Consistency and accuracy in the simulation of two-phase flows with phase change using sharp interface capturing methods

Victor Boniou, Schmitt Thomas, Aymeric Vié

Consistency and accuracy in the simulation of two-phase flows with phase change using sharp interface capturing methods

Victor Boniou[a,*], Thomas Schmitt[a], Aymeric Vié[a,b]

[a] *Laboratoire EM2C UPR 288, CNRS, CentraleSupélec, Université Paris-Saclay, 3, rue Joliot-Curie 91192 Gif-sur-Yvette cedex France*
[b] *Fédération de Mathématiques de CentraleSupélec, CNRS, 3, rue Joliot-Curie 91192 Gif-sur-Yvette cedex France*

Abstract

In this paper, the modelling of phase change in an incompressible two-phase flow solver is detailed without restricting the numerical methods to a specific interface capturing method. The main challenge when including phase-change is the handling of flux discontinuities at the interface when advancing temperature and species mass fraction. An accurate and second order discretization is proposed for any Eulerian representation of the interface either by adding a sharp source term or by imposing a boundary condition at the interface. As the accuracy and convergence rate of such solver are driven by the reconstruction of the evaporation rate $\dot{m}$, particular attention is devoted to the reconstruction of gradient normal to the interface. Several methodologies are proposed to compute second-order gradients at the interface location adapted to any interface representation. Applying such techniques to a second-order accurate field leads to an expected first order accuracy of $\dot{m}$ but with remarkable accuracy improvements using ghost cell methods with quadratic extrapolation. Then, several phase-change procedures are built by combining a selection of numerical methods to handle flux discontinuities and evaluate gradients. The procedures are investigated on planar phase-change simulations to bring out inconsistent combination choices. Finally, a multidimensional evaporation test case is presented to show the final accuracy and limitations of phase-change modelling in today two-phase flow solvers.

Keywords: Volume-of-fluid, Level-set, Phase change, Incompressible flows, Cartesian grids

1. Introduction

Evaporation is a natural phenomenon ubiquitous in industrial processes such as food drying [1], spray cooling [2], or spray combustion [3]. In the literature, authors

*Corresponding author
Email address: victor.boniou@centralesupelec.fr (Victor Boniou)

employ different methods to solve two-phase flows, which can be classified into two main interface representations

1. The diffuse interface methods (DIM) define the interface as a region where quantity variations are stiff but continuous. Then, they do not need to represent the interface location explicitly. These methods are often used to study compressible flows as they rely on enriched thermodynamic treatments in the interfacial region where classical physics is not valid.
   DIM encompass the multifluid methods [4] which describes the interface as an artificially diffused region and the phase-field methods [5] based on an advanced thermodynamic description of the interface.
2. The sharp interface methods (SIM) use an infinitely thin interface as a moving surface with a given position. With this representation, the fluid properties are discontinuous, and the interface needs to be located accurately to apply the correct jump conditions.
   Front tracking (FT) methods [6] explicitly transport Lagrangian markers all belonging to the interface while interface capturing methods such as Volume-of-fluid (VOF) [7] and Level Set (LS) [8] rely on an Eulerian representation of the interface through the transport of a colour function.

As this work aims to study low Mach evaporating two-phase flows, compressible formulations are unnecessary. Thus, this work focuses on SIMs.

When evaporation comes into place, the numerical methods based on SIM must be adapted to treat additional discontinuities, which constitutes challenging issues. In the late '90s, the pioneer works of Juric et al. [9] with an FT method, Welch et al. [10] using VOF or Son et al. [11] with LS have paved the way for simulations with phase change. These works only contained the boiling phenomenon as the set of equations, and the phase-change modelling was limited to mono-component mixtures.
Some years later, Tanguy et al. [12] proposed an LS solver for evaporation simulations with a bi-component gas phase. The method performed simulations of convected droplets in 2D axisymmetric configurations. One year later, a solver based on VOF proposed by Schlottke et al. [13] performed 3D simulations of evaporation of convected droplets subject to high deformations.
Since these contributions, the domain of two-phase flow simulations has been very active with a noticeable gain in interest for the simulation of evaporation [14, 15, 16, 17, 18, 19, 20]. This increasing research activity resulted in considerable improvements in the robustness and accuracy of the methods in the last decade, which allowed to simulate more complex configurations such as evaporation in turbulence [21, 22, 23], combustion [24], or supersonic flows [25].
The literature presented here shows a vast spectrum of numerical methods, indicating no consensus on the best methodology to achieve evaporation simulations. This paper initially chooses to consider Eulerian representations of the interface and a one-fluid representation of the velocity in mass and momentum equations. The aim is then to bring some clarifications on the different methodologies which are available in the literature to face the main challenges emerging from the integration of phase change in an incompressible two-phase flow solver:

1. The computation of the evaporation rate $\dot{m}$, which is the pillar of the phase-change phenomenon: as for the evaluation of κ to obtain an appropriate surface tension modelling, the evaluation of $\dot{m}$ requires specific attention to describing phase change adequately.
2. The treatment of the velocity jump $[\mathbf{u}]_\Gamma$ in the continuity equation, which cannot be reduced to a simple divergence-free condition $\nabla \cdot \mathbf{u} = 0$ using a one-fluid representation of the velocity.
3. The temperature T and species mass fraction Y also need to be solved to close the system of equations. The main difficulty arises in the flux jump at the interface due to phase change.

These numerical aspects are often treated differently depending on the method employed to capture the interface. While the interface capturing method defines the transported color function c (the liquid volume fraction f for VOF or an implicit function ϕ or ψ for level Set), it does not impose any interface representation.
Indeed, an explicit representation of the interface Σ can be obtained equally from any c. The only difference resides in the method and constraint applied to Σ: for geometric VOF, it is possible to reconstruct Σ with the constraint that the volume enclosed matches exactly with the transported f while for algebraic VOF or Level-Set, Σ is reconstructed based on the iso-contour defining the interface position c^{-1}. On the other hand, an implicit representation, the signed distance to the interface ϕ, can also be obtained equally from any c. From VOF or Conservative Level-Set, this signed distance can be reconstructed from c^{-1}, while in Standard Level-Set, ϕ is directly transported.
It is then clear that limiting an interface capturing method to a single interface representation shrinks the spectrum of numerical methods that allow reconstructing the critical information of the interface topology (normal, curvature, surface, Dirac, Heaviside). Nonetheless, using explicit and implicit interface representations from any interface capturing method does not guarantee an accurate or consistent method. The outcome is highly dependent on the reconstruction techniques employed and the color function it is based on.
This paper investigates these issues from an interface representation perspective (Σ or ϕ) instead of an interface capturing method perspective (color function c). First, the governing equations of incompressible two-phase flows in Sec. 2. Then, the handling of flux jumps and the gradient reconstruction are studied both from explicit and implicit representations of the interface in 3 and 4 respectively. Finally, the most accurate methods are selected and employed to simulate evaporation test cases either using VOF or Standard Level-Set in Sec. 5 followed by a conclusion on the study in Sec. 6.

2. Physical description

A two-phase flow with phase change can be described by a set of governing equations and jump conditions. The jump conditions need additional physical considerations to close the terms such as the curvature κ and the evaporation rate $\dot{m}$. Finally, an interface representation makes the link between the governing equations valid in the phases taken separately and the jump conditions only valid at the interface. Both fluids

are here supposed to be incompressible with a mono-component liquid phase and a gas phase consisting in the mixture of the vapour of the condensable species and inert gas.

2.1. Incompressible flow governing equations

The set of governing equations for bi-component incompressible flows is:

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

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u} \otimes \mathbf{u})\right) = -\nabla P + \nabla \cdot (2\mu \mathbf{D}) + \rho \mathbf{f}_v \quad , \tag{1b}$$

$$\rho c_p\left(\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T\right) = \nabla \cdot (k \nabla T) \quad , \tag{1c}$$

$$\rho\left(\frac{\partial Y}{\partial t} + \mathbf{u} \cdot \nabla Y\right) = \nabla \cdot (\rho \mathcal{D}_v \nabla Y) \quad , \tag{1d}$$

where $\mathbf{u}$ is the velocity, P is the pressure, T is the temperature and Y is the species mass fraction in the gas. The fluid properties are denoted by ρ the density, μ the dynamic viscosity, k the thermal conductivity, c_p the specific heat at constant pressure and $\mathcal{D}_v$ the species diffusivity of the fluid. Finally, $\mathbf{f}_v$ is the body force and $\mathbf{D}$ the rate-of-deformation tensor.

2.2. Interface jump conditions

The two phases are coupled through jump conditions at the interface

$$[\mathbf{u}]_\Gamma \cdot \mathbf{n}_\Gamma = \dot{m}\left[\frac{1}{\rho}\right]_\Gamma \quad , \tag{2a}$$

$$\dot{m}\,[\mathbf{u}]_\Gamma \cdot \mathbf{n}_\Gamma - [P]_\Gamma + 2\,[\mu \mathbf{D} \cdot \mathbf{n}_\Gamma]_\Gamma \cdot \mathbf{n}_\Gamma = \sigma\kappa \quad , \tag{2b}$$

$$\dot{m}\,[\mathbf{u}]_\Gamma \cdot \mathbf{t}_\Gamma + 2\,[\mu \mathbf{D} \cdot \mathbf{n}_\Gamma]_\Gamma \cdot \mathbf{t}_\Gamma = 0 \quad , \tag{2c}$$

$$[k \nabla T \cdot \mathbf{n}_\Gamma]_\Gamma = \dot{m}\,[h_t]_\Gamma \quad , \tag{2d}$$

$$\rho_g \mathcal{D}_v \left.\nabla Y\right|_\Gamma \cdot \mathbf{n}_\Gamma = \dot{m}\,(1 - Y_\Gamma) \quad . \tag{2e}$$

where the jump operator $[\Phi]_\Gamma$ is defined such that $[\Phi]_\Gamma = \Phi_l - \Phi_g$, the interface normal vector $\mathbf{n}_\Gamma$ is defined by the outward pointing liquid normal $\mathbf{n}_\Gamma = \mathbf{n}_l = -\mathbf{n}_g$, the interface tangential vector is $\mathbf{t}_\Gamma$, σ is the surface tension, κ is the curvature, $\dot{m}$ is the evaporation rate and Y_Γ is the species mass fraction at the interface. The enthalpy jump $[h_t]_\Gamma$ is defined by

$$[h_t]_\Gamma = -\mathscr{L}_{vap} + \left[c_p\right]_\Gamma (T_\Gamma - T_{sat}) \tag{3}$$

with $\mathscr{L}_{vap}$ the specific heat of boiling, T_Γ the interface temperature and T_{sat} the saturation temperature of the pure liquid.

2.3. Phase-change closures

The phase-change procedure needs a closure for the evaporation rate $\dot{m}$, the interface temperature T_Γ and the species mass fraction at the interface Y_Γ. While $\dot{m}$ appears in all jump conditions related to phase change, the interface temperature and species mass fraction can be used explicitly at different stages to impose boundary conditions or jump conditions at the interface and compute accurate gradients at the interface. A general system of equation can be written to determine these quantities by using the jump conditions and a thermodynamic relation between T_Γ and Y_Γ

$$\begin{cases} \dot{m} = \dfrac{[k\nabla T \cdot \mathbf{n}_\Gamma]_\Gamma}{[h]_\Gamma} = \mathcal{M}_T(T_\Gamma) \quad , & \text{(4a)} \\ \dot{m} = \dfrac{\rho_g \mathcal{D}_v \, \nabla Y|_\Gamma \, \mathbf{n}_\Gamma}{Y_\Gamma - 1} = \mathcal{M}_Y(Y_\Gamma) \quad , & \text{(4b)} \\ Y_\Gamma = \dfrac{P_s\left(T_\Gamma\right) M_v}{P_s\left(T_\Gamma\right) M_v + \left(P_{ref} - P_s\left(T_\Gamma\right)\right) M_0} = \mathcal{R}(T_\Gamma) \quad , & \text{(4c)} \end{cases}$$

with M_v and M_0 the molar mass of vapour and inert gas respectively. The relation Eq.(4c) is obtained by assuming that the interface is at thermodynamic equilibrium, pressure, temperature and chemical potential are equal in the two phases. The saturation vapour pressure P_s is obtained from the Rankine relation

$$P_s = P_{ref} \exp\left(-\frac{\mathscr{L}_{vap}}{R}\left(\frac{1}{T_\Gamma} - \frac{1}{T_{sat}}\right)\right) \quad . \tag{5}$$

In this integration, $\mathscr{L}_{vap}$ and R are supposed constant with temperature, which is not always true (see [26] for some extended studies) and T_{sat} is taken at a reference pressure P_{ref}. Note that some authors use different relations such as in FS3D [13, 20] for which authors prefer the Wagner equation [27]. Other authors [22, 28] use the Antoine equation [29] instead. These other formulations rely on empirical corrections to take into account dependencies of $\mathscr{L}_{vap}$ with the temperature.
For general vaporization problems, Eqs. (4) need to be solved, and can be reformulated into an implicit equation on T_Γ

$$T_\Gamma = \mathcal{R}^{-1}\left(\mathcal{M}_Y^{-1}\left(\mathcal{M}_T(T_\Gamma)\right)\right) \quad . \tag{6}$$

Numerically, $\mathcal{M}_T$ and $\mathcal{M}_Y$ contain also the normal gradients $T_l^{(1)} = \nabla T_l \cdot \mathbf{n}_\Gamma$, $T_g^{(1)} = \nabla T_g \cdot \mathbf{n}_\Gamma$ and $Y^{(1)} = \nabla Y \cdot \mathbf{n}_\Gamma$ which are not known a priori and needs an explicit reconstruction with T_Γ and Y_Γ appearing in the stencil. This can be done by simple differencing as in [30, 18] or by fitting an Erf function as in [31]. However, Palmore et al. [18] pointed out the necessity to have a high order treatment of $\dot{m}$ to obtain converging methods. This is why, in most of the methodologies presented in the literature, $\dot{m}$ is obtained directly from $\mathcal{M}_T$ or $\mathcal{M}_Y$ by using more sophisticated normal gradient evaluations.

2.3.1. Evaporation rate

Letting aside the general phase-change system of Eq. (4), one of the expressions of $\dot{m}$ has to be used arbitrarily. While this approach leads to the loss of the discrete equality $\mathcal{M}_T = \mathcal{M}_Y$ it gives more degree of freedom on the computation of $\dot{m}$.

Most of the evaporation solvers [12, 13, 30, 31, 19] rely on $\mathcal{M}_Y$ to compute $\dot{m}$, as it only requires one gradient evaluation. While it gives a straightforward framework for the simulation of general unsteady evaporation process, it suffers from ill-posedness when the interface conditions are close to $T_\Gamma = T_{sat}$ and $Y_\Gamma = 1$. This has been first observed in [30] which motivated the resolution of the full system Eq. (4) instead to obtain $\dot{m}$. They noticed that this problem arises in the case of high evaporation rate only. This behaviour has been widely explored in [17] where it was found that $\mathcal{M}_Y$ is prompt to large errors when $Y_\Gamma > 0.9$.

A way around is to compute $\dot{m}$ from $\mathcal{M}_T$ as in all the boiling solvers [11, 32, 33, 34, 35, 36, 37]. It can be generalized to evaporation simulations as in [38, 15, 18] by considering an interface temperature $T_\Gamma \neq T_{sat}$ to handle both evaporation and boiling without further considerations.

Finally, in the work presented in [17], a switch between the two regimes is proposed to benefit from both formulations and avoid the numerical issues previously encountered. If T_Γ is below a certain value T_c, the interface is supposed to be far enough from boiling and $\mathcal{M}_Y$ is used. When T_c is reached, the method switches to the boiling limit where $\dot{m}$ is computed from $\mathcal{M}_T$ instead.

Other authors also considered the use of both formulations in the same solver [14, 16]. But no switch is possible during a given simulation, the formulation is chosen with respect to the test case simulated.

2.3.2. Temperature and species mass fraction at the interface

As discussed previously, T_Γ and Y_Γ can be retrieved from solving the general system Eq. (4). However, if $\dot{m}$ is evaluated from a methodology of Sec. (2.3.1), the relevance of solving iteratively Eq. (4) is limited as the system will not be verified ($\mathcal{M}_T \neq \mathcal{M}_Y$). This is why most of the phase-change solvers [13, 15, 12, 17, 19] disregard Eq. (4a) and Eq. (4b). Therefore, another equation is required in addition to the Clausius-Clayperon relation $Y_\Gamma = \mathcal{R}(T_\Gamma)$ to obtain T_Γ and Y_Γ. In practice, this is done by either deducing T_Γ from the computed temperature field T or Y_Γ from the computed species mass fraction field Y through extrapolation (first order extrapolation [13, 15] or second order extrapolation [12, 17, 19]).

2.4. Two-phase flow representation

Now that the physical problem with the phase-change closures have been described, the discretization of the governing equations defined by Eq. (1) must include the jump conditions from Eq. (2) using an interface capturing method.

2.4.1. Interface representation

The interface is represented in an Eulerian fashion using a color function c which is transported using

$$\frac{\partial c}{\partial t} + \mathbf{u}_\Gamma \cdot \nabla c = 0 \tag{7}$$

with $\mathbf{u}_\Gamma$ the interface velocity.
From this color function, all the information related to the interface topology can be retrieved such as the normal $\mathbf{n}_\Gamma$, the Heaviside H_Γ, the Dirac distribution δ_Γ or the curvature κ. Note that this paper presents the different numerical techniques without restricting them to a specific interface capturing method. However, two approaches will be considered in the conclusions and in the investigation of full phase-change procedures in Sec. 5.1.1. They are selected to be as representative as possible of the state-of-the art of VOF and Level-Set frameworks:

- An approach applied to VOF with an explicit representation of the interface and quantities defined at the phase barycenter;
- An approach applied to Standard Level-Set (SLS) with an implicit representation of the interface and quantities defined at the cell center.

