



**HAL**  
open science

## Numerical simulation of boiling on unstructured grids

Guillaume Sahut, Giovanni Ghigliotti, Guillaume Balarac, Manuel Bernard,  
Vincent Moureau, Philippe Marty

► **To cite this version:**

Guillaume Sahut, Giovanni Ghigliotti, Guillaume Balarac, Manuel Bernard, Vincent Moureau, et al.. Numerical simulation of boiling on unstructured grids. *Journal of Computational Physics*, 2021, pp.110161. 10.1016/j.jcp.2021.110161 . hal-03126543

**HAL Id: hal-03126543**

**<https://hal.science/hal-03126543>**

Submitted on 19 Nov 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Numerical simulation of boiling on unstructured grids

Guillaume Sahut<sup>a,\*</sup>, Giovanni Ghigliotti<sup>a</sup>, Guillaume Balarac<sup>a,b</sup>, Manuel Bernard<sup>a</sup>,  
Vincent Moureau<sup>c</sup>, Philippe Marty<sup>a</sup>

<sup>a</sup>*Univ. Grenoble Alpes, CNRS, Grenoble INP, LEGI, 38000 Grenoble, France*

<sup>b</sup>*Institut Universitaire de France (IUF), Paris, France*

<sup>c</sup>*Normandie Univ., UNIROUEN, INSA Rouen, CNRS, CORIA, 76000 Rouen, France*

---

## Abstract

A numerical method for the simulation of multiphase flows with phase change on unstructured grids is presented. Based on the work of Tanguy et al. [31] using a Level Set/Ghost Fluid Method coupling for two-dimensional axisymmetric cartesian grids, we extend the method to two- and three-dimensional unstructured grids with the aim of taking a step towards realistic boiling simulations in industrial context. The mass transfer rate at the interface accounting for phase change is computed as a function of the liquid and vapor heat fluxes at the interface and of the latent heat of the fluid by means of a new framework improving the accuracy of differential operators. The mass transfer rate is then used in the projection method to solve Navier-Stokes equations and in the advection equation of the Level Set function to account for the interface movement due to phase change. An immersed Dirichlet boundary condition is imposed at the interface to ensure that boiling always occurs at saturation temperature. We demonstrate the accuracy of our method first in the case of a static growing bubble with a fixed constant mass transfer rate, and second with a mass transfer rate computed from the thermal fluxes at the interface. In both cases, the bubble radius at final time converges with grid refinement towards the theoretical value.

*Keywords:* Boiling, Two-phase flows, Multiphase flows, Phase change, Bubble growth, Unstructured grids, Simplicial grids, Heat and mass transfer, Mass transfer rate, Level Set method, Ghost Fluid Method, Node-centered Finite Volume

---

---

\*Corresponding author

*Email address:* `guillaume.sahut@univ-grenoble-alpes.fr` (Guillaume Sahut)

*Preprint submitted to Elsevier*

*November 19, 2020*

## 1. Introduction

Boiling is a physical phenomenon which plays an important role in various industrial processes. It occurs especially in steam cycles of thermal power stations, nuclear power plants, two-phase loops used in cooling of electronic components and spray cooling processes. It is also used in the condenser heat exchangers of every refrigerating machine or heat pump. The study of boiling proves to be a major stake not only from a theoretical and academic point of view but also from a technical and industrial point of view, where any improvement of boiling heat transfer would have a major impact given the number of applications.

Boiling, and in general phase-changing flows, are a specific case of two-phase flows. From a computational point of view, two families of approaches exist to simulate two-phase flows. The first approach consists in using Lagrangian markers to explicitly identify the position of the interface. Methods belonging to this family are referred to as front-tracking methods. The second consists in identifying the interface position through the use of an auxiliary field, that is advected within an Eulerian framework. Methods of this kind are referred to as front-capturing methods. The two most prominent examples of front-capturing methods are the Volume Of Fluid method (VOF) [13], based on the advection of a volume fraction function, and the Level Set method (LS) [25], based on the advection of a function representing the signed distance function to the interface.

In addition to the method used to locate the interface, numerical methods to compute two-phase flows can be differentiated on the strategy deployed to handle discontinuities (both in physical properties of the fluids and boundary conditions) at the interface. A first approach is to alleviate the discontinuity by spreading the variation through a few cells across the interface. This eases the numerical solution of the system of PDEs, but may introduce unphysical spreading of the discontinuity. Methods of this kind are denoted smooth interface methods. Another approach consists in retaining the discontinuity at the discretization level. While this seems satisfactory from a modelling point of view, a certain number of numerical techniques need to be deployed in order to retain the discontinuity despite solving PDEs that naturally tend to smoothen the solution. Such methods are denoted sharp interface methods. One of the most prominent realizations of sharp interface methods is the Ghost Fluid Method (GFM) proposed by Fedkiw et al. [9]. In the GFM, each physical field is divided in one field per phase, and each field of each phase is artificially extended beyond the interface in a continuous fashion. The extended values are called ghost values. They allow the computation of derivatives across the interface, since the discontinuity is artificially removed for the field of a given phase.

Direct Numerical Simulations (DNS) of boiling flows were pioneered by Welch [35] in 1995. The author developed a two-dimensional, moving-mesh finite-volume method for a single, weakly deformable bubble. A simple interface model based on surface tension and surface energy is used. The mass transfer rate at the interface is computed using moving grid triangles. The computations are performed on interface-dependent triangular grids. The results showed basic capabilities to track interfaces with phase change. Son and Dhir [29] used the Continuum Surface Force model [4] to compute the mass transfer rate and the velocity discontinuity at the interface in the context of nucleate boiling to investigate bubble release pattern depending on the heat flux. The interface is captured by a Level Set method [25]. Juric and Tryggvason [16] presented a numerical method to simulate liquid-vapor phase change in which the interface is followed by a Front-Tracking

method [34, 15, 32]. Interfacial source terms for surface tension, mass transfer and latent heat are added as indicator functions of the interface. The authors used finite differences on cartesian grids to simulate film boiling and bubble departure. Based on the work of Welch [35], Welch and Wilson [36] presented in 2000 two-dimensional numerical simulations of incompressible two-phase flows with phase change, using a VOF-based interface tracking method in conjunction with a mass transfer model and a model for surface tension. The authors derived a one-dimensional analytical test problem featuring a thin thermal layer propagating with the moving phase interface. The authors emphasized that this test problem isolates the ability of a method to accurately calculate the thermal layers responsible for driving the mass transfer in boiling flows. In 2007, Tanguy et al. [30] used a Level Set function to capture the interface coupled to the Ghost Fluid Method [9] to compute discontinuities at the interface, in order to simulate vaporizing flows (liquid-to-vapor phase change with multi-component species) in two-dimensional cartesian grids. The method is validated against various test-cases of vaporizing droplets yielding good agreement with analytical results. Also in 2007, Gibou et al. [10] used the same coupling for two-dimensional film boiling simulation, again with good accuracy. These two references demonstrate the efficiency of the Level Set/Ghost Fluid Method coupling, and thus mark a milestone in two-phase flow simulations with phase change. In 2014, Tanguy et al. [31] proposed a comparison of different numerical methods suited to simulations of two-phase flows with phase change. The liquid-vapor interface is captured using the Level Set method. The accuracies of the Ghost Fluid Method (sharp interface) and the delta function method (smooth interface) are compared to compute the normal velocity jump condition. The authors showed that smoothing the velocity jump condition at the interface could lead to a misleading mass prediction, whereas the Ghost Fluid Method performed well. Moreover, the authors demonstrated that high-order extrapolation methods on the thermal field allowed performing accurate and robust simulations for a thermally controlled bubble growth. Simulations of the growth of static and rising bubbles are presented. The computations are performed on two-dimensional axisymmetric cartesian grids.

