI community, thus far.

In this study, we work to address the research gaps identified in the preceding paragraphs. For this purpose, we develop a sophisticated diffuse interface framework for simulating boiling flows. The methods of this framework were chosen to support large-scale, massively parallel simulations *by design*. The aims of this study are (i) to derive a consistent mixture formulation for the ACDI equation for application to boiling flows, (ii) to develop an improved FFT-based pressure solution scheme, (iii) to find an efficient yet versatile phase change formulation for the chosen phase field model, and (iv) to incorporate these features in a robust time-stepping algorithm.

The remainder of this paper is structured as follows. First, we derive a suitable set of mixture equations in Section 2. For the numerical solution of the equations (Section 3), we present the time-stepping methodology in Section 3.2, which we complete by developing an FFT pressure solution scheme (Section 3.3). In Section 3.4, we modify an existing kinetic phase change model for the ACDI application. The corresponding solution algorithm is presented in Section 3.6. After discussing the methods, we present a comprehensive testing campaign to highlight the achieved improvements (Section 4). Finally, in Section 5, we summarize the main findings and compare them with the aims of the study.

2. Governing equations

This paper considers an immiscible two-phase flow, where the phases are separated by a diffuse interface, in which both phases coexist. Using the volume fraction α_i and the density of the ith phase ρ_i, we define partial densities $\tilde{\rho}_i$ and mixture density ρ as

$$\tilde{\rho}_i = \alpha_i \rho_i \quad \text{and} \quad \rho = \sum \tilde{\rho}_i, \tag{1}$$

where $\sum \alpha_i = 1$. Both phases are assumed incompressible; thus, $\nabla \cdot \vec{u}_i = 0$. A volume averaging approach is chosen for the mixture velocity. Consequently, $\vec{u}$ denotes the velocity resulting from volume averaging

$$\vec{u} = \sum_i \alpha_i \vec{u}_i. \tag{2}$$

A favourable feature of volume averaging is that $\vec{u}$ is divergence-free in the absence of phase change; therefore, it is used by many authors in the phase field community [36,37,70]. In general, $\vec{u}_1 \neq \vec{u}_2$, which is why we introduce the differential velocity $\Delta\vec{u} = \vec{u}_1 - \vec{u}_2$, following the notation of [13]. This allows the phase velocities to be expressed in terms of the mixture quantities

$$\vec{u}_1 = \vec{u} + \alpha_2 \Delta\vec{u} \quad \text{and} \quad \vec{u}_2 = \vec{u} - \alpha_1 \Delta\vec{u}. \tag{3}$$

Using these mixture quantities, we derive the governing equations by (i) starting from the conservation equation for each phase, (ii) deriving the mixture equation, and (iii) making appropriate modelling choices for the considered problem. In the following subsections, this procedure is applied to the mass and phase field transport equations, the momentum transport equation, and the energy transport equation.

2.1. Mass and phase field transport

The mass conservation in each phase reads as

$$\frac{\partial \tilde{\rho}_i}{\partial t} + \nabla \cdot (\tilde{\rho}_i \vec{u}_i) = \mathcal{M}_i, \tag{4}$$

where $\rho_i = const.$, and the source $\mathcal{M}_i$ denotes the interfacial mass transfer density of the ith phase with $\sum_i \mathcal{M}_i = 0$.

We introduce the phase field variable ϕ being equal to the volume fraction α_1, which corresponds to the gaseous phase for the remainder of this paper. Consequently, $\alpha_1 = \phi$ and $\alpha_2 = (1-\phi)$. Dividing (4) by ρ_i for $i = 1$, we arrive at the evolution equation for the phase field variable ϕ,

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\phi \vec{u}_1) = \frac{\mathcal{M}_1}{\rho_1}. \tag{5}$$

The subscript 1 of the mass transfer term $\mathcal{M}_1$ is dropped for the remainder of this paper for convenience, thus $\mathcal{M}_1 = \mathcal{M}$ and $\mathcal{M}_2 = -\mathcal{M}$. Applying the mixture relations from Eqs. (3) to (5) results in the evolution equation for the phase field variable ϕ expressed in terms of mixture quantities,

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\phi \vec{u}) = \frac{\mathcal{M}}{\rho_1} - \nabla \cdot (\phi(1-\phi)\Delta\vec{u}) = \frac{\mathcal{M}}{\rho_1} + \nabla \cdot \vec{\mathcal{R}}. \tag{6}$$

In Eq. (6), we introduce the term $\vec{\mathcal{R}} = -\phi(1-\phi)\Delta\vec{u}$. The differential velocity $\Delta\vec{u}$ is typically associated with a sharpening or diffusion of the interface region. This difference is not necessarily connected to a physical interpretation. For non-phase-change problems, this is often explained with counter-acting the numerical diffusion (i.e. resharpening). In the presence of phase change, an analogy may be drawn to the interface normal velocity jump. In general, $\Delta\vec{u}$ is not accessible and a separate treatment is required for $\vec{\mathcal{R}}$, which we discuss at the end of this section. Summing the transport equations for both α_i, we obtain the relation for the divergence of $\vec{u}$

$$\sum_i \left[\frac{\partial \alpha_i}{\partial t} + \nabla \cdot (\alpha_i \vec{u}_i)\right] = \nabla \cdot \vec{u} = \mathcal{M}\left(\frac{1}{\rho_1} - \frac{1}{\rho_2}\right). \tag{7}$$

Similar to Eq. (7), the mass transport is the sum of Eq. (4) for both phases

$$\sum_i \left[\frac{\partial \tilde{\rho}_i}{\partial t} + \nabla \cdot (\tilde{\rho}_i \vec{u}_i)\right] = \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = (\rho_1 - \rho_2)\nabla \cdot \vec{\mathcal{R}} = \nabla \cdot \vec{\mathcal{F}}. \tag{8}$$

Here, $(\rho_1 - \rho_2)\vec{\mathcal{R}}$ is denoted $\vec{\mathcal{F}}$, which is the differential mass flux between the two phases. As such, Eqs. (5) and (8) have the same structure as those found in the conservative diffuse interface literature, e.g. [25,33], but carry an additional phase change term $\mathcal{M}$. To close the system, we need an expression for the term $\vec{\mathcal{R}}$ since the differential velocity is, in general, unknown. Using the accurate conservative diffuse interface method [1], we set

$$\vec{\mathcal{R}} = \Gamma\left[\epsilon \nabla \phi - \frac{1}{4}\left(1 - \tanh^2\left(\frac{\Psi}{2\epsilon}\right)\right)\vec{n}\right]. \tag{9}$$

For situations involving phase change, the closure of $\vec{\mathcal{R}}$ could be further augmented by calculating the interface velocity jump due to mass transfer. This velocity jump can be approximated using an interface normal integral of $\mathcal{M}$. For computational efficiency, calculating this integral was not considered. A satisfactory closure was obtained using Eq. (9) as shown through the testing campaign in Section 4. For the remainder of this paper we will refer to $\vec{\mathcal{R}}$ as the regularization term [1]. In Eq. (9), Γ is the regularization speed, ϵ a measure for the interface thickness, $\vec{n}$ the interface normal vector, and Ψ the level-set function. Typically Γ is related to the largest velocity in the domain $u_{\max}$, i.e. $\Gamma = \Gamma^* |u_{\max}|$, where Γ^* is dimensionless factor. Further, ϵ is related to the grid spacing Δx and a dimensionless factor ϵ^* so that $\epsilon = \epsilon^* \Delta x$. Special choices for the values of those parameters were made by [1] to enable the discretization with central differencing schemes. In Section 3.4, we present an extension to boiling flows. Finally, the level-set function Ψ is calculated [1] from

$$\Psi = \epsilon \ln\left(\frac{\phi + \varepsilon}{1 - \phi + \varepsilon}\right). \tag{10}$$

The level-set function is also used to calculate the interface normal vector $\vec{n} = \nabla\Psi / |\nabla\Psi|$. For numerical stability reasons, a small number ε is added to Eq. (10). Note that ε can vary depending on the implementation. However, in this study, robust calculations of Ψ were obtained by using values in the order of 10^{-16}. This completes the derivation of the phase field evolution equation. In the next step, the focus is turned to the momentum balance of the mixture.

2.2. *Momentum transport*

To derive the mixture momentum transport equation, we start at the transport equations for each phase separately. This strategy was also followed in [10,13,36,39] and, more recently, in [32,40]. For the ith phase, the linear momentum balance equation is given by

$$\frac{\partial \alpha_i \rho_i \vec{u}_i}{\partial t} + \nabla \cdot (\alpha_i \rho_i \vec{u}_i \otimes \vec{u}_i) = -\alpha_i \nabla p + \nabla \cdot \tau_i + \alpha_i \rho_i \vec{g} + \vec{I}_i + \vec{E}_i, \tag{11}$$

where τ_i is the viscous stress tensor $\tau_i = \alpha_i \mu_i (\nabla \vec{u}_i + \nabla \vec{u}_i^T)$, μ_i the constant dynamic viscosity, $\vec{g}$ the earth's gravitational force, and $\vec{I}_i$ represents all phase interaction forces. Note that $\sum_i \vec{I}_i = \vec{0}$. The term $\vec{E}_i$ represents the contributions of the surface energy in the context of Allen-Cahn or Cahn-Hillard models [32]. More generally, $\vec{E}_i$ represents the effect of surface tension (see [10]). Further we assume that both phases share locally the same pressure p. The sum of the left-hand side (LHS) of Eq. (11) over the phase index i may be expressed in terms of the mixture quantities ρ, $\vec{u}$, $\vec{\mathcal{R}}$, and $\vec{\mathcal{F}}$ as

$$\begin{aligned}\sum_i \left[\frac{\partial \alpha_i \rho_i \vec{u}_i}{\partial t} + \nabla \cdot (\alpha_i \rho_i \vec{u}_i \otimes \vec{u}_i)\right] =& \frac{\partial \rho \vec{u}}{\partial t} + \nabla \cdot \left(\rho \vec{u} \otimes \vec{u}\right) - \nabla \cdot \left(\vec{\mathcal{F}} \otimes \vec{u}\right) \\ & - \frac{\partial \vec{\mathcal{F}}}{\partial t} - \nabla \cdot \left(\vec{u} \otimes \vec{\mathcal{F}}\right) + \nabla \cdot \left(\left(\frac{\rho_1}{\phi} + \frac{\rho_2}{1 - \phi}\right)\vec{\mathcal{R}} \otimes \vec{\mathcal{R}}\right) \\ =& \frac{\partial \rho \vec{u}}{\partial t} + \nabla \cdot \left(\left(\rho \vec{u} - \vec{\mathcal{F}}\right) \otimes \vec{u}\right) + \vec{C}.\end{aligned} \tag{12}$$

Eq. (12) is the complete LHS that describes the coupling of the momentum equation with the diffuse interface model. The second line of Eq. (12) represents the terms commonly neglected when a volume-averaged mixture velocity is used. For conciseness, these terms are replaced by $\vec{C}$ in the third line. Without the terms summarized by $\vec{C}$, Eq. (12) is identical to the LHS of the momentum equation used by the CDI community [25,33]. Note that $\vec{C}$ is solely active within the diffuse interface region and vanishes for the pure gas and liquid phases. These terms originate from the diffuse interface model and represent a correction accounting for the additional mass flux due to the interface regularization $\vec{\mathcal{R}}$ (see Eq. (8)).

Neglecting the terms $\partial\vec{\mathcal{F}}/\partial t + \nabla\cdot\left(\vec{u}\otimes\vec{\mathcal{F}}\right)$ (Eq. (12)) is commonly justified by the assumption that the "relative momenta [...] are negligible when computed relative to the gross motion of the fluid" [34]. Subsequently, this assumption was followed in numerous studies, e.g. [37,39,71]. In other works [25,33,52], this simplification is utilized implicitly. As shown in the literature [33,52], this assumption leads to another possible (kinetic energy conservative) coupling of phase field models with the Navier Stokes Equations (NSE). Nevertheless, formally, the term $\partial\vec{\mathcal{F}}/\partial t + \nabla\cdot\left(\vec{u}\otimes\vec{\mathcal{F}}\right)$ is not zero [32,40] and recent numerical experiments [70] show that keeping the terms led to accurate results in two-phase flows.

The last term of the second line of Eq. (12) received different treatment in the literature. Though often neglected, it is worth noting that in [13] the term is recognized as part of the pressure. In [39] this term is incorporated into the stress tensor. For the present study, this term is retained in $\vec{C}$ and its relevance will be evaluated in the context of boiling flows in the subsequent sections.

When using a density averaged (barycentric) mixture velocity $\vec{v} = \sum_i(\tilde{\rho}_i\vec{u}_i)\rho^{-1}$, we can reformulate Eq. (12) in a more concise form through the relation $\rho\vec{v} = \rho\vec{u} - \vec{\mathcal{F}}$ as follows

$$\begin{aligned}&\frac{\partial\rho\vec{u}}{\partial t} + \nabla\cdot\left(\rho\vec{u}\otimes\vec{u}\right) - \nabla\cdot\left(\vec{\mathcal{F}}\otimes\vec{u}\right) - \frac{\partial\vec{\mathcal{F}}}{\partial t} - \nabla\cdot\left(\vec{u}\otimes\vec{\mathcal{F}}\right) + \nabla\cdot\left(\left(\frac{\rho_1}{\phi} + \frac{\rho_2}{1-\phi}\right)\vec{\mathcal{R}}\otimes\vec{\mathcal{R}}\right)\\ &\quad= \frac{\partial\rho\vec{v}}{\partial t} + \nabla\cdot\left(\rho\vec{v}\otimes\vec{v}\right) + \nabla\cdot\left(\left(\frac{\rho_1\rho_2}{\rho\phi(1-\phi)}\right)\vec{\mathcal{R}}\otimes\vec{\mathcal{R}}\right).\end{aligned} \tag{13}$$

Eq. (13) is equivalent to the momentum balance found in [10]. A drawback of using $\vec{v}$ is that the mixture velocity is not divergence-free when $\mathcal{M} = 0$, because $\nabla\cdot\vec{v} = \nabla\cdot\vec{u} - \nabla\cdot(\rho^{-1}\vec{\mathcal{F}})$. However, formulating the momentum balance LHS in terms of a volume-averaged velocity $\vec{u}$ or a density-averaged velocity $\vec{v}$ leads to equivalent mixture formulations [70]. For conciseness, we will occasionally use $\vec{v}$ in this study, bearing in mind that $\vec{u}$ is the primary solution quantity.

For the mixture formulation of the right-hand side (RHS) of Eq. (11), we will discuss each term separately. Evidently, $\sum_i(-\alpha_i\nabla p + \alpha_i\rho_i\vec{g} + \vec{I}_i) = -\nabla p + \rho\vec{g}$, but modelling $\sum_i(\tau_i)$ in terms of mixture quantities is not trivial. Although complete formulations [32,36] or approximations [10] are available, we choose to use

$$\sum_i(\tau_i) \approx \tau = \mu(\nabla\vec{u} + \nabla\vec{u}^T - 2\nabla\cdot\vec{u}\mathbb{I}). \tag{14}$$

Here, $\mathbb{I}$ is the identity tensor and $\mu = \sum_i \alpha_i\mu_i$. Note that $2\nabla\cdot\vec{u}\mathbb{I}$ removes the spurious pressure contribution of the normal viscous stress component [5,72].

Lastly, the term $\sum_i \vec{E}_i$ of Eq. (11) represents the surface tension effect. Since the conservative diffuse interface method does not obey a known energy functional like the Cahn-Hillard or Allen-Cahn models [42], we replace $\sum_i \vec{E}_i$ with a general forcing vector $\vec{f}_{ST}$ to model the effect of surface tension. Selection of a model for $\vec{f}_{ST}$ is sensitive, as spurious currents can significantly impact the overall result quality [48]. In Section 4, we will compare different surface tension models and provide best practice guidelines.

The final momentum balance equation in terms of mixture quantities reads

$$\frac{\partial\rho\vec{u}}{\partial t} + \nabla\cdot\left(\left(\rho\vec{u} - \vec{\mathcal{F}}\right)\otimes\vec{u}\right) + \vec{C} = -\nabla p + \nabla\cdot\tau + \rho\vec{g} + \vec{f}_{ST}, \tag{15}$$

where

$$\vec{C} = -\frac{\partial\vec{\mathcal{F}}}{\partial t} - \nabla\cdot\left(\vec{u}\otimes\vec{\mathcal{F}}\right) + \nabla\cdot\left(\left(\frac{\rho_1}{\phi} + \frac{\rho_2}{1-\phi}\right)\vec{\mathcal{R}}\otimes\vec{\mathcal{R}}\right). \tag{16}$$

2.3. Energy transport

To derive the energy transport, we start at the partial enthalpy density $\tilde{\rho}_i h_i$ as the transported, conserved quantity. Here, h_i is the enthalpy of the ith phase, which we define as $h_i = c_{p,i}T_{abs} + h_{0,i}$. We utilize the absolute temperature T_{abs} and the heat capacity $c_{p,i}$ (at constant pressure). As a reference enthalpy, the enthalpy of formation $h_{0,i}$ at 0 K is used. Further we assume $c_{p,i}$ to be constant. The transport equation for $\tilde{\rho}_i h_i$ is then [13]

$$\frac{\partial\tilde{\rho}_i h_i}{\partial t} + \nabla\cdot(\tilde{\rho}_i h_i\vec{u}_i) = \nabla\cdot(\alpha_i\lambda_i\nabla T_{abs}). \tag{17}$$

Summing Eq. (17) for all i and expressing the result in mixture quantities results in

$$\frac{\partial\rho h}{\partial t} + \nabla\cdot(\rho h\vec{u}) = \nabla\cdot(\lambda\nabla T_{abs}) + \nabla\cdot\left[(\rho_1 h_1 - \rho_2 h_2)\vec{\mathcal{R}}\right]. \tag{18}$$

Here, the mixture enthalpy density is $\rho h = \sum_i \tilde{\rho}_i h_i$ and the mixture heat conductivity $\lambda = \sum_i \alpha_i\lambda_i$ with λ_i assumed constant.

Introducing a relative temperature $T = T_{abs} - T_{sat}$ based on the (constant) saturation temperature T_{sat} and the latent heat at saturation temperature $L_{sat} = h_1(T_{sat}) - h_2(T_{sat})$, we can rewrite (18) as

$$\frac{\partial\rho c_p T}{\partial t} + \nabla\cdot(\rho c_p T\vec{u}) = \nabla\cdot(\lambda\nabla T) - \mathcal{M}L_{sat} + (\rho_1 c_{p,1} - \rho_2 c_{p,2})\nabla\cdot(T\vec{\mathcal{R}}). \tag{19}$$

Taking $\rho c_p = \sum_i \tilde{\rho}_i c_{p,i}$ out of the derivatives leads us to the following transport equation for the temperature

$$\rho c_p\left(\frac{\partial T}{\partial t} + \vec{u}\cdot\nabla T\right) = \nabla\cdot(\lambda\nabla T) - L\mathcal{M} + \Delta\rho c_p\vec{\mathcal{R}}\cdot\nabla T. \tag{20}$$

In Eq. (20), $L = f(T) = L_{sat} + (c_{p,1} - c_{p,2})T$ is the temperature-dependent latent heat and $\Delta\rho c_p = \rho_1 c_{p,1} - \rho_2 c_{p,2}$. The term $\vec{\mathcal{R}}$ is kept in Eq. (20) for consistency. Nevertheless various phase field studies do not seem to require this contribution [19,22,23,30]. It is worth mentioning that in some works [13,14,73,74], the contribution of $\vec{\mathcal{R}}$ is included in the temperature transport. However, in the latter works, a different closure procedure was used.

Mathematically, Eqs. (18)–(20) are equivalent, but they differ when implemented numerically. In this paragraph, we discuss the choice of the equation in detail. Clearly, from an energy conservation point of view, Eqs. (18) or (19) are preferred. For the discretization of the advection term, a different scheme has to be selected than that for the phase field transport Eq. (6). In general the temperature field may contain steep gradients or jumps (e.g. at the interface), which limit the use of central differencing schemes for the spatial discretization. As such, specialized flux limiter or WENO schemes [75] are required.

Applying different spatial discretization schemes for transporting physical properties can lead to inconsistencies. Here, the transport of the physical property ρc_p is implied by both Eqs. (18) and (19), although the transport of this quantity is already defined by Eq. (6). This is inconsistent, as the same physical property would be transported twice, with two different numerical schemes. We overcome this problem by using Eq. (20) to separate the temperature transport (Eq. (20)) from the transport of physical properties (Eq. (6)). Choosing Eq. (20) for simulating the thermal energy transport can lead to conservation errors due to its non-conservative form. This error is discussed further in Section 4.4. Note that, apart from Eq. (20), all numerical implementations use central differencing methods for discretizing spatial derivatives.

The final energy balance equation has the form

$$\rho c_p \left(\frac{\partial T}{\partial t} + \vec{u} \cdot \nabla T \right) = \nabla \cdot (\lambda \nabla T) + \nabla \cdot \vec{j} - L\mathcal{M} + \Delta\rho c_p \vec{\mathcal{R}} \cdot \nabla T, \tag{21}$$

with

$$\vec{j} = -\vec{n}\mathcal{M}L\epsilon \left(\frac{\lambda_1}{\lambda_2} - 1 \right). \tag{22}$$

In Eq. (21), the term $\vec{j}$ was artificially added. This term is called the *anti-trapping current* [76]. In the context of phase field modelling, $\vec{j}$ is used to improve convergence to the sharp interface limit [13,76–78]. For the present application, we used a simplified version of the anti-trapping current proposed in [13]. This simplification is based on the assumption that the heat transport from the liquid to the interface is high compared to the heat coming from the vapour side. The reader is referred to the work in Ref. [13] for further details.

3. Numerical implementation

The following section presents a discrete scheme for solving the governing equations. First, we briefly discuss the spatial operators required to discretize the equations in space. The remainder of the section is primarily devoted to elaborating on the temporal integration. Finally, the numerical implementation of the phase change and the associated impact on the time-stepping is discussed.

3.1. Spatial discretization

We now discuss the discretization in space. For the discrete representation of ϕ, $\vec{u}$, p, and T, we decompose the computational domain into uniform cubic cells of volume Δx^3. In each cell, the solution variables are located in a staggered arrangement, such that scalars (i.e. ϕ, p, and T) are defined at the cell centres and vectors (e.g. $\vec{u}$) at the cell faces. Fig. 1 illustrates this situation for two dimensions; the extension to three dimensions is straightforward.