In addition to the discretization errors related to imposing flux discontinuities and evaluating normal gradient, the transport errors from VOF and SLS are not equivalent and need to be compared to give the full picture of numerical methods design for phase-change simulations.

2.4.2. Momentum conservation

The jump condition associated to momentum is treated using a one-fluid formulation as in the majority of incompressible two-phase flow solvers of the literature. This leads to the following momentum equation with an additional term due to surface tension located at the interface through a Dirac distribution δ_Γ:

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) = -\nabla P + \nabla \cdot (2\mu \mathbf{D}) + \rho \mathbf{f}_v + \sigma \kappa \mathbf{n}_\Gamma \delta_\Gamma \quad . \tag{8}$$

The resolution of Eq. (8) is not the scope of this paper as it has already been broadly investigated in the literature. Moreover, phase change does not impact algorithms developed for isothermal two-phase flows as the one-fluid formulation takes into account the new contribution in Eq. (2b) implicitly through the surface tension. The numerical methods used in the solver are detailed in Appendix 8.3.

2.4.3. Other transport equations

The jump conditions associated to phase change result in flux discontinuities located at the interface:

- The mass conservation Eq. (1a) appears indirectly in the pressure equation such that
$$\nabla \cdot \left(\frac{1}{\rho} \nabla P \right) = \nabla \cdot \mathbf{u}^* \quad , \tag{9}$$
with $\left[\frac{1}{\rho} \nabla P \cdot \mathbf{n}_\Gamma \right]_\Gamma = [\mathbf{u}]_\Gamma \cdot \mathbf{n}_\Gamma = \dot{m} \left[\frac{1}{\rho} \right]_\Gamma$.
- The temperature transport Eq. (1c) is rewritten as
$$\nabla \cdot (k \nabla T) = \rho c_p \left(\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T \right) \quad , \tag{10}$$
with $[k \nabla T \cdot \mathbf{n}_\Gamma]_\Gamma = \dot{m} [h_t]_\Gamma$.

- The species mass fraction transport Eq. (1d) is rewritten as

$$\nabla \cdot (\rho \mathcal{D}_v \nabla Y) = \rho \left(\frac{\partial Y}{\partial t} + \mathbf{u} \cdot \nabla Y \right) \quad , \tag{11}$$

  with $[\rho \mathcal{D}_v \nabla Y \cdot \mathbf{n}_\Gamma]_\Gamma = \dot{m} [Y]_\Gamma$.

The resolution of these equations requires to impose a flux discontinuity $[\beta \nabla \Phi \cdot \mathbf{n}_\Gamma]_\Gamma = b_\Gamma$ on a Laplacian operator $\nabla \cdot (\beta \nabla \Phi)$. All the jump conditions depend on the evaporation rate $\dot{m}$ which needs a numerical reconstruction. Then, two numerical challenges arises when designing numerical methods to solve these transport equations: how to impose the flux discontinuity with a sharp and accurate method and how to reconstruct accurately $\dot{m}$ at the interface ?
The velocity jump is imposed using a source term to be consistent with the momentum treatment. For temperature and species mass fraction, an investigation is performed in Sec. 3 to obtain an optimal method able to give a sharp representation of the quantities and an accurate discretization. The reconstruction of $\dot{m}$ requires the computation of normal gradients of liquid and gas temperature or species mass fraction at the interface location. The high accuracy of $\dot{m}$ is a prerequisite for accurate phase-change simulations as it appears in the transport of all important fluid quantities. This challenge is detailed in Sec. 4.
Finally, Sec. 5.1.3 investigates several combinations of the handling of flux discontinuities and the evaluation of $\dot{m}$ from Eq. (4a) or Eq. (4b).

3. Imposing flux discontinuities

This section focuses on the first numerical challenge of phase-change simulations resulting in the resolution of the Poisson equation with a jump condition. Given the general Poisson equation

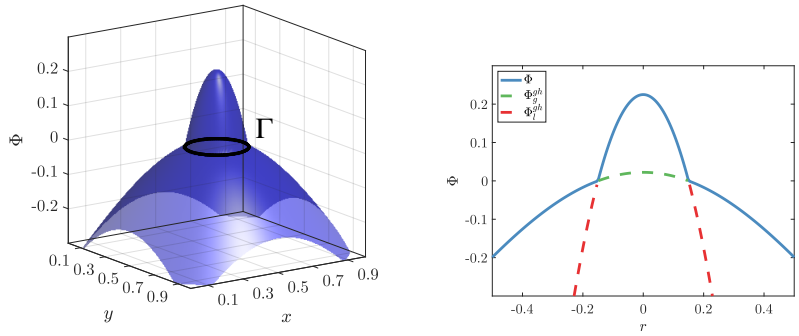
$$\nabla \cdot (\beta \nabla \Phi) = S \quad , \tag{12}$$

with the associated jump condition $[\beta \nabla \Phi \cdot \mathbf{n}_\Gamma]_\Gamma = b_\Gamma$ at the interface. This problem is analogous to solve Eqs. (9), (10) or (11). The jump condition b_Γ can be handled with a source term or an immersed boundary condition (IBM) at the interface location.
When the cell is not cut by the interface, the following 2D finite volume discretization is used for Eq. (12)

$$F^{(\Phi)}_{i+\frac{1}{2},j} A_{i+\frac{1}{2},j} - F^{(\Phi)}_{i-\frac{1}{2},j} A_{i-\frac{1}{2},j} + F^{(\Phi)}_{i,j+\frac{1}{2}} A_{i,j+\frac{1}{2}} - F^{(\Phi)}_{i,j-\frac{1}{2}} A_{i,j-\frac{1}{2}} = S_{i,j} V_{i,j} \quad , \tag{13}$$

with A the face area (length in 2D) and V the cell volume (area in 2D).
In this section, a 2D static diffusion problem is considered to evaluate the different methodologies to solve Eq. (12) with a flux jump.
The study is based on the test case defined in [34] and illustrated in Fig. 1 for a domain $\Omega = \Omega_l \cap \Omega_g$ of dimensions $[1 \times 1]$ where Ω_l is the interior and Ω_g the exterior of the circle of radius $R = 0.2$ centered at $(0.5, 0.5)$

$$\Phi = \begin{cases} \exp\left(-r^2\right) \exp - \left(-R^2\right) & \text{if } r > R \\ 10 \left(R^2 - r^2\right) & \text{if } r \leq R \end{cases} \quad , \tag{14}$$



(a) Surface contour of Φ with the interface Γ represented in black.

(b) Evolution of Φ with respect to r (in blue) with ghost values (in red and green).

Figure 1: Illustration of Φ with a surface contour (left) and a 2D plot with respect to r (right)

with r the reduced radius such that $r = \sqrt{(x-0.5)^2 + (y-0.5)^2}$.
The associated static diffusion problem with $\beta_l = 10$ and $\beta_g = 1$ is written as

$$\nabla \cdot (\beta \nabla \Phi) = \begin{cases} \left(4r^2 - 4\right)\exp\left(-R^2\right) & \text{if } r > R \\ -400 & \text{if } r \leq R \end{cases} \quad , \tag{15}$$

with either a flux jump or an IBM based on the interface value Φ_Γ or/and the interface flux $\Phi_\Gamma^{(1)} = \nabla\Phi \cdot \mathbf{n}_\Gamma$. They are imposed using the exact values

$$[\beta \nabla\Phi \cdot \mathbf{n}_\Gamma]_\Gamma = -200R + 2R\exp\left(-R^2\right) \quad , \quad \Phi_\Gamma = 0 \quad , \quad \Phi^{(1)}_{\Gamma,l} = -20R \quad , \quad \Phi^{(1)}_{\Gamma,g} = -2R\exp\left(-R^2\right) \quad . \tag{16}$$

The error is then computed using L_2 and L_∞ norms for a range of number of cell per diameter from $N_D = 3.2$ to $N_D = 409.6$.

3.1. Source term method

This section considers the resolution of Eq. (12) in the whole domain by imposing $[\beta\nabla\Phi \cdot \mathbf{n}_\Gamma]_\Gamma$ as a source term. Then, Eq. (12) is rewritten as

$$F^{(\Phi)}_{i+\frac{1}{2},j} A_{i+\frac{1}{2},j} - F^{(\Phi)}_{i-\frac{1}{2},j} A_{i-\frac{1}{2},j} + F^{(\Phi)}_{i,j+\frac{1}{2}} A_{i,j+\frac{1}{2}} - F^{(\Phi)}_{i,j-\frac{1}{2}} A_{i,j-\frac{1}{2}} = \left(S_{i,j} + b_{\Gamma,i,j}\delta_\Gamma\right) V_{i,j} \quad . \tag{17}$$

The accuracy of the problem resolution then depends on the approximation of the Dirac distribution δ_Γ.
In practice, a good approximation of δ_Γ is as sharp as possible with a good accuracy on the associated interface area.
This Dirac distribution can be retrieved either based on explicit or implicit representations of the interface through the notional relations

$$\delta_\Gamma = \frac{A_\Gamma}{V_C} = |\nabla H_\Gamma(c)| = \mathbf{n}_\Gamma \cdot \nabla H_\Gamma(c) \tag{18}$$

with A_Γ the interface surface, V_C the cell volume and $H_\Gamma(c)$ an Heaviside function. Studies on the approximation of δ_Γ have already been performed using VOF [39] or Level-Set [40]. They both agree that using an explicit representation of the interface always leads to better approximations of δ_Γ. In this section, this statement is moderated by giving a full picture of δ_Γ approximations and their corresponding accuracy. Then, the accuracy metric is studied based on the relative error E_{area} between the exact perimeter $A_{exact} = 2\pi R$ and the reconstructed perimeter $A_{reco} = \sum_{i=1}^{N_C} A_{\Gamma,i}$ of the circle of radius $R = 0.2$. The final metric is $\langle E_{area} \rangle$ the mean of E_{area} over the 100 circles randomly located in the domain to meet as many configurations as possible.

3.1.1. Implicit representation

In the VOF framework, direct differentiation of the volume fraction f can provide an interface density. The most popular formula [41, 42] that was used in the context of algebraic VOF is the following:

$$\delta_\Gamma = |\nabla f| \quad . \tag{19}$$

The issue of such approximation is that it is a global formulation and can suffer from local inaccuracies. Moreover, it provides finite values of interface area even in cells which does not contain the interface. This last aspect is not compatible with a sharp representation of the interface and can lead to shape deformations through phase change [39]. A more general formula of the form $\mathscr{F}(f)|\nabla f|$ such that $\int_0^1 \mathscr{F}(f)df = 1$ was proposed in [43] to sharpen the approximation of δ_Γ. Taking $\mathscr{F}(f) = 1$ leads to the AF0 formula Eq. (19) while two other choices AF1 and AF2 use the sharper formulae $\mathscr{F}(f) = 2f$ and $\mathscr{F}(f) = 6f(1-f)$ respectively. Even if those formulae have been originally introduced in the VOF framework, they can be extended to any Heaviside $H_\Gamma(c)$. In this study, algebraic formulae are applied to $H^\epsilon_\Gamma(f)$ and $H^\epsilon_\Gamma(\phi)$ defined as

$$H^\epsilon_\Gamma(f) = f \quad , \qquad H^\epsilon_\Gamma(\phi) = \psi = \frac{1}{2}\left(\tanh\left(\frac{\phi}{2\epsilon}\right)\right) \quad . \tag{20}$$

Another interesting formulation is the reinterpretation of the ghost fluid method (GFM) of [44] as a source term with a sharp Dirac distribution based on H^0_Γ. While this reinterpretation was made on the pressure jump for surface tension force [45], it can be extended to a flux jump using H^0_Γ defined at the faces.

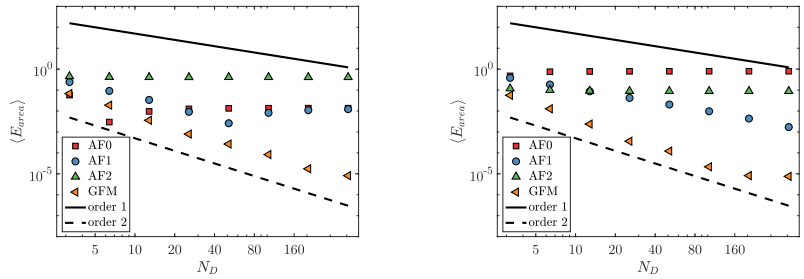
$$\delta_\Gamma = \mathbf{n}_\Gamma \cdot \nabla H^0_\Gamma \quad , \tag{21}$$

The 1D discretization is then

$$\delta_{\Gamma,i} = \left(H^0_\Gamma\left(\frac{1}{2}(c_{i+1} + c_i)\right) - H^0_\Gamma\left(\frac{1}{2}(c_i + c_{i-1})\right)\right)\frac{n_{x,i}}{\Delta x} \quad , \tag{22}$$

with $n_{x,i}$ the first component of the normal to the interface, $H^0_\Gamma(c) = 1$ in the liquid and $H^0_\Gamma(c) = 0$ in the gas.

In Fig. 2a, convergence of $\langle E_{area} \rangle$ is presented for the algebraic formulae applied to the volume fraction f. The sharpest version AF2 provides highly erroneous area estimations while AF1 shows convergence in the low resolution. In the high



(a) From f **(b)** From ψ

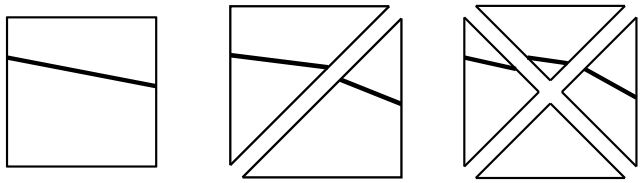
Figure 2: Mesh convergence of $\langle E_{area} \rangle$ for algebraic formulae applied to f and ψ

resolution limit, AF0 and AF1 gives similar results. However, none of these formulae converges with mesh refinement. On the other hand, the GFM formulation presents a second-order convergence which is always more accurate than the algebraic formulae considered here.
Fig. 2b shows $\langle E_{area} \rangle$ convergence of those formulae applied to ψ defined by Eq. (20) which is a smoother version of f. The smoothing of H_Γ leads to higher errors on AF0, however it provides a convergent behaviour for AF1.
Overall, the conclusions are the same in the literature [40, 39]: using algebraic formulae is not suitable for accurate and sharp evaluation of the interface area even when the Haeviside is smoother. On the other hand, the GFM formulation using a sharp Heaviside leads to a second-order approximation of δ_Γ.
It is then clear that an implicit representation of the interface can also lead to sharp and convergent approximations of δ_Γ if the appropriate formulation is used.

3.1.2. Explicit representation

The other way to define δ_Γ is to directly rely on the explicit representation of the interface in a given computational cell. This methodology naturally leads to a sharp representation of δ_Γ as it will be located only in cells containing the interface. The following study aims to explain the different methodologies to reconstruct the interface and compare their accuracy.
The reconstruction of a second-order explicit representation of the interface can be done either using a cell decomposition (CD) [46] based on the iso-contour of c or a piecewise linear interface construction (PLIC) [47] based on the volume fraction f.
The explicit reconstructions of the interface are defined as:

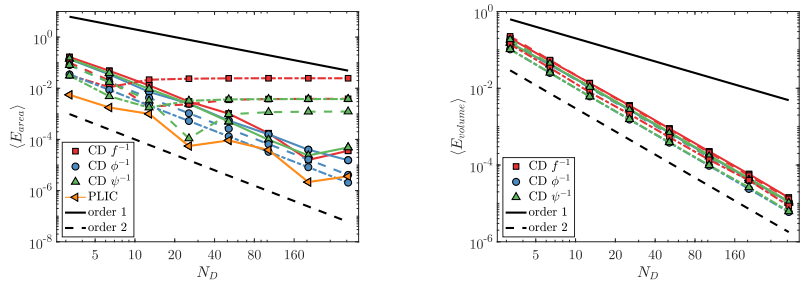
- CD0, the marching cube method [46].
- CD1, the simplex decomposition in two simplices [48].
- CD2, the simplex decomposition in four simplices [21].
- PLIC, the piecewise linear interface construction [47].



(a) CD0 and PLIC **(b)** CD1 **(c)** CD2

Figure 3: Cell decomposition level illustration, the thick line represent a piece of interface

The difference between these decompositions is illustrated in Fig. 3. While CD0 and PLIC lead to a single linear reconstruction in the cell (Fig. 3a), CD1 allows two slopes (Fig. 3b) and CD2 three slopes (Fig. 3c). Then, the level of decomposition can provide a subcell representation of the interface curvature if a simplex decomposition is used. However, PLIC reconstructs the interface such that the liquid volume enclosed in the cell corresponds exactly to the volume fraction f. It is then relevant to observe the volume error E_{volume} provided by the interface reconstruction for CD which is not exact in volume. The area and volume errors from these explicit reconstructions of the interface are presented in Fig. 4.



(a) Convergence of $\langle E_{area} \rangle$ **(b)** Convergence of $\langle E_{volume} \rangle$

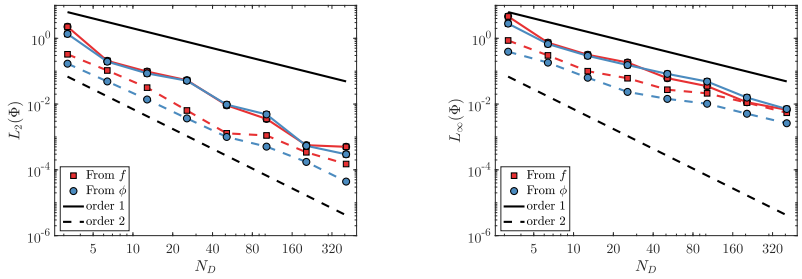
Figure 4: Mesh convergence of $\langle E_{area} \rangle$ and $\langle E_{volume} \rangle$ for the different cell decomposition levels CD0 (——), CD1 (- - -) and CD2 (-···-). PLIC is added in $\langle E_{area} \rangle$ but does not appear for $\langle E_{volume} \rangle$ as it is exact in volume.