Based on the work of Tanguy et al. [31], our goal is to present and assess a numerical method for the simulation of multiphase flows with phase change on unstructured grids. The numerical developments have been implemented in the YALES2 code, a community-based code developed by a network of several French labs within the Scientific Interest Group (GIS) “SUCCESS”. YALES2 is a numerical code whose general purpose is the solving of fluid mechanics problems. It is designed for three-dimensional unstructured grids in order to simulate single-phase and two-phase flows in complex geometries. Our methodology has been designed for unstructured grids, with distributed memory parallelism in order to take advantage of distributed computational power. Our configuration of interest is a vapor bubble surrounded by a liquid phase. Both the vapor and liquid phases are of the same chemical composition. We focus on the growth of the bubble due to the liquid phase heated above the saturation temperature. In addition to the complexity of the physical phenomena encountered at the interface when phase change is not considered (action of surface tension and parameter discontinuity, density and viscosity, as well as pressure discontinuity between the two phases), boiling contributes to the pressure discontinuity and also implies a velocity discontinuity at the interface between the liquid and vapor phases. Classical numerical methods for the simulation of two-phase flows are not designed to take into account a discontinuous velocity field

whose discontinuity is moreover coupled to the temperature field. Substantial efforts are then required to extend existing algorithms in order to simulate boiling flows, especially on unstructured grids. We present these developments along with some promising results concerning numerical simulations of boiling flows on unstructured grids. Section 2 introduces the physics of boiling, emphasizing the coupling between the Navier-Stokes equations and the heat equation by means of the velocity and pressure discontinuities at the interface computed as functions of the mass transfer rate. Section 3 presents our numerical method for the simulation of multiphase flows without phase change on unstructured grids in two and three dimensions. Section 4 extends the method to take into account phase change. Section 5 presents numerical results for the validation of the method in two and three dimensions, first with a fixed uniform and constant mass transfer rate, and second with a mass transfer rate computed as a function of the heat fluxes at the interface. Finally, Section 6 concludes this work and highlights some perspectives.

## 2. Governing equations and interface conditions

The liquid and vapor phases are assumed non-miscible and both phases are assumed incompressible. The subscripts ‘liq’ and ‘vap’ are used to name physical properties in the liquid and vapor phases, respectively. These properties are uniform in each phase but are usually different from one phase to the other (e.g. density, viscosity...). The Sharp Interface model is used in which the interface between the two phases is infinitely thin [21]. Consequently, physical properties admit a discontinuity at the interface denoted by

$$[A]_{\Gamma} = A_{\text{liq}} - A_{\text{vap}}, \quad (1)$$

where  $A$  is a per-phase uniform physical field and discontinuous at the interface  $\Gamma$ , and  $A_{\text{liq}}$  and  $A_{\text{vap}}$  are taken in a given infinitesimal neighborhood of the interface.

### 2.1. Physical model

In each phase, mass conservation, or the incompressibility hypothesis, reads

$$\nabla \cdot \mathbf{u} = 0, \quad (2)$$

where  $\mathbf{u}$  is the fluid velocity in the given phase. While this hypothesis is very closely satisfied for the liquid phase, care must be taken when applying it to the vapor phase. Within the boiling context, we assume that the vapor temperature does not depart significantly from the saturation temperature, so that thermal dilatation can be neglected. Indeed, in the test cases presented in this paper, this is the case.

Following the notations of Nguyen et al. [24] and Gibou et al. [10], we denote the interface normal velocity by  $\mathbf{W} = D\mathbf{n}$ , where  $\mathbf{n}$  is the interface normal vector. Mass conservation across the interface reads (see also [33])

$$\dot{m} = \rho_{\text{liq}} (\mathbf{u}_{\text{liq}} \cdot \mathbf{n} - D) = \rho_{\text{vap}} (\mathbf{u}_{\text{vap}} \cdot \mathbf{n} - D), \quad (3)$$

where  $\dot{m}$  is the mass transfer rate (mass per unit of time and surface) across the interface accounting for phase change,  $\rho$  is the density and  $\mathbf{n}$  is the interface normal vector pointing

to the liquid phase. As a result, the magnitude of the interface normal velocity  $D$  is given by

$$D = \mathbf{u}_{\text{liq}} \cdot \mathbf{n} - \frac{\dot{m}}{\rho_{\text{liq}}} = \mathbf{u}_{\text{vap}} \cdot \mathbf{n} - \frac{\dot{m}}{\rho_{\text{vap}}}, \quad (4)$$

leading to the discontinuity of the normal velocity at the interface, given by

$$[\mathbf{u} \cdot \mathbf{n}]_{\Gamma} = \dot{m} \left[ \frac{1}{\rho} \right]_{\Gamma}. \quad (5)$$

Moreover, in order to specify that the tangential components of the velocity are continuous across the interface, it is more convenient to rewrite Eq. (5) as [24, 10, 31]

$$[\mathbf{u}]_{\Gamma} = \dot{m} \left[ \frac{1}{\rho} \right]_{\Gamma} \mathbf{n}. \quad (6)$$

In each phase, momentum conservation leads to the Navier-Stokes equations

$$\frac{\partial \rho \mathbf{u}}{\partial t} = \nabla \cdot (\mathbf{T} - \rho \mathbf{u} \mathbf{u}) + \mathbf{f}, \quad (7)$$

where  $\mathbf{T}$  is the symmetric stress tensor given by

$$\mathbf{T} = -P\mathbf{I} + 2\mu\mathbf{S}, \quad (8)$$

$P$  is the pressure and  $\mu$  is the dynamic viscosity of the considered phase,  $\mathbf{I}$  is the identity tensor,  $\mathbf{S}$  is the symmetric rate-of-strain tensor given by

$$\mathbf{S} = \frac{1}{2} \left( \nabla \mathbf{u} + (\nabla \mathbf{u})^{\text{T}} \right), \quad (9)$$

and  $\mathbf{f}$  is the total body force applied to the fluid particle. Neglecting the gravitational forces and assuming a uniform surface tension coefficient [33], the total body force reduces to

$$\mathbf{f} = \sigma \kappa \mathbf{n} \delta_{\Gamma}, \quad (10)$$

where  $\sigma$  is the surface tension coefficient,  $\kappa$  is the interface curvature and  $\delta_{\Gamma}$  is a multidimensional Dirac distribution localized at the interface [18]. Equations (7) and (10) constitute the Whole-Domain Formulation of Navier-Stokes equations [31, 18]. Referring to Fig. 1, we define point  $P$  moving at the interface velocity  $\mathbf{W} = D\mathbf{n}$ , i.e. the relative velocity of the fluid with respect to  $P$  is  $\mathbf{u} - D\mathbf{n}$ . Rewriting Eq. (7) in the reference frame moving with point  $P$  leads to

$$\frac{\partial}{\partial t} \{ \rho (\mathbf{u} - D\mathbf{n}) \} = \nabla \cdot \{ \mathbf{T} - \rho (\mathbf{u} - D\mathbf{n}) (\mathbf{u} - D\mathbf{n}) \} + \sigma \kappa \mathbf{n} \delta_{\Gamma}, \quad (11)$$

where

$$\mathbf{S} = \frac{1}{2} \left\{ \nabla (\mathbf{u} - D\mathbf{n}) + (\nabla (\mathbf{u} - D\mathbf{n}))^{\text{T}} \right\}. \quad (12)$$