The governing equations derived in Section 2 allow the use of central differencing schemes for almost all spatial derivatives. As described in Section 2.3, the exception is the transport equation for thermal energy, where a 5th-order WENO scheme [75] is employed. Therefore, a single neighbouring grid point is required for the discrete operators using second-order central schemes and two points for those using the WENO scheme (i.e. for T). For inter-process communication in parallel computing, this reduces data exchange between processes, and consequently, an improved scalability.

3.2. Temporal discretization

In this section, we present the numerical integration of the variables ϕ, $\vec{u}$,p,T in time. First, we use a second-order explicit Adams-Bashforth scheme to compute intermediate values between time steps n and $n+1$ for the fields ϕ and T. This state is denoted with the superscript * and obtained by solving

$$\phi^* = \phi^n + \Delta t^{n+1}\left(f_{t,1}\mathcal{B}_\phi^n + f_{t,2}\mathcal{B}_\phi^{n-1}\right), \tag{23}$$

$$T^* = T^n + \Delta t^{n+1}\left(f_{t,1}\mathcal{B}_T^n + f_{t,2}\mathcal{B}_T^{n-1}\right). \tag{24}$$

According to the Adams-Bashforth scheme, the coefficients $f_{t,1}$,$f_{t,2}$ are

$$f_{t,1} = 1 + \frac{\Delta t^{n+1}}{2\Delta t^n} \quad \text{and} \quad f_{t,2} = -\frac{\Delta t^{n+1}}{2\Delta t^n}. \tag{25}$$

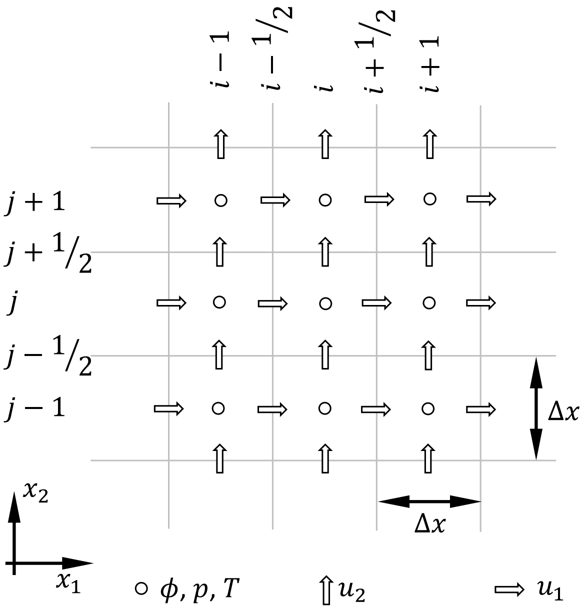


Fig. 1. Schematic visualization for deployed finite volume discretization in two dimensions.

Furthermore, a term $\mathcal{B}^n_c$ denotes the rate of change for a quantity c at time step n, i.e., $(\partial c/\partial t)^n = \mathcal{B}^n_c$. Following this procedure for ϕ and T and rearranging Eqs. (6) and (21), we write $\mathcal{B}^n_\phi$ and $\mathcal{B}^n_T$ as

$$\mathcal{B}^n_\phi = -\nabla\cdot(\phi^n\vec{u}^n) + \nabla\cdot\vec{\mathcal{R}}^n \tag{26}$$

$$\mathcal{B}^n_T = -\vec{u}^n\cdot\nabla T^n + \frac{1}{(\rho c_p)^*}\left(\nabla\cdot(\lambda^n\nabla T^n) + \nabla\cdot\vec{j}^n + \Delta\rho c_p\vec{\mathcal{R}}^n\cdot\nabla T^n\right). \tag{27}$$

In Eqs. (26) and (27), the contribution of the phase change term $\mathcal{M}$ was ignored when computing the intermediate stage *. In Section 3.4 we discuss calculating $\mathcal{M}$ as a function of the intermediate variable values. This step enhances the numerical stability. After calculating $\mathcal{M}^{n+1}$, the final values ϕ^{n+1} and T^{n+1} are obtained as

$$\phi^{n+1} = \phi^* + \Delta t^{n+1}\frac{\mathcal{M}^{n+1}}{\rho_1}, \quad \text{and} \quad T^{n+1} = T^* + \Delta T^{n+1}_{PC}. \tag{28}$$

Here ΔT^{n+1}_{PC} is the temperature increment due to phase change (see Section 3.4).

The velocity field is advanced in time using Eq. (15). Starting with the intermediate momentum $(\rho\vec{v})^*$ as

$$(\rho\vec{v})^* = (\rho\vec{v})^n + \Delta t^{n+1}\left[\left(f_{t,1}\vec{\mathcal{B}}^n_{\rho\vec{v}} + f_{t,2}\vec{\mathcal{B}}^{n-1}_{\rho\vec{v}}\right) + \rho^{n+1}\vec{g} + \vec{f}^{n+1}_{ST} - \nabla p^n\right]. \tag{29}$$

The surface tension $\vec{f}_{ST}$ and the effect of gravity through $\vec{g}$ are excluded from the Adams-Bashforth scheme since their values for time step $n+1$ are already known. Additionally, we decompose the new pressure $p^{n+1} = p^n + \psi^{n+1}$ and add the contribution of the old pressure p^n in Eq. (29) to the intermediate momentum. We recall that $(\rho\vec{v})^n = \rho^n\vec{u}^n - \vec{\mathcal{F}}^n$. The term $\vec{\mathcal{B}}^n_{\rho\vec{v}}$, containing the advection and viscus effects, is defined as

$$\vec{\mathcal{B}}^n_{\rho\vec{v}} = -\nabla\cdot\left(\rho^n\vec{v}^n\otimes\vec{v}^n\right) - \nabla\cdot\left(\left(\frac{\rho_1\rho_2}{\rho^n\phi^n(1-\phi^n)}\right)\vec{\mathcal{R}}^n\otimes\vec{\mathcal{R}}^n\right) + \nabla\cdot\tau^n. \tag{30}$$

3.3. *Pressure solution*

This section presents calculating $\vec{u}^{n+1}$ and the pressure increment $\psi^{n+1} = p^{n+1} - p^n$ from the intermediate momentum $(\rho\vec{v})^*$. For simulations using the CDI method, solving the PPE is the most expensive step in terms of computational cost [8,79]. Consequently, we devote special attention to this solution procedure. The intuitive method for obtaining ψ^{n+1} and $\vec{u}^{n+1}$ may be to use a pressure correction step [80]

$$\nabla\cdot\left(\frac{1}{\rho^{n+1}}\nabla\psi^{n+1}\right) = \frac{1}{\Delta t^{n+1}}\left[\nabla\cdot\vec{v}^* - \mathcal{M}^{n+1}\left(\frac{1}{\rho_1} - \frac{1}{\rho_2}\right) + \nabla\cdot\left(\frac{\vec{\mathcal{F}}^{n+1}}{\rho^{n+1}}\right)\right]. \tag{31}$$

Using $(\rho\vec{v})^* = \rho^{n+1}\vec{v}^*$, one sets

$$\vec{u}^{n+1} = \vec{v}^* - \frac{\Delta t^{n+1}}{\rho^{n+1}}\nabla\psi^{n+1} + \frac{\vec{\mathcal{F}}^{n+1}}{\rho^{n+1}} \quad , \quad p^{n+1} = p^n + \psi^{n+1}. \tag{32}$$

Solving Eq. (31) with an iterative scheme is expensive for large density ratios, as this requires specialized preconditioners [81] and results in slow convergence rates. FFT-based solvers can offer a substantial speed-up potential and are frequently utilized for multiphase flows [29,59,61]. As such, we explore the use of FFT-based solvers to reduce the computational cost.

Using an FFT-solver requires rewriting $\nabla\cdot({\rho^{n+1}}^{-1}\nabla\psi^{n+1})$ to obtain a constant-coefficient matrix as proposed in [56,57]. In the following, we will: (i) repeat the method proposed in [57] to outline the shortcomings for boiling applications, and (ii) propose an alternative pressure solution scheme for boiling flows. For details on the FFT usage, we refer the readers to the work in [58].

3.3.1. Density splitting and pressure extrapolation

This section reports the method employed in [57]. This is included to motivate seeking an alternative formulation for boiling flow applications. The main idea is based on the approximation [56]

$$\frac{1}{\rho^{n+1}}\nabla\psi^{n+1} = \frac{1}{\rho_0}\nabla\psi^{n+1} + \left(\frac{1}{\rho^{n+1}} - \frac{1}{\rho_0}\right)\nabla\psi^{n+1} \approx \frac{1}{\rho_0}\nabla\psi^{n+1} + \left(\frac{1}{\rho^{n+1}} - \frac{1}{\rho_0}\right) f_{ex}\nabla\psi^{n}, \tag{33}$$

where $\rho_0 = \min(\rho_1, \rho_2) = constant$ and the extrapolation coefficient $f_{ex} = \Delta t^{n+1}/\Delta t^n$. As such, the density is split into a constant and a variable part (first step). For the approximation of the variable part, ψ^{n+1} is replaced with the extrapolated pressure increment $f_{ex}\psi^n$ of the previous time step (second step). Furthermore, in line with [60] and [57], we find the velocity and pressure field at the next time step through

$$\vec{v}^{**} = \vec{v}^{*} - \Delta t^{n+1}\left[\left(\frac{1}{\rho^{n+1}} - \frac{1}{\rho_0}\right) f_{ex}\nabla\psi^{n}\right], \tag{34}$$

$$\nabla^2\psi^{n+1} = \frac{\rho_0}{\Delta t^{n+1}}\left[\nabla\cdot\vec{v}^{**} - \mathcal{M}^{n+1}\left(\frac{1}{\rho_1} - \frac{1}{\rho_2}\right) + \nabla\cdot\left(\frac{\vec{\mathcal{F}}^{n+1}}{\rho^{n+1}}\right)\right], \tag{35}$$

$$\vec{u}^{n+1} = \vec{v}^{**} - \frac{\Delta t^{n+1}}{\rho_0}\nabla\psi^{n+1} + \frac{\vec{\mathcal{F}}^{n+1}}{\rho^{n+1}}, \quad \text{and} \quad p^{n+1} = p^n + \psi^{n+1}. \tag{36}$$

Here, $\vec{v}^{**}$ represents a second intermediate velocity field that contains the extrapolated pressure contribution of the variable density part. While the above procedure has gained popularity, it was shown in [62] that using the approximation in Eq. (33) is problematic for interfacial pressure jumps. This is the case in the presence of surface tension in combination with a density jump at a phase interface, where spurious pressure oscillations were reported [62]. In that context a specialized scheme for treating forcing terms on the RHS of the momentum equation was proposed in order to remove oscillations.

In boiling flow scenarios, interfacial pressure jumps are caused by the recoil pressure, which generates similar oscillations [61]. In [61], the same scheme is used for additional forcing terms to correct the recoil pressure [82]. This led to a significant reduction of oscillations, although some oscillations remained during the simulation start-up phase.

Moreover, the additional requirements on the CFL number, reported in [62], increase the computational cost when using Eqs. (34)–(36). Acknowledging the recent advancements, we propose an alternative pressure solution procedure to overcome the aforementioned drawbacks.

3.3.2. Proposed FFT scheme

The key idea of the proposed solution stems from the works of Juric and Tryggvason [5], and Shin and Juric [83]. According to their scheme, the intermediate momentum $(\rho\vec{v})^*$ is not divided by the density, when assembling the RHS of Eq. (31). Adapted to the present work, the scheme used in [5] reads

$$\nabla^2\psi^{n+1} = \frac{1}{\Delta t^{n+1}}\left[\nabla\cdot(\rho\vec{v})^* - \nabla\cdot(\rho\vec{v})^{n+1}\right], \tag{37}$$

$$\vec{u}^{n+1} = \frac{(\rho\vec{v})^* - \Delta t^{n+1}\nabla\psi^{n+1} + \vec{\mathcal{F}}^{n+1}}{\rho^{n+1}}. \tag{38}$$

This procedure avoids the splitting approach from Eq. (33) and naturally returns a constant coefficient matrix suitable for applying FFT solvers. However, using Eq. (37) poses two difficulties: (i) momentum $(\rho\vec{v})^{n+1}$ needs to be approximated, and (ii) divergence of the velocity $\vec{u}^{n+1}$ is not necessarily equal to $\mathcal{M}^{n+1}({\rho_1}^{-1} - {\rho_2}^{-1})$ after the correction step (38).

We address these difficulties of Eq. (37) by deploying three solution stages, each including a matrix inversion of a constant coefficient matrix which can be efficiently solved using FFT.

Stage 1: In the first stage the momentum for the time step t^{n+1} is estimated. This is done through the splitting approach, as in Section 3.3.1. Quantities emerging from this estimation stage are marked by a tilde $\widetilde{\cdot}$. Formally,

$$\vec{v}^{**} = \vec{v}^{*} - \Delta t^{n+1}\left[\left(\frac{1}{\rho^{n+1}} - \frac{1}{\rho_0}\right) f_{ex}\nabla\psi^{n}\right], \tag{39}$$

$$\nabla^2\widetilde{\psi} = \frac{\rho_0}{\Delta t^{n+1}}\left[\nabla\cdot\vec{v}^{**} - \mathcal{M}^{n+1}\left(\frac{1}{\rho_1} - \frac{1}{\rho_2}\right) + \nabla\cdot\left(\frac{\vec{\mathcal{F}}^{n+1}}{\rho^{n+1}}\right)\right], \tag{40}$$

$$\text{and} \quad \widetilde{\rho\vec{v}} = \rho^{n+1}\vec{v}^{**} - \Delta t^{n+1}\frac{\rho^{n+1}}{\rho_0}\nabla\widetilde{\psi}. \tag{41}$$

Stage 2: The momentum estimate $\widetilde{\rho\vec{v}}$ is used as an approximation for the momentum $(\rho\vec{v})^{n+1}$ at the next time step. Modifying Eq. (37) allows solving the second Poisson equation without the splitting approach. Using this *momentum-based* PPE, we calculate the improved pressure increment $\widehat{\psi}$ as a better estimate of the pressure increment ψ^{n+1}. Consequently we solve

$$\nabla^2\widehat{\psi} = \frac{1}{\Delta t^{n+1}}\left[\nabla\cdot(\rho\vec{v})^* - \nabla\cdot\widetilde{\rho\vec{v}}\right]. \tag{42}$$

Stage 3: As mentioned above, the improved pressure increment $\widehat{\psi}$ does not guarantee a divergence-free velocity field, which is why the final step of the pressure solution is required. Analogously to stage 1, using $\widehat{\psi}$ instead of ψ^n for the splitting, we compute p^{n+1} and $\vec{u}^{n+1}$ by solving

$$\vec{v}^{***} = \vec{v}^* - \Delta t^{n+1}\left[\left(\frac{1}{\rho^{n+1}} - \frac{1}{\rho_0}\right)c\nabla\widehat{\psi}\right], \tag{43}$$

$$\nabla^2\psi^{n+1} = \frac{\rho_0}{\Delta t^{n+1}}\left[\nabla\cdot\vec{v}^{***} - \mathcal{M}^{n+1}\left(\frac{1}{\rho_1} - \frac{1}{\rho_2}\right) + \nabla\cdot\left(\frac{\vec{\mathcal{F}}^{n+1}}{\rho^{n+1}}\right)\right], \tag{44}$$

$$\vec{u}^{n+1} = \vec{v}^{***} - \frac{\Delta t^{n+1}}{\rho_0}\nabla\psi^{n+1} + \frac{\vec{\mathcal{F}}^{n+1}}{\rho^{n+1}}, \tag{45}$$

$$\text{and}\quad p^{n+1} = p^n + \psi^{n+1}. \tag{46}$$

Here, $\vec{v}^{***}$ denotes the improved prediction velocity containing the contribution of the variable density with the pressure increment $\widehat{\psi}$. In Eq. (43) we used a constant c to limit the pressure increment $\widehat{\psi}$ based on $\widetilde{\psi}$. This limiter is calculated as follows

$$c = \min\left(1, \frac{|\min(\widetilde{\psi})|}{|\min(\widehat{\psi})|}, \frac{|\max(\widetilde{\psi})|}{|\max(\widehat{\psi})|}\right). \tag{47}$$

This limiter c is necessary to stabilize the method. Note that $\widehat{\psi}$ and $\widetilde{\psi}$ are linked through the momentum estimation $\widetilde{\rho\vec{v}}$. Consequently, Stages 1–3 are only valid if $\widehat{\psi}$ and $\widetilde{\psi}$ are similar, whereas larger deviations will cause the solution to diverge.

This FFT scheme is subsequently referred to as the Momentum-based Pressure treatment for Density Jumps (short FFT-MPDJ) to highlight the feature of using Stage 2. In Section 4, we evaluate the proposed FFT-MPDJ scheme and compare it to the established method in [57,60].

3.4. *Phase change model*

A crucial part of this study is selecting an adequate numerical procedure to model phase change. Large-scale simulations of boiling flows require a phase change model that ensures: (i) numerical efficiency, (ii) scalability, and (iii) robustness. Although *heat conduction models* are generally free from empirical parameters [3], they are computationally more expensive due to their need to calculate temperature gradients across the interface. Thus, we use a simpler *kinetic model* loosely based on [68,69], but modified for the present phase field model.

As shown in Section 3.2, the numerical integration of ϕ and T is split into two steps, which are deployed to improve numerical stability. The temperature increment due to phase change is defined as $\Delta T_{PC}^{n+1} = T^{n+1} - T^*$. Ideally, at a sharp interface, the phase change would happen instantaneously, so that $T^{n+1} = 0$ (i.e. at saturation temperature) and consequently

$$\Delta T_{PC}^{n+1} = (T_{sat} - T_{abs}^*) = -T^*. \tag{48}$$

With ΔT_{PC}^{n+1}, the mass transfer term $\mathcal{M}$ can be easily calculated from Eq. (21) resulting in

$$\mathcal{M}^{n+1} = -\frac{(\rho c_p)^*\Delta T_{PC}^{n+1}}{L(T^*)\Delta t^{n+1}}, \tag{49}$$

which reassembles the formulation in [68] and [69]. However, there are drawbacks of using Eq. (48) to calculate ΔT_{PC}^{n+1} in the context of the present study. Namely:

- the accuracy of the model strongly depends on the time step, as pointed out in [84],
- Eq. (48) is not restricted to the interfacial region, and
- no weighting of ΔT_{PC}^{n+1} accounts for the interfacial jumps in $(\rho c_p)^*$. As indicated by Eq. (49) the same temperature increment results in different mass transfer rates on either side of the interface (due to the multiplication with $(\rho c_p)^*$). This leads to an unbalanced distribution of $\mathcal{M}^{n+1}$.

To address these drawbacks, we use the following modified version of Eq. (48)

$$\Delta T_{PC}^{n+1} = -T^*\theta_t\theta_{pc}\theta^*_{\rho c_p}\theta^*_{\phi}. \tag{50}$$

As seen in Eq. (50), we maintain the structure of Eq. (48) so that ΔT_{PC}^{n+1} is still proportional to T^*. But, we multiply T^* by four non-dimensional factors θ. Note that $\theta^*_{\phi} = \theta_{\phi}(\phi^*)$ and $\theta^*_{\rho c_p} = \theta_{\rho c_p}(\phi^*)$.

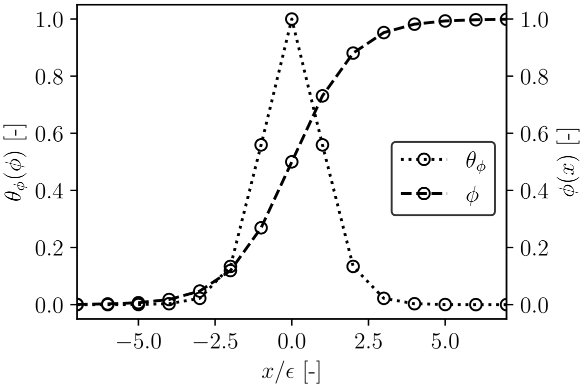


Fig. 2. Visualization of the discretized interface indicator in Eq. (51), used to restrict the mass transfer $\mathcal{M}$ to the interface region. The interface indicator θ_ϕ is shown together with the phase field variable ϕ, as function of the non-dimensional interface distance x/ϵ. Circles represent the values computed at cell-centers, where the cells have a uniform size of Δx.

The factor θ_ϕ restricts the mass transfer to the interface region. We choose θ_ϕ as

$$\theta_\phi = \frac{1}{2}\left(1 + \cos\left(2\pi|\phi - 0.5|\right)\right). \tag{51}$$

The distribution in the discrete case is found in Fig. 2. In theory, the choice of θ_ϕ is arbitrary, provided $\theta_\phi(\phi = 0) = \theta_\phi(\phi = 1) = 0$ and $\theta_\phi(\phi = 0.5) = 1$. However, we use Eq. (51) for its high concentration around $\phi = 0.5$, which we found to be advantageous (see Fig. 2).

With the factor θ_t, the temperature increment ΔT^{n+1}_{PC} is linked to a characteristic phase change timescale t_{pc}, decoupling the phase change model from the chosen time step size Δt^{n+1}. We assume that the phase change is governed by the heat conduction in the liquid phase, thus,

$$t_{pc} = \frac{\Delta x^2}{a_2}, \quad \text{and} \quad \theta_t = \frac{\Delta t^{n+1}}{t_{pc}}, \tag{52}$$

where $a_i = \lambda_i/(\rho_i c_{p,i})$ is the thermal diffusivity.

Accounting for the jump in ρc_p we use $\theta_{\rho c_p}$ defined as

$$\theta_{\rho c_p} = \frac{1}{\frac{\rho_1 c_{p,1}}{\rho_2 c_{p,2}}\alpha_1 + \alpha_2}. \tag{53}$$

As seen in Eq. (53), $\theta_{\rho c_p}$ weights ΔT^{n+1}_{PC} stronger on the vapour side (proportional to the ratio of ρc_p), resulting in a balanced distribution of $\mathcal{M}$. Lastly, θ_{pc} is a model parameter that we found to be independent of (i) the simulation setup and (ii) the physical properties of the fluid. For the remainder of the study, we fix $\theta_{pc} = 0.65$.

Comparing Eqs. (48) and (50) reveals the requirement $\theta_t \theta_{pc} \theta^*_{\rho c_p} \theta^*_\phi < 1$ to avoid overshooting the saturation temperature. This leads to an implicit phase change time step constraint Δt_{pc} given by

$$\Delta t_{pc} = \frac{1}{\theta_{pc}} \frac{\lambda_1}{\lambda_2} \frac{\Delta x^2}{a_1}. \tag{54}$$

Note that Eq. (54) is written in terms of the heat conduction timescale for the vapor $\Delta x^2 a_1^{-1}$, which is usually more restrictive than that of the liquid phase. Therefore, Δt_{pc} will only affect Δt^{n+1} when the ratio $\lambda_1 \lambda_2^{-1}$ is small. In contrast to [69], no iteration loop over the energy transport is necessary when following the strategy described above. For the verification of the phase change model, a series of simulations is performed and presented in Section 4.