In Fig. 4a, a second-order convergence of the area evaluation is observed for the use of CD0 with small effects of the choice of c^{-1} on the results. However, f^{-1} and ψ^{-1} saturate for the last resolution point displayed using CD0 while they saturate for coarser meshes using CD1 and even coarser meshes using CD2. On the other hand, all the decompositions applied to ϕ^{-1} show a straight second-order convergence with accuracy magnitude increased by the level of decomposition. CD2 is then the best choice when ϕ is transported. The saturation is not observed for $\langle E_{volume} \rangle$ in Fig. 4b where a second-order convergence of the volume is obtained for all simplex decomposition from any c^{-1} with only limited improvement of volume accuracy (all slopes almost collapse).

From these observations, a conclusion can be drawn: the evaluation of area seems to be more prompt to errors compared to the volume and increasing the decomposition level amplifies these errors for f^{-1} and ψ^{-1}. In practice, the intersections are obtained from linear interpolation of c. As f and ψ are far from being linear, they are more prompt to intersection errors which are accumulated by increasing the level of decomposition. Finally, using the inherent PLIC reconstruction leads to the most accurate area evaluation while being exact in volume, it is then the best choice when f is transported.

3.1.3. Application to the diffusion problem

Here, the selected methods of Sec. 3.1.1 and 3.1.2 are compared on the diffusion problem. The implicit representation of the interface is based on the GFM formulation of Eq. (21) using either f or ϕ. For the explicit representation, PLIC is used from f and CD2 is used from ϕ.



(a) Convergence of $L_2(\Phi)$ **(b)** Convergence of $L_\infty(\Phi)$

Figure 5: Mesh convergence of $L_2(\Phi)$ and $L_\infty(\Phi)$ for the static diffusion problem using an implicit (——) or an explicit (- - -) representation of the interface.

Fig. 5 shows second-order convergence in $L_2(\Phi)$ and first-order convergence in $L_\infty(\Phi)$ for all the δ_Γ approximation choices. It was already shown in the literature that GFM [49] and Dirac approximation from explicit reconstruction [40] both lead to first order convergence in $L_\infty(\Phi)$ and second-order convergence in $L_2(\Phi)$ when based on ϕ. This is explained by the main error contribution introduced by the source term approach which is the mixing of phase quantities at the interface. The maximum error is then located at the interface and is even more important when $[\beta]_\Gamma$ is large. An important result is that GFM is not the most accurate methodology to impose a jump condition as a source term at the interface even if it is often used in the literature.
On the other hand, the choice of f or ϕ to retrieve an interface representation does not seem to affect drastically the accuracy of the schemes while using an explicit representation always lead to more accurate results.
Now, it is interesting to investigate the immersed boundary method which allows to avoid the mixing of quantities and is independent of the magnitude of $[\beta]_\Gamma$.

3.2. Immersed boundary method

Imposing boundary conditions at the interface can be done using the cut-cell method [50] (CCM) which uses explicitly the boundary defined by the interface reconstruction or the ghost-cell method [51] (GCM) which defines ghost values to implicitly impose the boundary condition at the interface.
In this case, Eq. (12) is divided into two subproblems in Ω_l and Ω_g coupled with a boundary condition at the interface defined as

$$a\Phi_\Gamma + b\Phi_\Gamma^{(1)} = c \quad , \tag{23}$$

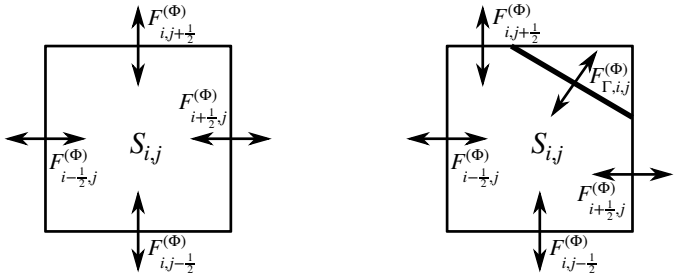
with a, b and c defined such that

1. If $a \neq 0$ and $b = 0$, it reduces to a Dirichlet boundary condition
2. If $a = 0$ and $b \neq 0$, it reduces to a Neumann boundary condition
3. If $a \neq 0$ and $b \neq 0$, it corresponds to a Robin boundary condition

Note that a, b and c are not known a priori for a phase-change solver application and are reconstructed.

3.2.1. Cut-cell method

The method was first presented as the Cartesian grid method [50] and then applied to the study of compressible flows [52] and the resolution of the Poisson equation in 2D [53] and 3D [54].



(a) Pure cell **(b)** Interfacial cell

Figure 6: Illustration of the flux definitions for a pure cell and interfacial cell using CCM

In the case of a cell containing the interface, an additional contribution is due to the interface and the configuration of Fig. 6b can be discretized by

$$F^{(\Phi)}_{i+\frac{1}{2},j}A_{f,i+\frac{1}{2},j} - F^{(\Phi)}_{i-\frac{1}{2},j}A_{f,i-\frac{1}{2},j} + F^{(\Phi)}_{i,j+\frac{1}{2}}A_{f,i,j+\frac{1}{2}} - F^{(\Phi)}_{i,j-\frac{1}{2}}A_{f,i,j-\frac{1}{2}} + F^{(\Phi)}_{\Gamma,i,j}A_{\Gamma,i,j} = S_{i,j}f_{i,j}V_{i,j} \quad , \tag{24}$$

with A_f the wetted areas, $F^{(\Phi)}_{\Gamma,i,j}$ the interface flux, $A_{\Gamma,i,j}$ the interface area and $f_{i,j}$ the volume fraction of the known part of the domain in the cell. In Eq. (24), the face and interface fluxes need to be evaluated.
Two compact discretizations are considered here with Φ either stored at the cell center (cc) [55] or at the phase barycenter (pb) $\mathbf{x}_{cm,i,j}$ [18].

The discretization proposed in [55] for $F_f^{(\Phi)}$ is written as

$$F_{i-\frac{1}{2},j}^{(\Phi,cc)} = \beta_{i-\frac{1}{2},j} \frac{\Phi_{i,j} - \Phi_{i-1,j}}{\Delta x} \quad , \tag{25}$$

with Φ defined at the cell center. Note that higher order flux can be retrieved by taking into account the wetted face area with additional interpolations as in [53, 54]. However it loses the compactness of the discretization which is a drawback for an implicit treatment.
In [18], $F_f^{(\Phi)}$ is based on the following expression

$$F_{i-\frac{1}{2},j}^{(\Phi,pb)} = \beta_{i-\frac{1}{2},j} \left(\Phi_{i,j} - \Phi_{i-1,j}\right) \frac{\Delta x_{cm}}{\|\mathbf{x}_{cm,i,j} - \mathbf{x}_{cm,i-1,j}\|_2^2} \quad , \tag{26}$$

with $\Phi_{i,j}$ defined at the cut-cell barycenter $\mathbf{x}_{cm,i,j}$ and Δx_{cm} the distance in the x direction of both cell barycenters. This expression is not second-order accurate, but allows to keep a symmetric discretization of the gradient.
If the cut-cell is defined from a PLIC reconstruction which is not continuous, $A_{f,i-\frac{1}{2},j}$ need to be defined from $A^-_{f,i-\frac{1}{2}}$ the wetted face area of the PLIC reconstruction from cell C_{i-1} and $A^+_{f,i-\frac{1}{2}}$ the wetted face area of the PLIC reconstruction from cell C_i. In the work of [18], they found that using $\min\left(A_f^-, A_f^+\right)$ was providing the most stable results. A drawback of this method is that it can lead to cells with a non-zero interface area $A_{\Gamma,i,j}$ and zero wetted areas which is clearly inconsistent. As a circumvent, this work proposes to define the wetted areas from a weight average of both wetted areas

$$A_{f,i-\frac{1}{2}} = \frac{W_{i-1,j} A^-_{f,i-\frac{1}{2}} + W_{i,j} A^+_{f,i-\frac{1}{2}}}{W_{i-1,j} + W_{i,j}} \quad , \tag{27}$$

with $W = f(1-f)$. This interpolation gives more importance to the wetted areas computed from interface cells close to $f = 0.5$ which are expected to provide interface reconstructions less prompt to errors.
Then, the definition of $F_{\Gamma,i,j}^{(\Phi)} = \beta_{\Gamma,i,j} \Phi_\Gamma^{(1)}$ depends on the type of boundary condition that needs to be prescribed at the interface.
To apply Dirichlet boundary condition Φ_Γ, one need to compute $F_{\Gamma,i,j}^{(\Phi)}$ from an approximation. In [18], a compact first-order gradient approximation is built from $\mathbf{x}_\Gamma$ and $\mathbf{x}_{cm}$

$$F_{\Gamma,i,j}^{(\Phi)} = \beta_{\Gamma,i,j} \frac{\Phi_{i,j} - \Phi_\Gamma}{\|\mathbf{x}_{cm} - \mathbf{x}_\Gamma\|_2} \quad . \tag{28}$$

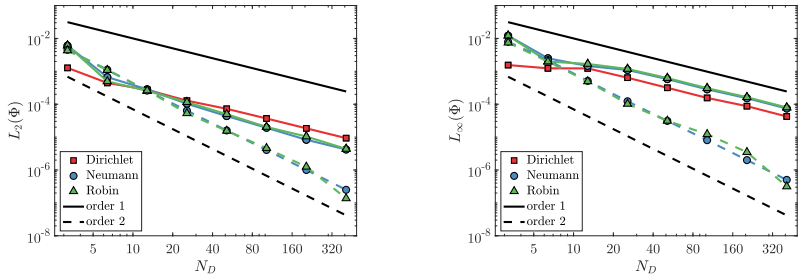
This expression allows to handle the boundary condition fully implicitly while keeping a symmetric matrix.
For Neumann boundary condition, the choice is straightforward and has been used in several works [53, 54, 56, 57]. The flux is directly prescribed by the Neumann condition

$$F_{\Gamma,i,j}^{(\Phi)} = \beta_{\Gamma,i,j} \Phi_\Gamma^{(1)} \quad . \tag{29}$$

The Robin boundary condition is very interesting for evaporation applications and has been presented by [55] for the resolution of the Stefan problem. This takes the form

$$F^{(\Phi)}_{\Gamma,i,j} = \beta_{\Gamma,i,j} \frac{c_{i,j} - a_{i,j}\Phi_{i,j}}{b_{i,j}} \quad , \tag{30}$$

with the first-order approximation $\Phi_\Gamma = \Phi_{i,j}$.
Using CCM with quantities defined at cell center or phase barycenter leads to the following $L_2(\Phi)$ and $L_\infty(\Phi)$ errors convergence displayed in Fig. 7.



(a) Convergence of $L_2(\Phi)$ **(b)** Convergence of $L_\infty(\Phi)$

Figure 7: Mesh convergence of $L_2(\Phi)$ and $L_\infty(\Phi)$ for the static diffusion problem with CCM using phase barycenter quantities (——) or cell center quantities (- - -)

When quantities are defined at the cell center, the method leads to second-order accuracy. However, when using quantities defined at the phase barycenter, the resulting accuracy is between first and second order because of the face flux approximation Eq. (26). More specifically, the Dirichlet methodology leads to a first-order convergence which is due to the first-order flux approximation at the interface from Eq. (28). When a Neumann boundary condition is applied, the flux at the interface is exact and the method is between first and second-order. The Robin boundary condition is slightly less accurate than the Neumann boundary condition which is expected as at is a mixing between Dirichlet and Neumann. Then, the phase barycenter arrangement seems to be a higher constraint on numerical methods to obtain higher order accuracy.

3.2.2. Ghost cell method

In parallel to CCM, another approach has been developed in the late 90's to handle Poisson equations on irregular domains based on a ghost cell method (GCM).
The first algorithm was proposed in [51] to apply Dirichlet boundary conditions at the interface. Then, other works such as [58] proposed an extension of GFM to impose boundary conditions instead of quantity jumps at the interface.
The following second-order finite difference discretization is proposed for the 1D version of Eq. (12) where the interface lies between x_i and x_{i+1}

$$\frac{\beta_{i+\frac{1}{2}} \frac{\Phi^{gh}_{i+1} - \Phi_i}{\Delta x} - \beta_{i-\frac{1}{2}} \frac{\Phi_i - \Phi_{i-1}}{\Delta x}}{\Delta x} = S_i \quad , \tag{31}$$

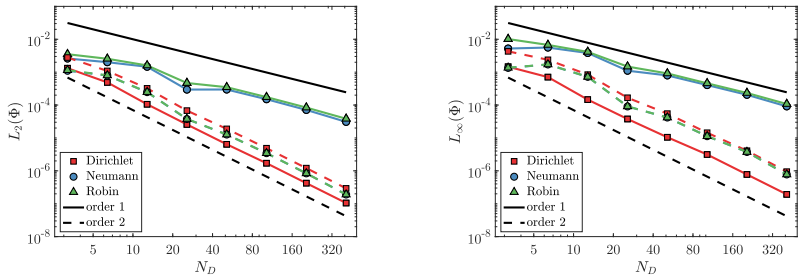
with Φ_{i+1}^{gh} the ghost value of Φ in the unknown domain.
Then, a second-order approximation of Φ_{i+1}^{gh} can be defined as $\Phi_{i+1}^{gh} = \frac{\Phi_\Gamma + (\theta-1)\Phi_i}{\theta}$ using linear extrapolation. Higher order extrapolations were already used for the resolution of Stefan problem [51] or for fourth order accuracy when solving Laplace and heat equations in [59]. However, they lead to a non-symmetric linear system. As for the cell volume in CCM, expressions including θ are not well-defined for arbitrary small values of θ. This issue can be handled by fixing a threshold θ_c for which the interface x_Γ is too close to x_i and imposing $\Phi_i = \Phi_\Gamma$ does not alter the overall accuracy and boundness of the method [58]. Another way to handle such singularity is to use a new stencil which does not include Φ_i in the discretization [34]. Then the denominator is bounded by Δx.
The convergent behaviour was first observed numerically without any evidence from Taylor expansion analysis, however in [60] a theoretical proof is provided.
This approach is easily extended to multidimensional Poisson equation by applying the same methodology for the other directions. However, application to other boundary conditions such as Neumann or Robin is not straightforward and has been addressed only recently [61, 62].
The extrapolation used to defined Φ_{i+1}^{gh} can be generalized to any boundary condition type using the following second-order extrapolation

$$\Phi_{i+1}^{gh} = \Phi_i + \frac{(c\Delta x - a\Delta x \Phi_i)\, n_x}{b + a\theta\Delta x n_x} \quad . \tag{32}$$

Note that in this relation, $a = 1$, $b = 0$ and $c = \Phi_\Gamma$ leads to the second-order extrapolation initially proposed in [58].
In [61], authors demonstrated that using a dimensional-splitting is inherently first-order and not even consistent for the gradient evaluation due to the the normal direction nature of such boundary condition type. They propose a new procedure for imposing a Robin boundary condition in the normal direction to the interface using a ghost-cell method.
Instead of approximating Φ^{gh}, the normal derivative $\Phi^{(1)}$ is approximated as

$$\Phi^{(1)} = \frac{c - a\Phi_i}{b + a\theta\Delta x} \quad . \tag{33}$$

This derivative is then extrapolated constantly in a narrow band following [63]. This derivative is then used in order to define second order accurate ghost cells Φ^{gh} in the interface vicinity following the linear extrapolation of [63]. Finally, the standard discretization of Eq. (31) is used for the Laplacian with Φ_{i+1}^{gh} defined by the procedure above.
Note that the procedure to impose Robin boundary conditions has been extended to general boundary conditions in [62] by fixing $b = 0$ for Dirichlet boundary conditions or $a = 0$ for Neumann boundary conditions.
In Fig. 8, the error with GCM either from a dimensional-splitting or a normal direction discretization is shown. As pointed out by [61], the use of dimensional-splitting leads to first-order accuracy when applied to Neumann or Robin boundary conditions while the normal direction discretization provides second-order



(a) Convergence of $L_2(\Phi)$ **(b)** Convergence of $L_\infty(\Phi)$

Figure 8: Mesh convergence of $L_2(\Phi)$ and $L_\infty(\Phi)$ for the static diffusion problem with GCM using a dimensional-splitting (——) or a normal direction discretization (- - -)

accuracy for all boundary condition types. However, the Dirichlet boundary condition is less accurate than the dimensional-splitting approach.
Then, it is mandatory to use a normal direction discretization for Neumann or Robin boundary conditions to retrieve consistent gradient approximation while it is more interesting to use dimensional-splitting for Dirichlet boundary conditions.

3.3. Methods selection for VOF and SLS framework

To conclude this investigation, several approaches have been compared based on explicit or implicit interface representation of the interface from f or ϕ. As the VOF framework considered in this paper uses a phase barycenter arrangement of scalars, CCM is used with an accuracy between first and second-order. Note that this method could be improved to obtain second-order accuracy which is not the scope of this paper.
For the SLS framework, the source term method is not as accurate and sharp as IBM. Moreover, looking at methods to impose Neumann or Robin boundary conditions, CCM and GCM based on cell center arrangement of scalars are both second-order accurate with similar accuracy magnitude. It is then more interesting to use the recent GCM based on normal direction discretization to apply Neumann or Robin boundary conditions instead of using the CCM which requires to reconstruct an explicit interface. It is more accurate to rely on the dimensional-splitting discretization to apply Dirichlet boundary conditions.
With the above discretization choices, the SLS framework relying on GCM is more accurate than the VOF framework based on the phase barycenter CCM.