The pressure discontinuity at the interface is obtained by integrating Eq. (11) over the volume  $V$  of thickness  $\delta\epsilon$  surrounding point  $P$  and taking the limit of the integral when  $\delta\epsilon$  tends to zero. Indeed, since  $V$  is a thin band centered on the interface, this limit gives

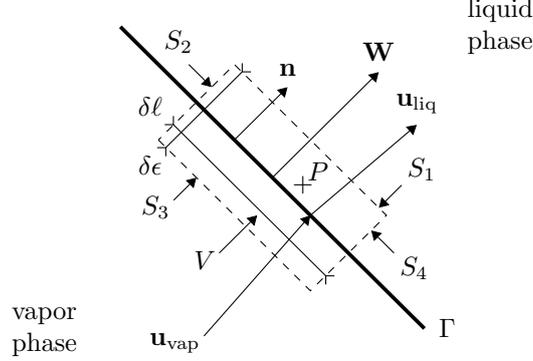


Figure 1: The volume  $V$  is a thin band of width  $\delta\epsilon$  and length  $\delta\ell$  centered on the interface  $\Gamma$  moving at velocity  $\mathbf{W} = D\mathbf{n}$ , and delimited by the surface  $S = S_1 \cup \dots \cup S_4$ . The point  $P$  is at rest with respect to the interface.

the expression of the discontinuity of each volumetric term appearing in Eq. (11). This demonstration has been given in [33] without phase change. We now extend it to take phase change into account. In order for the volume  $V$  to follow the motion of point  $P$ , we use the Reynolds transport theorem to integrate the left-hand side of Eq. (11) over  $V$ , i.e.

$$\int_V \frac{\partial}{\partial t} \{\rho(\mathbf{u} - D\mathbf{n})\} dV = \frac{d}{dt} \int_V \rho(\mathbf{u} - D\mathbf{n}) dV - \oint_S (D\mathbf{n} \cdot \mathbf{n}_S) \rho(\mathbf{u} - D\mathbf{n}) dS. \quad (13)$$

In the limit  $\delta\epsilon \rightarrow 0^+$ , the first term of the right-hand side of Eq. (13) vanishes. The complete integral of Eq. (11) over  $V$  then reads

$$0 = - \oint_S \rho(\mathbf{u} - D\mathbf{n}) \{(\mathbf{u} - 2D\mathbf{n}) \cdot \mathbf{n}_S\} dS - \oint_S P \mathbf{n}_S dS + 2 \oint_S \mu \mathbf{S} \mathbf{n}_S dS + \int_\Gamma \sigma \kappa \mathbf{n} d\gamma. \quad (14)$$

Moreover, the integrals over  $S$  in Eq. (14) reduce to the sum of the integrals over  $S_1$  and  $S_3$ . Since  $S_1$  is located in the liquid phase and  $S_3$ , in the vapor phase, and since  $\mathbf{n}_{S_1} = \mathbf{n}$  and  $\mathbf{n}_{S_3} = -\mathbf{n}$ , Eq. (14) is rewritten

$$0 = - [\rho(\mathbf{u} - D\mathbf{n})(\mathbf{u} \cdot \mathbf{n} - 2D)]_\Gamma - [P]_\Gamma \mathbf{n} + 2 [\mu \mathbf{S}]_\Gamma \mathbf{n} + \sigma \kappa \mathbf{n}, \quad (15)$$

where  $\delta\ell$  is taken small enough for  $\kappa$  to be assumed uniform on the portion of the interface included in  $V$ . Taking the normal component of Eq. (15) and simplifying the first term of the right-hand side by means of Eq. (3) leads to

$$[P]_\Gamma = \sigma \kappa + 2 [\mu \mathbf{n} \cdot \mathbf{S} \mathbf{n}]_\Gamma - \left[ \rho(\mathbf{u} \cdot \mathbf{n} - D)^2 \right]_\Gamma, \quad (16)$$

where  $\mathbf{S}$  is initially given by Eq. (12) and can be reduced here to Eq. (9) since  $\mathbf{n}$  is uniform on  $S_1$  and  $S_3$ . Finally, using Eqs (3) and (5), the last term of the right-hand side of Eq. (16) is rewritten

$$\left[ \rho(\mathbf{u} \cdot \mathbf{n} - D)^2 \right]_\Gamma = \dot{m}^2 \left[ \frac{1}{\rho} \right]_\Gamma, \quad (17)$$

leading to the pressure discontinuity at the interface,

$$[P]_{\Gamma} = \sigma\kappa + \left[ \mu \mathbf{n} \cdot \left( \nabla \mathbf{u} + (\nabla \mathbf{u})^T \right) \mathbf{n} \right]_{\Gamma} - \dot{m}^2 \left[ \frac{1}{\rho} \right]. \quad (18)$$

In the Whole-Domain Formulation, the contribution of surface tension forces to the pressure discontinuity at the interface is already taken into account by means of the total body force  $\mathbf{f}$  in Eqs. (7) and (10). Moreover, as stated in [18], independently of the formulation, the contribution of viscous effects is already taken into account when computing the divergence of the stress tensor  $\mathbf{T}$  in Eq. (7). Only the contribution of the mass transfer rate needs to be added separately. The main difficulty of this formulation is the need of an accurate discretization of subgrid quantities, often leading to the smearing of the interface [4, 31].

Another formulation of the Navier-Stokes equations for two-phase flows is the Jump-Condition Formulation [31, 18] in which surface tension forces are not directly included in the equations. In this formulation, Eq. (7) is rewritten

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u}\mathbf{u}) = -\frac{\nabla P}{\rho} + \frac{1}{\rho} \nabla \cdot \left( \mu \left( \nabla \mathbf{u} + (\nabla \mathbf{u})^T \right) \right), \quad (19)$$

where the density is moreover extracted from the derivatives, emphasizing that it is constant and uniform in each phase. The contributions of surface tension forces and mass transfer rate to the pressure discontinuity at the interface given by Eq. (18) are added as external ingredients. In this work, we use the Jump-Condition Formulation for compatibility with the Sharp Interface model [21].