3.5. Time step size and adaptive Γ^*

To apply the CDI method to flows involving boiling phenomena, a central difficulty needs to be addressed: the profile of the phase field function ϕ loses its equilibrium hyperbolic tangent shape because of the strong divergence of $\vec{u}$ at the interface. In other words, the source term $\mathcal{M}$ causes the profile of ϕ to thicken or even lose its shape entirely. This occurs when $\mathcal{M} > 0$ and the speed of regularization term $\vec{\mathcal{R}}$ is insufficient. Given the use of central differencing schemes in space, [24,25] suggest that $\epsilon^* = 1$ and $\Gamma^* = 1$, while [1] reports that using Eq. (9) would allow for even lower values of ϵ^*. Let us recall the diffuse interface time step constraint

$$\Delta t_\phi = \min_i \left[\frac{1}{\max\left\{ \left(\frac{6\Gamma\epsilon}{\Delta x^2}\right) - \left(\frac{\partial u_i}{\partial x_i}\right), 0 \right\}} \right] \tag{55}$$

from [25], which we use as a conservative estimate for the time step constraint. However, Δt_{ϕ} might not be the limiting factor. When Δt_{ϕ} is not restrictive (e.g. because $\Delta t_{pc} < \Delta t_{\phi}$), the regularization speed Γ might be too slow to maintain the hyperbolic tangent interface shape. For those cases, we propose an adaptive change of Γ^* to keep $\Delta t^{n+1} = C_{CFL} \Delta t_{\phi}$ at all times. The remaining time step constraints for the present simulations are: (i) the convective time step constraint $\Delta t_{\vec{u}}$, (ii) the momentum diffusion constraint Δt_{μ}, (iii) the heat conduction constraint Δt_{λ}, (iv) the capillary time step constraint Δt_{σ}, and (v) the constraint due to gravitational acceleration $\Delta t_{\vec{g}}$. These are given by [85]

$$\Delta t_{\vec{u}} = \frac{\Delta x}{\max(|u_1| + |u_2| + |u_3|)}, \tag{56}$$

$$\Delta t_{\mu} = \frac{\Delta x^2}{6} \min_i \left(\frac{\rho_i}{\mu_i} \right), \tag{57}$$

$$\Delta t_{\vec{g}} = \sqrt{\frac{\Delta x}{|\vec{g}|}}, \tag{58}$$

$$\Delta t_{\sigma} = \sqrt{\frac{\Delta x^2 \min_i \rho_i}{\sigma \max|\kappa|}}, \tag{59}$$

$$\Delta t_{\lambda} = \frac{\Delta x^2}{6} \min_i \left(a_i^{-1} \right). \tag{60}$$

Here, κ is the interface curvature calculated from $\kappa = \nabla \cdot \vec{n}$. Following [85], Eqs. (56)–(59) can be combined into

$$\Delta t_{\Sigma} = \frac{2}{\frac{1}{\Delta t_{\vec{u}}} + \frac{1}{\Delta t_{\mu}} + \sqrt{\left(\frac{1}{\Delta t_{\vec{u}}} + \frac{1}{\Delta t_{\mu}} \right)^2 + \frac{4}{\Delta t_{\vec{g}}^2} + \frac{4}{\Delta t_{\sigma}^2}}}. \tag{61}$$

In contrast to the available literature on the conservative diffuse interface method, we solely define a minimum regularization speed $\Gamma^*_{\min}$ and let Γ^* adjust to the time step. Using Eq. (55), this is done by calculating

$$\Delta t_{\phi,min} = \min_i \left[\frac{1}{\max\left\{ \left(\frac{6\Gamma^*_{min} |u_{max}| \epsilon}{\Delta x^2} \right) - \left(\frac{\partial u_i}{\partial x_i} \right), 0 \right\}} \right]. \tag{62}$$

In case $\Delta t_{\phi,min} > \min(\Delta t_{\Sigma}, \Delta t_{\lambda}, \Delta t_{pc})$, we rearrange Eq. (55) to obtain the maximum allowed regularization speed as

$$\Gamma^* = \frac{\Delta x^2}{6|u_{max}|\epsilon} \left[\frac{1}{\min(\Delta t_{\Sigma}, \Delta t_{\lambda}, \Delta t_{pc})} + \min_i \left(\frac{\partial u_i}{\partial x_i} \right) \right]. \tag{63}$$

If $\Delta t_{\phi,min}$ is the restricting time step constraint, then $\Gamma^* = \Gamma^*_{\min}$. Regardless, the time step Δt^{n+1} is adjusted according to

$$\Delta t^{n+1} = C_{CFL} \min(\Delta t_{\phi,min}, \Delta t_{\Sigma}, \Delta t_{\lambda}, \Delta t_{pc}), \tag{64}$$

where C_{CFL} is the CFL safety factor, which is chosen to be $0.4 \le C_{CFL} \le 0.5$ for all test cases in Section 4. For the application to boiling flows, we found values in the range $1 \le \Gamma^*_{\min} \le 2$ and $\epsilon^* = 1$ to return favourable results.

3.6. Algorithm

The procedure introduced in Sections 3.2–3.5 is shown as a pseudo code in Algorithm 1. We report the exact order in which we solve the conservation equations from Section 2.

This algorithm exhibits several distinct features, which are summarized and highlighted as follows:

- Phase change model: As described in Section 3.4, we presented a modified *kinetic model* using only the local superheat, which is designed to be (i) efficient, (ii) robust, and (ii) free from simulation-dependent parameters. We also derived the relevant time step constraint (Eq. (54)) for a stable simulation.
- Momentum balance equation: Staying in line with the conservative diffuse interface method framework, we included additional terms (see Eq. (16)) related to the regularization term $\vec{\mathcal{R}}$ following a similar strategy as [10,32]. Note that, no assumptions on the relevance of these terms have been made.
- Pressure solution scheme: For the present study, we implement the FFT-MPDJ solver specialized for flows subjected to phase change. Through additional solution stages, we eliminate the drawbacks commonly reported by other studies [61,62] that are using the density splitting approach (Section 3.3.1). Through the design of the pressure scheme, no pressure oscillations related to interfacial pressure jumps occur.
- Adaptive Γ^*: Phase change poses a new requirement on the speed of the regularization term, which is not always satisfied by a fixed coupling to the maximum velocity. As shown in Section 3.5, we present a methodology that continuously updates Γ^* to keep the regularization speed at the level of the smallest timescale.

In the remainder of this paper, Algorithm 1 is tested against various benchmark simulations, through which we test the numerical scheme, and also discuss the effect and relevance of the above-mentioned features.

Algorithm 1 Proposed procedure for solving the governing equations.

```
Set initial conditions $\phi(t=0)$, $T(t=0)$, $p(t=0)$ , and $\vec{u}(t=0)$
Set the physical properties $\rho, c_p, \lambda, \mu$
Calculate initial $\Psi$ using Eq. (10)
Calculate $\vec{\mathcal{R}}$ from Eq. (9)
while $t < t_{end}$ do
    Obtain intermediate phase field value $\phi^*$ from Eq. (23)
    Update physical properties $\rho^*, c_p^*$
    Obtain intermediate temperature $T^*$ from Eq. (24)
    Calculate $\Delta T_{PC}^{n+1}$ using Eq. (50)
    Obtain $\mathcal{M}^{n+1}$ using Eq. (49)
    Final states of $\phi^{n+1}$ and $T^{n+1}$ through Eq. (28)
    Update $\Psi^{n+1}, \vec{n}^{n+1}, \kappa^{n+1}$
    Update the physical properties $\rho^{n+1}, c_p^{n+1}, \lambda^{n+1}, \mu^{n+1}$
    Calculate prediction momentum $(\rho\vec{v})^*$ from Eq. (29)
    Update $\vec{\mathcal{R}}^{n+1}$ with Eq. (9)
    Approximate $(\rho\vec{v})^{n+1}$ with $\widetilde{\rho\vec{v}}$ by Eq. (39) - Eq. (41)
    Obtain pressure estimate $\hat{\psi}$ by solving Eq. (42)
    Find final $p^{n+1}$ and $\vec{u}^{n+1}$ through Eq. (43) - Eq. (46)
    Update time step size $\Delta t^{n+1}$ with Eq. (64)
    if $\Delta t_{\phi,min} > \min(\Delta t_{\Sigma}, \Delta t_{\lambda}, \Delta t_{pc})$ then
        $\Gamma^*$ with Eq. (63)
    else
        Use $\Gamma^*_{\min}$
    end if
end while
End of simulation
```

4. Simulations

In this section, we solve the governing equations with the method introduced above for a range of benchmark simulations. We conduct simulations including and excluding phase change, while the complexity of the flow problems increases towards the end of the section.

4.1. One-dimensional recoil pressure jump

The first benchmark case, inspired by [10], is a simple one-dimensional domain in which a moving coordinate system follows an interface (see Fig. 3(a)), so that the phase field variable ϕ is constant in time (Fig. 3(b)). A velocity jump U is prescribed by defining the mass transfer term as $\mathcal{M} = U|\nabla\phi|\left(\rho_1^{-1} - \rho_2^{-1}\right)^{-1}$, implying that the flow field is fully defined by the continuity equation (Eq. (7)). Following [10], the coordinate system moves at interface velocity $u_{int} = -U(\rho_2/\rho_1 - 1)^{-1}$. Accordingly, the phase velocities in the moving reference frame are $u_1 = U - u_{int}$ and $u_2 = -u_{int}$ (see Fig. 3(a)). Consequently, the theoretical recoil pressure jump at the interface is given by $\Delta p_{th} = p_1 - p_2 = -u_{int}^2\rho_2(\rho_2/\rho_1 - 1)$. As the phase field ϕ and the density ρ are constant in the moving frame, we can integrate Eq. (8) to obtain an exact expression for the regularization term $\vec{\mathcal{R}} = (\rho\vec{u} + u_{int}\rho_2)(\rho_1 - \rho_2)^{-1}$. We now conduct two different tests: one with $U = const.$ and another with $U = f(t)$. For all simulations, we use a grid of size $\Delta x = 10^{-3}$ m in a domain of length $l = 0.1$ m, and set the densities to $\rho_1 = 0.001$ kg m^{-3} and $\rho_2 = 1$ kg m^{-3}.

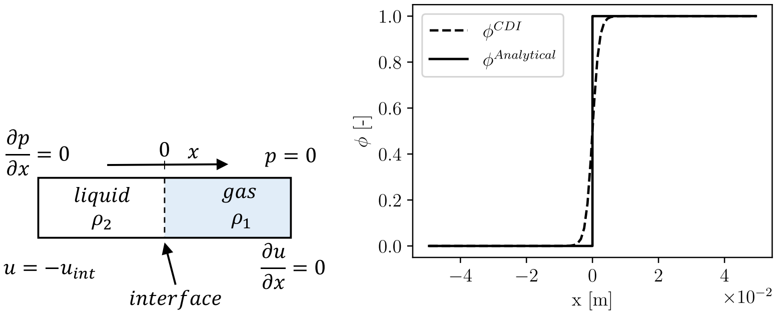
4.1.1. Pressure solver comparison

In the first test, we set $U = 0.5$ m/s (see Fig. 4(a)) and monitor the behaviour of the pressure solver when calculating the pressure jump. Solvers (e.g. iterative or direct) that invert the matrix in Eq. (31) can reliably return the pressure distribution seen in Fig. 4(b) after one iteration, since all other quantities are constant. However, as shown in [61], simulating this pressure jump with FFT solvers using the density splitting (Section 3.3.1) can be challenging due to pressure oscillations.

We then compare the FFT-MPDJ solver (Section 3.3.2) with the FFT solver in [57] by measuring the relative pressure jump error versus the iteration count. The relative error in Δp_{th} is visualized in Fig. 5 for the two solvers under investigation; the FFT-MPDJ solver reaches an relative deviation from Δp_{th} of $10^{-8}\%$ in less than 10 iterations. In contrast, the density splitting approach in [57] exhibits strong oscillations at the start of the simulation, only reaching an acceptable deviation from Δp_{th} after completing $\sim 10^4$ iterations. Once reaching a steady state, both solvers return the pressure distribution shown in Fig. 4(b).

4.1.2. Evaluation of the momentum balance equation

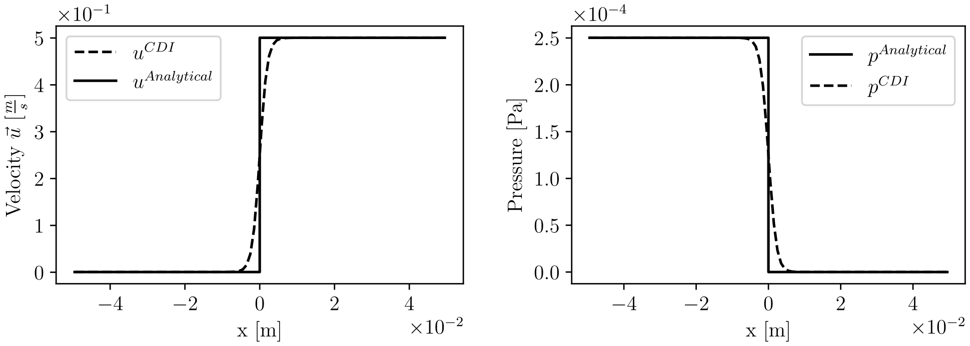
In the second stage of the recoil pressure experiments, a variable velocity jump is applied and the pressure equation is solved with the FFT-MPDJ solver. This enables us to evaluate the relevance of $\vec{C}$ (see Eq. (16)). As seen in, e.g., Fig. 4(a), the interface speed is small



(a) Flow configuration

(b) Phase field profile

Fig. 3. Basic simulation settings for the case of the one-dimensional recoil pressure jump.



(a) Velocity jump

(b) Pressure jump

Fig. 4. Numerical results of the one-dimensional recoil pressure jump test case when prescribing a constant interfacial velocity jump $U = 0.5$ m/s and $\rho_2/\rho_1 = 1000$. The velocity (a) and the pressure (b) have a smoothed jump due to the diffused interface representation. Solid lines represent the analytical jump conditions.

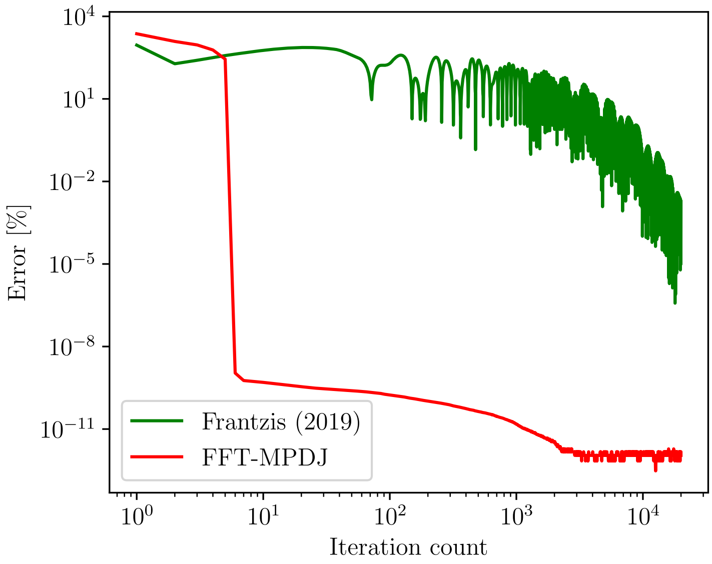
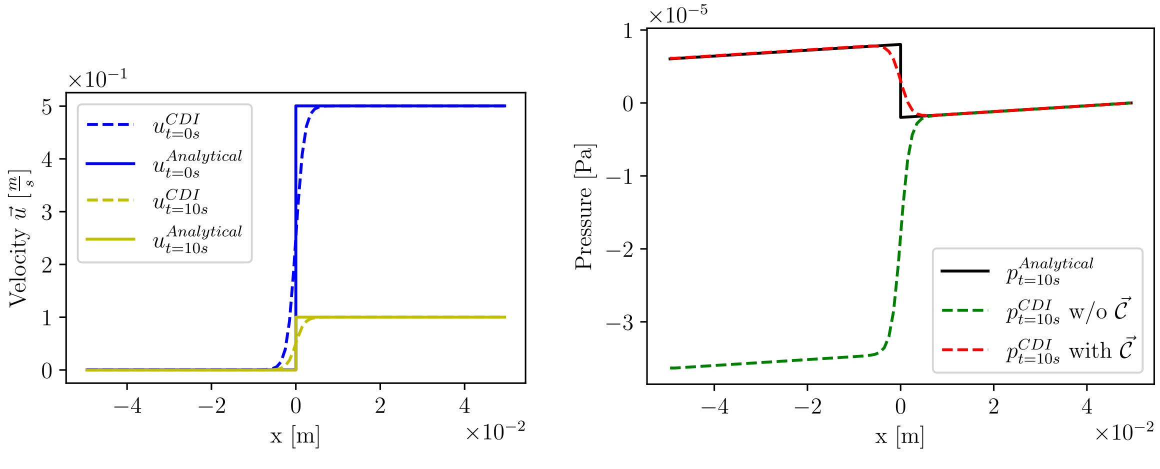


Fig. 5. Relative error as measured by Δp_{th} over the iteration count for different PPE solvers.



(a) Velocity jump (b) Pressure jump

Fig. 6. Numerical results of the one-dimensional recoil pressure jump test case when prescribing a variable interfacial velocity jump $U = U(t) = [0.5 - 0.04t]$ m/s and $\rho_2/\rho_1 = 1000$. The velocity (a) jump is shown at the start ($t = 10$ s) and at the end ($t = 10$ s) of the simulation. Since $\vec{u}$ is fully determined by the continuity equation (Eq. (7)), neglecting $\vec{C}$ has no impact on the velocity $\vec{u}$. The pressure at $t = 10$ s is shown in (b), where a strong dependence of $\vec{u}$ on $\vec{C}$ is seen. Analytical results are represented by solid lines.

compared to the velocity jump $U = \Delta\vec{u}$, which is proportional to $\vec{\mathcal{R}}$; thus, the additional terms (i.e. $\vec{C}$) are more prominent. Fig. 6(a) illustrates the change in the velocity field from $t = 0$ s to $t = 10$ s when the velocity jump is prescribed as $U = U(t) = [0.5 - 0.04t]$ m/s. We carry out two simulations: (i) one including $\vec{C}$, and (ii) one excluding. Fig. 6(b) shows the pressure distribution at time $t = 10$ s, where the red dashed line represents the simulation with $\vec{C}$ and the green dashed line the one without. The analytical solution (black solid line) shows the pressure jump Δp_{th} along with the slope $\partial p/\partial x = \rho_2 \partial u_{int}/\partial t$ in each phase. While the simulation including the additional terms matches the analytical solution, a major discrepancy is seen for the simulation where $\vec{C}$ was neglected. Not only becomes the magnitude (green dashed) of the jump incorrect, but also its sign, underlining the importance of $\vec{C}$ for boiling flow phenomena.

Both of the recoil pressure tests demonstrate the improvements achieved through the methodology developed in the present study. First, the deployed FFT-MPDJ solver exhibits superior properties for calculating the recoil pressure jump at interfaces subjected to phase change. Second, for capturing the pressure distribution accurately, this test case shows the importance of keeping the additional terms the momentum equation (Eq. (15)), which arise from the differential velocity in the interfacial region. Although these two features significantly improve the results for the one-dimensional phase change tests, an obvious question arises: how do they affect the simulation of three-dimensional bubble dynamics in the absence of phase change? This is addressed in the next subsection.

4.2. *Three-dimensional rising bubble*

Simulating the dynamics of bubbles rising due to gravity is a common benchmark. Specifically, one may test the interface capturing model or the pressure solution scheme [56,60,62,70,86]. In detail, we consider a three-dimensional setup using the geometry from [87]. Refer to Fig. 7 for a schematic diagram of this flow configuration. At $t = 0$, a spherical bubble of diameter D_0 is initialized at location $x_1 = x_2 = x_3 = D_0$. The no-slip condition is applied at all boundaries of the domain.

This benchmarks involves two different sets of fluid properties, which differ in the gas density and viscosity, as well as the surface tension (Table 1). The relevant non-dimensional numbers are the Reynolds number $Re = \rho_1 u_{ref} l_{ref} \mu_1^{-1}$, the Weber number $We = \rho_1 u_{ref}^2 l_{ref} \sigma^{-1}$, and the Froude number $Fr = u_{ref}/\sqrt{\|\vec{g}\| l_{ref}}$ [60]. The selected reference quantities are

$$l_{ref} = D_0, \quad t_{ref} = \sqrt{\frac{l_{ref}}{\|\vec{g}\|}}, \quad u_{ref} = \sqrt{l_{ref}\|\vec{g}\|}, \quad p_{ref} = \frac{u_{ref}\mu_2}{l_{ref}}. \tag{65}$$

The values in Table 1 show that a laminar flow is expected in both cases. In 'Case B' the influence of the surface tension is lower, whereas for 'Case A' the inertial effects are comparable to the effect of the surface tension. In this section, all non-dimensionalized results are marked with an asterix. For this benchmark, the *Continuum Surface Force* CSF [43] is used. A detailed description of this model is found in Section 4.5.1.

4.2.1. *Pressure solver comparison*

In the first series of simulations, we switch off $\vec{C}$ and compare the pressure solver in [57] with the FFT-MPDJ solver of the present study (see data in Fig. 8). Simulations are carried out for two uniform grid spacings (i) with $\Delta x = D_0/32$ and (ii) with $\Delta x = D_0/64$. Both

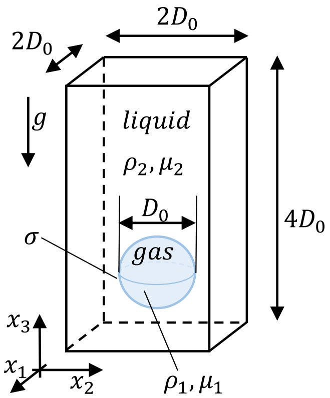


Fig. 7. Simulation setup for the bubble rising benchmark.

Table 1
Corresponding fluid properties (see [87]).

Name	$\frac{\rho_2}{\rho_1}$	$\frac{\mu_2}{\mu_1}$	Re	We	Fr
Case A	10	10	35	1	1
Case B	1000	100	3.5	0.125	1

are applied to two sets of fluid properties ’Case A’ and ’Case B’. Note that the density splitting approach’s (Section 3.3.1) capability of simulating bubble dynamics has been verified repeatedly in the literature [56,60]. Accordingly, the FFT-MPDJ solver should ideally return the same results as the common density splitting FFT [57].