4. Evaluating normal gradients

The imposition of flux discontinuity can now be handled by one of the methods presented above. They all require an accurate approximation of the normal gradient at the interface which is essential for phase-change simulations as it is used for computation of $\dot{m}$. This numerical aspect has been broadly investigated in the literature for phase-change simulations as it conditions the accuracy of the predicted

mass variation during a simulation. The main challenge of this computation is to provide high-order accuracy with limited information. Several approaches can respond to these constraints such as one-sided differences and ghost cell differences. In this section, these methods are evaluated and improvements are proposed to obtain an accurate gradient computation using different interface representations.
For a quantitative investigation of the gradient reconstruction accuracy, the test case defined in Sec. 3 is reused. The errors for a phase normal gradient $\Phi^{(1)}_{\Gamma,p}$ with $p = l, g$ at the interface is computed as the normalized L_2 error norm. Note that it is interesting to investigate both gas and liquid normal gradients as their related fields are not of the same nature.

4.1. One-sided differences

The idea behind one-sided differences is to consider the available information of the single phase and include the interface value Φ_Γ in the gradient stencil. This implies the use of the interface location $\mathbf{x}_\Gamma$ from an interface representation (see Appendix 8.1). First, one-sided gradients can be evaluated with dimensional-splitting from $\Phi^{(1)}_\Gamma = \nabla\Phi \cdot \mathbf{n}_\Gamma$. In the Level Set framework, one-sided differences can be designed using the signed distance ϕ [30] where, in a cell C_i with the interface lying between x_i and x_{i-1}, the gradient is computed as

$$\left.\frac{\partial\Phi}{\partial x}\right|_{\Gamma,i} = \frac{\Phi_i - \Phi_{\Gamma,i}}{\Delta x_\Gamma} \quad , \tag{34}$$

with Δx_Γ the distance between x_i and x_Γ. In [38], a higher-order finite difference defined in cell C_i is used if the interface lies between x_i and x_{i-1}

$$\left.\frac{\partial\Phi}{\partial x}\right|_{\Gamma,i} = \frac{-\Delta x_\Gamma^2\Phi_{i-1} + \left(\Delta x_\Gamma^2 - \Delta x^2\right)\Phi_i + \Delta x^2\Phi_{\Gamma,i}}{\Delta x\Delta x_\Gamma\left(\Delta x + \Delta x_\Gamma\right)} \quad . \tag{35}$$

In [34], an improvement is proposed to handle the singular case $\Delta x_\Gamma \approx 0$

$$\left.\frac{\partial\Phi}{\partial x}\right|_{\Gamma,i} = \frac{\left(\Delta x^2 - \Delta x_\Gamma^2\right)\Phi_{i-2} + \left(-4\Delta x^2 + \Delta x_\Gamma^2\right)\Phi_{i-1} + 3\Delta x^2\Phi_{\Gamma,i}}{\Delta x\left(\Delta x + \Delta x_\Gamma\right)\left(2\Delta x + \Delta x_\Gamma\right)} \quad . \tag{36}$$

This new stencil is used only when $\Delta x_\Gamma < \epsilon_c$ defined as $\epsilon_c = 10^{-3}\Delta x$. Actually, it is possible to use any arbitrary order finite differences for this gradient computation. For instance, fourth-order one-sided gradients are used in [28] .
A discretization based the PLIC reconstruction has been proposed in the VOF framework [13]

$$\left.\frac{\partial\Phi}{\partial x}\right|_{\Gamma,i} = \frac{\Phi_{i+1} - \Phi_{\Gamma,i}}{\Delta x + \Delta x_\Gamma} \quad , \tag{37}$$

In this expression, the computational point is always chosen such that $\Delta x + \Delta x_\Gamma > \Delta x$ to avoid any troublesome behavior when Δx_Γ is arbitrarily small.
In practice, the evaporation rate evaluation $\dot{m}$ have to be performed at the interface position $\mathbf{x}_\Gamma$. As the finite differences defined by Eq. 37 is defined at the cell center x_i, it contains an inherent first order error. Eq. 34, Eq. 35 and Eq. 36 are then of huge interest as they correspond to a Taylor expansions around x_Γ.

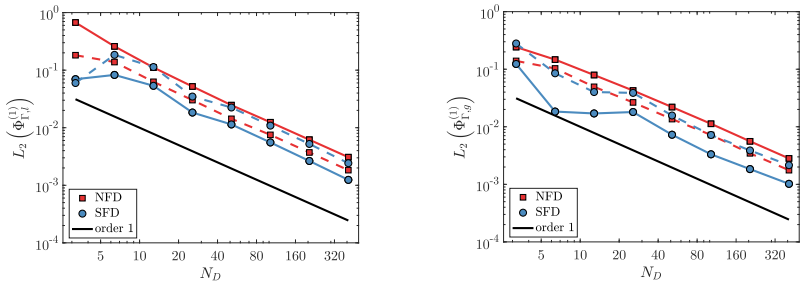
Other methods use normal finite differences instead of dimensional-splitting. In the Front-Tracking framework, a method based on a fixed normal distance to the interface is presented in [16] to prevent any singular behaviour for gradient discretizations when δx_Γ is arbitrary small. This takes the form of a first-order one-sided difference

$$\Phi^{(1)}_{\Gamma,i} = \frac{\widetilde{\Phi}(\delta) - \Phi_{\Gamma,i}}{\delta} \quad , \tag{38}$$

with $\widetilde{\Phi}(\delta)$ retrieved from linear interpolation of the cells adjacent to C_i at the point which is located at a normal distance δ from the interface. δ can be chosen between Δx and $2\Delta x$ without noticeable impact on the accuracy. It has also been used in a VOF framework relying on the PLIC reconstruction barycenter [35]. In [64], a second-order discretization is proposed

$$\Phi^{(1)}_{\Gamma,i} = \frac{\widetilde{\Phi}(2\delta) - 4\widetilde{\Phi}(\delta) + 3\Phi_{\Gamma,i}}{2\delta} \quad , \tag{39}$$

with $\widetilde{\Phi}(2\delta)$ and $\widetilde{\Phi}(\delta)$ obtained from linear interpolation.
A last approach was presented in [65, 66, 36] to avoid using the mixed cells values which are averaged. The normal gradient is computed in a set of pure cells by first-order derivatives. Then, the gradients in the mixed cells are retrieved either using simple average of neighbour pure cells [65, 66] or using a weighting [36].
In the present comparison, both normal finite differences (NFD) and dimensional-splitting finite differences (SFD) are tested either with first or second-order discretization. In practice, the first-order and second-order discretizations of NFD method correspond to Eq. (38) and Eq. (39) respectively based on a PLIC reconstruction. SFD method uses the first order discretization of Eq. (34) and the second-order discretization Eq. (36) with Δx_Γ computed from Level-Set (see Appendix 8.1). These choices are representative of the current methods of the literature.



(a) Convergence of $L_2\left(\Phi^{(1)}_{\Gamma,l}\right)$ **(b)** Convergence of $L_2\left(\Phi^{(1)}_{\Gamma,g}\right)$

Figure 9: Mesh convergence of $L_2\left(\Phi^{(1)}_{\Gamma,l}\right)$ and $L_2\left(\Phi^{(1)}_{\Gamma,g}\right)$ for normal finite differences (NFD) and dimensional-splitting finite differences (SFD) using first-order (——) or second-order discretization (- - -).

In Fig. 9, all methods exhibit a first-order convergence for the gradient. This rate is expected for FND-OS1 and SFD-OS1 while it is not trivial for FND-OS2 and

SFD-OS2. In fact, using the location of the interface in the stencil, which has been demonstrated to be second-order accurate in Appendix 8.1 prevents any one-sided discretization to reach high-order accuracy. This point was reported in [34] where a higher-order representation of the interface using Gradient Augmented Level Set allowed to recover a second-order accuracy of the gradients.
Moreover, the interface value Φ_Γ is not known a priori for general phase-change simulations. The quantity is reconstructed with a limited order of accuracy which could induce even more error in the gradient computation. This aspect is not discussed in most of the work using the one-sided gradient method as it is mainly used in boiling applications where T_Γ is imposed at the saturation temperature of the pure liquid.
These limitations can be circumvented by keeping a uniform discretization based on a ghost cell method. Then, neither the interface position nor the interface value are required.

4.2. Ghost cell method

The dimensional-splitting GCM presented in Sec. 3.2.2 can also be employed to compute the interface value and normal gradient. In [12, 17, 19], ghost cells are defined in the unknown part of the domain, and linear interpolation and central differences are used to retrieve the quantity and the gradient respectively. This gives the second-order interpolation of Φ_Γ at $\mathbf{x}_\Gamma$

$$\Phi_\Gamma = (1-\theta)\Phi_{p,i} + \theta\Phi^{gh}_{p,i+1} \quad , \tag{40}$$

with $\theta = |\phi_i|/(|\phi_i| + |\phi_{i+1}|)$.
In 2D or 3D configuration, Φ^x_Γ, Φ^y_Γ (and Φ^z_Γ) are obtained from Eq. (40) and projected to the correct value Φ_Γ with

$$\Phi_\Gamma = \Phi^x_\Gamma n_x^2 + \Phi^y_\Gamma n_y^2 \quad . \tag{41}$$

The second-order difference located at the cell center is written as

$$\left.\frac{\partial\Phi}{\partial x}\right|_{\Gamma,i} = \frac{\Phi^{gh}_{i+1} - \Phi_{i-1}}{2\Delta x} \quad . \tag{42}$$

Note that in this expression, $\left.\frac{\partial\Phi}{\partial x}\right|_{\Gamma,i}$ is not defined at the interface but at $\mathbf{x}_i$. In [14], the gradient are upwind to give a gradient evaluation closer to the interface

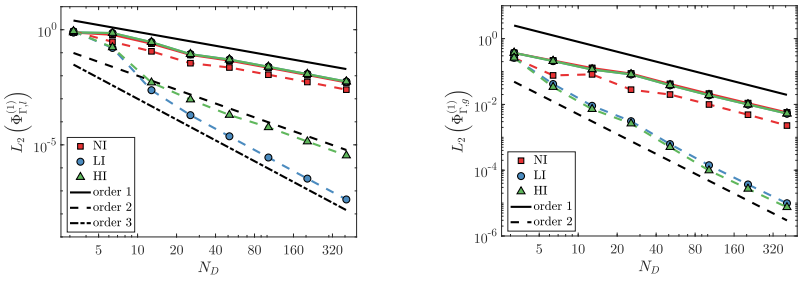
$$\left.\frac{\partial\Phi}{\partial x}\right|_{\Gamma,i} = \frac{\Phi^{gh}_{i+1} - \Phi_i}{\Delta x} \quad . \tag{43}$$

Even if this is still not providing $\nabla\Phi$ exactly at the interface position, it reduces the first-order error of a standard central difference located at the cell center by centering the gradient at the closest cell face to the interface.
The gradients computed from uniform discretization are defined at the cell center and contain an inherent first-order error which cannot be overcome by improving the stencil of the gradient evaluation.

For higher-order interface gradients, an interpolation to the interface position is then required. This can be done either by linear or harmonic interpolation applied to $\Phi^{(1)}$ (when interface lies between x_{i-1} and x_i)

$$\Phi^{(1)}_{\Gamma,i} = \frac{\Phi^{(1)}_i|\phi_{i-1,j}| + \Phi^{(1)}_{i-1}|\phi_i|}{|\phi_i| + |\phi_{i-1}|} \quad , \qquad \Phi^{(1)}_{\Gamma,i} = \Phi^{(1)}_i \Phi^{(1)}_{i-1} \frac{|\phi_i| + |\phi_{i-1}|}{\Phi^{(1)}_i|\phi_i| + \Phi^{(1)}_{i-1}|\phi_{i-1}|} \quad . \tag{44}$$

In practice, the gradient is first computed at the cell center x_i. Then, the interface gradient is only computed in cells where ϕ changes sign. In a multi-dimensional case, ϕ can change sign in several directions, in this case, $\Phi^{(1)}_\Gamma$ is retrieved by weight average of all the interpolated values based on squared normal components. Note that this interpolation step is only required when using quadratic interpolation or higher-order. In the case of linear interpolation, the ghost cells are already defined such that the gradient is constant in the normal direction, then the evaluation of the gradient between two cells apart from an interface cells are expected to be substantially equal. For the comparison, no interpolation (NI), linear interpolation (LI) or harmonic interpolation (HI) of the gradient with ghost cells defined either by linear extrapolation (LE) or quadratic extrapolation (QE) are investigated. The exact signed distance to the interface is used in this study. However, a second-order signed distance is expected to lead to the same accuracy order with a potential shift in accuracy magnitude.



(a) Convergence of $L_2\left(\Phi^{(1)}_{\Gamma,l}\right)$ **(b)** Convergence of $L_2\left(\Phi^{(1)}_{\Gamma,g}\right)$

Figure 10: Mesh convergence of $L_2\left(\Phi^{(1)}_{\Gamma,l}\right)$ and $L_2\left(\Phi^{(1)}_{\Gamma,g}\right)$ for gradients either based on LE (——) or QE (- - -) with no additional interpolation (NI), linear interpolation (LI) or harmonic interpolation (HI).

A first-order trend is presented in Fig. 10 for gradients evaluated from both LE and QE. This is an expected result for LE as the extrapolated field is second-order and the gradient computed using centered finite differences leads to a first-order gradient approximation. On the other hand, QE improves accuracy while maintaining a first-order convergence rate. From the same reasoning as for the linear extrapolation, a second-order gradient would be expected from a centered finite difference on a third order field.
In fact, the first-order trend is due to the location of the gradient evaluation. As expected, an interpolation process does not improve the accuracy of gradient from LE. Indeed, they all collapse to the same error trend.

For QE, the gradient is not constant anymore in the normal direction and an interpolation is expected to improve the accuracy of the evaluation. In Fig. 10, it is clear that both LI and HI improve the accuracy and rate of convergence of the normal gradient evaluation. Surprisingly, a super-convergence is observed in Fig. 10a for LI. This can be explained by the quadratic nature of the Φ_l field which leads to an exact extrapolation using QE, and an exact interpolation of the gradients using LI. For the exponential functional of Φ_g, the expected second-order convergence is retrieved in Fig. 10a.
As a conclusion, using QE for Φ^{gh} and LI or HI for $\Phi_\Gamma^{(1)}$ leads to second-order normal gradients.

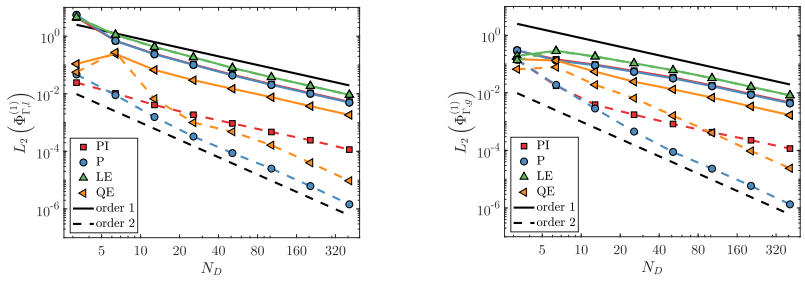
4.3. Least-square minimization

A last methodology is to compute the interface value and normal gradient from a functional fitting based on least-square minimization (LSM) of the field. This approach retrieves these quantities at the interface location without any additional interpolation and dimensional-splitting in opposition to GCM. Moreover it can be applied to both uniform and non-uniform (based on phase barycenter) discretizations without additional considerations.
In two-phase flow simulations, LSM was initially used to achieve convergence in the curvature evaluation for unstructured meshes [67] or from a distance prompt to reconstruction errors [68]. For phase-change simulations, this idea has already been used in [18] to fit a linear profile for T and Y. It was found that LSM allows mitigation of numerical errors by increasing the sample size of points in the stencil.
The Taylor expansion around the interface location $\mathbf{x}_{\Gamma,i}$ gives

$$\Phi(\mathbf{x}) = \Phi(\mathbf{x}_{\Gamma,i}) + \Delta\mathbf{x}^\intercal \nabla\Phi(\mathbf{x}_C) + \Delta\mathbf{x}^\intercal \nabla\nabla\Phi(\mathbf{x}_\Gamma)\Delta\mathbf{x} \quad , \tag{45}$$

with $\Delta\mathbf{x} = \mathbf{x} - \mathbf{x}_\Gamma$.
Keeping the first-order derivatives and neglecting second-order derivative corresponds to a linear functional fitting with an expected first-order accuracy at the point of interest while keeping the Taylor expansion up to second-order derivatives corresponds to a quadratic functional fitting with an expected second-order accuracy at the point of interest.
Note that other fitting can be used such as in [31] where the temperature evolution in the interface vicinity is fitted by an Erf profile.
In the following, three different stencils are proposed to solve this problem either based on linear or quadratic fit.
The first method is based on non-uniform arrangement and proposes to only use the points belonging to the considered phase with their values located at the phase barycenter. Then a cell located in the phase p has the following distance to the phase barycenter $\Delta\mathbf{x} = \mathbf{x}_{cm,p} - \mathbf{x}_{\Gamma,i}$ and the value $\Phi\left(\mathbf{x}_{cm,p}\right)$. An extension of this first system, referred as PI, can be build by adding the interface value in the stencil. Then the associated distance is $\Delta\mathbf{x} = 0$ and the corresponding value is $\Phi\left(\mathbf{x}_{\Gamma,i}\right) = \Phi_\Gamma$.
It has been shown in Sec. 4.1 that errors on the interface location could lead to important errors in the gradient evaluation when the interface value was explicitly

used in the stencil. To demonstrate this behaviour, another discretization, named P, is also proposed without the interface value in the least square minimization.
Finally, uniform discretization is also investigated. Then the distance is just $\Delta\mathbf{x} = \mathbf{x}_C - \mathbf{x}_\Gamma$ with values taken at the cell center $\Phi(\mathbf{x}_C)$. When the stencil implies values of mixed cells or the other phase, the values are retrieved either using LE or QE. The above discretizations PI, P, LE or QE are used either with linear (LSL) or quadratic (LSQ) functional fitting. A weight inspired from [36] is used to give more importance to stencil points in the normal direction to the interface. The test case performed here rely on a phase barycenter arrangement of Φ to be representative of a gradient evaluation used in the VOF framework.