Energy conservation can be formulated by means of different primitive variables as the internal energy or the enthalpy. The internal energy is favored to describe thermodynamics phenomena in closed systems, and temperature (or enthalpy) is more commonly used for description of quasi-isobaric phenomena (usually open systems) [31]. In this work, we focus on simulations of boiling in open systems, we then express energy conservation with the temperature variable. In each phase, energy conservation reads

$$\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T = \frac{1}{\rho c_p} \nabla \cdot (\lambda \nabla T), \quad (20)$$

where  $T$  is the temperature,  $c_p$  is the heat capacity at constant pressure and  $\lambda$  is the thermal conductivity of the considered phase. Without phase change, energy conservation implies that the heat flux at the interface  $-\lambda \nabla T|_{\Gamma}$  is entirely transmitted from one phase to the other, i.e. one has  $[-\lambda \nabla T \cdot \mathbf{n}]_{\Gamma} = 0$ . When phase change is considered, a portion of the heat flux, equal to  $[-\lambda \nabla T \cdot \mathbf{n}]_{\Gamma}$ , is absorbed by the interface neighborhood in order to provide the latent heat  $L_v$  required to break liquid molecular bonds and transform some quantity  $\dot{m}$  of liquid into vapor per interface surface and time units. The remaining part of the heat flux is transmitted from one phase to the other. This phenomenon leads to the following power balance,

$$[-\lambda \nabla T \cdot \mathbf{n}]_{\Gamma} = \dot{m} L_v, \quad (21)$$

accounting for the latent heat of vaporization as an energy sink at the interface. Equation (21) acts as a definition of the mass transfer rate  $\dot{m}$ . Moreover, we make the assumption

that the temperature is continuous at the interface and that the interface temperature  $T_\Gamma$  is equal to the uniform and constant saturation temperature  $T_{\text{sat}}$  of the fluid, i.e.

$$T_\Gamma = T_{\text{sat}}. \quad (22)$$

## 2.2. Interface capturing method

The liquid-vapor interface is captured using the Level Set method [25]. The signed distance function  $\phi$  to the interface  $\Gamma$  is defined at time  $t$  for  $\mathbf{x}$  belonging to the computational domain  $\Omega$  as

$$\phi(t, \mathbf{x}) = \begin{cases} -\min_{\mathbf{x}_\Gamma \in \Gamma(t)} \|\mathbf{x} - \mathbf{x}_\Gamma\|, & \text{if } \mathbf{x} \in \text{vapor phase,} \\ 0, & \text{if } \mathbf{x} \in \Gamma(t), \\ \min_{\mathbf{x}_\Gamma \in \Gamma(t)} \|\mathbf{x} - \mathbf{x}_\Gamma\|, & \text{if } \mathbf{x} \in \text{liquid phase,} \end{cases} \quad (23)$$

where the set  $\{\mathbf{x} \in \Omega \mid \phi(t, \mathbf{x}) = 0\}$  captures the interface at time  $t$ . In order to accurately capture the interface throughout the simulation, the signed distance function  $\phi$  is advected by the interface velocity  $\mathbf{u}_\Gamma$  using the standard advection equation

$$\frac{\partial \phi}{\partial t} + \mathbf{u}_\Gamma \cdot \nabla \phi = 0, \quad (24)$$

where  $\mathbf{u}_\Gamma$  is equivalently given by [24, 10, 31]

$$\mathbf{u}_\Gamma = \mathbf{u}_{\text{liq}} - \frac{\dot{m}}{\rho_{\text{liq}}} \mathbf{n} \quad (25)$$

and

$$\mathbf{u}_\Gamma = \mathbf{u}_{\text{vap}} - \frac{\dot{m}}{\rho_{\text{vap}}} \mathbf{n}. \quad (26)$$

Equations (25) and (26) denote that the transport of the interface can be formulated in both phases. The interface normal vector  $\mathbf{n}$  is given by

$$\mathbf{n} = \frac{\nabla \phi}{\|\nabla \phi\|} \quad (27)$$

and the interface curvature  $\kappa$ , by

$$\kappa = -\nabla \cdot \mathbf{n}, \quad (28)$$

which is rewritten in the form given by Goldman [11],

$$\kappa = \frac{\nabla \phi^T H(\phi) \nabla \phi - \|\nabla \phi\|^2 \text{Tr}(H(\phi))}{\|\nabla \phi\|^3}, \quad (29)$$

where  $H(\phi)$  is the hessian matrix of  $\phi$  and  $\text{Tr}$  is the trace operator. If one expresses the interface velocity by means of Eq. (26), then Eq. (24) is rewritten

$$\frac{\partial \phi}{\partial t} + \mathbf{u}_{\text{vap}} \cdot \nabla \phi = \frac{\dot{m}}{\rho_{\text{vap}}}, \quad (30)$$

assuming that  $\|\nabla \phi\| = 1$ . In order to solve Eq. (30) on grid nodes, the interface velocity  $\mathbf{u}_\Gamma$ , while physically defined only at the subgrid interface, is computed on grid nodes.

### 3. Numerical method to simulate multiphase flows without phase change on unstructured grids

This work is based on YALES2, a library of finite-volume-based solvers developed for realistic turbulent two-phase flow simulations with low Mach numbers [22]. The YALES2 solver has been validated for various applications such as combustion [23, 12], bio-mechanics [5], hydro-electricity [37], or wind energy [2]. It is able to handle efficiently unstructured meshes with several billions of elements, thus enabling the Large Eddy Simulation (LES) of laboratory and semi-industrial configurations.

YALES2 uses a level set method to capture the interface motion [8]. The step of time advection of the level set function does not preserve the profile of the signed distance function to the interface. After advection of the level set function, the function  $\phi$  is then reshaped, or reinitialized, to the signed distance function to the advected interface using the geometrical approach proposed by Janodet et al. [14]. The interface curvature is computed using Eq. (29) in which high-order differential operators provided by Bernard et al. [3] are used.

Another important ingredient of the method is the pressure-velocity coupling. The incompressible Navier-Stokes equation (19) is solved in both liquid and vapor phases using the projection method based on fractional time steps developed by Chorin [6] and improved by Kim and Moin [17]. The methodology is first presented without phase change. The extension to phase change will be detailed in the next section. Knowing  $\mathbf{u}^n$  and  $P^{n-\frac{1}{2}}$ , the first step of the projection method used to define  $\mathbf{u}^{n+1}$ , the velocity field at next iteration, is the computation of a velocity predictor  $\mathbf{u}^*$  given by

$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} = -\nabla \cdot (\mathbf{u}^n \mathbf{u}^n) + \frac{1}{\rho^n} \nabla \cdot \left( \mu^n \left( \nabla \mathbf{u}^n + (\nabla \mathbf{u}^n)^T \right) \right), \quad (31)$$

which is *a priori* not divergence-free. The second step is the resolution of a Poisson equation to compute the pressure  $P^{n+\frac{1}{2}}$ , given by

$$\nabla \cdot \left( \frac{\nabla P^{n+\frac{1}{2}}}{\rho^n} \right) = \frac{\nabla \cdot \mathbf{u}^*}{\Delta t}, \quad (32)$$

where  $P^{n+\frac{1}{2}}$  is the only unknown. The last step is the correction of the velocity predictor  $\mathbf{u}^*$  by means of the updated pressure  $P^{n+\frac{1}{2}}$  to obtain the updated velocity  $\mathbf{u}^{n+1}$ , i.e.

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^*}{\Delta t} = -\frac{\nabla P^{n+\frac{1}{2}}}{\rho^n}, \quad (33)$$

ensuring  $\nabla \cdot \mathbf{u}^{n+1} = 0$  by Eq. (32). In Eqs. (31-33), the density and viscosity, despite being uniform and constant in each phase, are time-dependent on one given grid node since their values are altered by the passage of the interface. The left-hand side of Eq. (32) is composed of two differential operators, the gradient operator applied to  $P$  and the divergence operator applied to  $\nabla P/\rho$ . The computation of a derivative in one phase should be performed using only the physical quantities defined in the said phase, but the numerical discretization of the differential operator requires using also grid nodes located in the other phase. Then the jump of the quantity at the interface has to be subtracted in the discretization. Consequently, one has to compute  $[P]_\Gamma$  and  $[\nabla P/\rho]_\Gamma$ .