The quantities of interest are (i) the centre of mass of the gaseous phase (left panels of Fig. 8), (ii) the sphericity of the bubble (middle panels of Fig. 8) defined as $A(t=0)/A(t)$, where A is the surface area of the isocontour of the phase field $\phi = 0.5$, and (iii) the bubble rise velocity (right panels of Fig. 8). The data in Fig. 8(a) confirm that the results obtained from both solvers are indistinguishable. For the higher density ratio ’Case B’, the agreement remains very good, although a slight difference can be observed in the sphericity for both grid resolutions (Fig. 8(b)).

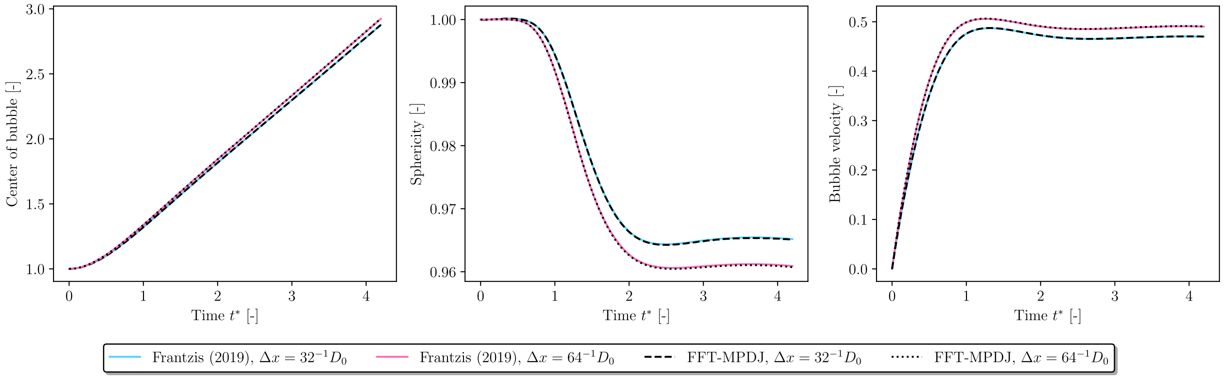
In the next step, we investigate the source of the small differences seen for the high density ratio by examining the pressure field evolution versus simulation runtime. We consider the pressure along a line parallel to the x_3-axis cutting through the centre of the bubble. Qualitatively, the pressure should exhibit a linear increase towards the lower domain boundary (hydrostatic pressure), interrupted at the (upward-moving) bubble location, due to its lower density and its velocity.

The pressure distribution was collected 32 times until $t^* = 1.34$ for both solvers. The result for $\Delta x = D_0/32$ is seen in Fig. 9, where the solver from [57] (left) is compared to the present FFT-MPDJ solver (middle), using the initial condition $p^*(t=0) = 0$. The pressure field for the density splitting scheme requires almost the entire observation interval to adjust to the hydrostatic pressure. In contrast, the FFT-MPDJ solver adapts by $t^* = 0.04$. As can be seen on the right panel of Fig. 9, both solvers return the same pressure distribution after the start-up phase. Long start-up phases for adjusting the pressure field to the flow are deemed to have a minor impact on the bubble dynamics depicted in Fig. 8(b).

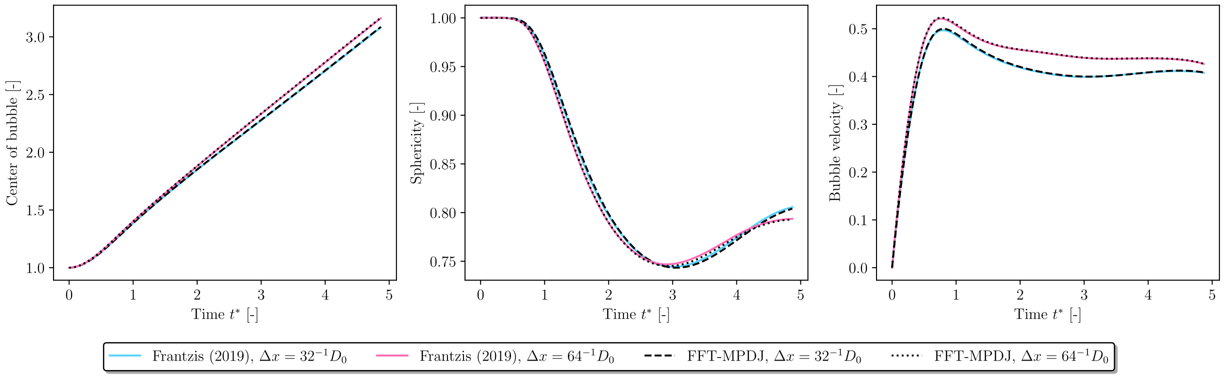
4.2.2. *Evaluation of the momentum balance equation*

The second series of simulations aims to investigate the effect of the additional momentum transport terms (i.e. $\vec{C}$) on the bubble dynamics in the absence of phase change. In the literature [34], the transport of relative momenta is often neglected. This is reasonable when comparing the differential mass flux $\vec{F}$ to the momentum $\rho\vec{u}$ for pure interface transport when excluding phase change. To quantify the effect, we conduct simulations for ’Case A’ and ’Case B’ using the FFT-MPDJ solver and the previous two resolutions. For each case, the results obtained including and excluding the additional terms are compared (see Fig. 10). We also include the benchmark data published in [87] as further reference. The reference data was obtained through the codes: (i) DROPS (level-set, finite element method) [88], (ii) NASt3D (level-set, finite difference method) [89] , and (iii) OpenFOAM (VOF, finite volume method) [90]. We refer the interested reader to the publication by [87] for further simulation details. The results for both density ratios, together with the reference data, are shown in Fig. 10. In this figure, it becomes apparent that for all simulations, an improvement is seen when $\vec{C}$ is included. The main effect is seen in the sphericity (middle), where the results clearly improve towards the reference data when $\vec{C}$ is used. Slight improvements are also visible throughout the simulations for the rise velocity (right panels of the same figure).

To conclude, with this non-phase change benchmark, we demonstrate that the proposed FFT-MPDJ solver can accurately simulate bubble dynamics in three dimensions. Additionally, we show that the our approach is superior for adjusting the pressure field to the flow problem at high density ratios. Lastly, this benchmark demonstrates that the modified momentum balance equation, incorporating the additional terms from Eq. (16), also exhibits clear, albeit minor, improvements when simulating bubble dynamics in the absence of phase change. Despite these improvements, we acknowledge that there is merit in the common assumption of neglecting



(a) Results for Case A with $\rho_2/\rho_1 = 10$



(b) Results for Case B with $\rho_2/\rho_1 = 1000$

Fig. 8. Comparison of the pressure solution scheme by [57] (blue and pink solid lines) with the FFT-MPDJ solver (Section 3.3.2, black dashed and dotted lines) for simulating bubble dynamics without phase change. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

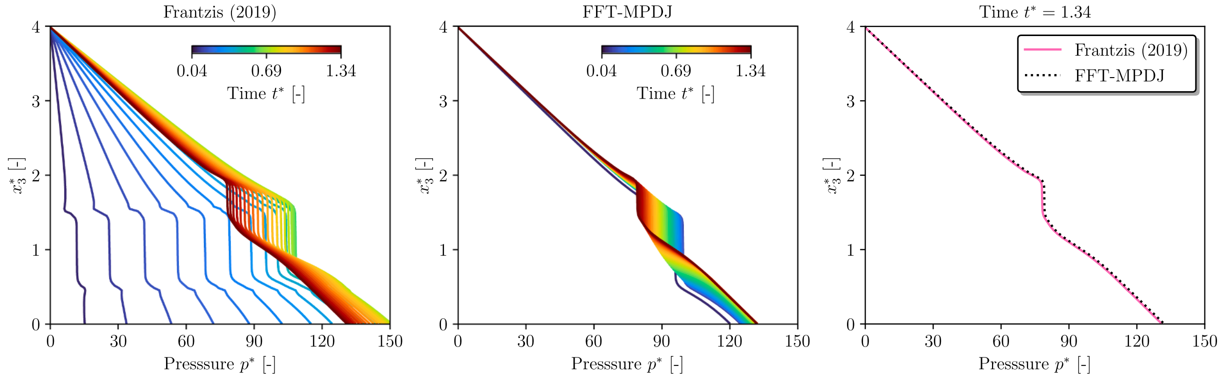
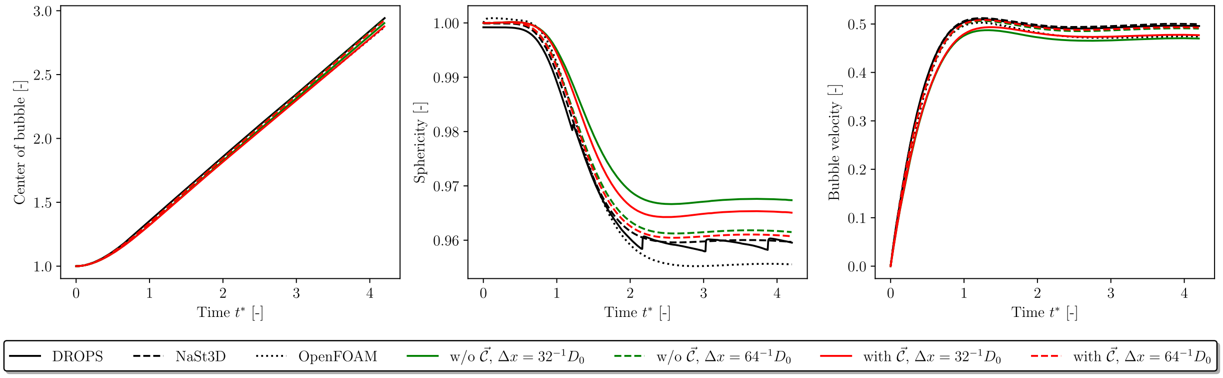


Fig. 9. Evolution of the pressure distribution along the x_3 direction through the bubble centerline for Case B ($\rho_2/\rho_1 = 1000$) with $\Delta x = 32^{-1}D_0$. Comparison between two pressure solution schemes (left and middle panel). The pressure distribution at $t^* = 1.34$ obtained with either scheme is compared on the right panel.

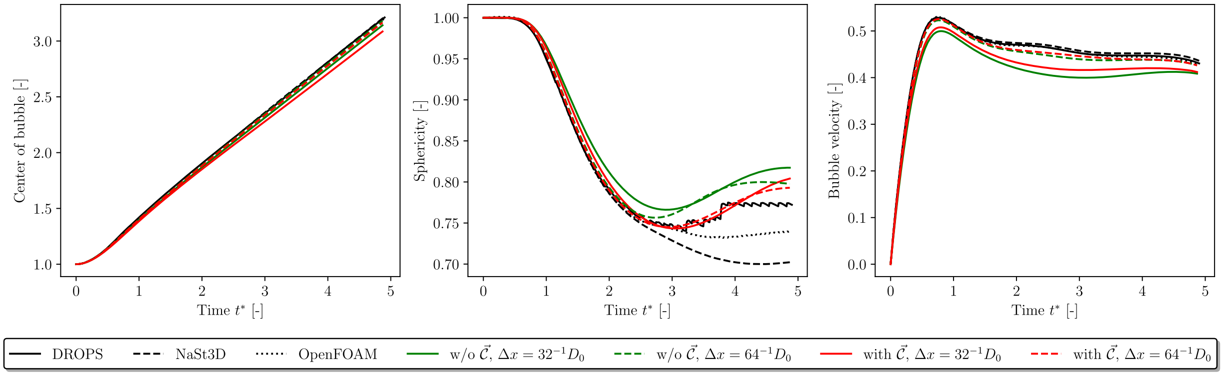
$\vec{C}$ for non-phase change problems. Based on the findings of these first benchmarks, the FFT-MPDJ solver and the full momentum balance equation are used for the simulations in the following sections.

4.3. Stefan problem

The Stefan problem is a one-dimensional phase-change problem in which a superheated wall heats a growing gaseous layer, pushing away the adjacent liquid phase. This situation is visualized in Fig. 11(a), where a typical temperature profile is sketched.



(a) Results for Case A with $\rho_2/\rho_1 = 10$



(b) Results for Case B with $\rho_2/\rho_1 = 1000$

Fig. 10. Comparing the effect of $\vec{C}$ on bubble dynamics without phase change with respect to reference simulations (black lines) from [87].

Table 2
Physical properties of various two-phase fluid systems.

Property	Unit	Fluid$_A$	Fluid$_B$	Fluid$_C$	Water[1]	Ethanol[1]
ρ_1	kg m^{-3}	0.01	0.25	5	0.597	1.647
ρ_2	kg m^{-3}	1	2.5	200	958.4	736.44
μ_1	kg m^{-1} s^{-1}	n/a	n/a	0.005	$1.26 \cdot 10^{-5}$	$1.02 \cdot 10^{-5}$
μ_2	kg m^{-1} s^{-1}	n/a	n/a	0.1	$2.80 \cdot 10^{-4}$	$4.49 \cdot 10^{-4}$
$c_{p,1}$	J kg^{-1} K^{-1}	200	10	200	2030	1806
$c_{p,2}$	J kg^{-1} K^{-1}	n/a	10	n/a	4216	3185
λ_1	W m^{-1} K^{-1}	0.005	0.0035	1	0.025	0.0199
λ_2	W m^{-1} K^{-1}	n/a	0.0015	n/a	0.679	0.1654
L_{sat}	J kg^{-1}	10^4	10^2	10^4	$2.26 \cdot 10^6$	$8.499 \cdot 10^5$
T_{sat}	°C	n/a	n/a	n/a	100	78.2
σ	N m^{-1}	n/a	n/a	0.1	0.059	0.0174

[1] at 1013 hPa

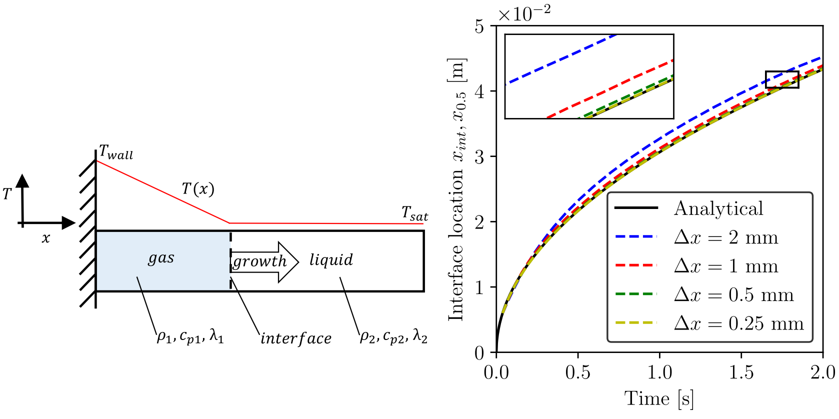
The distance of the interface from the wall x_{int} has an analytical solution [91] of the form

$$x_{int}(t) = 2\beta\sqrt{a_1 t} \quad \text{, and} \quad \beta \exp(\beta^2)\mathrm{erf}(\beta) = \frac{c_{p,1}(T_{wall} - T_{sat})}{L(T_{sat})\sqrt{\pi}}, \tag{66}$$

where β is the growth constant, which depends on the wall superheat $T_{wall} - T_{sat}$ and the physical properties of the vapour. This case is a classical benchmark for phase change models [7,92].

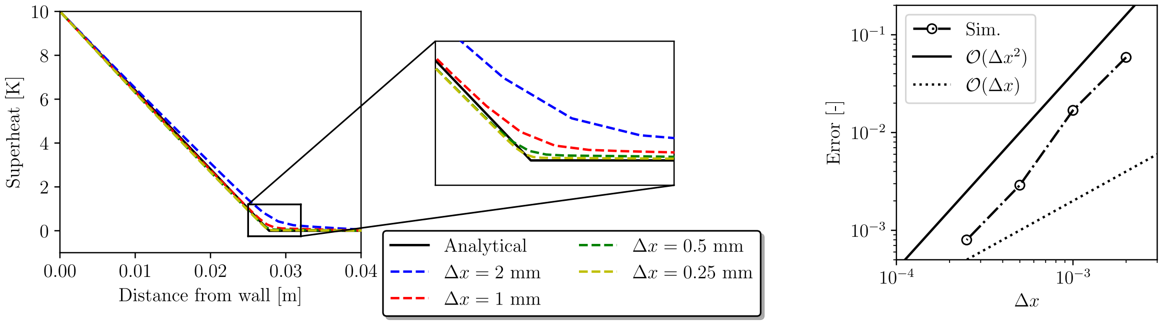
Using the parameter settings from [92], i.e. $T_{wall} - T_{sat} = 10$ K, and the physical properties of Fluid$_A$ (see Table 2), a growth constant of $\beta = 0.3064$ is numerically determined. For the simulation, a domain of length $l = 0.05$ m is discretized with four different grids using the grid spacings $\Delta x = 2$ mm, 1 mm, 0.5 mm, and 0.25 mm, respectively. All simulations are initialized at physical time $t = 38.34$ ms, with the interface ($\phi = 0.5$) located at $x_{0.5} = 6$ mm and the corresponding analytical (linear) temperature profile.

The interface location $x_{0.5}$ is plotted over time t, together with the analytical solution, in Fig. 11(b). With decreasing grid spacing, convergence to the analytical solution of Eq. (66) is achieved. Results with a grid spacing of $\Delta x = 2$ mm exhibit a notable deviation



(a) (b)

Fig. 11. (a) Schematic sketch of the Stefan problem with (b) the results for the interface location over time for four different grid spacings.



(a) Temperature profiles at time $t = 0.823$ s (b) Error convergence

Fig. 12. Stefan problem: (a) Temperature profiles in comparison to the analytical solution, and (b) error convergence using Eq. (67).

from the analytical solution. Towards smaller grid spacings, a good agreement appears at $\Delta x = 1$ mm, becoming almost identical to the analytical solution at $\Delta x = 0.5$ mm and $\Delta x = 0.25$ mm.

The results in [92] (sharp interface, VOF) report grid convergence at $\Delta x = 2$ mm. In contrast, the method presented in this study requires roughly twice the number of grid points for convergence. This confirms a typical trend of the increased resolution requirement for diffuse-interface methods compared to sharp-interface methods [8].

In Fig. 12(a), we compare the temperature distributions obtained from the different grid spacings to the analytical solution at time $t = 0.823$ s. The temperature profiles confirm the findings in Fig. 11(b). With $\Delta x = 2$ mm the temperature greatly deviates from the analytical (linear) temperature distribution. Using $\Delta x = 1$ mm a good agreement is observed and further decreasing the grid spacing results in minor improvements.

For quantifying the error reduction with the grid spacing, an error measure is introduced. The error is quantified using the difference of the simulated interface position $x_{0.5}$ where $\phi = 0.5$, and the analytical position x_{int} as

$$\text{Error} = \sqrt{\frac{\sum_t \left(x_{int} - x_{0.5}\right)^2}{\sum_t x_{int}^2}}. \tag{67}$$

In Fig. 12(b) this error measure is shown for the different grid spacings. The error reduces at an approximately second-order rate for the Stefan problem.

4.4. The sucking interface problem

The sucking interface problem (or absorption problem) describes a situation where a growing gaseous layer pushes a liquid away from a wall. Unlike the Stefan problem, the liquid is superheated, with the gaseous phase and the wall remaining at saturation conditions. The heat reaches the interface from the liquid side, where a thermal layer is developing. This situation is depicted in Fig. 13.

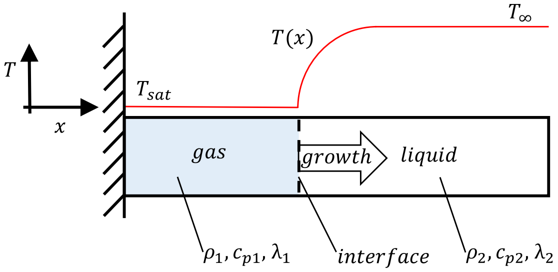


Fig. 13. Setup of the sucking interface problem.

According to the analytical solution, the interface location x_{int} has a square-root dependence with respect to time t,

$$x_{int}(t) = 2\beta\sqrt{a_1 t}. \tag{68}$$

However, for the sucking interface problem, the growth constant β is obtained from a different transcendental equation [91] that determines β

$$\exp(\beta^2)\mathrm{erf}(\beta)\left[\beta - \frac{(T_\infty - T_{sat})c_{p,1}\lambda_2\sqrt{a_1}\exp\left(-\beta^2\frac{\rho_1^2 a_1}{\rho_2^2 a_2}\right)}{L_{sat}\lambda_1\sqrt{\pi a_2}\ \mathrm{erfc}\left(\beta\frac{\rho_1\sqrt{a_1}}{\rho_2\sqrt{a_2}}\right)}\right] = 0. \tag{69}$$

The corresponding analytic temperature profile of the liquid phase $T_2(x,t)$ reads

$$T_2(x,t) = T_\infty - \left[\frac{(T_\infty - T_{wall})}{\mathrm{erfc}\left(\beta\frac{\rho_1\sqrt{a_1}}{\rho_2\sqrt{a_2}}\right)}\right]\mathrm{erfc}\left(\frac{x}{2\sqrt{a_2 t}} + \frac{\beta(\rho_1-\rho_2)}{\rho_2}\sqrt{\frac{a_1}{a_2}}\right). \tag{70}$$

This flow problem is tested with two liquids, namely (i) a generic Fluid$_B$, and (ii) saturated Water at 100 °C (see Table 2), posing different challenges to our phase change model.

The first set of simulations with Fluid$_B$ replicates the settings used in [93], where the liquid phase has a far-field temperature $T_\infty - T_{sat} = 2$ K. Using Eq. (69), 2 K above the saturation temperature leads to the growth constant value $\beta \approx 0.2689$. The discretization of the computational domain ($l = 1$ m) is conducted using cell sizes $\Delta x = 20$ mm, 10 mm, 5 mm, 2.5 mm, and 1.25 mm. At physical time $t = 4$ s, the simulation is started, with the centre of the phase field being located at $x_{int} = 0.0402$ m. In this simulation, the temperature of the liquid is initialized using Eq. (70), with the gas being saturated.