(a) Convergence of $L_2\left(\Phi^{(1)}_{\Gamma,l}\right)$ **(b)** Convergence of $L_2\left(\Phi^{(1)}_{\Gamma,g}\right)$

Figure 11: Mesh convergence of $L_2\left(\Phi^{(1)}_{\Gamma,l}\right)$ and $L_2\left(\Phi^{(1)}_{\Gamma,g}\right)$ for different discretizations using LSL (——) or LSQ (- - -).

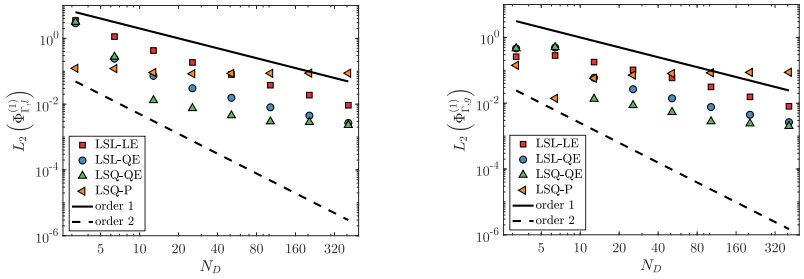
The results are presented in Fig. 11 where LSL is first-order accurate while LSQ is second-order unless for LE and PI. As for the results of NI, using LSL in combination with QE improves the accuracy magnitude but does not allow second-order accuracy. LE errors collapse as the slope is already imposed by the linear extrapolation, then using LSL or LSQ leads to the same accuracy.
The discretizations P and PI do not provide the same accuracy for high resolutions. Indeed, the combination of LSQ with PI is second-order accurate for coarse to medium meshes while it decreases to first-order convergence for high resolutions while combining LSQ with P lead to second-order accuracy for all resolutions. This is explained by the second-order error introduced by the interface location approximation already observed in Sec. 4.1.
Finally, using only one-sided values of Φ (P) is more accurate than using the extrapolated values (QE) combined with LSQ. In fact, P only implies exact values of Φ while QE introduces a third order error on more than half of the values used in the least square minimization.

4.4. Methods selection for VOF and SLS framework

For completeness, the same gradient methodologies are applied to a second-order scalar field obtained from the static diffusion problem presented in Sec. 3. This last study is relevant for evaporation problems where the scalar field is not exact and

evolves in time through diffusion and convection. As demonstrated in Sec. 3, the handling of flux jumps is second-order accurate at best. Then, a first-order accuracy is expected on gradient evaluation. For a phase-change solver, it means that $\dot{m}$ is evaluated at first-order from the gradient reconstruction.
The most accurate gradient evaluations retained in the above study are directly applied on the scalar field obtained from the static diffusion problem with Dirichlet boundary conditions of Sec. 3. This problem is analogous to the resolution of the energy equation with computation of the evaporation rate $\dot{m}$ from $\mathcal{M}_T$.

VOF framework with a combination of CCM and LSM. Using an explicit representation of the interface with quantities defined at cell barycenters, the scalar field $\overline{\Phi}^{VOF}$ obtained from Sec. 3.2.1 is between first and second-order. Then, the normal gradient is expected to be at most first-order accurate.

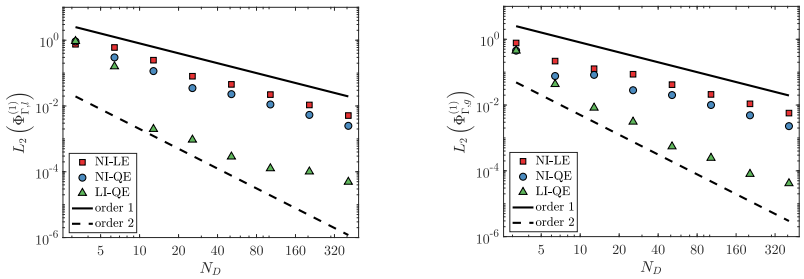


(a) Convergence of $L_2\left(\Phi^{(1)}_{\Gamma,l}\right)$ **(b)** Convergence of $L_2\left(\Phi^{(1)}_{\Gamma,g}\right)$

Figure 12: Mesh convergence of $L_2\left(\Phi^{(1)}_{\Gamma,l}\right)$ and $L_2\left(\Phi^{(1)}_{\Gamma,g}\right)$ for the different least square approaches applied to $\overline{\Phi}^{VOF}$

The results are presented in Fig. 12 for the LSL-LE, LSL-QE, LSQ-QE and LSQ-P methods. The accuracy is drastically decreased compared to the results presented in Sec. 4.3. The LSQ-P method is not convergent anymore while other methods show a convergent behaviour. LSL provides a solid first-order convergence of the normal with a more accurate computation using QE. However, LSQ leads to error saturation for high resolution while it is the most accurate method for low to medium meshes. In fact, trying to fit a quadratic functional is not the most suited method for its lack of monotonicity as discussed in [31] where an Erf functional was fitted instead. From these considerations, LSL is the most robust method and will be used hereafter.

SLS framework with GCM. With an implicit representation of the interface, the scalar field $\overline{\Phi}^{SLS}$ obtained from Sec. 3.2.2 is second-order accurate. Then, the normal gradient is expected to be first-order accurate.
In Fig. 13, the error convergence of NI-LE, NI-QE and LI-QE are represented. All methods have an asymptotic first-order convergence rate while LI-QE is essentially second-order for low to medium resolutions. The use of quadratic extrapolation



(a) Convergence of $L_2\left(\Phi^{(1)}_{\Gamma,l}\right)$ **(b)** Convergence of $L_2\left(\Phi^{(1)}_{\Gamma,g}\right)$

Figure 13: Mesh convergence of $L_2\left(\Phi^{(1)}_{\Gamma,l}\right)$ and $L_2\left(\Phi^{(1)}_{\Gamma,g}\right)$ for the different combination of field extrapolation and gradient interpolation applied to $\overline{\Phi}^{SLS}$

improves gradient accuracy while performing the additional linear interpolation improves it even more.

Recap of method choices. From the above considerations, the LE and QE lead to first-order gradient evaluations while the use of quantities restrained in a single-phase close to the interface did not converge for a VOF framework Moreover, LSQ was proven to be less robust in the literature and did not lead to substantial accuracy improvements in the coarse and fine meshes. Then LSL combined with LE or QE is used to obtain first-order accurate gradient close to the interface.
In a SLS framework, the use of QE is a huge improvement in accuracy if combined with a linear interpolation of the gradient at the interface location. The difference is more important compared to QE applied to a VOF framework, as $\overline{\Phi}^{SLS}$ is already more accurate than $\overline{\Phi}^{VOF}$. Another reason is the difference of accuracy for QE described in Appendix 8.2 if normal are computed from the PLIC reconstruction or from ϕ directly. As observed in Sec. 3 for imposing flux discontinuity, the SLS framework is again more accurate than the VOF framework for evaluating normal gradients.
However, the same conclusion is not clear for phase-change applications from this investigation on a static test case. Indeed, it has been demonstrated here that the introduction of new errors in the scalar field can lead to different convergence behaviour for the gradient evaluation using LE and QE. This is then interesting to pursue this investigation on dynamic test cases where important transport errors are also introduced and can impact gradient evaluations and the handling of flux discontinuities at the interface using VOF and SLS frameworks.
In the following section, these methods are investigated for all test cases and are referred to VOF-LE, VOF-QE, SLS-LE and SLS-QE respectively.

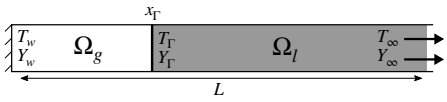


Figure 14: Illustration of the planar Stefan problem

Case	Frank sphere	Boiling	Evaporation
c	**on**	**on**	**on**
$\dot{m}$	**on**	**on**	**on**
T_l	**on**	off	**on**
T_g	off	**on**	**on**
u	off	**on**	**on**
Y	off	off	**on**
T_Γ, Y_Γ	off	off	**on**

Table 1: Activated equations for the Stefan problems

5. Full phase-change procedure

In this section, the goal is to study different phase-change procedures built from a combination of methods evaluated in Sec. 3 and 4. Apart from the choice of flux discontinuity imposition and normal gradient evaluation, all other numerical methods used in the solver are detailed in Appendix 8.3.
For this purpose, planar Stefan flow and spherical Stefan flow problems are investigated.
The planar solutions allow to consider 1D simulations where the transport of VOF is exact in space while it is quasi-exact for SLS. Moreover, the quantity reconstructions at the interface are not prompt to geometric errors which can be prevalent as demonstrated in Appendix 8.1. In the 3D test case, topology errors are also included and allow to demonstrates interest of the reconstruction choices in more complex applications.

5.1. Planar Stefan problems

The planar Stefan flow problems considered here are successively introduced with different levels of coupling between mass, momentum, energy and species given in Tab. 1. This allows a step-by-step validation of the solver along with the impact of the solver choices for different degree of complexity in the test case investigated. All the presented planar Stefan flow problems consist of the same domain of length L, with wall boundaries at the left T_w, Y_w and outflow boundaries at the right T_∞, Y_∞ as illustrated in Fig. 14. The gas phase is located between the wall and the interface while the liquid domain is supposed infinite starting from the interface position. Phase change occurs because of the jump in energy flux or species flux at the interface and creates the motion of the interface. The corresponding analytical position x_Γ and velocities u_Γ, u_l and u_g derived as [59, 69, 34, 18]

$$x_\Gamma = 2\gamma\sqrt{\lambda t} \quad , \qquad u_\Gamma = \gamma\sqrt{\frac{\lambda}{t}} \quad , \qquad u_l = \gamma\sqrt{\frac{\lambda}{t}}\left(1-\frac{\rho_g}{\rho_l}\right) \quad , \qquad u_g = 0 \quad . \tag{46}$$

with γ a diffusion layer and λ the thermal diffusivity such that $\lambda = k/(\rho c_p)$.

The analytical solution of temperature and species depends on the Stefan problem and is given in the corresponding sections. The fluid properties and physical set-up used in these problems are given in Tab. 2 and 3 respectively.

Case	Frank sphere		Boiling		Evaporation	
Phase	Liquid	Gas	Liquid	Gas	Liquid	Gas
ρ (kg/m^3)	1	1	1000	1	1000	1
μ (Pa · s)	1	1	$1 \cdot 10^{-2}$	$1 \cdot 10^{-5}$	$1 \cdot 10^{-2}$	$1 \cdot 10^{-5}$
k (W/m/K)	1	1	$1 \cdot 10^{-1}$	$1 \cdot 10^{-2}$	$1 \cdot 10^{-1}$	$1 \cdot 10^{-2}$
c_p (J/kg/K)	1	1	1000	1000	1000	1000
λ (m^2/s)	1	1	$1 \cdot 10^{-7}$	$1 \cdot 10^{-5}$	$1 \cdot 10^{-7}$	$1 \cdot 10^{-5}$
M (kg/mol)	–	–	–	–	–	0.018
$\mathscr{L}_{vap}$ (J/kg)	1	–	$1 \cdot 10^{6}$	–	$1 \cdot 10^{6}$	–
$\mathcal{D}_v$ (m^2/s)	–	–	–	–	–	$1 \cdot 10^{-5}$
T_{sat} (K)	0	–	373.15	–	373.15	–
σ (N/m)	1	–	0.01	–	0.01	–

Table 2: Physical properties of fluids considered in the planar Stefan problems

Case	Frank sphere	Boiling	Evaporation
L (m)	10	$1 \cdot 10^{-3}$	$1 \cdot 10^{-3}$
t_0 (s)	1	0.1	$1 \cdot 10^{-2}$
t_f (s)	1.5	0.2	0.1
x_Γ (m)	0.86	$1.4119 \cdot 10^{-4}$	$7.3204 \cdot 10^{-5}$
T_Γ (K)	0	373.15	296.12
T_∞ (K)	0.5	373.15	323.15
T_w (K)	0	383.15	296.12
Y_Γ	–	–	0.22106
Y_w	–	–	0.2
Y_∞	–	–	1
γ	0.43	0.0706	0.11575

Table 3: Physical set-up for the planar Stefan problems

In the following, metrics are defined to quantify the errors associated to the 1D test cases. The $L_1(\Phi_p)$ and $L_\infty(\Phi_p)$ errors of the phase scalars with $p = l, g$ are defined as

$$L_1\left(\Phi_p\right) = \frac{\frac{1}{N_p}\sum_{i=1}^{N_p}\left|\Phi_{p,i,exact} - \Phi_{p,i,num}\right|}{\Phi_\infty} \quad , \qquad L_\infty\left(\Phi_p\right) = \max_i \frac{\left|\Phi_{p,i,exact} - \Phi_{p,i,num}\right|}{\Phi_\infty} \quad . \tag{47}$$

N_p corresponds to the number of cells belonging to the phase p. $\Phi_{p,i,exact}$ is provided by the analytical solution while $\Phi_{p,i,num}$ is the numerical value provided by the solver at the end of the simulation.

The mass error and the phase velocity u_p with $p = l, g, \Gamma$ error can be expressed as

$$E(x_\Gamma) = \frac{\left|x_{\Gamma,exact} - x_{\Gamma,num}\right|}{L} \quad , \qquad E(u_p) = \frac{\max\limits_i \left|u_{p,exact} - u_{p,i,num}\right|}{u_{p,exact}} \quad , \tag{48}$$

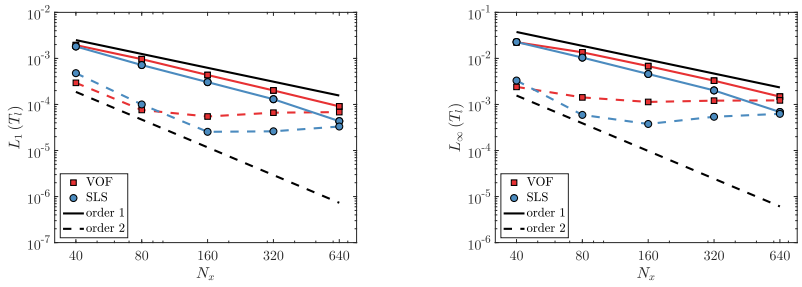
with L the domain length, $x_{\Gamma,exact}$ and $u_{p,exact}$ the exact interface position and phase velocity respectively, $x_{\Gamma,num}$ the numerical value at the end of the simulation and $u_{p,i,num}$ the numerical value in C_i at the end of the simulation.
It is very important that the different quantities of the physical set-up given in Tab. 3 are taken with a large number of significant digits to retrieve convergent behaviours. Indeed, the errors are already very low in 1D test cases and then become very sensitive to initialization errors.

5.1.1. Planar Frank sphere

The first test case aims to evaluate the accuracy of both $\dot{m}$ and the diffusion problem with a moving boundary condition. The planar Frank Sphere is a class of problem where two fluids of same properties (see Tab. 2) are separated by an interface. It has been used in [59] to assess a fourth order accurate scheme for the heat equation. In this mathematical problem, the momentum equation is not relevant as $[\rho]_\Gamma = 0$, the velocity is set to zero and does not evolve with time. Moreover, the species mass fraction equation is not considered neither because both fluids have the same properties, thus $Y = 1$ in the domain. Then the temperature at the interface is at saturation and is fixed to $T_\Gamma = T_{sat}$ given in Tab. 2. As the wall temperature T_w is also set to T_{sat}, the gas temperature is not solved neither and is simply set to $T_g = T_{sat}$. The evolution of liquid temperature is then given by the expression

$$T_l = T_w + \frac{T_\infty - T_\Gamma}{\mathrm{erfc}\,(\gamma)} \left(\mathrm{erfc}\,(\gamma) - \mathrm{erfc}\left(\frac{x}{2\sqrt{\lambda_l t}}\right)\right) \quad , \tag{49}$$

with the values of T_w, T_∞ and T_Γ given in Tab. 3 and λ_l given in Tab. 2. Therefore, phase change occurs because of the jump in energy flux at the interface and is driven by $\frac{\partial T_l}{\partial x}\Big|_\Gamma$. The simulation starts with the analytical temperature of T_l at $t_0 = 1$ s with the initial position of the interface $x_\Gamma = 0.86$ m and stops at $t_f = 1.5$ s.
In Fig. 15, the errors associated to the planar Frank sphere problem are provided. First, $L_1(T_l)$ and $L_\infty(T_l)$ exhibits a second-order trend for the coarse meshes in Fig. 15a and Fig. 15b while it saturates from $N_x = 160$ when using quadratic extrapolations (VOF-QE and SLS-QE). In fact, using a quadratic extrapolation on a second-order field at most does not lead to second-order accuracy as illustrated in Sec. 4.4 which could explain this saturation. On the other hand, the gradient evaluations based on linear extrapolation (VOF-LE and SLS-LE) converge between first and second-order. It can be observed that no convergence is retrieved for $E(x_\Gamma)$ and $E(u_\Gamma)$ in Fig. 15c and Fig. 15d when using QE. In fact, the first mesh N_x already provide a position evolution which is very close to the exact solution when using QE. Therefore, the error related to LE is predominant for the LE-based cases which allows the first-order convergence. For QE, as the error related to gradient reconstruction is drastically reduced, another source of error could become predominant with mesh



(a) Convergence of $L_1(T_l)$

(b) Convergence of $L_\infty(T_l)$

(c) Convergence of $E(x_\Gamma)$

(d) Convergence of $E(u_\Gamma)$

Figure 15: Mesh convergence of errors for the planar Frank sphere problem using LE (——) or QE (- - -).

resolution leading to the non-convergent behaviour. This source of error could come from the boundary condition applied to liquid temperature which is not at infinity or from the time integration.
Overall, even if QE-based gradient reconstructions do not converge in the high resolution limit, it provides more accurate quantities compared to LE-based gradient reconstructions for low resolutions. Also, SLS is globally more accurate than VOF for all metrics.
From these observations, the choice of extrapolation is not clear. On the one hand, LE provides a convergent behaviour but on the other hand, QE is more accurate for all meshes considered here. However, for a test case with more error sources as the planar boiling, it is not clear if QE still provides more accurate results. At this stage, no conclusion on the best choice can be made and both extrapolations are still compared in the next study about planar boiling.