Without phase change, and neglecting the viscous stress tensor jump, the pressure jump condition at the interface is given by the Laplace pressure, i.e.

$$[P]_{\Gamma} = \sigma\kappa. \quad (34)$$

The jump of the quantity  $\nabla P/\rho$  can be formally defined as

$$\left[ \frac{\nabla P}{\rho} \right]_{\Gamma} = \frac{1}{\rho_{\text{liq}}} \nabla P_{\Gamma}^{\text{liq}} - \frac{1}{\rho_{\text{vap}}} \nabla P_{\Gamma}^{\text{vap}}, \quad (35)$$

where  $P_{\Gamma}^{\text{liq}}$  and  $P_{\Gamma}^{\text{vap}}$  are taken infinitely close to the interface, respectively in the liquid and vapor phases. Without phase change, Eq. (35) can be simplified using the continuity of the velocity at the interface. Indeed, by linearity of the jump operator, Eq. (19) implies

$$\left[ \frac{\partial \mathbf{u}}{\partial t} \right]_{\Gamma} + [\nabla \cdot (\mathbf{u}\mathbf{u})]_{\Gamma} = \left[ -\frac{\nabla P}{\rho} \right]_{\Gamma} + \left[ \frac{1}{\rho} \nabla \cdot \left( \mu \left( \nabla \mathbf{u} + (\nabla \mathbf{u})^{\text{T}} \right) \right) \right]_{\Gamma}, \quad (36)$$

where the continuity of the velocity and its derivatives at the interface implies that the two jumps of the left-hand side are null. Moreover, as stated in [7], the two jumps of the right-hand side are directed toward the interface normal vector and given by

$$\left[ \frac{\mathbf{n} \cdot \nabla P}{\rho} \right]_{\Gamma} = \left[ \frac{\mu \mathbf{n} \cdot \nabla^2 \mathbf{u}}{\rho} \right]_{\Gamma}. \quad (37)$$

Dalmon et al. [7] also emphasize that, similarly to Eq. (18) for the pressure jump, the contribution of viscous effects to the jump of the pressure gradient is already taken into account when computing the divergence of the stress tensor  $\mathbf{T}$  in Eq. (7). Indeed, the jump of the pressure gradient can be decomposed as

$$\left[ \frac{\mathbf{n} \cdot \nabla P}{\rho} \right]_{\Gamma} = \left[ \frac{\mathbf{n} \cdot \nabla P}{\rho} \right]_{\Gamma}^{\text{predictor, Eq. (31)}} + \left[ \frac{\mathbf{n} \cdot \nabla P}{\rho} \right]_{\Gamma}^{\text{Poisson, Eq. (32)}}, \quad (38)$$

where the exponents of the right-hand side indicate the step of the projection method in which the corresponding jump component is imposed. Consequently, provided that the velocity predictor  $\mathbf{u}^*$  is computed by Eq. (31), i.e. with  $\mu$  inside the divergence, then the remaining part of the jump of the pressure gradient that has to be enforced in Eq. (32), i.e. the second term of the right-hand side of Eq. (38), simplifies to [7]

$$\left[ \frac{\nabla P}{\rho} \right]_{\Gamma}^{\text{Poisson, Eq. (32)}} = \mathbf{0}. \quad (39)$$

To detail the numerical implementation, we refer to Fig. 2 representing the control volume  $\mathcal{V}_p$  associated to grid node  $\mathbf{x}_p$  crossed by the interface  $\Gamma$ . As a preliminary, we give the spatial discretization of the divergence of the velocity used in the finite volume framework of YALES2 to solve Eqs. (31) and (32) at node  $\mathbf{x}_p$  belonging to the liquid phase. One has

$$\nabla \cdot \mathbf{u}|_p = \frac{1}{\mathcal{V}_p} \sum_{i=1}^N \frac{\mathbf{u}_p + \mathbf{u}_{q_i}}{2} \cdot \mathbf{A}_{p,q_i}, \quad (40)$$