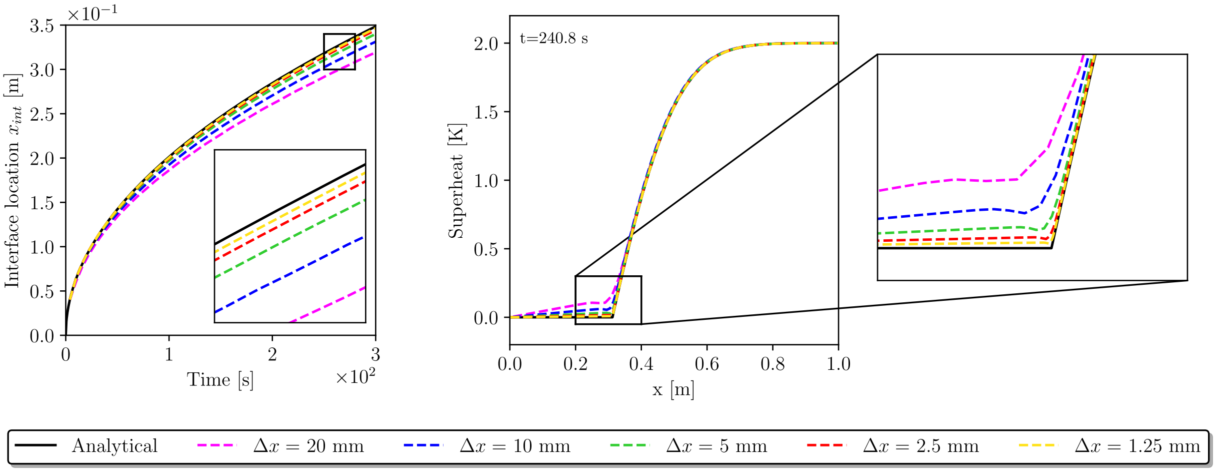
As commonly observed for this benchmark [49,93], the front speed is underestimated for coarse grids. This is reflected on the left side of Fig. 14(a). Larger deviations are observed for grid spacings $\Delta x = 20$ mm and $\Delta x = 10$ mm. Therefore, the resolution was increased, resulting in good agreement with grid spacing $\Delta x = 5$ mm and finer. The error reduction with decreasing grid spacing is visualized in Fig. 15(a) where Eq. (67) is used to quantify the error. For this set of fluid properties, the error reduces at a first-order rate.

Additional insights are obtained by examining the temperature distribution at a physical time $t = 240.8$ s (Fig. 14(a), middle and right plots). For the two coarse grids, a significant deviation from the analytical temperature is observed in the gaseous phase. In the liquid phase, away from the interface, all profiles collapse onto the analytical solution. Slight superheat appearing in the gas could be a result of the anti-trapping current $\vec{j}$ (see Eq. (22)). For Fluid$_B$ the heat conductivities are $\lambda_1 > \lambda_2$. This causes $\vec{j}$ to switch sign, leading to additional heat transport into the gas. This could also explain the reduced error reduction rate seen in Fig. 15(a). However, for most 'real' fluids, $\lambda_1 < \lambda_2$ (see Table 2). Therefore we regard the results concerning this fluid as an exception.

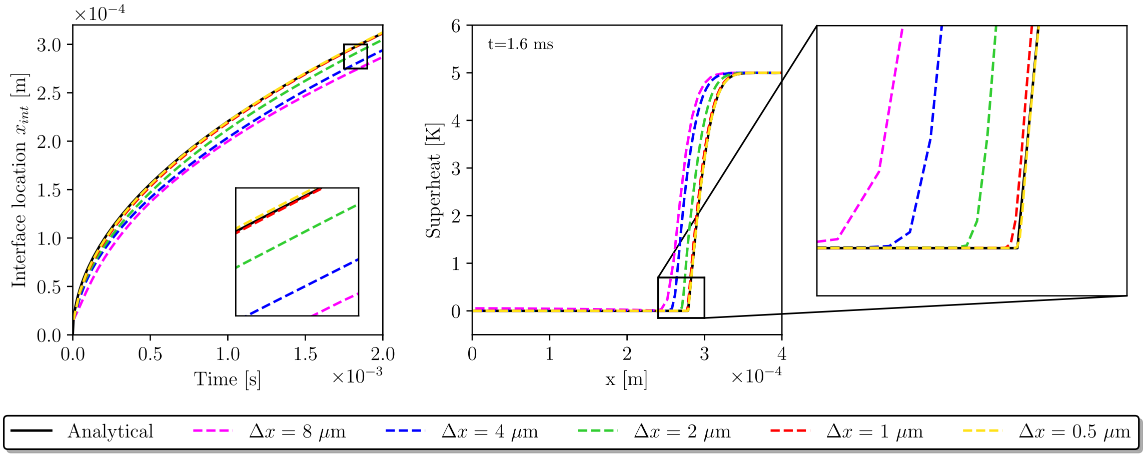
In the second set of simulations, the fluid properties of saturated water at 100 °C are considered. For these simulations, the far-field temperature of the liquid is 5 K above the saturation temperature. The water properties exhibit a high density ratio, a high heat capacity, and fast growth rates. These conditions allow us to test the limits of the phase change module (here, $\beta \approx 0.767$). We use the five grids with $\Delta x = 8$ μm, $\Delta x = 4$ μm, $\Delta x = 2$ μm, $\Delta x = 1$ μm, and $\Delta x = 0.5$ μm, and a domain of length $l = 0.4$ mm.

Fig. 14(b) shows the results when starting the simulation at time $t = 5$ μs and using the corresponding analytical solution as the initial condition. On the left, the interface position versus time is shown. With an increased grid resolution, the result quality is visibly improved. The rate at which the results converge to the analytical solution is shown in Fig. 15(b). The convergence rate is slightly improved compared to the case employing the fluid properties of Fluid$_B$, but slower than that seen for the Stefan problem. Lower resolutions under-predict the growth rate in the start-up phase, resulting in a permanent offset relative to the analytical solution for the remainder of the simulation runtime. This indicates that, at lower resolutions, the initial thermal profile is under-resolved, resulting in erroneous interface speeds. The middle and left plots show the steep temperature profile, characteristic of the selected physical properties. Here, $\lambda_1 < \lambda_2$, which is also reflected in the temperature profiles, where no spurious temperature distribution is found in the gaseous phase (compare with Fig. 14(a)).

The utilized temperature transport equation (Eq. (20)) has a non-conservative form, and thus, the enthalpy error accumulation over time is investigated. We consider the sucking interface problem with the Water properties to construct a critical scenario. This is

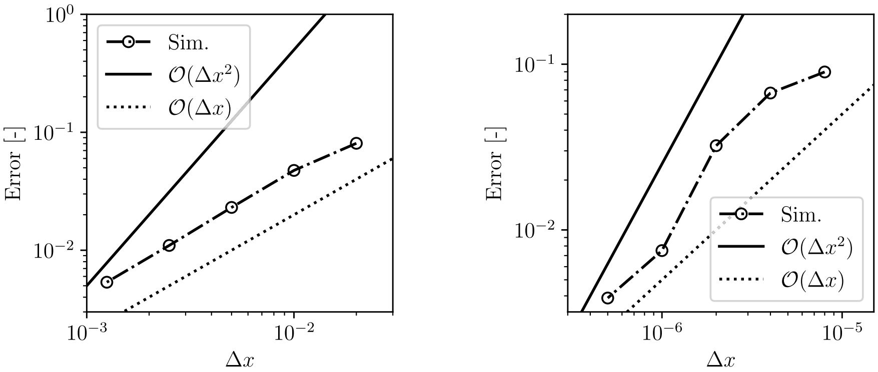


(a) Fluid$_B$, T_∞ = 2 K



(b) Water, T_∞ = 5 K

Fig. 14. Results of the sucking interface problem.



(a) Fluid$_B$, T_∞ = 2 K

(b) Water, T_∞ = 5 K

Fig. 15. Error convergence rates for both fluids in the sucking interface problem calculated from Eq. (67).

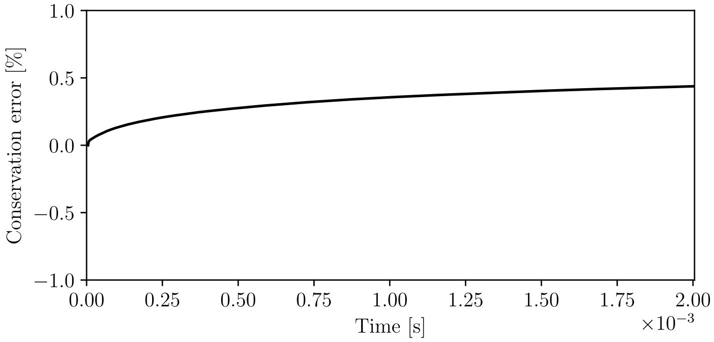


Fig. 16. Conservation error calculated from the total enthalpy in the domain using Eq. (71) and a grid spacing of $\Delta x = 2$ µm.

due to the large contrasts in fluid properties and the presence of steep temperature gradients on the interface. For this test, we monitor the enthalpy over time to evaluate its change. As the left boundary is an adiabatic wall, this evaluation solely needs to account for the outflow of liquid on the outflow boundary $\partial\Omega_O$ in the boundary normal direction $\vec{n}_O$ (right of Fig. 13). The total enthalpy is calculated from

$$H_{tot}(t) = \int_{\Omega} \rho h + \int_0^t \int_{\partial\Omega_O} \rho h(\vec{n}_O \cdot \vec{u}). \tag{71}$$

In Fig. 16 the change of $H_{tot}(t)$ in relation to its value at simulation start is plotted in percent versus the simulation runtime. The maximum observed conservation error is 0.4 %. Therefore, we conclude that gain or loss of thermal energy plays a negligible role in the presented framework for the problems considered in this study.

In the subsequent sections, the phase change model is subjected to more complex cases in two and three dimensions.

4.5. Three-dimensional bubble growth

In three dimensions, an important benchmark is the study of a bubble growing in a superheated liquid at zero gravity. This problem can reduce to a spherically symmetric flow, for which a known analytical solution is available [94] and serves as reference for the following numerical tests. The analytical solution has a square-root dependency on time t, such that the radial interface position R_{int} is

$$R_{int}(t) = 2\beta\sqrt{a_2 t}. \tag{72}$$

For the three-dimensional bubble growth [94], β is governed by the following transcendental equation

$$\begin{aligned} &\frac{\rho_2 c_{p,2}(T_\infty - T_{sat})}{\rho_1\left(L_{sat} + (c_{p,2} - c_{p,1})(T_\infty - T_{sat})\right)} \\ &= 2\beta^2 \int_0^1 \exp\left[-\beta^2\left(\left(1-\chi\right)^{-2} - 2\chi\left(1 - \rho_1/\rho_2\right) - 1\right)\right] d\chi. \end{aligned} \tag{73}$$

In all subsequent simulations, we consider 1/8th of the bubble, by placing an initial bubble in the corner of a cubic domain. This setup is sketched in Fig. 17, where the domain boundaries intersecting the bubble are symmetry boundaries, and the opposite ones are outflow boundaries. Using properties of water (see Table 2), and a far-field temperature of $T_\infty - T_{sat} = 5$ K, leads to a growth constant of $\beta \approx 15.073$ when solving Eq. (73).

The cubic computational domain is discretized using uniform grids of sizes $\Delta x = 125$ µm, $\Delta x = 62.5$ µm, $\Delta x = 31.3$ µm, and $\Delta x = 15.6$ µm. In the computational domain of length $l = 4$ mm, a bubble is placed with an initial radius of $R_0 = l/4$, corresponding to a physical time $t = 6.55$ ms at simulation start. The temperature field is initialized with the analytical temperature profile [94]

$$\begin{aligned} T_2(r, R_{int}) &= T_\infty - \frac{2\beta^2\rho_1}{\rho_2 c_{p,2}}\left(L_{sat} + (c_{p,2} - c_{p,1})(T_\infty - T_{sat})\right) \\ &\times \int_{1-\frac{R_{int}}{r}}^1 \exp\left[-\beta^2\left(\left(1-\chi\right)^{-2} - 2\chi\left(1 - \rho_1/\rho_2\right) - 1\right)\right] d\chi. \end{aligned} \tag{74}$$

Note that for the first set of simulations, surface tension is disabled, as it does not (from an analytical point of view, see Eqs. (72)–(74)) impact the bubble growth, itself.

Fig. 18(a) shows the size of the bubble over time for all grids in comparison to the analytical solution. A strong dependency on the grid resolution is observed, resulting in significant deviations from the analytical growth rate for grids using $\Delta x = 125$ µm and $\Delta x = 62.5$ µm. Good agreement with Eq. (72) is only achieved by using a grid spacing of $\Delta x = 15.6$ µm. In Fig. 18(c), the

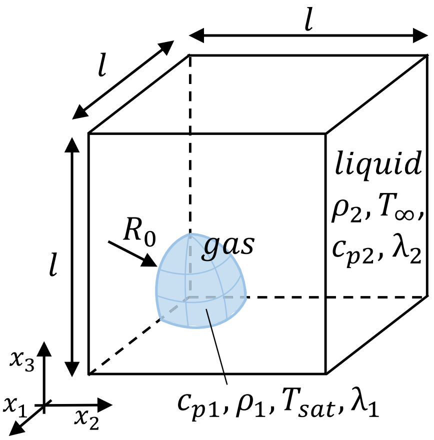


Fig. 17. Setup for simulating three dimensional bubble growth.

error convergence for the considered grid spacings is shown when using the error definition of Eq. (67). Across all grid spacings, an approximately second-order error reduction rate can be observed. The reason for the high resolution requirements is evident in Fig. 18(b), which depicts the analytical solutions steep temperature gradients. Although the results obtained from $\Delta x = 15.6$ μm to $\Delta x = 62.5$ μm seem to be similar, they still exhibit distinct differences in the temperature gradient close to the interface, which explains the superior performance when using $\Delta x = 15.6$ μm. More details on the results when using $\Delta x = 15.6$ μm are found in Fig. 19, where we visualize the velocity field alongside the temperature distribution (left) and the distribution of the mass transfer $\mathcal{M}$ (right). The velocity field jumps in the radial direction across the interface, whereas the gas velocity inside the bubble stays at rest. Additionally, $\mathcal{M}$ has a smooth distribution, verifying the modelling choices made in Section 3.4. Interestingly, for this flow, $\mathcal{M}$ is slightly shifted towards the liquid side. This slight asymmetry of $\mathcal{M}$ could be a result of the temperature transport contribution of the regularization terms $\vec{\mathcal{R}}$ through Eq. (21). However, a negative impact was not observed in this study.

4.5.1. *Evaluation of surface tension models*

Here, the same computational settings are considered, but the surface tension is also included, i.e., $\vec{f}_{ST} \neq 0$. In the context of the CDI method, two classes of surface tension models are used: (i) the continuum surface force (CSF), and (ii) energy-based models. The CSF model [43] takes the simple form

$$\vec{f}_{ST,CSF} = \sigma\kappa\nabla\phi, \tag{75}$$

where σ is the surface tension coefficient and κ the interface curvature is defined as $\kappa = \nabla\cdot\vec{n}$. The energy-based models for CDI are based on the free-energy functional taken from the Allen-Chan model [42]. The generic energy-based method [13,42,45] reads

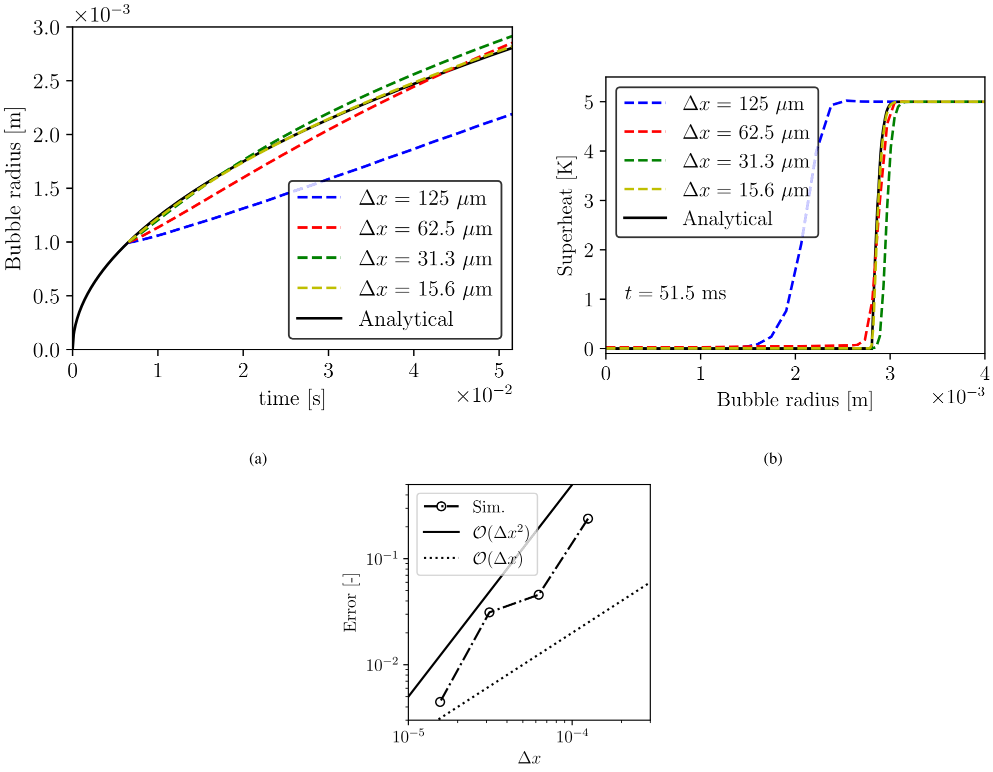
$$\vec{f}_{ST,EB} = \xi\nabla\phi, \quad \xi = \frac{6\sigma}{\epsilon}\left(\frac{\partial\Phi}{\partial\phi} - \epsilon^2\nabla^2\phi\right), \quad \text{and} \quad \Phi = \frac{1}{2}\left[\phi(1-\phi)\right]^2, \tag{76}$$

where ξ denotes the chemical potential. The chemical potential uses the free energy Φ of the phase field variable $\phi \in [0,1]$. For applications without phase change, the energy-based method is deemed superior to the CSF due to its improved convergence rate [42,47].

In general, including the effect of surface tension in interface-resolved simulations commonly introduces spurious (or parasitic) currents due to unbalanced forces at the gas-liquid interface [95]. In the context of boiling flows, this issue is often amplified by the velocity jump across the interface [48]. Fig. 20(a) shows the spurious currents (using the CSF) at the interface (left), which can disturb the thermal layer around the bubble through advection (close-up). This amplification leads to erroneous growth rates (Fig. 20(b)). Consequently, selection of the surface tension model is a deciding factor for the simulation success. We therefore take this opportunity to compare the performance of several surface tension models applied to boiling flows using the CDI method to (i) guide future studies, and (ii) select the best-performing model for the following sections.

The spurious flow structures that develop during the simulation are investigated with three different grids and visualized in Fig. 21. Here, the direction and magnitude of the velocity field is shown for time $t = 29$ ms. The first, second, and third rows correspond to the grid resolutions with $\Delta x = 125$ μm, $\Delta x = 62.5$ μm, and $\Delta x = 31.3$ μm, respectively. On the left, the results for $\sigma = 0$ are included as a reference. Both surface tension models (Eqs. (75) and (76)) introduce severe flow structures. The resolution changes the pattern of the spurious flow, where the results obtained with lower resolutions seem to be less affected. Refined grids exhibit larger velocity magnitudes of the spurious currents, which affect the overall growth rate of the bubble (compare with Fig. 20(b)). For all grid resolutions, the energy-based surface tension model performs worse than the CSF. But neither of the models (Eqs. (75), (76)) provide satisfactory results for the study of bubble growth.

In the literature, several variations of both surface tension model classes exist. Seeking a robust model for the present application, we explore selected variations [51–53]. For the energy-based surface tension model, researchers reformulated the expression for the



(a)

(b)

(c)

Fig. 18. Numerical results for 3D bubble growth without gravity for four different grids. The bubble radius over time (a), and the radial temperature distribution at time $t = 51.5$ ms (b) together with the analytical solutions Eqs. (72) and (74) are shown.The error convergence using Eq. (67) of the simulated bubble radius compared to the analytical solution Eq. (72) is shown in panel (c).

surface tension stress such that the forcing term differs on a discrete level, but remains mathematically equivalent. For example, Eq. (76) was reformulated in Ref. [52] as

$$\vec{f}^{*}_{ST,EB} = \frac{6\sigma}{\epsilon}\left(\frac{\partial\Phi}{\partial\phi} - \epsilon^2\nabla^2\phi\right)\nabla\phi = \frac{6\sigma}{\epsilon}\left(\nabla\Phi - \epsilon^2(\nabla^2\phi)\nabla\phi\right). \tag{77}$$

They refer to this formulation as the *conservative* method by rearranging the derivatives. A different strategy is followed by [53], where the surface tension is expressed in terms of the level-set function Ψ (Eq. (10)). Expressing $\vec{f}_{ST,EB}$ in terms of the level-set function yields

$$\vec{f}^{**}_{ST,EB} = \frac{6\sigma}{\epsilon}\left(\phi(1-\phi)(1-2\phi)(1-|\Psi|^2) - \epsilon\phi(1-\phi)\nabla^2\Psi\right). \tag{78}$$

We compare the formulations in Eqs. (76)–(78), obtaining the flow structures visualized in Fig. 22. This figure shows the significant impact of the discrete representation of the surface tension formulation on the spurious currents. While the results obtained through the standard method and the level-set formulation [53] are comparable, a noticeable worsening is observed when using the *conservative* formulation. None of the considered formulations are capable of reducing the spurious flow structures to an acceptable level. Acknowledging the advantageous properties of the energy-based model class for non-phase change problems, we disregard these models for the remainder of this study.