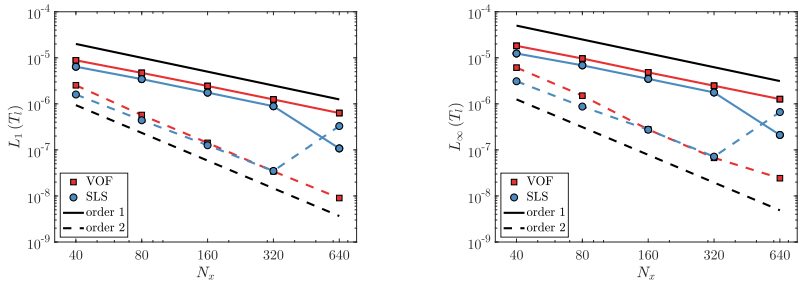
5.1.2. Planar boiling

A step further to general vaporization is to consider a physical problem with a density jump between both phases. Then, the momentum equation is solved with a velocity jump caused by phase change at the interface. The planar boiling problem was first considered in [69] and investigated in [34, 18] where liquid ans gas are considered to be the fluid and its vapour with fluid properties of Tab. 2. Then, the interface is at the

saturation conditions of the pure liquid and the species equation is not relevant to solve. The interface temperature is fixed to $T_\Gamma = T_{sat}$. The temperature at infinity is set such that $T_\infty = T_{sat}$, therefore, the liquid temperature is set to $T_l = T_{sat}$ and does not require to be solved in opposition to the Frank sphere problem. The evolution of gas temperature is then given by the expression

$$T_g = T_w + \frac{T_\Gamma - T_w}{\text{erf}\,(\gamma)} \,\text{erf}\left(\frac{x}{2\sqrt{\lambda_g t}}\right) \quad , \tag{50}$$

with the values of T_w and T_Γ given in Tab. 3 and λ_g given in Tab. 2. Phase change still occurs because of the jump in energy flux at the interface and is driven by the evaluation of $\left.\frac{\partial T_g}{\partial x}\right|_\Gamma$. The simulation starts with the analytical temperature of T_g at $t_0 = 0.1$ s with the initial position of the interface $x_\Gamma = 1.4119 \cdot 10^{-4}$ m and stops at $t_f = 0.2$ s.



(a) Convergence of $L_1(T_g)$

(b) Convergence of $L_\infty(T_g)$

(c) Convergence of $E(x_\Gamma)$

(d) Convergence of $E(u_l)$

Figure 16: Mesh convergence of errors for the planar boiling problem using LE (——) or QE (- - -).

In Fig. 16, the errors associated to the planar boiling problem are provided. As opposed to the Frank sphere problem, a clear second-order convergence is observed in Fig. 16a , Fig. 16b and Fig. 16c for $L_1(T_l)$, $L_\infty(T_l)$ and $E(x_\Gamma)$ when using QE unless for the $N_x = 640$ simulation using SLS-QE. For LE, the expected first-order convergence is observed. The order of convergence is not clear for $E(u_l)$ in Fig. 16d even if a second-order tendency can also been observed for quadratic extrapolations whereas it is first-order for linear extrapolations.

For that test case, the use of QE leads to a serious improvement in accuracy. It is quite surprising as the system of equations induces more sources of error due to the mass and momentum equations and quadratic extrapolations are not expected to improve the order of convergence as demonstrated in Sec. 4.4. However, the absence of transport and topology errors in planar test cases could explain this super-convergence. Now, it is interesting to pursue the same study for evaporation where all equations and closures have to be explicitly solved.

5.1.3. Planar evaporation

Finally, the general evaporation problem of [18] is presented where the gas part of the domain is a mixture of inert gas and liquid vapour. Thus, the species mass fraction equation evolves in time and needs to be solved in the gas. The analytical solution for species mass fraction evolution is

$$Y = Y_w + \frac{Y_\Gamma - Y_w}{\text{erf}\,(\gamma)} \,\text{erf}\left(\frac{x}{2\sqrt{\mathcal{D}_v t}}\right) \quad , \tag{51}$$

with Y_w and Y_Γ given in Tab. 3 and $\mathcal{D}_v$ given in Tab. 2. Note that γ is the same layer for species and temperature as the Lewis number $Le = \lambda_g/\mathcal{D}_v = 1$ in this case.
In such a case, the quantities at the interface are not imposed explicitly and need to be retrieved from the phase-change procedure. The exact value of T_Γ and Y_Γ can be derived analytically with the additional relation from Eq. (4) as in [18]:

$$T_\Gamma = T_w + \frac{\mathscr{L}_{vap}}{c_{p,g}} \sqrt{\frac{\mathcal{D}_v}{\lambda_g}} \frac{Y_\Gamma - Y_w}{Y_\Gamma - 1} \quad , \tag{52}$$

As T_Γ is computed numerically, it does not match exactly the initial liquid temperature $T_l = T_\infty$. Then the liquid temperature also needs to be solved to take into account the potential derivations of T_Γ from T_∞.
The evolution of gas temperature is still given by Eq. (50) with the values of T_w and T_Γ given in Tab. 3 and λ_g given in Tab. 2. Phase change still occurs because of the jump in energy flux at the interface and is driven by the evaluation of $\left.\frac{\partial T_g}{\partial x}\right|_\Gamma$. The simulation starts with the analytical temperature of T_g at $t_0 = 0.01$ s with the initial position of the interface $x_\Gamma = 7.3204 \cdot 10^{-5}$ m and stops at $t_f = 0.1$ s.

Full phase change procedure study. As the interface quantities T_Γ and Y_Γ are not imposed anymore and needs an explicit reconstruction at the interface, this last test case allows to investigate the phase-change procedure.
The overall phase change procedure needs the combination of three important features:

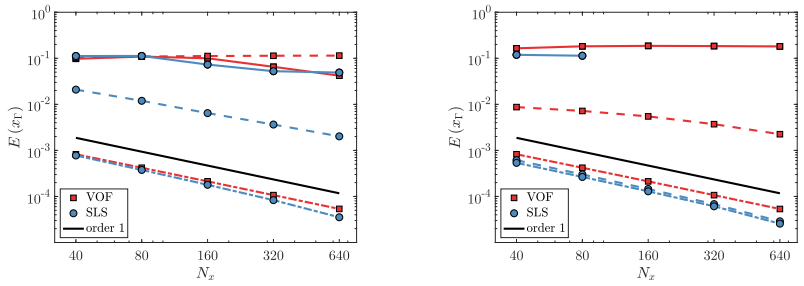
1. The computation of $\dot{m}$;
2. The computation of T_Γ and Y_Γ;
3. The boundary condition at the interface for T_l, T_g and Y.

In [15], the choice of applying a Robin boundary condition on the species instead of a classic Dirichlet boundary condition Y_Γ is not innocuous. The reason for that is the

tight coupling between the scalars and the computation of $\dot{m}$, T_Γ and Y_Γ. The coupling of $\dot{m}$ with pressure and interface regression is not discussed as it is expected to be a fairly low coupling compared to energy and species equations.
The aim of this study is to bring out the impact of the different combinations of phase change procedure. Some key points of the following discussion are already notified in [18].
First, the Dirichlet boundary condition applied to Y is investigated by trying different quantity reconstruction at the interface. Then, the same study is performed with Robin boundary condition applied to Y for comparison.
Different reconstructions of Y_Γ and T_Γ are investigated for this choice of embedded boundary conditions at the interface:

- A first-order extrapolation $Y_\Gamma = Y_C$ leading to method $\mathcal{Q}_1(Y)$;
- A second-order extrapolation Eq. (41) leading to method $\mathcal{Q}_2(Y)$;
- An iterative process based on the general system Eq. (4) with the gradient reconstruction of [18] and [30] for VOF and SLS respectively. This leads to method $\mathcal{Q}_3(T_l, T_g, Y)$.

In the results shown here, the evaporation rate is computed either from the temperature $\mathcal{M}_1(T_l, T_g)$ or the species mass fraction $\mathcal{M}_2(Y)$.
All the analysis relies on the interface position error $E(x_\Gamma)$ as it is sufficient to detect inconsistent behaviour of the phase-change procedure.
First, the phase change procedure is designed to impose Dirichlet boundary conditions for all scalars T_l, T_g and Y. The boundary conditions are then direct functions of the quantities at the interface $\mathcal{Q}$, this is expressed as $T_l|_{\partial\Omega}(\mathcal{Q})$, $T_g\big|_{\partial\Omega}(\mathcal{Q})$ and $Y|_{\partial\Omega}(\mathcal{Q})$.

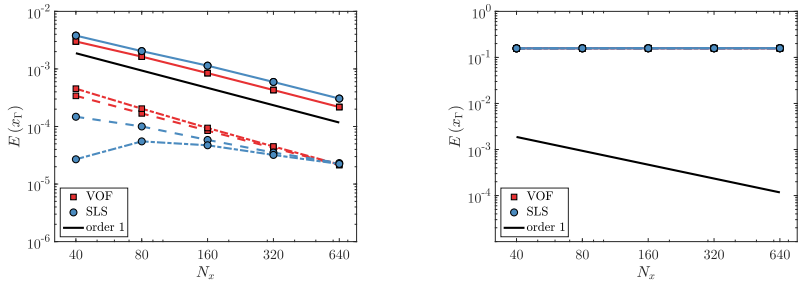


(a) $\mathcal{M}_1(T_l, T_g)$ **(b)** $\mathcal{M}_2(Y)$

Figure 17: Mesh convergence of $E(x_\Gamma)$ for the planar evaporation problem with $\mathcal{Q}_1$ (——), $\mathcal{Q}_2$ (- - -), $\mathcal{Q}_3$ (-···-) and different $\mathcal{M}$ for Dirichlet boundary conditions applied to Y

In Fig. 17, the mesh convergence of $E(x_\Gamma)$ is given. As expected, the reconstruction based on $\mathcal{Q}_1(Y)$ leads to an inconsistent scheme both with $\mathcal{M}_1(T_l, T_g)$ and $\mathcal{M}_2(Y)$ (see Fig. 17a and Fig. 17b respectively). This is due to the strong coupling between $\mathcal{Q}_1(Y)$ and $Y|_{\partial\Omega}(\mathcal{Q})$ as Y_Γ is directly set to the closest point to the interface. Then the diffusion operator of Y is inconsistent in the case where the cell center belongs to the gas phase: in such case, the interface flux is always $F_\Gamma^{(Y)} = (Y_C - Y_C)/\Delta x = 0$.

However, the extrapolation procedure allows to retrieve convergent behaviours. For SLS, using $\mathcal{Q}_2(Y)$ relaxes the coupling but can lead to inconsistent fluxes too in the limit cases where the interface is close to a cell center. For VOF, analysis on $\mathcal{Q}_2(Y)$ is not as simple because of the least square system solved to extrapolate the values, however relaxation of the coupling is observed using $\mathcal{M}_2(Y)$ while it is inconsistent using $\mathcal{M}_1(T_l, T_g)$.
Finally, the use of $\mathcal{Q}_3(T_l, T_g, Y)$ drastically lowers the coupling as the quantities depend on all scalar fields and not only on Y. Then the results are improved and retrieve a convergent behaviour.
The same phase change procedure is used but with a Robin boundary condition for Y with the mesh convergence of $E(x_\Gamma)$ given in Fig. 18.



(a) $\mathcal{M}_1(T_l, T_g)$ **(b)** $\mathcal{M}_2(Y)$

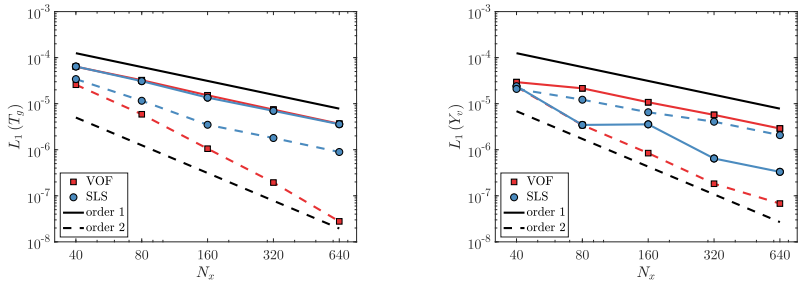
Figure 18: Mesh convergence of $E(x_\Gamma)$ for the planar evaporation problem with $\mathcal{Q}_1$ (——), $\mathcal{Q}_2$ (- - -), $\mathcal{Q}_3$ (-···-) and different $\mathcal{M}$ for Robin boundary conditions applied to Y

This new choice of boundary condition creates a dependency on the evaporation rate computed with $\mathcal{M}$ instead of the quantities at the interface. The boundary condition applied to Y is then a function of $Y|_{\partial\Omega}(\mathcal{M})$ either computed from $\mathcal{M}_1(T_l, T_g)$ or $\mathcal{M}_2(Y)$. As illustrated in Fig. 18a, this modification of the boundary condition allows to be consistent using any of the quantity methodologies for both VOF and SLS if the evaporation rate is retrieve from $\mathcal{M}_1(T_l, T_g)$. This is explained by the low coupling between the boundary condition obtained from the temperature fields and the quantity reconstruction obtained from extrapolation of Y. However, when $\mathcal{M}_2(Y)$ is used instead, a tight coupling is created between the boundary condition applied on Y and the field Y itself. This always leads to inconsistent schemes as shown in Fig. 18b.
The above study is summarized in Tab. 4 where all well-suited procedures for phase change are represented with ✓ while inconsistent procedures are tagged by ✗.
Note that a well-suited method leads to convergent behaviour but with huge disparity in the accuracy. In the present solver, the choice has been to use $\mathcal{M}_1(T_l, T_g)$ by default, because it gives a natural way to handle both boiling and evaporation without problems. Also, using Robin boundary conditions for species with the $\mathcal{Q}_2$ reconstruction seems to be the best trade off between implementation complexity, efficiency and accuracy.

		Dirichlet		Robin	
		$\mathcal{M}_1(T_l, T_g)$	$\mathcal{M}_2(Y)$	$\mathcal{M}_1(T_l, T_g)$	$\mathcal{M}_2(Y)$
VOF	$\mathcal{Q}_1(Y)$	X	X	✓	X
	$\mathcal{Q}_2(Y)$	X	✓	✓	X
	$\mathcal{Q}_3(T_l, T_g, Y)$	✓	✓	✓	X
SLS	$\mathcal{Q}_1(Y)$	X	X	✓	X
	$\mathcal{Q}_2(Y)$	✓	✓	✓	X
	$\mathcal{Q}_3(T_l, T_g, Y)$	✓	✓	✓	X

Table 4: Sum-up of well-suited (✓) and inconsistent (X) procedures for phase change.

Gradient reconstruction study. Now that the phase change procedure has been chosen, the study of Sec. 5.1.1 and 5.1.2 can be performed to compare LE and QE.



(a) Convergence of $L_1(T_g)$

(b) Convergence of $L_1(Y)$

(c) Convergence of $E(x_\Gamma)$

(d) Convergence of $E(u_l)$

Figure 19: Mesh convergence of errors for the planar evaporation problem with gradients based on LE (——) and QE (- - -).

Again, VOF-QE leads to a second-order convergence for all error metrics as shown in Fig. 19. However, this is not the case for SLS-QE which is between first and second-order accuracy. By looking at Fig. 19b, the QE method is less accurate than LE causing the lack of second-order convergence for the other quantities as they are all coupled. The difference between VOF and SLS convergence rates could be explained by the exact transport of VOF while SLS is still prompt to spatial errors. On the other hand, LE leads to an expected first-order convergence of the quantities as

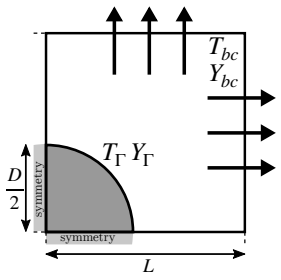


Figure 20: Illustration of the static droplet evaporation

Case	Stefan spherical	
Phase	Liquid	Gas
ρ (kg/m^3)	700	1
μ (Pa · s)	$3.26 \cdot 10^{-4}$	$1 \cdot 10^{-5}$
k (W/m/K)	$1.61 \cdot 10^{-1}$	$5.2 \cdot 10^{-2}$
c_p (J/kg/K)	2000	1000
M (kg/mol)	0.058	0.029
$\mathscr{L}_{vap}$ (J/kg)	–	$2.3 \cdot 10^6$
D_m (m^2/s)	–	$5.2 \cdot 10^{-5}$
T_{sat} (K)	329	–
σ (N/m)	0	–

Table 5: Physical properties of fluids in the static droplet evaporation

already presented in [18].
Overall, the same conclusions can be drawn for the general evaporation test case: using quadratic extrapolation to reconstruct the gradient at the interface drastically improves the results and provides a better convergence rate.