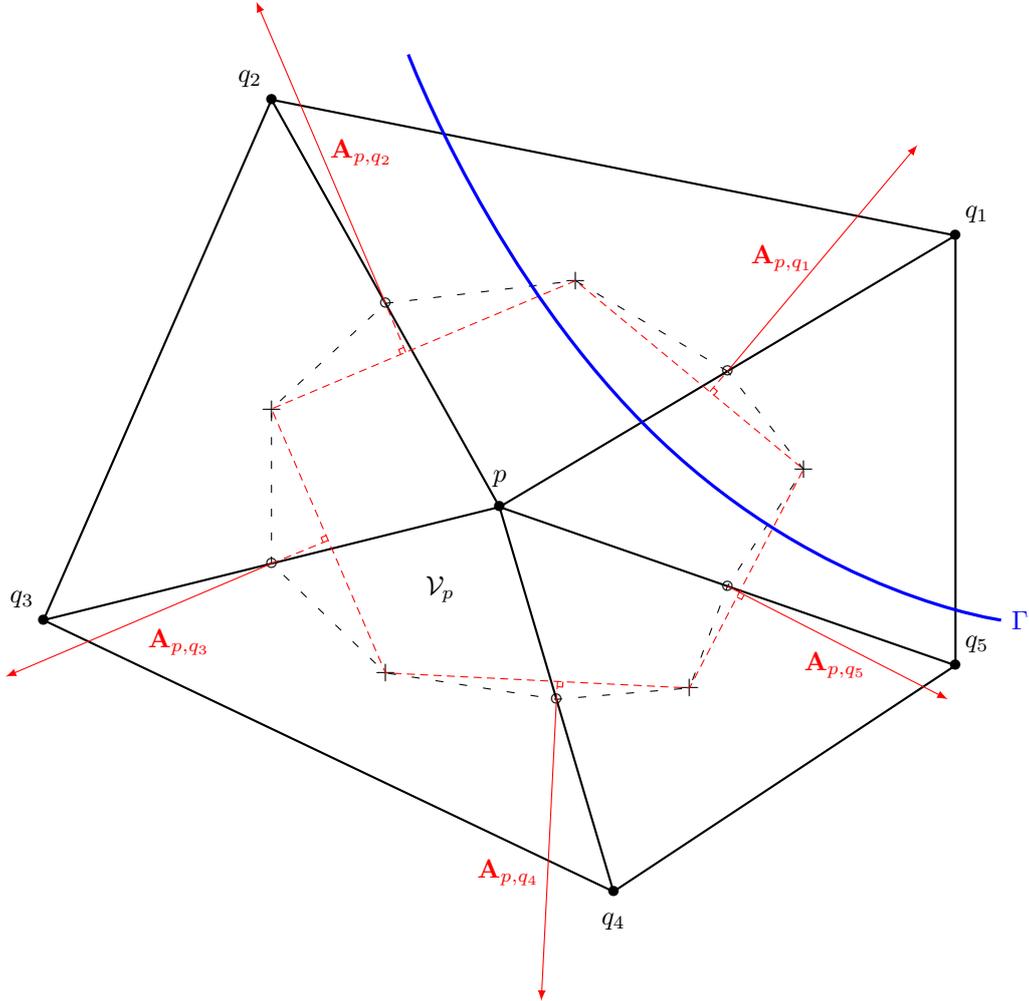


Figure 2: Control volume  $\mathcal{V}_p$  associated to node  $p$ . The neighboring nodes of  $p$  are numbered counterclockwise. The barycenter of each element and the middle of each pair of nodes are respectively represented by the symbols  $+$  and  $o$ . The control volume associated to node  $p$  is represented in simple black dashes. To each pair of nodes are associated two facets of the control volume, each linking the barycenter of a neighboring element to the middle of the pair. The sum of the fluxes of a physical field at the two facets is equal to the flux of the same physical field at the “face” linking the two barycenters of the neighboring elements to the pair, represented in dense red dashes. If this face is denoted  $S_i$  and if  $\mathbf{n}_i$  is its outward normal vector to the control volume of  $p$ , then the vector  $\mathbf{A}_{p,q_i}$  represented in red is equal to  $S_i \mathbf{n}_i$ . For more clarity,  $S_i$  and  $\mathbf{n}_i$  are not represented. The interface  $\Gamma$  is represented in blue.

where  $N$  is the number of neighboring nodes of node  $\mathbf{x}_p$ ,  $\mathbf{A}_{p,q_i}$  is the oriented face of the control volume  $\mathcal{V}_p$  as illustrated in Fig. 2, and where the phase and temporal indices are omitted. We also give the spatial discretization of the laplacian of the pressure used to

solve Eq. (32) at node  $\mathbf{x}_p$ , one has

$$\nabla \cdot \left( \frac{\nabla P^{n+\frac{1}{2}}}{\rho^n} \right) \Big|_p = \frac{1}{V_p} \sum_{i=1}^N \frac{1}{\rho_{p,q_i}^n} \nabla_{p,q_i} P^{n+\frac{1}{2}} \cdot \mathbf{A}_{p,q_i}, \quad (41)$$

where

$$\rho_{p,q_i}^n = \begin{cases} \rho_{\text{liq}} & \text{if } \mathbf{x}_{q_i} \in \text{liquid phase,} \\ \rho_{\Gamma} := \theta \rho_{\text{liq}} + (1 - \theta) \rho_{\text{vap}} & \text{otherwise,} \end{cases} \quad (42)$$

$\theta \in [0; 1]$  is the distance between  $\mathbf{x}_p$  and the interface along the pair  $(\mathbf{x}_p, \mathbf{x}_{q_i})$ , relatively to the length of the pair, and  $\nabla_{p,q_i} P^{n+\frac{1}{2}}$  is the pressure gradient along the pair  $(\mathbf{x}_p, \mathbf{x}_{q_i})$  defined below. We now describe the method used to impose the jumps at the interface given by Eqs. (34) and (39) in Eq. (41). The only node pair crossed by the interface is the pair  $(\mathbf{x}_p, \mathbf{x}_{q_1})$  where node  $\mathbf{x}_p$  is located in the liquid phase and node  $\mathbf{x}_{q_1}$  in the vapor phase. Following the spirit of a sharp interface method, the jumps must be imposed at the intersection point between this node pair and the interface. In order to do so, we use the Ghost Fluid Method (GFM) [9], and two pressure fields (one per phase) are considered. This is only an intermediate step necessary for the derivation. Once the jumps are applied at the interface, only one pressure field remains in the whole domain, with liquid pressure values in the liquid phase and vapor pressure values in the vapor phase (see Eq. (51)). A liquid pressure field,  $P^{\text{liq}}$ , is defined in the liquid phase and in a vapor region close to the interface, in order to be able to define differential operators in the liquid neighborhood of the interface (e.g., at node  $\mathbf{x}_p$ ). Similarly, a vapor pressure field,  $P^{\text{vap}}$ , is defined in the vapor phase and in a liquid region close to the interface. Since we discretize the pressure laplacian at node  $\mathbf{x}_p$  which is located in the liquid phase, we use the liquid pressure field in Eq. (41), even at node  $\mathbf{x}_{q_1}$ . Consequently, the pressure jump at node  $\mathbf{x}_{q_1}$  must be computed and applied in Eq. (41) to remove the (liquid) pressure discontinuity on the same node in the gradient operator. Equations (35) and (39) lead to

$$[\nabla P]_{\Gamma} = \frac{[\rho]_{\Gamma}}{\rho_{\text{liq}}} \nabla P_{\Gamma}^{\text{liq}}. \quad (43)$$

As a result, the first-order Taylor expansion of the pressure jump at grid node  $\mathbf{x}_{q_i}$  from the pressure jump at the interface reads

$$[P]_{q_i} = [P]_{\Gamma} + (1 - \theta) \frac{[\rho]_{\Gamma}}{\rho_{\text{liq}}} \Delta \mathbf{x}_{p,q_i} \cdot \nabla P_{\Gamma}^{\text{liq}}. \quad (44)$$

Then the pressure gradient  $\nabla_{p,q_i} P^{\text{liq}}$  used in Eq. (41) is given along any node pair  $(\mathbf{x}_p, \mathbf{x}_{q_i})$  located in the liquid phase by

$$\nabla_{p,q_i} P^{\text{liq}} = \frac{P_{q_i}^{\text{liq}} - P_p^{\text{liq}}}{\|\Delta \mathbf{x}_{p,q_i}\|^2} \Delta \mathbf{x}_{p,q_i}, \quad (45)$$

and along any node pair crossed by the interface, as for the pair  $(\mathbf{x}_p, \mathbf{x}_{q_1})$ , by

$$\nabla_{p,q_1} P^{\text{liq}} = \frac{P_{q_1}^{\text{liq}} - P_p^{\text{liq}}}{\|\Delta \mathbf{x}_{p,q_1}\|^2} \Delta \mathbf{x}_{p,q_1}, \quad (46)$$

$$= \frac{P_{q_1}^{\text{vap}} - P_p^{\text{liq}}}{\|\Delta \mathbf{x}_{p,q_1}\|^2} \Delta \mathbf{x}_{p,q_1} + \frac{[P]_{q_1}}{\|\Delta \mathbf{x}_{p,q_1}\|^2} \Delta \mathbf{x}_{p,q_1}, \quad (47)$$

$$= \frac{P_{q_1}^{\text{vap}} - P_p^{\text{liq}}}{\|\Delta \mathbf{x}_{p,q_1}\|^2} \Delta \mathbf{x}_{p,q_1} + \frac{[P]_{\Gamma}}{\|\Delta \mathbf{x}_{p,q_1}\|^2} \Delta \mathbf{x}_{p,q_1} + (1 - \theta) \frac{[\rho]_{\Gamma}}{\rho_{\text{liq}}} \frac{\Delta \mathbf{x}_{p,q_1} \cdot \nabla P_{\Gamma}^{\text{liq}}}{\|\Delta \mathbf{x}_{p,q_1}\|^2} \Delta \mathbf{x}_{p,q_1}, \quad (48)$$

where the 'liq' superscript is added to emphasize that Eq. (41) is solved using only pressure values compatible with the liquid phase (either physical values in the liquid phase, or vapor values in the vapor phase to which the pressure jump is subtracted), and where the temporal dependence is removed for clarity. We make the assumption that the pressure gradient  $\nabla P_{\Gamma}^{\text{liq}}$  is directed along node pair  $(\mathbf{x}_p, \mathbf{x}_{q_1})$  and is thus equal to  $\nabla_{p,q_1} P^{\text{liq}}$ . Using Eq. (46), Eq. (48) is then rewritten

$$\frac{\rho_{\Gamma}}{\rho_{\text{liq}}} \frac{P_{q_1}^{\text{liq}} - P_p^{\text{liq}}}{\|\Delta \mathbf{x}_{p,q_1}\|^2} \Delta \mathbf{x}_{p,q_1} = \frac{P_{q_1}^{\text{vap}} - P_p^{\text{liq}}}{\|\Delta \mathbf{x}_{p,q_1}\|^2} \Delta \mathbf{x}_{p,q_1} + \frac{[P]_{\Gamma}}{\|\Delta \mathbf{x}_{p,q_1}\|^2} \Delta \mathbf{x}_{p,q_1}. \quad (49)$$