Regarding the CSF model, a promising variation of Eq. (75) was proposed in [51], where the gradient of a sharpened phase indicator ϕ' replaces the term $\nabla\phi$. Therefore, the CSF variation by [51] reads

$$\vec{f}^{*}_{ST,CSF} = \sigma\kappa\nabla\phi', \tag{79}$$

where ϕ' is the sharpened phase indicator calculated from

$$\phi' = \frac{1}{1-\eta}\left[\min\left(\max\left(\phi, \frac{\eta}{2}\right), 1-\frac{\eta}{2}\right) - \frac{\eta}{2}\right], \tag{80}$$

with η being the sharpening factor. The authors of Ref. [51] suggested using $\eta = 0.5$ for VOF methods. We reevaluate this value in the context of the ACDI method. For this purpose, we study the development of spurious currents for $\eta = 0.5$ and $\eta = 0.75$ and compare

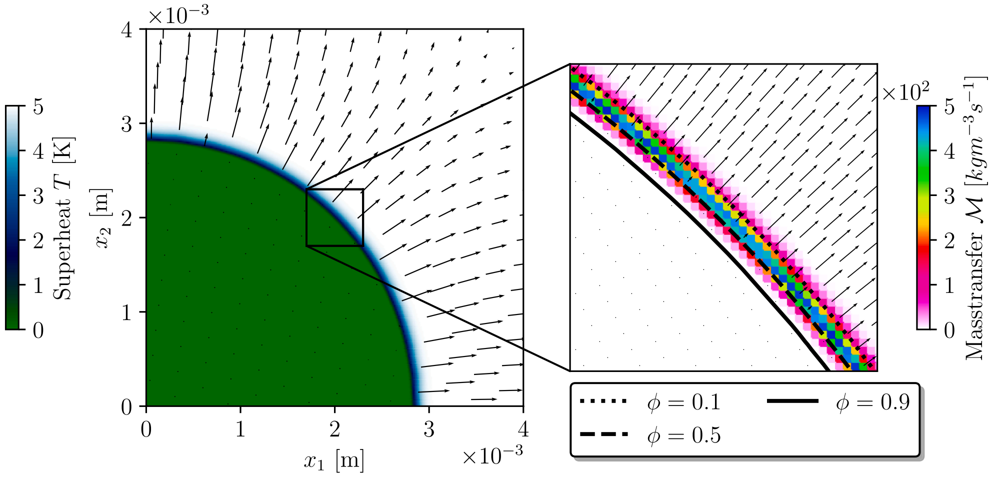
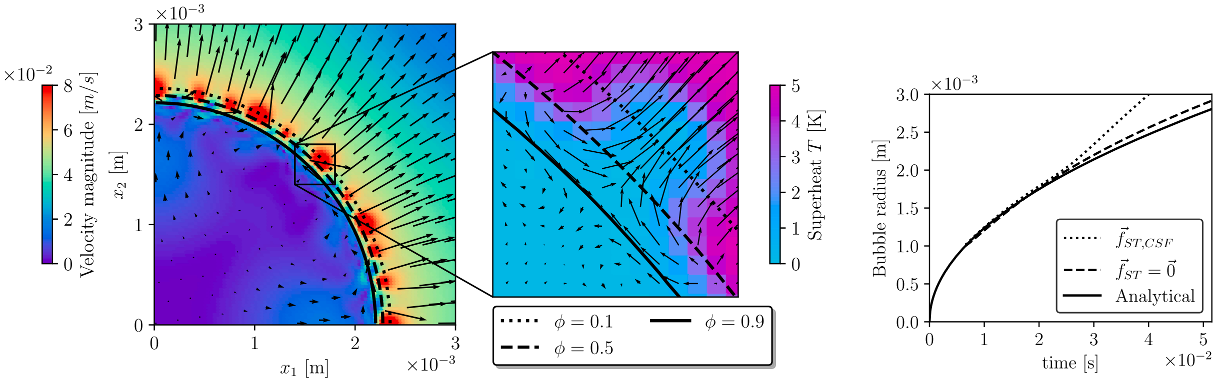


Fig. 19. Results on a slice through the bubble centre. Temperature distribution and qualitative velocity field at time $t = 51.5$ ms (left). The closeup view (right) shows the distribution of the mass transfer $\mathcal{M}$ with respect to the interface location. In the closeup, the black lines correspond to the isocontours of the phase field variable.



(a)

(b)

Fig. 20. Demonstration of the effect of spurious currents on the bubble growth. A slice through the bubble centre (a) visualizes the spurious velocities in the gas and liquid phase at $t = 29$ ms. The close-up shows how the spurious velocities transport the thermal boundary layer around the bubble. Comparison of the growth rate (b) with and without the influence of the spurious currents. Results corresponding to $\Delta x = 31.3$ μm and using the CSF model.

them to the standard CSF model. The effect of η is visualized in Fig. 23, where ϕ' is compared to ϕ. It is shown that for $\eta = 0.5$ and $\epsilon = \Delta x$, the gradient of ϕ' only differs from zero for 3-4 cell edges, whereas for $\eta = 0.75$, it only differs from zero for 2-3 cell edges. As a result, $\nabla\phi'$ is significantly more concentrated around $\phi = 0.5$.

We now investigate the benefit of this increased concentration for spurious currents. The effect of the sharpened model on the bubble growth is studied with two values of η. Fig. 24 summarizes the results obtained using $\Delta x = 62.5$ μm, with the standard CSF results included as a reference. We observe that spurious currents drastically reduce with increased η. In addition, spurious flow patterns observed with the sharpened CSF do not increase over the simulation runtime. We may argue that η should be set as large as possible while still allowing for a stable simulation. From our experiments we found that $\eta = 0.75$ provides a good compromise reducing the spurious currents (see Fig. 24) and exhibiting stable and robust numerical behaviour (given $\epsilon^* = 1$). Hence, the expression in Eq. (79) with $\eta = 0.75$ is implemented for all remaining simulations in this study due to its superior performance in reducing spurious currents.

4.6. Two-dimensional film boiling

Our next step is assessing the framework's capability of simulating complex boiling flows in multi-dimensional spaces. In this section, we consider the simulation of film boiling in two dimensions. In this case, a horizontal, heated plate with a fixed temperature T_{wall} is separated from the liquid phase by a growing gas layer (Fig. 25(a)). The gas layer grows until the interface becomes unstable due to gravity, leading to the detachment of a rising bubble.

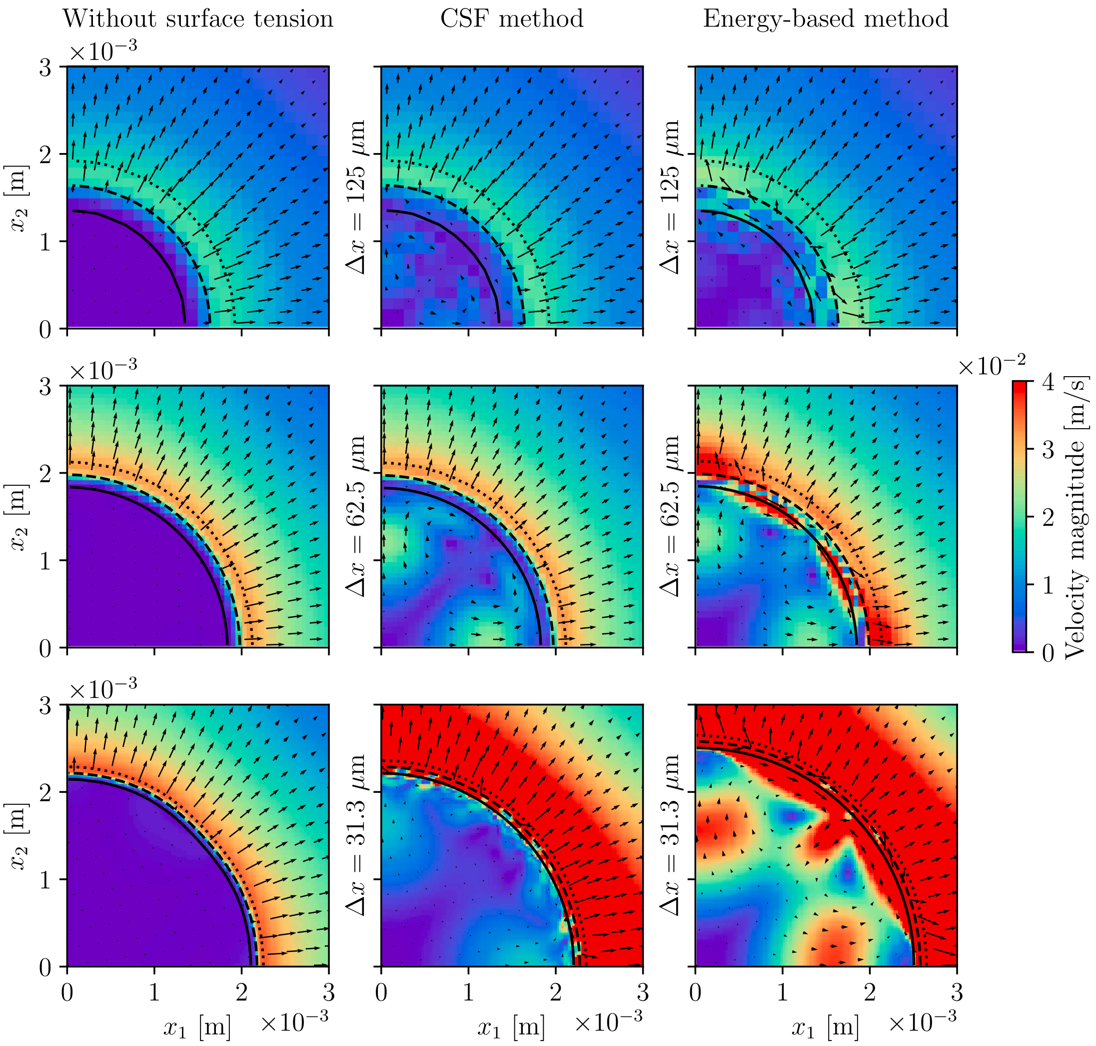


Fig. 21. Visualization of spurious flow structures at $t = 29$ ms. Slice through the bubble centre for different surface tension models and three resolutions. The solid, dashed and dotted lines mark the isocontour lines of the phase field at $\phi = 0.9$, 0.5, and 0.1, respectively.

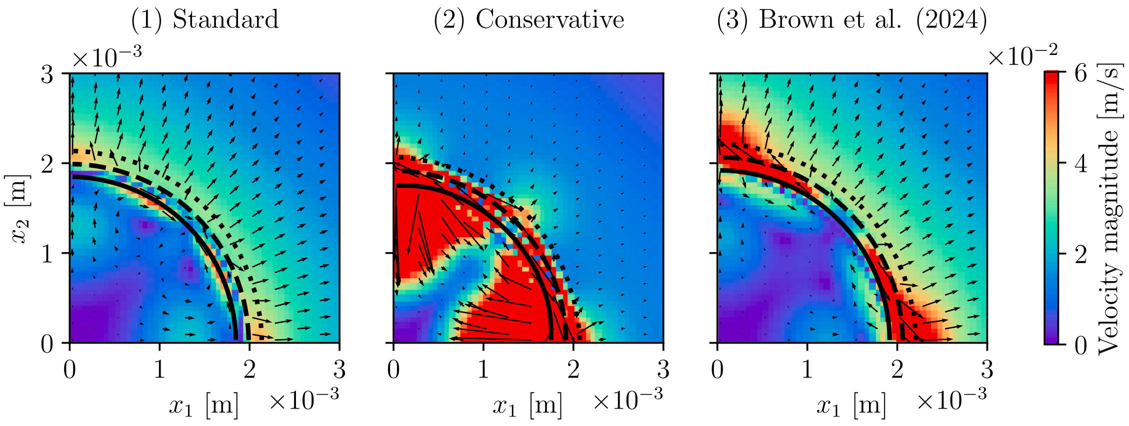
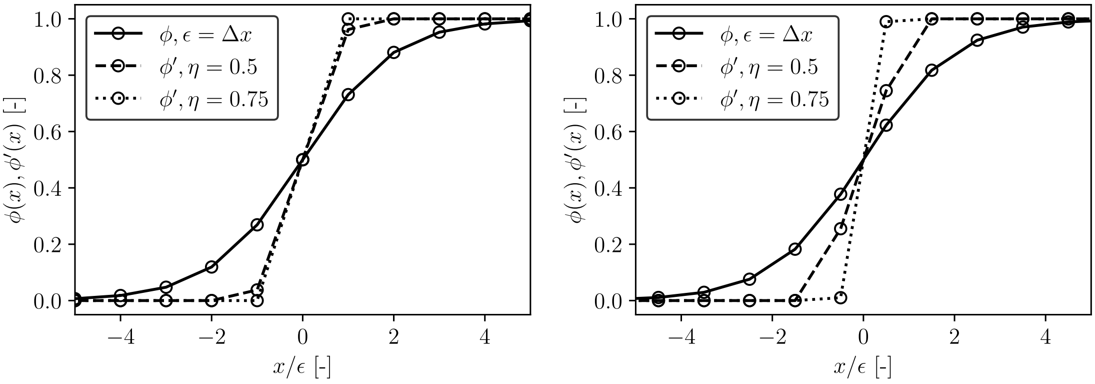


Fig. 22. Velocity field slicing through the bubble at $t = 29$ ms obtained by using $\Delta x = 62.5$ µm. Comparison of different formulations of the energy-based models: (i) standard energy-based formulation using Eq. (76); (ii) the conservative formulation [52] using Eq. (77); (iii) the level-set based formulation [53] using Eq. (78). The solid, dashed and dotted lines mark the isocontour lines of the phase field at $\phi = 0.9$, 0.5, and 0.1, respectively.



(a) $\phi = 0.5$ at cell centre

(b) $\phi = 0.5$ at cell edge

Fig. 23. Phase field variable ϕ and sharpened phase indicator ϕ' plotted over the dimensionless interface distance x/ϵ. Comparison of different sharpening factors η for different locations of $\phi = 0.5$, with circles marking the cell centres.

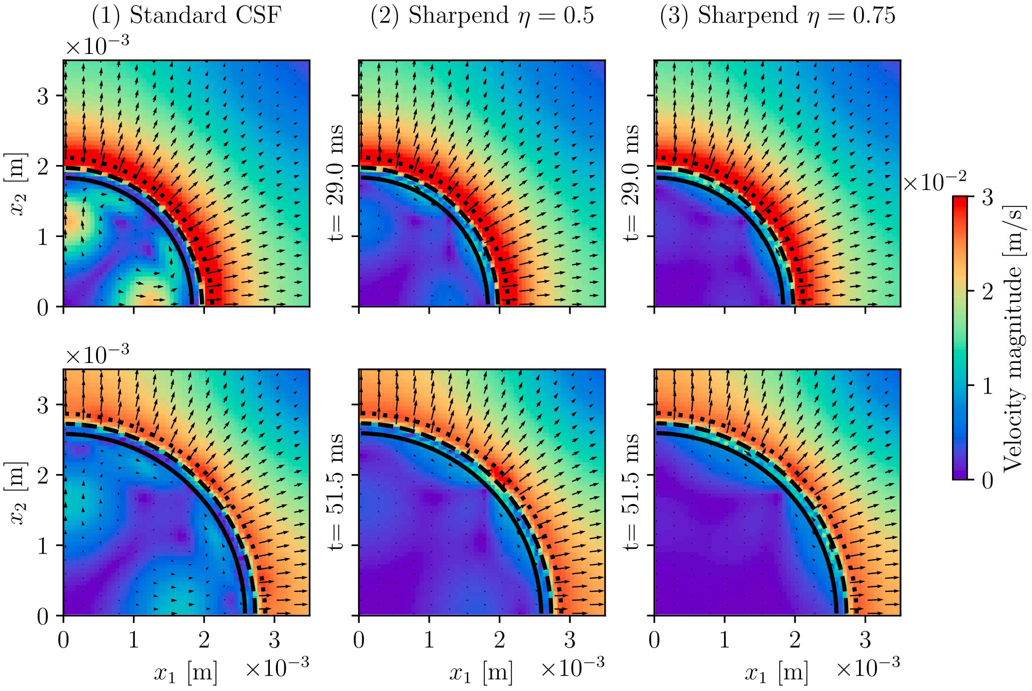


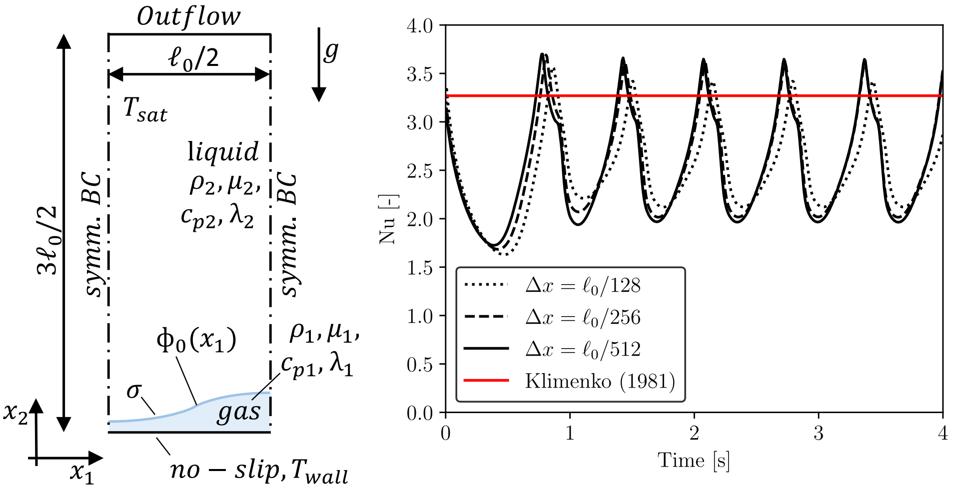
Fig. 24. Comparison of the sharpened CSF method [51], i.e. Eq. (79), using different sharpening factors η with the standard CSF, Eq. (75). The evolution of the spurious currents is visualized by including two time steps, i.e., $t = 29$ ms (top row) and $t = 51.5$ ms (bottom row). Results were obtained by using $\Delta x = 62.5$ μm.

The length scale ℓ_0, characterizing the instability, is the two-dimensional Taylor wavelength [96],

$$\ell_0 = 2\pi\sqrt{\frac{3\sigma}{\|\vec{g}\|(\rho_2 - \rho_1)}}. \tag{81}$$

Note that ℓ_0 is equal to $\pi\sqrt{12}$ times the capillary length $l_c = \sqrt{\sigma\|\vec{g}\|^{-1}(\rho_2 - \rho_1)^{-1}}$. Exploiting the symmetry of the detaching bubble, half the domain is simulated. To reduce the computational cost, we truncate the domain, in the x_2 direction, to a size of $3\ell/2$ and apply an outflow boundary condition at the top. We employ the outflow boundary method proposed in [97], which allows bubbles of a wide range of Weber numbers to exit the domain. Thus, an ordinary homogeneous Dirichlet boundary condition for the pressure at the outflow becomes applicable, as required by the FFT solver [58].

In line with previous studies [7,19,92,98], the physical properties of Fluid$_C$ are utilized (see Table 2). In contrast to these studies, we use a wall superheat $T_{wall} - T_{sat}$ of only 1 K. We chose this modification, as phase field models seem to struggle to allow detach-



(a) (b)

Fig. 25. Schematic sketch of the film boiling simulation (a). Nusselt number over time (b) for three different meshes. The results for Nu correspond to using Fluid$_C$ and $T_{wall} - T_{sat} = 1$ K.

Table 3
Two-dimensional film boiling. Time averaged Nusselt numbers compared to Nu_K from Eq. (84).

Grid spacing Δx	$\ell_0/128$	$\ell_0/256$	$\ell_0/512$
$\overline{Nu}$	2.538	2.583	2.562
$\overline{Nu}/Nu_K - 1$	−22.4 %	−21.0 %	−21.7 %

ments of vapour bubbles with these fluid properties in two dimensions [19]. The initial conditions [92] for the phase field ϕ are set by placing the centre of the interface ($\phi = 0.5$) along the path described by the function ϕ_0 according to

$$\phi_0 = \frac{\ell_0}{32} + \frac{\ell_0}{128} \cos\left(\frac{2\pi x_1}{\ell_0}\right), \tag{82}$$

to trigger an early detachment of a bubble with this first instability. The initial temperature field decreases linearly from $T(x_2 = 0) = T_{wall} - T_{sat}$ to $T(x_2 = \phi_0) = 0$ in the x_2 direction. After startup, the simulations are run until $t = 4$ s to capture approximately six bubble detachment cycles. For the discretization of the computational domain, we use three (uniform) grids with $\Delta x = \ell_0/128$, $\Delta x = \ell_0/256$, and $\Delta x = \ell_0/512$.

The simulations are evaluated considering the temporal evolution of the Nusselt number Nu, which quantifies the heat transfer. To obtain the instantaneous Nusselt number, we compute

$$Nu = \frac{l_c}{(T_{wall} - T_{sat})} \frac{\int_{\partial\Omega_{wall}} \left.\frac{\partial T}{\partial x_2}\right|_{x_2=0} dx_1}{\int_{\partial\Omega_{wall}} dx_1}. \tag{83}$$

Here, the spatially averaged temperature gradient at the bottom domain boundary is non-dimensionalized by the wall superheat and the capillary length scale l_c. To compare the results to the literature, in particular the well-established empirical correlation by Klimenko [96], we compute the time-averaged Nusselt number $\overline{Nu}$. For the given parameters, this empirical Nusselt number, denoted by Nu_K, is

$$Nu_K = 0.19\mathrm{Gr}^{\frac{1}{3}}\mathrm{Pr}^{\frac{1}{3}} f, \text{with} \quad \mathrm{Gr} = \frac{l_c \|\vec{g}\| \rho_1^2}{\mu_1^2}\left(\frac{\rho_2}{\rho_1} - 1\right), \quad f = 0.89\zeta^{\frac{1}{3}}, \tag{84}$$

where $\mathrm{Pr} = c_{p,1}\mu_1\lambda_1^{-1}$ and $\zeta = c_{p,1}(T_{wall} - T_{sat})L_{sat}^{-1}$.

Fig. 25(b) shows the temporal evolution of the instantaneous Nusselt number for all examined grid resolutions. As reference, the red solid line in Fig. 25(b) represents the value of the empirical correlation Nu_K. Apart from the start-up phase, the two finest grids result in the same Nusselt numbers (Fig. 25(b)). The coarse grid exhibits smaller amplitudes in Nu, yet yields the same bubble detachment frequency as the other two finer grids. For further comparison, we calculate the time-averaged Nusselt numbers $\overline{Nu}$, which are summarized in Table 3, along with the relative deviation from $\overline{Nu}$ to Nu_K. All grid resolutions show a discrepancy from Nu_K of at least 21 % (Table 3).