5.2. Static droplet evaporation

The static droplet evaporation is a spherical Stefan problem where topology evaluation takes an important role in the gradient and quantities reconstruction. The test case is also expected to be impacted by the ability of the interface tracking method to regress through transport. In 1D, the transport was almost exact, while in multidimensional test cases the interface capturing method has a huge impact. Here the goal is to assess the VOF and SLS solvers in a general multidimensional evaporation problem with convection effects limited to the Stefan flow induced by the evaporation process.
An acetone droplet of diameter $D = 0.1$ mm is placed in quiescent air and experience evaporation due to the gas conditions at infinity $T_\infty = 700$ K and $Y_\infty = 0$ with the fluid properties reported in Tab. 5. The problem of an isolated droplet evaporating in a gas at rest has been widely investigated in the literature. By assuming a quasi-steady gas phase consisting of the vapour and an inert gas, a pure liquid phase with uniform and stationary temperature, an analytical solution can be derived for temperature and species mass fraction fields along with droplet diameter evolution. The derivation of the solution can be found in [15, 21] and leads to the analytical solutions

$$T_g = T_\infty + \frac{\mathscr{L}_{vap}}{c_{p,g}} \left(1 - (1 - B_T)^{\left(1 - \frac{d}{D_\Gamma}\right)} \right) \quad , \tag{53}$$

$$Y = 1 + (Y_\infty + 1) \left(1 - (1 + B_M)^{\left(1 - \frac{d}{D_\Gamma}\right)} \right) \quad , \tag{54}$$

with $d = (x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2$ the diameter coordinate, D_Γ the droplet diameter, B_T and B_M the Spalding numbers related to heat and mass transfer

respectively such that

$$B_T = \frac{c_{p,g}\left(T_\infty - T_\Gamma\right)}{\mathscr{L}_{vap}} \quad , \qquad B_M = \frac{Y_\infty - Y_\Gamma}{Y_\Gamma - 1} \quad . \tag{55}$$

The temperature T_l is expected to be equal to the interface temperature $T_l = T_\Gamma$.
As for the planar Stefan flow, the value of T_Γ and Y_Γ can be derived analytically with the additional relation from Eq. (4a) and Eq. (4b)

$$T_\Gamma = T_\infty - \frac{\mathscr{L}_{vap}}{c_{p,g}}\left(1 - \left(\frac{Y_\infty - 1}{Y_\Gamma - 1}\right)^{Le}\right) \quad , \tag{56}$$

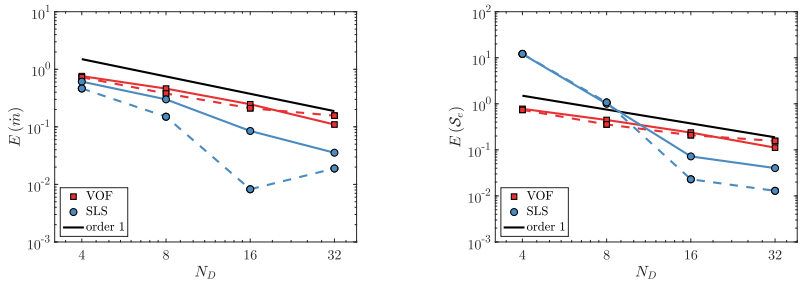
with $Le = \lambda_g / \mathcal{D}_v$ the Lewis number. By using and Clausius-Clayperon to close the system, $T_\Gamma = 294.94$ K and $Y_\Gamma = 0.43993$. The Spalding numbers can be deduced accordingly: $B_T = B_M = 0.7819$.
Finally, the diameter and evaporation rate evolutions are

$$D_\Gamma = \sqrt{D_0^2 - \frac{4\rho_g D_m}{\rho_l}\ln\left(1 + B_M\right)t} \quad , \qquad \dot{m} = \frac{2\rho_g \mathcal{D}_v \ln\left(1 + B_M\right)}{D_\Gamma} \quad . \tag{57}$$

The corresponding evaporation time for this test case with an initial diameter $D_0 = 1 \cdot 10^{-4}$ m is then $\tau_e = 0.029126$ s.
As the test case is performed in 3D on a cartesian grid, only a eighth of the domain is consider by using symmetries as illustrated in Fig. 20 to allow a convergence study with reasonable computational time. Moreover, the domain size is reduced to $[0, 4D_0]^3$ such that boundary conditions are not far enough from the droplet to be considered at infinity. Then, the boundary condition for temperature and species mass fraction are imposed by using the exact solution evolving in time $T_{bc}(t, \mathbf{x})$ and $Y_{bc}(t, \mathbf{x})$ as in [15] instead of forcing T_∞ and Y_∞ which would lead to erroneous results. At these boundaries, the flow is allowed to exit freely by imposing zero pressure boundaries.
The simulation is performed during $t_f/\tau_e = 2.5 \times 10^{-3}$ for four different meshes leading to $N_D = 4$ to $N_D = 32$ cells in the diameter for the different methods studied in the planar Stefan flow problem.
The error metrics used here are based on the slope of the d^2 law $\mathcal{S}_e = 1/\tau_e$ and the error on $\dot{m}$ defined as

$$E\left(\mathcal{S}_e\right) = \left|1 - \frac{1 - \left(\frac{D_{num}}{D_0}\right)^2}{\mathcal{S}_e t_f}\right| \quad , \qquad E\left(\dot{m}\right) = \frac{|\dot{m}_{exact} - \dot{m}_{num}|}{\dot{m}_{exact}} \quad , \tag{58}$$

with D_{num} and $\dot{m}_{num}$ the diameter obtained numerically and $\dot{m}_{exact}$ the exact evaporation rate at t_f.
As the study aims to quantify the accuracy of the phase-change procedure in a multi-dimensional configuration, the potential perturbations caused by the spurious behaviour of the liquid velocity close to the interface is mitigated by imposing $\kappa = 0$ and $\mathbf{u}_\Gamma = -\dot{m}/\rho_l \mathbf{n}_\Gamma$. However, the true value of $\mathbf{u}_l$ is still used in the transport of T_l.



(a) Convergence of $E(\dot{m})$ **(b)** Convergence of $E(\mathcal{S}_e)$

Figure 21: Mesh convergence of $E(\dot{m})$ and $E(\mathcal{S}_e)$ for the static droplet evaporation with gradients based on LE (——) and QE (- - -)

This simplification alleviate potential numerical issues arising from errors in the $\mathbf{u}_l$ reconstruction.

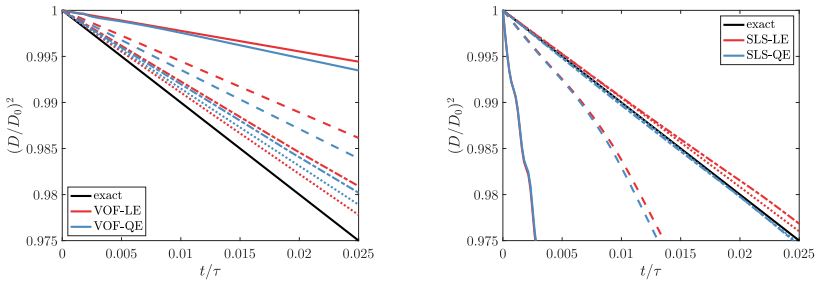
The convergences of $E(\dot{m})$ and $E(\mathcal{S}_e)$ are given in Fig. 21. The VOF-LE method converges to first-order for both metrics while VOF-QE loses accuracy for the finest grid $N_D = 32$. Otherwise, VOF-QE leads to a slight improvement in accuracy for the coarse meshes which is not important compared to the gain in the simulation of planar Stefan problems of Sec. 5.1.3. Then multidimensional effects are predominant for the VOF methodology and quadratic extrapolations do not seem to substantially improve results. This observation can explain the choice of linear extrapolation in the work of [18].

For SLS, Fig. 21a exhibits higher accuracy compared to VOF. Again, the conclusions are different from the planar test cases where VOF showed the most accurate results. This gives insight in the ability of Level Set to build more accurate reconstructions in multidimensional test cases as it was already the case for the normal and curvature. Moreover, the use of quadratic extrapolations lead to a significant gain in accuracy compared to the linear extrapolations as already observed in [15] using the same methodology.

However, the first two points in Fig. 21b exhibits errors higher than 100% of the expected slope $\mathcal{S}_e$ as shown in Tab. 6 which can be considered as inconsistent solutions. This can be explained by looking at the temporal evolution of the d^2 law in Fig. 22. The slopes are always under-predicted for VOF as shown in Fig. 22a while in SLS the slope is drastically over-predicted for $N_D = 4$ and $N_D = 8$ in Fig. 22b. Then SLS approach is unable to predict the regression correctly for low resolutions. This is explained by the reinitialization step which implies a mass loss which is more important than the regression due to phase change.

For completeness, a plot without evaporation has been added in Fig. 23 to show mass conservation convergence of the SLS method. It is obvious that the test cases with $N/D = 4$ and 8 cells in the diameter have a regression driven by the reinitialization. However, for better resolutions, the slope is very close to the expected regression when the reinitialization does not lead to significant mass loss.

To conclude, the multidimensional effects are more visible for VOF than SLS in term



(a) VOF **(b)** SLS

Figure 22: d2-law for VOF and SLS methods for $N_D = 4$ (——), $N_D = 8$ (- - -), $N_D = 16$ (-···-) and $N_D = 32$ (·····).

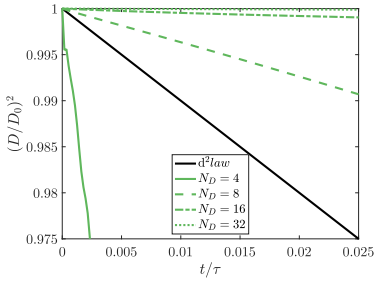


Figure 23: Effect of reinitialization on d2-law.

N_D	$E(\mathcal{S}_e)$ for VOF		$E(\mathcal{S}_e)$ for SLS	
	LE	QE	LE	QE
4	77%	74%	1200%	1200%
8	44%	36%	100%	100%
16	24%	21%	7.2%	2.3%
32	11%	15%	4.0%	1.3%

Table 6: Slope error $E(\mathcal{S}_e)$ at $t_f/\tau_e = 2.5 \times 10^{-3}$.

of accuracy of $\dot{m}$. This fact was well-known for the reconstruction of $\mathbf{n}$ and κ whereas it was not clear for $\dot{m}$ without this type of comparisons. However, the mass loss induced by reinitialization completely jeopardizes a good prediction of phase change through evaporation when using low resolutions while VOF is also inaccurate (more than 10% errors at the finest mesh in Tab. 6). Phase-change simulations still require high resolution to obtain accurate predictions of evaporation with the two frameworks presented in this paper which are representative of the state-of-the-art.

6. Conclusion

A thorough investigation of numerical methods applied to incompressible two-phase flow solvers with phase change has been conducted in this work.
First, it has been shown that imposing flux discontinuities at the interface with second-order accuracy can be achieved with any color function either using a source term approach or an immersed boundary method. However, immersed boundary methods allow to drastically improve the accuracy even if the discretization based on the phase barycenter is limited to first-order accuracy in this work.

Then, a methodology to retrieve normal gradients has been demonstrated at second-order accuracy close to the interface either using uniform or phase barycenter based arrangement of the quantities. More specifically, the GCM requires an additional interpolation at the interface position to avoid an inherent first-order error while LSM is used to deal with quantities defined at the phase barycenter. While this accuracy is obtained on exact fields, a last result is shown where these techniques are applied to a field prompt to second-order transport errors. In such case, all gradient evaluations are limited to first-order convergence.
Finally, full phase-change procedures are investigated on evaporation problems to also take into account transport errors. They are obtained by combining different IBM and computation of T_Γ and Y_Γ. The inherent inconsistency of some of the combination choices has been demonstrated while the most accurate phase-change procedure has been selected for both VOF and SLS frameworks. The interest of using quadratic extrapolations to evaluate accurate gradients has been enlighten on 1D and 3D Stefan problems.
Overall, the numerical experiments have demonstrated the complexity and subtlety in the manipulation of numerical methods to solve two-phase flows with phase change. One small error introduced at one of the steps to build the phase-change procedure can jeopardize the whole solver accuracy. Moreover, all the efforts put on the design of sophisticated numerical methods still lead to simulations which require high resolution to obtain acceptable predictions of mass evaporated during a simulation. Future research must focus on the extension of IBM methods to higher-order accuracy as already initiated in [59] for Dirichlet boundary conditions using GCM. In their work, fourth-order has been demonstrated on a class of static diffusion problems. For CCM, higher-order methods would require to augment the order of the explicit interface reconstruction by using either PROST [70] or QUASI [71] in the VOF framework or GALS [34] in the Level-Set framework. Finally, the problem of mass loss in Standard Level-Set could be solved by using more sophisticated reinitialization methods [72] or by coupling Level-Set with VOF [23].

7. Acknowledgements

The support of the ANR Project MIMETYC (ANR-17-CE22-0003) and SubSuperJet (ANR-14-CE22- 0014) is acknowledged. A part of this work was performed using HPC resources from the mésocentre computing center of Ecole CentraleSupélec and Ecole Normale Supérieure Paris- Saclay supported by CNRS and Région Ile-de-France.

8. Appendix

8.1. Interface location

The interface location error highly depend on the choice of interface representation:

- For a PLIC representation, the interface location in a given cell $\mathbf{x}_{\Gamma,i}$ is defined by the interface fragment barycenter as illustrated in Fig. 24a.

(a) VOF **(b)** Level-Set

Figure 24: Definition of $\mathbf{x}_{\Gamma,i}$ for VOF and Level-Set representations

- For a signed distance representation, the interface location is defined between two cell centres by linear interpolation. If the interface lies between $\mathbf{x}_{i-1}$ and $\mathbf{x}_i$ as in Fig. 24b then

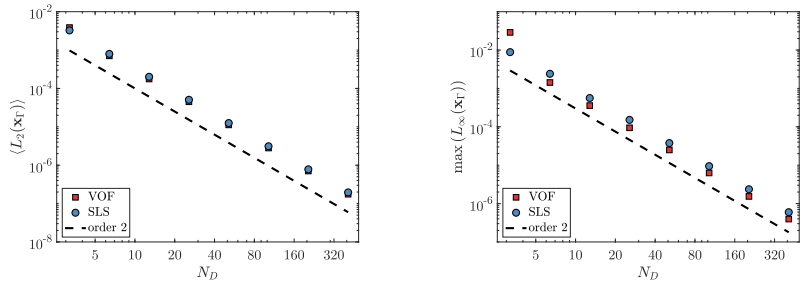
$$\mathbf{x}_{\Gamma,i} = \theta\mathbf{x}_{i-1} + (1-\theta)\mathbf{x}_i \quad , \tag{59}$$

  with $\theta = |\phi_i|/(|\phi_{i-1}| + |\phi_i|)$.

The error on the interface location is defined as in [34] : the exact signed distance at the interface is zero by definition while the computed signed distance based on the numerical interface location is

$$\phi\left(\mathbf{x}_{\Gamma,i}\right) = R - r(\mathbf{x}_{\Gamma,i}) \quad . \tag{60}$$

Then, the L_2 and L_∞ norms of $\phi\left(\mathbf{x}_{\Gamma,i}\right)$ are evaluated on 100 circles randomly located in the domain to meet as much configurations as possible. The final metric is $\langle L_2(\mathbf{x}_\Gamma)\rangle$ the mean of $L_2(\mathbf{x}_\Gamma)$ and $\max\left(L_\infty(\mathbf{x}_\Gamma)\right)$ the maximum of $L_\infty(\mathbf{x}_\Gamma)$ over all those configurations.



(a) Convergence of $\langle L_2(\mathbf{x}_\Gamma)\rangle$ **(b)** Convergence of $\max\left(L_\infty(\mathbf{x}_\Gamma)\right)$

Figure 25: Mesh convergence of $\langle L_2(\mathbf{x}_\Gamma)\rangle$ and $\max\left(L_\infty(\mathbf{x}_\Gamma)\right)$ based on VOF or Level-Set

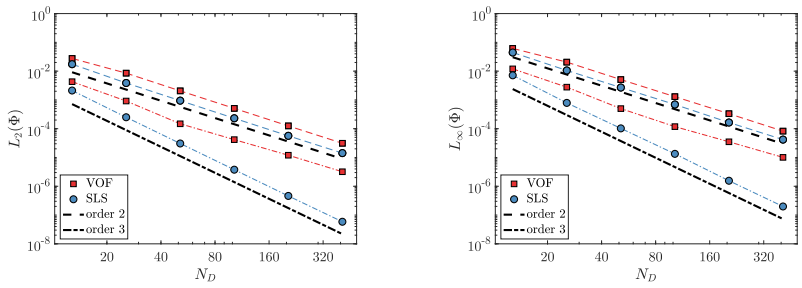
In Fig. 25, both PLIC and ϕ representations show a second-order accuracy as expected.

8.2. Extrapolation method

The main feature required for applying GCM is the extrapolation of scalars from one phase to another. The method used here is the PDE extrapolation of [63]. Here, only linear extrapolation (LE) and quadratic extrapolation (QE) are investigated either using PLIC or ϕ for the normal reconstruction. The extrapolation is performed from Ω_l to Ω_g delimited by the circle of radius $R = 0.2$ centered at $(0.5, 0.5)$ in a domain $[1 \times 1]$. The target scalar field Φ is defined as in [61] in the whole domain

$$\Phi = \exp\left(4\left(x - 0.5\right)\left(y - 0.5\right)\right) \quad . \tag{61}$$

The goal of this test case is to extrapolate the scalar in the unknown domain and compare its values to the target scalar field.



(a) Convergence of $L_2(\Phi)$ **(b)** Convergence of $L_\infty(\Phi)$

Figure 26: Mesh convergence of $L_2(\Phi)$ and $L_\infty(\Phi)$ for VOF and Level-Set using LE (- - -) or QE (-···-).

In Fig. 26 are presented $L_2(\Phi)$ and $L_\infty(\Phi)$ for linear and quadratic extrapolations. It is interesting to notice that the normal approximation has an impact on extrapolation accuracy. The PLIC representation provide less accurate normals compared to ϕ which impacts the accuracy of the extrapolated field in both $L_2(\Phi)$ and $L_\infty(\Phi)$ metrics. This discrepancy is increasing for quadratic extrapolation and can have impacts on the gradient evaluation.
The interested reader can refer to [73] for a more complete study of extrapolation accuracy.