Multiplying Eq. (49) by  $\mathbf{A}_{p,q_1}/\rho_{\Gamma}$  and rearranging the terms leads to

$$\frac{1}{\rho_{\Gamma}} \frac{P_{q_1}^{\text{vap}} - P_p^{\text{liq}}}{\|\Delta \mathbf{x}_{p,q_1}\|^2} \Delta \mathbf{x}_{p,q_1} \cdot \mathbf{A}_{p,q_1} = \frac{1}{\rho_{\text{liq}}} \frac{P_{q_1}^{\text{liq}} - P_p^{\text{liq}}}{\|\Delta \mathbf{x}_{p,q_1}\|^2} \Delta \mathbf{x}_{p,q_1} \cdot \mathbf{A}_{p,q_1} - \frac{1}{\rho_{\Gamma}} \frac{[P]_{\Gamma}}{\|\Delta \mathbf{x}_{p,q_1}\|^2} \Delta \mathbf{x}_{p,q_1} \cdot \mathbf{A}_{p,q_1}, \quad (50)$$

which, by identification of the first term of the right-hand side in Eqs. (32), (40) and (41), leads to

$$\frac{1}{\rho_{\Gamma}} \frac{P_{q_1}^{\text{vap}} - P_p^{\text{liq}}}{\|\Delta \mathbf{x}_{p,q_1}\|^2} \Delta \mathbf{x}_{p,q_1} \cdot \mathbf{A}_{p,q_1} = \frac{1}{2} \frac{\mathbf{u}_p^* + \mathbf{u}_{q_1}^*}{\Delta t} \cdot \mathbf{A}_{p,q_1} - \frac{1}{\rho_{\Gamma}} \frac{[P]_{\Gamma}}{\|\Delta \mathbf{x}_{p,q_1}\|^2} \Delta \mathbf{x}_{p,q_1} \cdot \mathbf{A}_{p,q_1}. \quad (51)$$

The right-hand side of Eq. (51) is the contribution (multiplied by the control volume  $\mathcal{V}_p$ ) of node pair  $(\mathbf{x}_p, \mathbf{x}_{q_1})$  to the computation of the pressure laplacian at node  $\mathbf{x}_p$  discretized in Eq. (41) where the pressure jump is computed and applied at the interface. For node pairs not crossed by the interface, the second term of the right-hand side of Eq. (51) vanishes. Equation (32) is then solved using the Deflated Pre-Conjugate Gradient (DPCG) solver [19, 20]. The same methodology is used to apply the pressure jump at the interface for the computation of the pressure gradient in the correction step of the velocity predictor given by Eq. (33).

#### 4. Numerical implementation to take into account phase change

The difference between two-phase flow simulations with and without phase change is the discontinuity of the velocity field at the interface in the case of phase change given by Eq. (6). Moreover, phase change requires the resolution of the heat equation in order to compute the mass transfer rate  $\dot{m}$  at the interface.

#### 4.1. Velocity-pressure coupling

With phase change, the projection method given by Eqs. (31), (32) and (33) is still applied in each phase. Nevertheless, since the velocity is discontinuous at the interface, special care has to be taken when differentiating across the interface. In order to overcome this difficulty, the Ghost Fluid Method (GFM) [9] is again used. The velocity field is divided into one field per phase. These velocity fields are defined in domains that overlap at least a few numerical cells around the interface. Both velocity fields are artificially extended (or extrapolated) beyond the interface with the method detailed below (the method does not depend on the GFM). Consequently, we define a liquid velocity field  $\mathbf{u}_{\text{liq}}$  and a vapor velocity field  $\mathbf{u}_{\text{vap}}$ . We define both velocity fields  $\mathbf{u}_{\text{liq}}$  and  $\mathbf{u}_{\text{vap}}$  in the whole domain. Physical values of the liquid (vapor) velocity are contained in the liquid (vapor) part of  $\mathbf{u}_{\text{liq}}$  ( $\mathbf{u}_{\text{vap}}$ ), and the vapor (liquid) part of  $\mathbf{u}_{\text{liq}}$  ( $\mathbf{u}_{\text{vap}}$ ) is used to store the liquid (vapor) ghost values.

The liquid velocity field is extrapolated across the interface in the direction of the interface normal vector  $\mathbf{n}_\Gamma$  solving the constant extrapolation equation given by Aslam [1],

$$\frac{\partial \mathbf{u}_{\text{liq}}}{\partial \tau} + \nabla \mathbf{u}_{\text{liq}} \cdot \mathbf{n}_\Gamma = 0, \quad (52)$$

where  $\tau$  is a pseudo-time emphasizing that this is purely a numerical step. Equation (52) is solved only on the vapor nodes of a narrow band around the interface. The same extrapolation is computed for  $\mathbf{u}_{\text{vap}}$  on the liquid nodes of the narrow band. Once Eq. (52) is solved for both liquid and vapor phases, the velocity divergence for both phases can be computed close to the interface in Eqs. (31) and (32). Moreover, the superposition at one node of the fields of the two phases enables the computation of the jump at this node.

In the projection method, the pressure jump at the interface is given by Eq. (18). Due to the discontinuity of the velocity at the interface, Eq. (39) does not hold anymore, and one has

$$[P]_{q_i} = [P]_\Gamma + (1 - \theta) \Delta \mathbf{x}_{p,q_i} \cdot \left( \rho_{\text{vap}} \left[ \frac{1}{\rho} \nabla P \right]_\Gamma + \frac{[\rho]_\Gamma}{\rho_{\text{liq}}} \nabla P_\Gamma^{\text{liq}} \right). \quad (53)$$