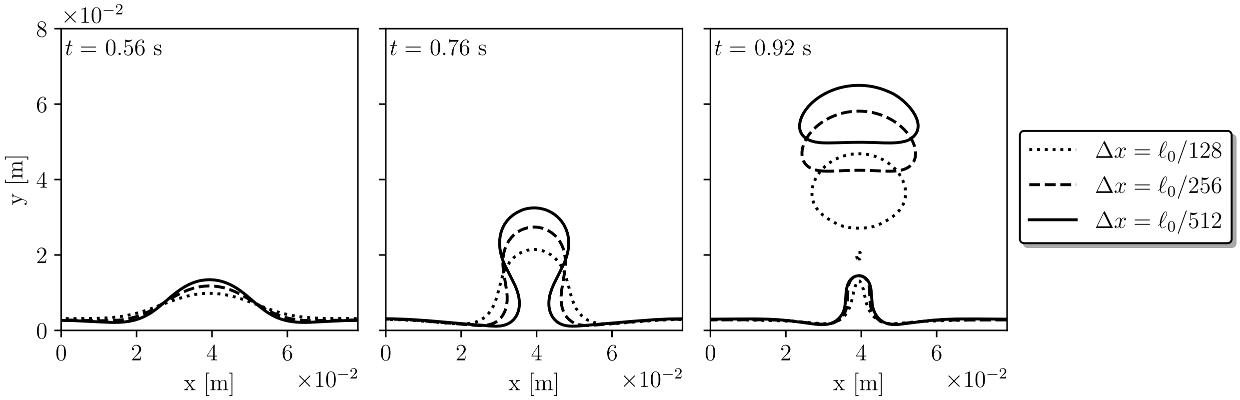
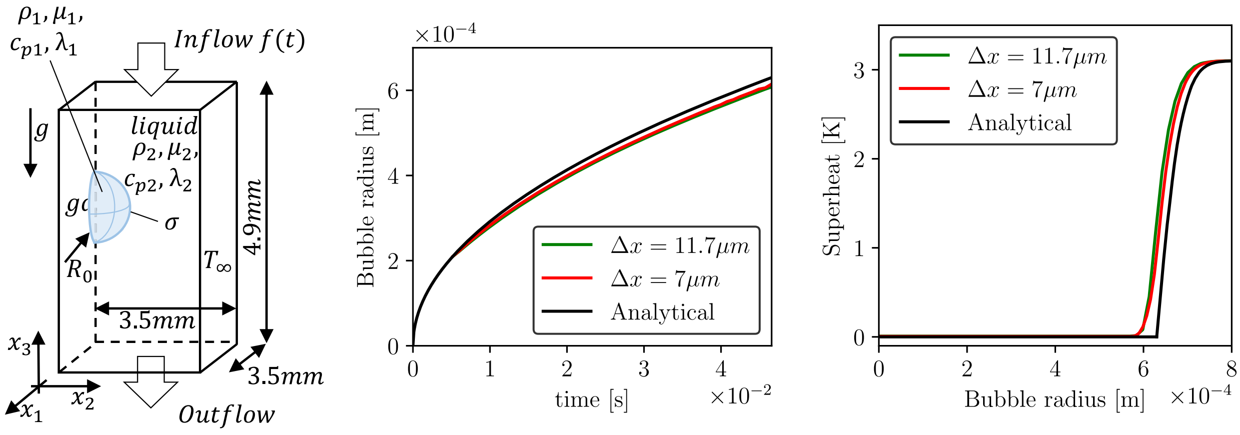


Fig. 26. Two-dimensional film boiling. Effect of resolution on the startup phase. Iso-contour of $\phi = 0.5$ for all grids at three different times.



(a) Setup (b) Bubble size for $\vec{g} = \vec{0}$ (c) Temperatures for $\vec{g} = \vec{0}$

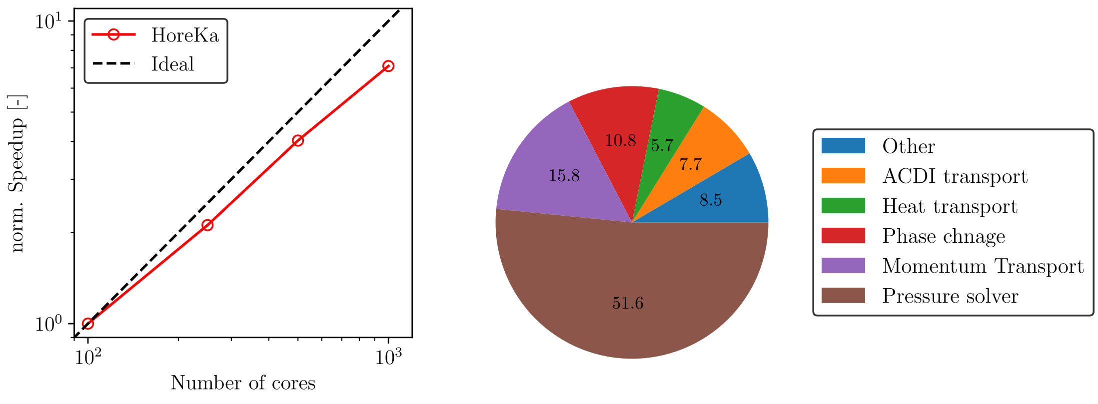
Fig. 27. (a) Schematics of the computational setup including dimensions and boundary conditions. The suitability of grid spacing $\Delta x = 11.6\mu m$ and $\Delta x = 7\mu m$ are confirmed by comparing with the zero gravity solutions of: (b) the bubble size in Eq. (72) and (c) temperature in Eq. (74).

Fig. 26 depicts the differences in results during the start-up phase at three different time steps. For the coarser grids, the departure of the first bubble is delayed (Fig. 26). The shapes of the detached bubbles are similar for the two finer grids, whereas that of the coarsest grid is significantly different. The difference in the bubble departure time, for the two finer grids, reduces in the remainder of the simulations, as indicated by the evolution of the Nusselt number in Fig. 25(b).

4.7. Bubble growth subjected to gravity

As a final benchmark, we consider a three-dimensional growing bubble in a superheated liquid, subjected to gravity. This situation was experimentally investigated for various fluids at different degrees of superheat in [99]. We compare our numerical predictions to the experimental results obtained for Ethanol (see Table 2) at 3.1 K superheat. Several numerical studies [14,49,61,73,100] have used the same experimental case to validate their implementations. This flow is symmetric about the vertical axis, allowing for a reduced computational cost by simulating a quarter of the domain. This situation is presented in Fig. 27(a), with the placement of the initial bubble of radius R_0 on one edge of the computational domain. Furthermore, the symmetry boundary condition is applied on the domain faces intersecting the bubble, while the free-slip boundary condition is imposed on the opposite faces. The coordinate system moves together with the rising bubble, allowing for the use of a significantly smaller domain (see [73]). This is enforced by dynamically adjusting the inflow velocity to keep the rising bubble at a constant location within the moving coordinate system. The set of boundary conditions is completed by using an outflow boundary at the bottom of the domain. For the validation, we adopt two uniform grids, with the grid spacings $\Delta x = 11.6$ µm and $\Delta x = 7$ µm. For all simulations, we use an initial bubble radius of $R_0 = 210$ µm, which is a common choice [14,49,61,73,100].

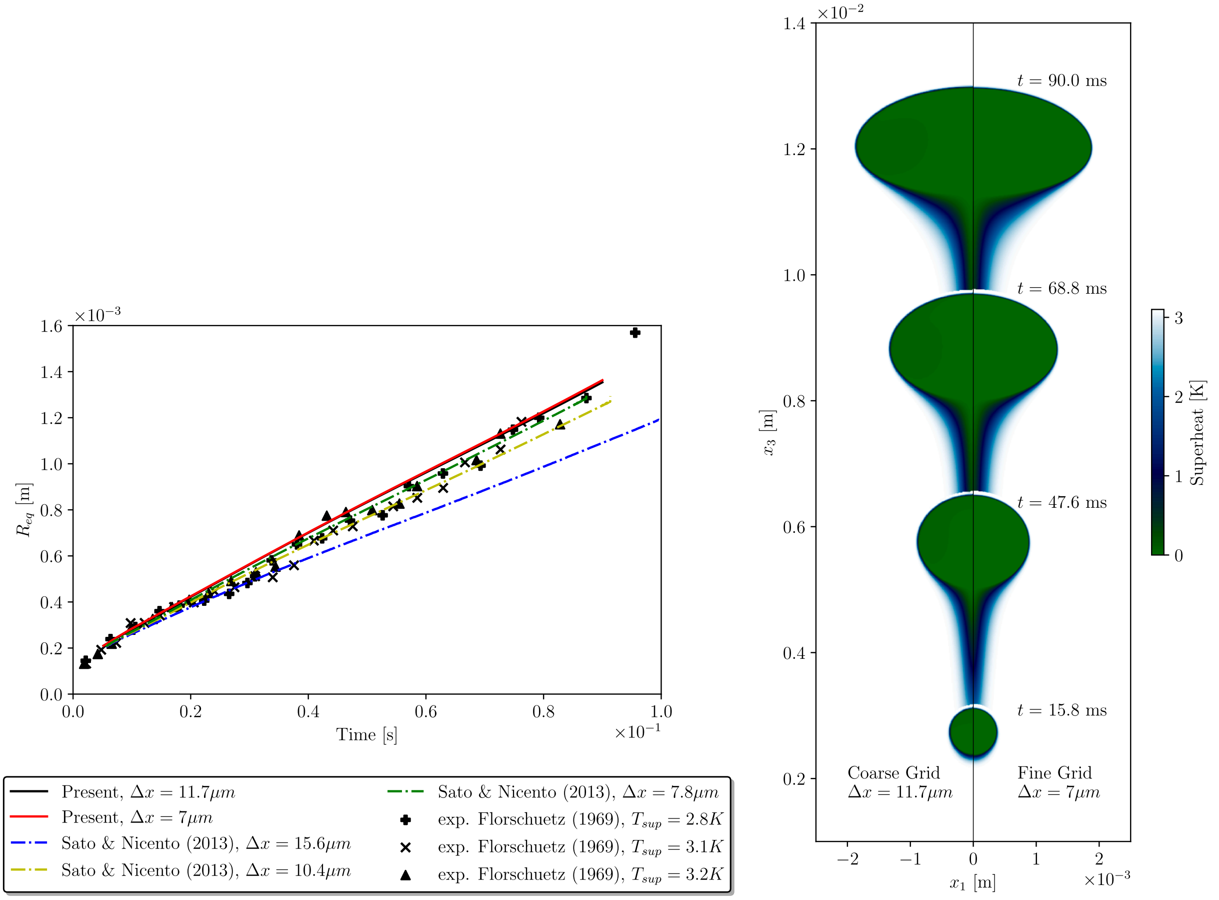
We first perform simulations with zero gravity to confirm the grid's capability to resolve the temperature field. The procedure described in Section 4.5 gives a bubble growth rate and temperature distribution with growth constant $\beta \approx 5.4969$. As seen in Fig. 27(b) and (c), similar results are obtained for both of the grid spacings, when gravity is neglected. The simulation shows good agreement



(a) Scaling evaluation

(b) Profiler analysis using 10^3 cores

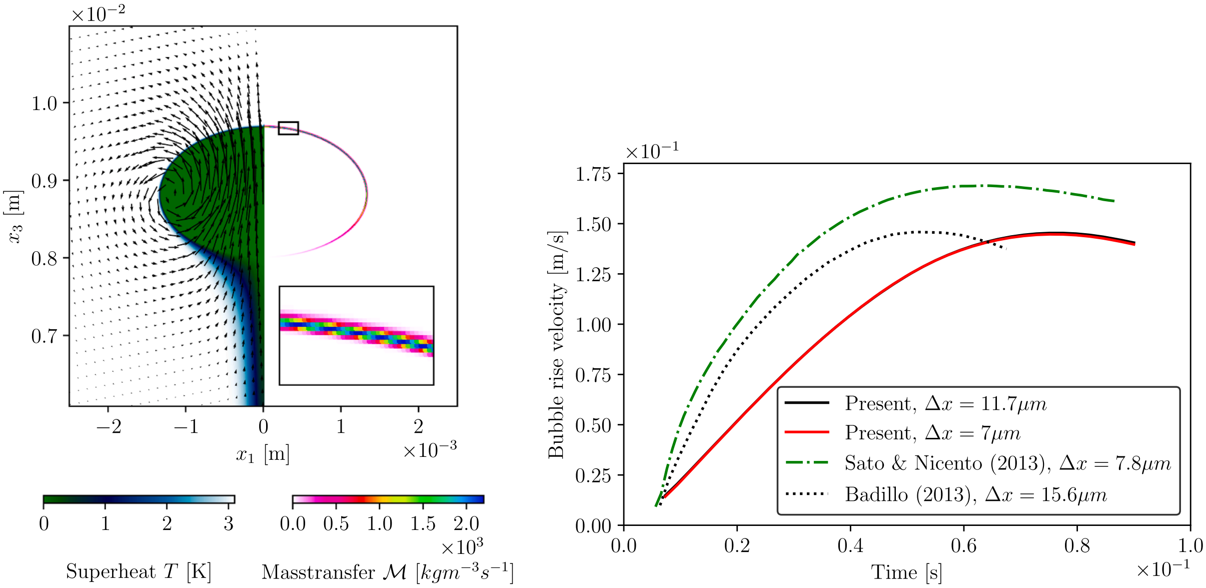
Fig. 28. Analysis of current implementation regarding (a) the scaling and (b) the profiler analysis.



(a) Evolution of bubble size over time

(b) Temperature distributions

Fig. 29. Bubble growth subjected to gravity. (a) Comparison of the time evolution of the bubble size against the experiments in [99] and the numerical studies of [49]. (b) Visualization of the temperature distribution at four selected time steps for both grid resolutions considered.



(a) Bubble at $t = 68.8ms$

(b) Bubble rise velocity

Fig. 30. (a) Close up view of the bubble shows the mass transfer density, velocity field, and temperature distribution for a selected time. (b) The bubble rise velocity is compared to the results by [14,49].

with the analytical solution for the bubble size, although the growth rate is slightly under-predicted (Fig. 27(b)). This is also reflected in the temperature profiles in Fig. 27(c), where the finer grid spacing shows minor improvements.

Before simulating the bubble growth subjected to gravity, the current implementation is evaluated regarding its performance in a high-performance computing (HPC) environment. The implementation uses Fortran95 and is based on the open-source codes FluTAS [60] and CaNS [58]. For the scaling test, the code was executed on the CPU nodes of the HoreKa cluster at the Karlsruhe Institute of Technology (KIT) using the Intel Xeon Platinum 8368 hardware. Considering the grid with $\Delta x = 7$ μm and four different numbers of cores ranging from 10^2 to 10^3 CPUs, the average time required for advancing the simulation by one time step was measured. The resulting normalized speed-up is visualized in Fig. 28(a). Here, good scaling properties were observed for the considered range of CPU cores. In Fig. 28(b), the proportion of time spent to complete each sub-task is visualized when using 10^3 CPUs.

In the next step we add gravity, $\|\vec{g}\| = 9.81$ m s^{-2}. Again, we use the analytical zero-gravity solution (Eqs. (72) and (74)) as an initial condition. This choice is justified by the assumption that buoyancy effects are negligible for very small bubbles, during the short startup phase. Using Eq. (72) and $R_0 = 210$ μm, this translates, again, into a physical time $t_0 = 5.17$ ms at the start of the simulation. For both grids, the simulation proceeds until the physical time $t = 90$ ms is reached.

To evaluate the results, we begin by comparing the simulated bubble size to the literature data. For that purpose, an equivalent bubble radius R_{eq} is utilized [99]. The equivalent radius is calculated as

$$R_{eq} = \frac{1}{4}(D_{12} + D_3), \tag{85}$$

where D_{12} is the bubble diameter in the x_1x_2-plane and D_3 is the one measured in the x_3 direction. Fig. 29(a) shows the evolution of R_{eq} over time, together with the experimental data by [99] for $T_\infty - T_{sat} = 2.8$ K, 3.1 K, and 3.2 K. Additionally, the numerical results from [49] are included for reference, using various grid spacings. As in the zero-gravity simulations, there is a weak dependence on grid spacing. For both simulations, the equivalent bubble radius is within the range of data points observed in the experiments, although more of the measurement points are scattered below the values predicted by the simulations. In comparison, our results are higher than those of [49] for all resolutions. However, our results align with [49] for the smallest grid spacing.

Fig. 29(b) displays the temperature distribution in the plane cutting through the bubble at four times for both considered grid resolutions (left and right half of the figure). We correct the data by accounting for the moving coordinate system such that all bubbles are displayed in the laboratory coordinate system. Fig. 29(b) shows how the bubble grows and rises, where green represents areas in which the fluid is at saturation temperature and white areas represent those at which $T_\infty - T_{sat} = 3.1$ K. The temperature distribution is characterized by a thin thermal layer at the top of the bubble and cooled liquid trailing behind the rising bubble. A clear separation between liquid and gas is observed, as indicated by the saturation temperature. Both grids result in similar temperature distributions and interface locations. For the fine grid (right), the wake of cooled liquid trailing behind the bubble seems to be less pronounced compared to the coarse grid (left).

To gain further insights, the bubble is examined in more detail at $t = 68.8$ ms. A close-up view is presented in Fig. 30(a), where the temperature distribution and the velocity vector field are shown on the left half. On the right, we display the distribution of the mass transfer density $\mathcal{M}$, along with a magnified view of a section in the top part of the bubble. As expected, the mass transfer is concentrated on the top of the bubble. At the bottom, almost no mass transfer occurs, due to the cooled liquid wake.

Another quantity of interest is the bubble rise velocity, which is commonly reported in the literature. The velocities are compared to those by [14] and [49] in Fig. 30(b). The simulations using grid spacing $\Delta x = 7$ μm and 11.6 μm, yield similar rise velocities. Between the two grids, a slight difference of 0.6 % is observed, where the velocities reach their peak. Compared to [14] and [49], our results exhibit a considerably slower bubble acceleration. However, the peak values are in agreement with [14], although they occur at different times. The velocity predicted in [49] shows a higher acceleration and a larger maximum. A possible explanation for the differences might be that the domain size and boundary conditions differ throughout the literature (e.g. smaller domain in [14] or different boundary conditions in [73]). A larger database of simulation results is required to claim evidence for this explanation. The experiment in [99] does not report this quantity.

Generally, quasi grid-independent results are achieved using $\Delta x = 11.6$ μm, which is comparable to numerical studies using a sharp-interface approach [49,61,100]. The study by Giustini and Issa [73] used $\Delta x = 3.12$ μm, which was arguably not necessary for the present study.

5. Conclusion

In this study, we present a framework to simulate boiling flows using the *Accurate Conservative Diffuse Interface* method for tracking the liquid-gas interface. The main objectives of our implementation are efficiency and consistency. We successfully develop and implement models that are computationally efficient, while consistently formulating the set of equations for the calculation of boiling flows.

To achieve consistency, we derive the governing equations of the mixture and identify the contributions of the regularization terms. As an outcome, we recognize the existence of additional contributions in the momentum balance equation that are often neglected. We consider them through an alternative formulation based on developments intended for Cahn-Hillard models. With this approach, we introduce a novel coupling of the momentum equation with the regularization terms, which significantly improves the accuracy of the results. Keeping all regularization terms is crucial for an accurate prediction of the interfacial pressure jump, especially for unsteady phase change. The use of this modified set of equations results in significant improvements, also when simulating bubble dynamics in the absence of phase change. From a more fundamental viewpoint, we demonstrate that the regularization terms can account for velocity differences among the two phases within the diffuse interface, and that neglecting these has a negative impact on the pressure jump at the interface. Thus, in contrast to previously proposed ACDI mixture equations, we conclude that including all regularization terms is mandatory to obtain accurate results in boiling scenarios.

To the best of our knowledge, our work is the first to implement this set of equations in the context of boiling flows in order to resolve complex, three-dimensional problems. These equations, based on the Accurate Conservative Diffuse Interface formulation, allows us to use central differencing schemes, which improves the scalability and efficiency.

Furthermore, we also enhance efficiency by utilizing a FFT-based solver for the pressure Poisson equation. We implement and test our alternative FFT-MPDJ scheme, which addresses the well-known shortcomings of existing solvers. Specifically, all simulation results demonstrate that our methodology: (i) avoids spurious pressure oscillations at the interface; (ii) provides a high accuracy in calculating the interfacial pressure jumps; and (iii) attains a faster adaptation to the flow in the startup phase.

We present a simple and efficient *kinetic* phase change model for the ACDI method. For this phase change model, we analyze the time-step constraint and introduce a dynamic adjustment of the regularization speed. The phase change model is tested against several boiling scenarios with a variety of fluid properties. We find the model to be free of problem-dependent parameters, and capable of converging to the analytical solution across all benchmarks. However, compared to Sharp Interface methods, the model required a slightly higher resolution to achieve convergence. Overall, we conclude that the phase change model is efficient, versatile, and, when combined with the dynamic regularization speed, robust.

Modelling surface tension is often connected with challenges for boiling flow simulations, mainly because of spurious currents around the phase interface. To address this issue, we investigate the performance of different energy-based and CSF models, using a bubble growth benchmark (Section 4.5). Although energy-based surface tension formulations are known to have advantages for problems when excluding phase change, we find them to cause increased spurious currents for boiling flows. It is concluded that a *sharpened* CSF model returns the lowest intensity of spurious currents.

Finally, we successfully simulate the dynamics of a single bubble subjected to phase change and gravity (Section 4.7). We validate that our framework is capable of accurately calculating real world boiling flows. We find good agreement between our results and experimental data for the bubble growth rate. The results are also consistent with other numerical studies using Sharp Interface methods at a comparable grid resolution.

For further investigation of boiling heat-transfer phenomena, accurate interface-resolved simulations play an important role. The results obtained in this study show that our developed framework is a suitable tool for that purpose. As such, we intend to apply the presented methodology to large-scale boiling flows for which the above-demonstrated properties are indispensable. In this context, additional extensions to this framework are planned to enable the numerical investigation of conjugate heat transfer phenomena.

CRediT authorship contribution statement

Lorenz Weber: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization; **Aritra Mukherjee:** Writing – review & editing, Supervision, Software, Methodology, Concep-

tualization; **Andreas G. Class:** Writing – review & editing, Supervision, Resources, Project administration, Funding acquisition, Conceptualization; **Luca Brandt:** Writing – review & editing, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization.

Data availability

Data will be made available on request.

Declaration of interests

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

Acknowledgements

The authors acknowledge support by the state of Baden-Württemberg through bwHPC. Furthermore, we are thankful for the financial support provided by the German Academic Exchange Service (DAAD) for partially funding research visits within this study. The support of the Framatome Professional School (FPS) is acknowledged. In addition, the authors are grateful for the Research Travel Grant through the Karlsruhe House of Young Scientists (KHYS) supporting the collaboration. We thank J. Yanez and C. Tazzeo for the suggestions to improve the quality of the manuscript.