8.3. Solver details

A brief survey of the solver used in this work is given. The momentum equation Eq. (1b) is solved with a one-field representation of the velocity using a classical projection method [74] to ensure mass conservation. The prediction step uses a mass-momentum consistent methodology for the convective term [75]. The convective fluxes are computed using WENO5 [76] and degenerate to first order upwind when the stencil implies points from both liquid and gas. The diffusion term follows second order discretization. The pressure jump condition is prescribed using GFM [44] with curvature computed from height function [77] in VOF or from finite differences in SLS.

For the energy conservation, Eq. (1c) is split into the transport of T_l and T_g coupled through the Dirichlet boundary condition T_Γ. In the same manner, the species mass fraction Eq. (1d) is considered only in the gas phase with a boundary condition at the interface. The convective terms are discretized using BQUICK [78] and degenerate to first order upwind close to the interface. For the diffusion treatment, two paradigms are presented for imposing the flux discontinuity. The first one makes use of a two-fluid formulation based on the PLIC reconstruction to solve scalars at their phase barycentre in a VOF framework with the method presented in Sec. 3.2.1. The other one uses a jump-condition formulation which divides the domain into liquid and gas based on the sign of the distance function in a SLS framework as detailed in Sec. 3.2.2.
The computation of $\dot{m}$ is performed using Eq. (4a) as described in Sec. 4.2. As the discretization is not uniform using the two-fluid formulation, LSM is used with VOF while standard finite differences are used for SLS. Then, $\dot{m}$ is constantly extrapolated in a narrow band around the interface as advised in [33].
Finally, the color function (either f for VOF or ϕ for SLS) is transported with the interface velocity $\mathbf{u}_\Gamma = \mathbf{u}_l - \frac{\dot{m}}{\rho_l}\mathbf{n}_\Gamma$. Here, $\mathbf{u}_l$ is obtained through the Stefan flow cancellation technique of [19] to assure the divergence-free property which is crucial for good mass conservation [12]. The VOF scheme used in this work is the dimensional-splitting method of [79] which allows to conserve mass up to machine precision while keeping a fairly simple and efficient implementation. For SLS, the standard method of [80] is used here with 2 iterations of the reinitialization after each time step.
All the discretization details are available in [73].

References

[1] Ashim K Datta. Porous media approaches to studying simultaneous heat and mass transfer in food processes. ii: Property data and representative results. Journal of food engineering, 80(1):96–110, 2007.

[2] Jungho Kim. Spray cooling heat transfer: The state of the art. International Journal of Heat and Fluid Flow, 28(4):753–767, 2007.

[3] William A Sirignano. Fuel droplet vaporization and spray combustion theory. Progress in Energy and Combustion Science, 9(4):291–322, 1983.

[4] Richard Saurel and Rémi Abgrall. A simple method for compressible multifluid flows. SIAM Journal on Scientific Computing, 21(3):1115–1145, 1999.

[5] John W Cahn and John E Hilliard. Free energy of a nonuniform system. i. interfacial free energy. The Journal of chemical physics, 28(2):258–267, 1958.

[6] Salih O Unverdi and Grétar Tryggvason. A front-tracking method for viscous, incompressible, multi-fluid flows. Journal of Computational Physics, 100(1):25–37, 1992.

[7] Cyril W Hirt and Billy D Nichols. Volume of fluid (vof) method for the dynamics of free boundaries. Journal of computational physics, 39(1):201–225, 1981.

[8] Stanley Osher and James A Sethian. Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations. Journal of computational physics, 79(1):12–49, 1988.

[9] Damir Juric and Grétar Tryggvason. Computations of boiling flows. International journal of multiphase flow, 24(3):387–410, 1998.

[10] Samuel W.J. Welch and John Wilson. A Volume of Fluid Based Method for Fluid Flows with Phase Change. Journal of Computational Physics, 160(2):662–682, 2000.

[11] Gihun Son and Vijay K Dhir. Numerical simulation of film boiling near critical pressures with a level set method. 1998.

[12] Sébastien Tanguy, Thibaut Ménard, and Alain Berlemont. A Level Set Method for vaporizing two-phase flows. Journal of Computational Physics, 2007.

[13] Jan Schlottke and Bernhard Weigand. Direct numerical simulation of evaporating droplets. Journal of Computational Physics, 227(10):5215–5237, 2008.

[14] Javed Shaikh, Atul Sharma, and Rajneesh Bhardwaj. On sharp-interface level-set method for heat and/or mass transfer induced Stefan problem. International Journal of Heat and Mass Transfer, 96:458–473, 2016.

[15] Lucia Rueda Villegas, Romain Alis, Mathieu Lepilliez, and Sébastien Tanguy. A Ghost Fluid/Level Set Method for boiling flows and liquid evaporation: Application to the Leidenfrost effect. Journal of Computational Physics, 316:789–813, 2016.

[16] Muhammad Irfan and Metin Muradoglu. A front tracking method for direct numerical simulation of evaporation process in a multiphase system. Journal of Computational Physics, 337:132–153, 2017.

[17] Min Chai, Kun Luo, Changxiao Shao, Haiou Wang, and Jianren Fan. A coupled vaporization model based on temperature/species gradients for detailed numerical simulations using conservative level set method. International Journal of Heat and Mass Transfer, 2018.

[18] John Palmore and Olivier Desjardins. A volume of fluid framework for interface-resolved simulations of vaporizing liquid-gas flows. Journal of Computational Physics, 399:108954, 2019.

[19] Nicolò Scapin, Pedro Costa, and Luca Brandt. A volume-of-fluid method for interface-resolved simulations of phase-changing two-fluid flows. Journal of Computational Physics, 407:109251, 2019.

[20] Jonathan Reutzsch, Corine Kieffer-Roth, and Bernhard Weigand. A consistent method for direct numerical simulation of droplet evaporation. Journal of Computational Physics, 413:109455, 2020.

[21] Romain Alis. Simulation numérique directe de gouttes et de groupes de gouttes qui s' évaporent dans un écoulement laminaire ou turbulent. PhD thesis, Toulouse, ISAE, 2018.

[22] Michael S. Dodd, Danyal Mohaddes, Antonino Ferrante, and Matthias Ihme. Analysis of droplet evaporation in isotropic turbulence through droplet-resolved DNS. International Journal of Heat and Mass Transfer, 172:121157, 2021.

[23] Leandro Germes Martinez, Benjamin Duret, Julien Reveillon, and Francois Xavier Demoulin. International Journal of Multiphase Flow A new DNS formalism dedicated to turbulent two-phase flows with phase change. International Journal of Multiphase Flow, 143(July):103762, 2021.

[24] Muhammad Irfan and Metin Muradoglu. A front tracking method for particle-resolved simulation of evaporation and combustion of a fuel droplet. Computers and Fluids, 174:283–299, 2018.

[25] Michael Kuhn and Olivier Desjardins. An all-Mach, low-dissipation strategy for simulating multiphase flows. Journal of Computational Physics, page 110602, 2021.

[26] Ireneo Kikic and Alessandro Vetere. Evaluation of several literature equations to predict the vaporization enthalpies at the normal boiling point. Fluid phase equilibria, 309(2):151–154, 2011.

[27] Robert C Reid, John M Prausnitz, and Bruce E Poling. The properties of gases and liquids. 1987.

[28] Lubomír Bureš and Yohei Sato. Direct numerical simulation of evaporation and condensation with the geometric VOF method and a sharp-interface phase-change model. International Journal of Heat and Mass Transfer, 173, 2021.

[29] Louis C Antoine. Thermodynamic vapor pressures: New relation between the pressures and the temperatures (thermodynamique, tensions des vapeurs: Novelle relation entre les tensions et les temperatures). CR Hebd. Seances Acad. Sci, 107(681):836, 1888.

[30] Gihun Son and Nahmkeon Hur. A coupled level set and volume-of-fluid method for the buoyancy-driven motion of fluid particles. Numerical Heat Transfer: Part B: Fundamentals, 42(6):523–542, 2002.

[31] Chen Ma and Dieter Bothe. Numerical modeling of thermocapillary two-phase flows with evaporation using a two-scalar approach for heat transfer. Journal of Computational Physics, 233:552–573, 2013.

[32] Liguo Chen, Duc Nguyen, Sanjoy Banerjee, and Frédéric Gibou. A level set based sharp interface method for the multiphase incompressible Navier-Stokes equations with phase change. Journal of Computational Physics, 222(2):536–555, 2007.

[33] Sébastien Tanguy, Michaël Sagan, Benjamin Lalanne, Frédéric Couderc, and Catherine Colin. Benchmarks and numerical methods for the simulation of boiling flows. Journal of Computational Physics, 264:1–22, 2014.

[34] Lakshman Anumolu and Mario F. Trujillo. Gradient augmented level set method for phase change simulations. Journal of Computational Physics, 353:377–406, 2018.

[35] Jie Zhang and Ming Jiu Ni. Direct numerical simulations of incompressible multiphase magnetohydrodynamics with phase change. Journal of Computational Physics, 375:717–746, 2018.

[36] LC Malan, AG Malan, Stéphane Zaleski, and PG Rousseau. A geometric vof method for interface resolved phase change and conservative thermal energy advection. Journal of Computational Physics, 426:109920, 2021.

[37] Guillaume Sahut, Giovanni Ghigliotti, Guillaume Balarac, Manuel Bernard, Vincent Moureau, and Philippe Marty. Numerical simulation of boiling on unstructured grids. Journal of Computational Physics, 432:110161, 2021.

[38] Yohei Sato and Bojan Ničeno. A sharp-interface phase change model for a mass-conservative interface tracking method. Journal of Computational Physics, 249:127–161, 2013.

[39] Gim Yau Soh, Guan Heng Yeoh, and Victoria Timchenko. An algorithm to calculate interfacial area for multiphase mass transfer through the volume-of-fluid method. International Journal of Heat and Mass Transfer, 100:573–581, 2016.

[40] Chohong Min and Frédéric Gibou. Robust second-order accurate discretizations of the multi-dimensional heaviside and dirac delta functions. Journal of Computational Physics, 227(22):9686–9695, 2008.

[41] Raja Banerjee. A numerical study of combined heat and mass transfer in an inclined channel using the vof multiphase model. Numerical Heat Transfer, Part A: Applications, 52(2):163–183, 2007.

[42] George Strotos, Manolis Gavaises, Andreas Theodorakakos, and George Bergeles. Numerical investigation of the evaporation of two-component droplets. Fuel, 90(4):1492–1507, 2011.

[43] Z Guo, David F Fletcher, and Brian S Haynes. A review of computational modelling of flow boiling in microchannels. The Journal of Computational Multiphase Flows, 6(2):79–110, 2014.

[44] Ronald P. Fedkiw, Tariq Aslam, Barry Merriman, and Stanley Osher. A Non-oscillatory Eulerian Approach to Interfaces in Multimaterial Flows (the Ghost Fluid Method). Journal of Computational Physics, 152(2):457–492, 1999.

[45] Stéphane Popinet. Numerical models of surface tension. Annual Review of Fluid Mechanics, 50:49–75, 2018.

[46] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. ACM siggraph computer graphics, 21(4):163–169, 1987.

[47] James E Pilliod. An analysis of piecewise linear interface reconstruction algorithms for volume-of-fluid methods. U. of Calif., Davis., 1992.

[48] Chohong Min and Frédéric Gibou. A second order accurate level set method on non-graded adaptive cartesian grids. Journal of Computational Physics, 225(1):300–321, 2007.

[49] Xu Dong Liu, Ronald P. Fedkiw, and Myungjoo Kang. A Boundary Condition Capturing Method for Poisson's Equation on Irregular Domains. Journal of Computational Physics, 160(1):151–178, 2000.

[50] James W Purvis and John E Burkhalter. Prediction of critical mach number for store configurations. AIAA Journal, 17(11):1170–1177, 1979.

[51] S. Chen, Barry. Merriman, Stanley. Osher, and Peter Smereka. A Simple Level Set Method for Solving Stefan Problems. Journal of Computational Physics, 135(1):8–29, 1997.

[52] Richard B Pember, John B Bell, Phillip Colella, William Y Curtchfield, and Michael L Welcome. An adaptive cartesian grid method for unsteady compressible flow in irregular regions. Journal of computational Physics, 120(2):278–304, 1995.

[53] Peter McCorquodale, Phillip Colella, and Hans Johansen. A Cartesian grid embedded boundary method for the heat equation on irregular domains. Journal of Computational Physics, 173(2):620–635, 2001.

[54] Peter Schwartz, Michael Barad, Phillip Colella, and Terry Ligocki. A Cartesian grid embedded boundary method for the heat equation and Poisson's equation in three dimensions. Journal of Computational Physics, 211(2):531–550, 2006.

[55] Joseph Papac, Frédéric Gibou, and Christian Ratsch. Efficient symmetric discretization for the Poisson, heat and Stefan-type problems with Robin boundary conditions. Journal of Computational Physics, 229(3):875–889, 2010.

[56] Yen Ting Ng, Chohong Min, and Frédéric Gibou. An efficient fluid-solid coupling algorithm for single-phase flows. Journal of Computational Physics, 228(23):8807–8829, 2009.

[57] M. Meyer, A. Devesa, Stefan Hickel, Xiangyu. Y. Hu, and Nikolaus. A. Adams. A conservative immersed interface method for Large-Eddy Simulation of incompressible flows. Journal of Computational Physics, 229(18):6300–6317, 2010.

[58] Frederic Gibou, Ronald P. Fedkiw, Li Tien Cheng, and Myungjoo Kang. A second-order-accurate symmetric discretization of the Poisson equation on irregular domains. Journal of Computational Physics, 176(1):205–227, 2002.

[59] Frédéric Gibou and Ronald Fedkiw. A fourth order accurate discretization for the Laplace and heat equations on arbitrary domains, with applications to the Stefan problem, volume 202. Elsevier, 2005.

[60] WP Jones and KR Menzies. Analysis of the cell-centred finite volume method for the diffusion equation. Journal of Computational Physics, 165(1):45–68, 2000.

[61] Min Chai, Kun Luo, Changxiao Shao, Haiou Wang, and Jianren Fan. A finite difference discretization method for heat and mass transfer with Robin boundary conditions on irregular domains. Journal of Computational Physics, 400:108890, 2020.

[62] Min Chai, Kun Luo, Haiou Wang, Shuihua Zheng, and Jianren Fan. Imposing mixed Dirichlet-Neumann-Robin boundary conditions on irregular domains in a level set/ghost fluid based finite difference framework. Computers and Fluids, 214:104772, 2021.

[63] Tariq D Aslam. A partial differential equation approach to multidimensional extrapolation. Journal of Computational Physics, 193(1):349–355, 2004.

[64] Michael Dodd. Direct numerical simulation of droplet-laden isotropic turbulence. PhD thesis, 2017.

[65] Christian Kunkelmann. Numerical modeling and investigation of boiling phenomena. PhD thesis, 2011.

[66] Gaël Guédon. Two-Phase Heat and Mass Transfer Modeling: Flexible Numerical Methods for Energy Engineering Analyses. (August):132, 2013.

[67] Emilie Marchandise, Philippe Geuzaine, Nicolas Chevaugeon, and Jean François Remacle. A stabilized finite element method using a discontinuous level set approach for the computation of bubble dynamics. Journal of Computational Physics, 225(1):949–974, 2007.

[68] Robert Chiodi and Olivier Desjardins. A reformulation of the conservative level set reinitialization equation for accurate and robust simulation of complex multiphase flows. Journal of Computational Physics, 343:186–200, 2017.

[69] Steffen Hardt and Frank Wondra. Evaporation model for interfacial flows based on a continuum-field representation of the source terms. Journal of Computational Physics, 227(11):5871–5895, 2008.

[70] Yuriko Renardy and Michael Renardy. Prost: a parabolic reconstruction of surface tension for the volume-of-fluid method. Journal of computational physics, 183(2):400–421, 2002.

[71] SV Diwakar, Sarit K Das, and Thirumalachari Sundararajan. A quadratic spline based interface (quasi) reconstruction algorithm for accurate tracking of two-phase flows. Journal of Computational Physics, 228(24):9107–9130, 2009.

[72] Daniel Hartmann, Matthias Meinke, and Wolfgang Schröder. The constrained reinitialization equation for level set methods. Journal of Computational Physics, 229(5):1514–1535, 2010.

[73] Victor Boniou. On the numerical simulation of evaporating two-phase flows using sharp interface capturing methods. PhD thesis, Université Paris-Saclay, 2021.

[74] Alexandre Joel Chorin. On the Convergence of Discrete Approximations to the Navier-Stokes Equations. Mathematics of Computation, 23(106):341, 1969.

[75] Olivier Desjardins and Vincent Moureau. Methods for multiphase flows with high density ratio. Center for Turbulence Research Proceedings of the Summer Program, pages 313–322, 2010.

[76] Chi-Wang Shu and Stanley Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. Journal of computational physics, 77(2):439–471, 1988.

[77] Stéphane Popinet. An accurate adaptive solver for surface-tension-driven interfacial flows. Journal of Computational Physics, 228(16):5838–5866, 2009.

[78] Marcus Herrmann, Guillaume Blanquart, and Venkat Raman. Flux corrected finite volume scheme for preserving scalar boundedness in reacting large-eddy simulations. AIAA journal, 44(12):2879–2886, 2006.

[79] Gabriel D Weymouth and Dick K-P Yue. Conservative volume-of-fluid method for free-surface simulations on cartesian-grids. Journal of Computational Physics, 229(8):2853–2865, 2010.

[80] Mark Sussman, Peter Smereka, Stanley Osher, et al. A level set approach for computing solutions to incompressible two-phase flow. 1994.