References

[1] S.S. Jain, Accurate conservative phase-field method for simulation of two-phase flows, J. Comput. Phys. 469 (2022) 111529. https://doi.org/10.1016/j.jcp.2022.111529
[2] I. Mudawar, Two-phase microchannel heat sinks: theory, applications, and limitations, J. Electron. Packag. 133 (4) (2011). https://doi.org/10.1115/1.4005300
[3] C.R. Kharangate, I. Mudawar, Review of computational studies on boiling and condensation, Int. J. Heat Mass Transfer 108 (2017) 1164–1196. https://doi.org/10.1016/j.ijheatmasstransfer.2016.12.065
[4] L. Bures, M. Bucci, Y. Sato, M. Bucci, A coarse grid approach for single bubble boiling simulations with the volume of fluid method, Comput. Fluids 271 (2024) 106182. https://doi.org/10.1016/j.compfluid.2024.106182
[5] D. Juric, G. Tryggvason, Computations of boiling flows, Int. J. Multiphase Flow 24 (3) (1998) 387–410. https://doi.org/10.1016/S0301-9322(97)00050-5
[6] G. Son, V.K. Dhir, Numerical simulation of film boiling near critical pressures with a level set method, J. Heat Transfer 120 (1) (1998) 183–192. https://doi.org/10.1115/1.2830042
[7] S.W. Welch, J. Wilson, A volume of fluid based method for fluid flows with phase change, J. Comput. Phys. 160 (2) (2000) 662–682. https://doi.org/10.1006/jcph.2000.6481
[8] S. Mirjalili, C.B. Ivey, A. Mani, Comparison between the diffuse interface and volume of fluid methods for simulating two-phase flows, Int. J. Multiphase Flow 116 (2019) 221–238. https://doi.org/10.1016/j.ijmultiphaseflow.2019.04.019
[9] D. Jamet, O. Lebaigue, N. Coutris, J. Delhaye, The second gradient method for the direct numerical simulation of liquid-vapor flows with phase change, J. Comput. Phys. 169 (2) (2001) 624–651. https://doi.org/10.1006/jcph.2000.6692
[10] Y. Sun, C. Beckermann, Diffuse interface modeling of two-phase flows based on averaging: mass and momentum equations 198 (2004) 281–308. https://doi.org/10.1016/j.physd.2004.09.003
[11] A. Onuki, Dynamic van der Waals theory, Phys. Rev. E 75 (2007). https://doi.org/10.1103/PhysRevE.75.036304
[12] T. Laurila, A. Carlson, M. Do-Quang, T. Ala-Nissila, G. Amberg, Thermohydrodynamics of boiling in a van der Waals fluid, Phys. Rev. E 85 (2012) 026320. https://doi.org/10.1103/PhysRevE.85.026320
[13] A. Badillo, Quantitative phase-field modeling for boiling phenomena, Phys. Rev. E 86 (2012) 041603. https://doi.org/10.1103/PhysRevE.86.041603
[14] A. Badillo, Technical Report, Heat Transfer and Thermal Engineering, 2013. https://doi.org/10.1115/IMECE2013-65925
[15] S.M. Allen, J.W. Cahn, Mechanisms of phase transformations within the miscibility gap of Fe-rich Fe-Al alloys, Acta Metall. 24 (5) (1976) 425–437. https://doi.org/10.1016/0001-6160(76)90063-8
[16] J.W. Cahn, J.E. Hilliard, Free energy of a nonuniform system. I. Interfacial free energy, J. Chem. Phys. 28 (2) (1958) 258–267. https://doi.org/10.1063/1.1744102
[17] R. Jafari, T. Okutucu-Özyurt, Phase-field modeling of vapor bubble growth in a microchannel, J. Comput. Multiphase Flows 7 (3) (2015) 143–158. https://doi.org/10.1260/1757-482X.7.3.143
[18] Y. Wang, J. Cai, Numerical investigation on bubble evolution during nucleate boiling using diffuse interface method, Int. J. Heat Mass Transfer 112 (2017) 28–38. https://doi.org/10.1016/j.ijheatmasstransfer.2017.04.109
[19] Z. Wang, X. Zheng, C. Chryssostomidis, G.E. Karniadakis, A phase-field method for boiling heat transfer, J. Comput. Phys. 435 (2021) 110239. https://doi.org/10.1016/j.jcp.2021.110239
[20] Y. Sun, C. Beckermann, Sharp interface tracking using the phase-field equation, J. Comput. Phys. 220 (2) (2007) 626–653. https://doi.org/10.1016/j.jcp.2006.05.025
[21] P.-H. Chiu, Y.T. Lin, A conservative phase field method for solving incompressible two-phase flows, J. Comput. Phys. 230 (1) (2011) 185–204. https://doi.org/10.1016/j.jcp.2010.09.021
[22] A. Tamura, K. Katono, Development of a phase-field method for phase change simulations using a conservative Allen-Cahn equation, J. Nucl. Eng. Radiat. Sci. 8 (2) (2021). https://doi.org/10.1115/1.4050209
[23] R. Haghani-Hassan-Abadi, A. Fakhari, M.H. Rahimian, Phase-change modeling based on a novel conservative phase-field method, J. Comput. Phys. 432 (2021) 110111. https://doi.org/10.1016/j.jcp.2021.110111
[24] S. Mirjalili, C.B. Ivey, A. Mani, A conservative diffuse interface method for two-phase flows with provable boundedness properties, J. Comput. Phys. 401 (2020) 109006. https://doi.org/10.1016/j.jcp.2019.109006
[25] S.S. Jain, A. Mani, P. Moin, A conservative diffuse-interface method for compressible two-phase flows, J. Comput. Phys. 418 (2020) 109606. https://doi.org/10.1016/j.jcp.2020.109606
[26] P. Moin, R. Verzicco, On the suitability of second-order accurate discretizations for turbulent flow simulations, Vortical Struct. Wall Turbul. 55 (2016) 242–245. https://doi.org/10.1016/j.euromechflu.2015.10.006
[27] L. Brown, S. Jain, P. Moin, A phase field model for simulating the freezing of supercooled liquid droplets, SAE Int. J. Adv. Current Pract. Mobility 6 (3) (2023) 1150–1157. https://doi.org/10.4271/2023-01-1454

[28] S. Mirjalili, S.S. Jain, A. Mani, A computational model for interfacial heat and mass transfer in two-phase flows using a phase field method, Int. J. Heat Mass Transfer 197 (2022) 123326. https://doi.org/10.1016/j.ijheatmasstransfer.2022.123326

[29] S.Z. Salimi, A. Mukherjee, M. Pelanti, L. Brandt, A low Mach number diffuse-interface model for multicomponent two-phase flows with phase change, J. Comput. Phys. 523 (2025) 113683. https://doi.org/10.1016/j.jcp.2024.113683

[30] N. Scapin, A. Shahmardi, W.H.R. Chan, S.S. Jain, S. Mirjalili, M. Pelanti, L. Brandt, A Mass-Conserving Pressure-Based Method for Two-Phase Flows with Phase Change, Technical Report, Stanford University, Center for Turbulence Research, 2022.

[31] A. Roccon, Boiling heat transfer by phase-field method, 2024, pp. 1–16. https://doi.org/10.1007/s00707-024-04122-7

[32] M.F. Eikelder, K.G. V.D. Zee, D. Schillinger, Thermodynamically consistent diffuse-interface mixture models of incompressible multicomponent fluids, J. Fluid Mech. 990 (2024). https://doi.org/10.1017/jfm.2024.502

[33] S. Mirjalili, A. Mani, Consistent, energy-conserving momentum transport for simulations of two-phase flows using the phase field equations, J. Comput. Phys. 426 (2021) 109918. https://doi.org/10.1016/j.jcp.2020.109918

[34] M.E. Gurtin, D. Polignone, J. Viñals, Two-phase binary fluids and immiscible fluids described by an order parameter, Math. Models Methods Appl. Sci. 06 (06) (1996) 815–831. https://doi.org/10.1142/S0218202596000341

[35] J. Lowengrub, L. Truskinovsky, Quasi-incompressible Cahn-Hilliard fluids and topological transitions, Proc. R. Soc. London. Ser. A: Math. Phys. Eng. Sci. 454 (1978) 2617–2654. https://doi.org/10.1098/rspa.1998.0273

[36] F. Boyer, A theoretical and numerical model for the study of incompressible mixture flows, Comput. Fluids 31 (1) (2002) 41–68. https://doi.org/10.1016/S0045-7930(00)00031-1

[37] 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 (03) (2012) 1150013. https://doi.org/10.1142/S0218202511500138

[38] O. Souček, V. Průša, J. Málek, K., On the natural structure of thermodynamic potentials and fluxes in the theory of chemically non-reacting binary mixtures, Acta Mech. 225 (2014) 3157–3186.

[39] M. Řehoř, Diffuse Interface Models in Theory of Interacting Continua, Technical Report, Mathematical Institute of Charles University and Interdisciplinary Center for Scientific Computing at Heidelberg University, 2018. Ph.D. thesis, https://doi.org/10.11588/heidok.00025676

[40] M.F.P. Eikelder, K.G. V.D. Zee, I. Akkerman, D. Schillinger, A unified framework for Navier-Stokes Cahn-Hilliard models with non-matching densities, Math. Models Methods Appl. Sci. 33 (01) (2023) 175–221. https://doi.org/10.1142/S0218202523500069

[41] R. Scardovelli, S. Zaleski, Direct numerical simulation of free-surface and interfacial flow, Annu. Rev. Fluid Mech. 31 (1999) 567–603. https://doi.org/10.1146/annurev.fluid.31.1.567

[42] S. Mirjalili, M.A. Khanwale, A. Mani, Assessment of an energy-based surface tension model for simulation of two-phase flows using second-order phase field methods, J. Comput. Phys. 474 (2023) 111795. https://doi.org/10.1016/j.jcp.2022.111795

[43] J. Brackbill, D. Kothe, C. Zemach, A continuum method for modeling surface tension, J. Comput. Phys. 100 (2) (1992) 335–354. https://doi.org/10.1016/0021-9991(92)90240-Y

[44] D. Jacqmin, Calculation of two-phase Navier-Stokes flows using phase-field modeling, J. Comput. Phys. 155 (1) (1999) 96–127. https://doi.org/10.1006/jcph.1999.6332

[45] J. Kim, A continuous surface tension force formulation for diffuse-interface models, J. Comput. Phys. 204 (2) (2005) 784–804. https://doi.org/10.1016/j.jcp.2004.10.032

[46] Q. He, N. Kasagi, Phase-field simulation of small capillary-number two-phase flow in a microtube, Fluid Dyn. Res. 40 (7-8) (2008) 497. https://doi.org/10.1016/j.fluiddyn.2008.01.002

[47] A. Roccon, F. Zonta, A. Soldati, Phase-field modeling of complex interface dynamics in drop-laden turbulence, Phys. Rev. Fluids 8 (2023). https://doi.org/10.1103/PhysRevFluids.8.090501

[48] S. Tanguy, M. Sagan, B. Lalanne, F. Couderc, C. Colin, Benchmarks and numerical methods for the simulation of boiling flows, J. Comput. Phys. 264 (2014) 1–22. https://doi.org/10.1016/j.jcp.2014.01.014

[49] Y. Sato, B. Ničeno, A sharp-interface phase change model for a mass-conservative interface tracking method, J. Comput. Phys. 249 (2013) 127–161. https://doi.org/10.1016/j.jcp.2013.04.035

[50] H.G. Lee, J. Kim, Regularized Dirac delta functions for phase field models, Int. J. Numer. Methods Eng. 91 (3) (2012) 269–288. https://doi.org/10.1002/nme.4262

[51] A.Q. Raeini, M.J. Blunt, B. Bijeljic, Modelling two-phase flow in porous media at the pore scale using the volume-of-fluid method, J. Comput. Phys. 213 (2) (2012) 5653–5668. https://doi.org/10.1016/j.jcp.2012.04.011

[52] Z. Huang, G. Lin, A.M. Ardekani, Consistent and conservative scheme for incompressible two-phase flows using the conservative Allen-Cahn model, J. Comput. Phys. 420 (2020) 109718. https://doi.org/10.1016/j.jcp.2020.109718

[53] R.L. Brown, S. Mirjalili, M.A. Khanwale, A. Mani, A Mass-Conserving Contact Line Treatment for Second-Order Conservative Phase Field Methods Based on the Generalized Navier Boundary Condition, arxiv:2412.16843 edition, 2024.

[54] A. Baldellou, X.A. Farré, A. Gorobets, F.X. Miquel, A. Oliva, Llena, Efficient strategies for solving the variable Poisson equation with large contrasts in the coefficients, ECCOMAS Congress, 2022, p. 2022. in: Collection of papers presented at the 8th.

[55] 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 (17) (2012) 5788–5804. https://doi.org/10.1016/j.jcp.2012.04.041

[56] M.S. Dodd, A. Ferrante, A fast pressure-correction method for incompressible two-fluid flows, J. Comput. Phys. 273 (2014) 416–434. https://doi.org/10.1016/j.jcp.2014.05.024

[57] C. Frantzis, D. Grigoriadis, An efficient method for two-fluid incompressible flows appropriate for the immersed boundary method, J. Comput. Phys. 376 (2019) 28–53. https://doi.org/10.1016/j.jcp.2018.09.035

[58] P. Costa, A FFT-based finite-difference solver for massively-parallel direct numerical simulations of turbulent flows, Comput. Math. Appl. 76 (8) (2018) 1853–1862. https://doi.org/10.1016/j.camwa.2018.07.034

[59] N. Scapin, P. Costa, L. Brandt, A volume-of-fluid method for interface-resolved simulations of phase-changing two-fluid flows, J. Comput. Phys. 407 (2020) 109251. https://doi.org/10.1016/j.jcp.2020.109251

[60] M. Crialesi-Esposito, N. Scapin, A.D. Demou, M.E. Rosti, P. Costa, F. Spiga, L. Brandt, Flutas: a GPU-accelerated finite difference code for multiphase flows, Comput. Phys. Commun. 284 (2023) 108602. https://doi.org/10.1016/j.cpc.2022.108602

[61] J. Poblador-Ibanez, N. Valle, B.J. Boersma, A momentum balance correction to the non-conservative one-fluid formulation in boiling flows using volume-of-fluid, J. Comput. Phys. 524 (2025) 113704. https://doi.org/10.1016/j.jcp.2024.113704

[62] P. Cifani, Analysis of a constant-coefficient pressure equation method for fast computations of two-phase flows at high density ratios, J. Comput. Phys. 398 (2019) 108904. https://doi.org/10.1016/j.jcp.2019.108904

[63] A. Mukherjee, S.G. Kandlikar, Numerical simulation of growth of a vapor bubble during flow boiling of water in a microchannel, Microfluid Nanofluidics 1 (2005) 137–145.

[64] F. Gibou, L. Chen, D. Nguyen, S. Banerjee, A level set based sharp interface method for the multiphase incompressible Navier-Stokes equations with phase change, J. Comput. Phys. 222 (2) (2007) 536–555. https://doi.org/10.1016/j.jcp.2006.07.035

[65] R.W. Schrage, I. Chapter, Introduction, Columbia University Press, New York Chichester, West Sussex, New York Chichester, West Sussex, 1953. cited 2025-05-11, https://doi.org/10.7312/schr90162-001

[66] W.H. Lee, A pressure iteration scheme for two-phase flow modeling, T.N. Veziroğlu, Multi-phase Transport: Fundamentals, Reactor Safety, Application, Hemisphere Publishing Corporation. Washington, DC. 1 (1980) 407–431.

[67] I. Tanasawa, Advances in Condensation Heat Transfer, 21, Elsevier, 1991. https://doi.org/10.1016/S0065-2717(08)70334-4

[68] A.S. Rattner, S. Garimella, Simple mechanistically consistent formulation for volume-of-fluid based computations of condensing flows, J. Heat Transfer 136 (7) (2014). https://doi.org/10.1115/1.4026808
[69] Z. Pan, J.A. Weibel, S.V. Garimella, A saturated-interface-volume phase change model for simulating flow boiling, Int. J. Heat Mass Transfer 93 (2016) 945–956. https://doi.org/10.1016/j.ijheatmasstransfer.2015.10.044
[70] M. Eikelder, D. Schillinger, The divergence-free velocity formulation of the consistent Navier-Stokes Cahn-Hilliard model with non-matching densities, divergence-conforming discretization, and benchmarks, J. Comput. Phys. 513 (2024) 113148. https://doi.org/10.1016/j.jcp.2024.113148
[71] S. Dong, An efficient algorithm for incompressible N-phase flows, J. Comput. Phys. 276 (2014) 691–728. https://doi.org/10.1016/j.jcp.2014.08.002
[72] M. Schreter-Fleischhacker, P. Munch, N. Much, M. Kronbichler, W.A. Wall, C. Meier, A consistent diffuse-interface model for two-phase flow problems with rapid evaporation, Adv. Model. Simul. Eng. Sci. 11 (1) (2024) 19.
[73] G. Giustini, R.I. Issa, Modelling of free bubble growth with interface capturing computational fluid dynamics, Exper. Comput. Multiphase Flow 5 (4) (2023) 357–364.
[74] A.K. Sadaghiani, Numerical and experimental studies on flow condensation in hydrophilic microtubes, Appl. Therm. Eng. 197 (2021) 117359. https://doi.org/10.1016/j.applthermaleng.2021.117359
[75] M. Castro, B. Costa, W.S. Don, High order weighted essentially non-oscillatory weno-z schemes for hyperbolic conservation laws, J. Comput. Phys. 230 (5) (2011) 1766–1792. https://doi.org/10.1016/j.jcp.2010.11.028
[76] A. Karma, Phase-field formulation for quantitative modeling of alloy solidification, Phys. Rev. Lett. 87 (2001) 115701. https://doi.org/10.1103/PhysRevLett.87.115701
[77] M. Ohno, K. Matsuura, Quantitative phase-field modeling for dilute alloy solidification involving diffusion in the solid, Phys. Rev. E 79 (2009). https://doi.org/10.1103/PhysRevE.79.031603
[78] Y. Sun, C. Beckermann, Phase-field modeling of bubble growth and flow in a Hele-Shaw cell, Int. J. Heat Mass Transfer 53 (15) (2010) 2969–2978. https://doi.org/10.1016/j.ijheatmasstransfer.2010.03.036
[79] S. Mirjalili, A Novel Diffuse Interface Method for Two Phase Flows and Application in Simulation of Micro Bubble Entrainment, Technical Report, Stanford University, 2019. Ph.D. thesis,
[80] A.J. Chorin, Numerical solution of the Navier-Stokes equations, Math. Comput. 22 (1968) 745–762. https://doi.org/10.1090/S0025-5718-1968-0242392-2
[81] M. Ferronato, Preconditioning for sparse linear systems at the dawn of the 21st century: history, current developments, and future perspectives, Int. Sch. Res. Not. 2012 (1) (2012) 127647. https://doi.org/10.5402/2012/127647
[82] M.F. Trujillo, Reexamining the one-fluid formulation for two-phase flows, Int. J. Multiphase Flow 141 (2021) 103672. https://doi.org/10.1016/j.ijmultiphaseflow.2021.103672
[83] S. Shin, D. Juric, Modeling three-dimensional multiphase flow using a level contour reconstruction method for front tracking without connectivity, J. Comput. Phys. 180 (2) (2002) 427–470. https://doi.org/10.1006/jcph.2002.7086
[84] H. Liu, J. Tang, L. Sun, Z. Mo, G. Xie, An assessment and analysis of phase change models for the simulation of vapor bubble condensation, Int. J. Heat Mass Transfer 157 (2020) 119924. https://doi.org/10.1016/j.ijheatmasstransfer.2020.119924
[85] M. Kang, R.P. Fedkiw, X.D. Liu, A boundary condition capturing method for multiphase incompressible flow, J. Sci. Comput. 15 (2000) 323–360. https://doi.org/10.1023/A:1011178417620
[86] X. Mao, V. Joshi, R. Jaiman, A variational interface-preserving and conservative phase-field method for the surface tension effect in two-phase flows, J. Comput. Phys. 433 (2021) 110166. https://doi.org/10.1016/j.jcp.2021.110166
[87] J. Adelsberger, P. Esser, M. Griebel, S. Gross, M. Klitz, A. Rüttgers, 3D incompressible two-phase flow benchmark computations for rising droplets, in: Proceedings of the 11th World Congress on Computational Mechanics (WCCM XI), the 11th World Congress on Computational Mechanics (WCCM XI), 2014.
[88] S. Gross, A. Reusken, in: Springer Series in Computational Mathematics, Berlin Heidelberg, Springer, 2011.
[89] R. Croce, M. Griebel, M.A. Schweitzer, Numerical simulation of bubble and droplet deformation by a level set approach with surface tension in three dimensions, Int. J. Numer. Methods Fluids 62 (9) (2010) 963–993. https://doi.org/10.1002/fld.2051
[90] Openfoam, 2013. https://openfoam.org/release/2-2-2/.
[91] I. Perez-Raya, S. Kandlikar, Chapter Three - Evaporation on a Planar Interface - Numerical Simulation and Theoretical Analysis of Heat and Mass Transport Processes, 48, Elsevier, 2016. https://doi.org/10.1016/bs.aiht.2016.08.005
[92] J.X.D. Sun, Q. Chen, Modeling of the evaporation and condensation phase-change problems with FLUENT, numerical heat transfer, Part B: Fund. 66 (4) (2014) 326–342. https://doi.org/10.1080/10407790.2014.915681
[93] M. Irfan, M. Muradoglu, A front tracking method for direct numerical simulation of evaporation process in a multiphase system, J. Comput. Phys. 337 (2017) 132–153. https://doi.org/10.1016/j.jcp.2017.02.036
[94] L. Scriven, On the dynamics of phase growth, Chem. Eng. Sci. 10 (1) (1959) 1–13. https://doi.org/10.1016/0009-2509(59)80019-1
[95] S. Popinet, Numerical models of surface tension, Annu. Rev. Fluid Mech. 50 (2018) 49–75. https://doi.org/10.1146/annurev-fluid-122316-045034
[96] V. Klimenko, Film boiling on a horizontal plate - new correlation, Int. J. Heat Mass Transfer 24 (1) (1981) 69–79. https://doi.org/10.1016/0017-9310(81)90094-6
[97] A. Dhruv, A vortex damping outflow forcing for multiphase flows with sharp interfacial jumps, J. Comput. Phys. 511 (2024) 113122. https://doi.org/10.1016/j.jcp.2024.113122
[98] D.Z. Guo, D.L. Sun, Z.Y. Li, Q.T. W., Phase change heat transfer simulation for boiling bubbles arising from a vapor film by the VOSET method, Numer. Heat Transfer, Part A: Appl. 59 (11) (2011) 857–881. https://doi.org/10.1080/10407782.2011.561079
[99] L. Florschuetz, C. Henry, A. Khan, Growth rates of free vapor bubbles in liquids at uniform superheats under normal and zero gravity conditions, Int. J. Heat Mass Transfer 12 (11) (1969) 1465–1489. https://doi.org/10.1016/0017-9310(69)90028-3
[100] L. Bureš, Y. Sato, Direct numerical simulation of evaporation and condensation with the geometric VOF method and a sharp-interface phase-change model, Int. J. Heat Mass Transfer 173 (2021) 121233. https://doi.org/10.1016/j.ijheatmasstransfer.2021.121233