Following the same methodology as in the previous section, Eq. (51) becomes

$$\begin{aligned} \frac{1}{\rho_\Gamma} \frac{P_{q_i}^{\text{vap}} - P_p^{\text{liq}}}{\|\Delta \mathbf{x}_{p,q_i}\|^2} \Delta \mathbf{x}_{p,q_i} \cdot \mathbf{A}_{p,q_i} &= \frac{1}{2} \frac{\mathbf{u}_p^{*,\text{liq}} + \mathbf{u}_{q_i}^{*,\text{liq}}}{\Delta t} \cdot \mathbf{A}_{p,q_i} - \frac{1}{\rho_\Gamma} \frac{[P]_\Gamma}{\|\Delta \mathbf{x}_{p,q_i}\|^2} \Delta \mathbf{x}_{p,q_i} \cdot \mathbf{A}_{p,q_i} \\ &\quad - (1 - \theta) \frac{\rho_{\text{vap}}}{\rho_\Gamma} \frac{\Delta \mathbf{x}_{p,q_i} \cdot \frac{[\mathbf{u}^*]_\Gamma - [\mathbf{u}^{n+1}]_\Gamma}{\Delta t}}{\|\Delta \mathbf{x}_{p,q_i}\|^2} \Delta \mathbf{x}_{p,q_i} \cdot \mathbf{A}_{p,q_i}, \quad (54) \end{aligned}$$

where Eq. (33) is used to replace  $[\nabla P / \rho]_\Gamma$  by  $([\mathbf{u}^*]_\Gamma - [\mathbf{u}^{n+1}]_\Gamma) / \Delta t$ . The right-hand side of Eq. (54) is the contribution (multiplied by the control volume  $\mathcal{V}_p$ ) of node pair  $(\mathbf{x}_p, \mathbf{x}_{q_i})$  to the computation of the pressure laplacian at node  $\mathbf{x}_p$  discretized in Eq. (41) where the pressure and velocity jumps are computed and applied at the interface in the case of phase change. For node pairs not crossed by the interface, the second and third

terms of the right-hand side of Eq. (54) vanish. Moreover, in Eq. (54), the jump of the velocity predictor, due to the jump of  $\mathbf{u}^n$ , is computed as

$$[\mathbf{u}^*]_\Gamma = \frac{[\mathbf{u}^*]_p + [\mathbf{u}^*]_{q_1}}{2}, \quad (55)$$

using the physical and ghost values discussed above, whereas the jump of the future velocity  $[\mathbf{u}^{n+1}]_\Gamma$  is computed using Eq. (6) which actually accounts for the phase change phenomenon by means of the mass transfer rate. The same methodology is used to apply the pressure and velocity jumps at the interface for the computation of the pressure gradient in the correction step of the liquid velocity predictor given by Eq. (33). Similarly, the projection method is applied in the vapor phase to update the vapor velocity field.

#### 4.2. Heat equation and mass transfer rate

The mass transfer rate  $\dot{m}$  is computed at time  $n+1$ . Since the mass transfer rate has physical meaning only at the interface, special care is taken for its computation. The mass transfer rate defined by Eq. (21) can be expanded to

$$\dot{m}_\Gamma = \frac{-\lambda_{\text{liq}} \nabla T_{\text{liq}}|_\Gamma + \lambda_{\text{vap}} \nabla T_{\text{vap}}|_\Gamma}{L_v} \cdot \mathbf{n}_\Gamma. \quad (56)$$

In order to compute  $\dot{m}_\Gamma$  precisely at the interface, it is necessary to evaluate accurately the temperature gradient of each phase at the interface position. The main difficulty is then to avoid differentiating across the interface. Indeed, the computation at the interface position or at nodes of the first layer, as  $\mathbf{x}_p$  in Fig. 2, would request informations from both sides of the interface, which is not possible. In this work, gradients are extended from nodes of the second layer,  $\mathbf{x}_{q_3}$  and  $\mathbf{x}_{q_4}$  in Fig. 2, to the interface location using Taylor series expansions. These expansions are based on high-order differential operators introduced in [3]. The computation of the liquid temperature gradient at the interface location is detailed hereinafter. The procedure is the same for the vapor phase. First of all, second-order accurate gradient  $\mathbf{G}^{\mathcal{O}2}$  and first-order accurate Hessian  $\mathbf{H}^{\mathcal{O}1}$  are computed at nodes of the second layer,  $\mathbf{x}_{q_3}$  and  $\mathbf{x}_{q_4}$ . These operators are then extrapolated at nodes of the first layer,  $\mathbf{x}_p$ , with second- and first-order accuracy, respectively,

$$\begin{cases} \mathbf{G}_{p,q_j}^{\mathcal{O}2}(T_{\text{liq}}) &= \mathbf{G}_{q_j}^{\mathcal{O}2}(T_{\text{liq}}) + \mathbf{H}_{q_j}^{\mathcal{O}1}(T_{\text{liq}}) \cdot (\mathbf{x}_p - \mathbf{x}_{q_j}), \\ \mathbf{H}_{p,q_j}^{\mathcal{O}2}(T_{\text{liq}}) &= \mathbf{H}_{q_j}^{\mathcal{O}1}(T_{\text{liq}}). \end{cases} \quad (57)$$

Differential operators at node  $\mathbf{x}_p$  are then obtained by averaging Taylor series expansions from nodes  $\mathbf{x}_{q_3}$  and  $\mathbf{x}_{q_4}$ . Once differential operators are known at nodes of the first layer, another expansion is performed from node  $\mathbf{x}_p$  to the interface location, to determine the liquid temperature gradient at the interface,

$$\nabla T_{\text{liq}}|_{p,\Gamma} = \mathbf{G}_{\Gamma,p}^{\mathcal{O}2}(T_{\text{liq}}) = \mathbf{G}_p^{\mathcal{O}2}(T_{\text{liq}}) + \mathbf{H}_p^{\mathcal{O}1}(T_{\text{liq}}) \cdot (\mathbf{x}_\Gamma - \mathbf{x}_p). \quad (58)$$

The same methodology applied in the vapor phase yields  $\nabla T_{\text{vap}}|_{q_1,\Gamma}$ . The interface normal vector is computed at the same interface location by

$$\mathbf{n}_{p,q_1;\Gamma} = (1 - \theta) \mathbf{n}_p + \theta \mathbf{n}_{q_1}. \quad (59)$$

The mass transfer rate is then computed precisely at the interface location crossing the node pair  $(\mathbf{x}_p, \mathbf{x}_{q_1})$  by

$$\dot{m}_{p,q_1} = \frac{-\lambda_{\text{liq}} \nabla T_{\text{liq}}|_{p,\Gamma} + \lambda_{\text{vap}} \nabla T_{\text{vap}}|_{q_1,\Gamma}}{L_v} \cdot \mathbf{n}_{p,q_1;\Gamma}. \quad (60)$$

Finally, since values of  $\dot{m}$  are needed on nodes to solve the level set advection equation and Navier-Stokes equations, the value of  $\dot{m}$  stored at node  $\mathbf{x}_p$  is computed as the simple average of the values  $\dot{m}_{p,q_j}$  computed on node pairs  $(\mathbf{x}_p, \mathbf{x}_{q_j})$  crossing the interface.

The heat equation is solved using the following semi-implicit scheme [10, 31]

$$\frac{\rho_i c_{p,i}}{\Delta t} T_i^{n+1} - \nabla \cdot (\lambda_i \nabla T_i^{n+1}) = \frac{\rho_i c_{p,i}}{\Delta t} T_i^n - \rho_i c_{p,i} \mathbf{u}_i^n \cdot \nabla T_i^n, \quad (61)$$

where the subscript  $i$  denotes the phase. As in [31], we make the assumption that boiling occurs at a uniform and constant saturation temperature. The immersed Dirichlet boundary condition given by Eq. (22) is then used at the interface when solving Eq. (61). In Eq. (61), the advection term  $\rho_i c_{p,i} \mathbf{u}_i^n \cdot \nabla T_i^n$  is included in the right-hand side as a source term. The term *semi*-implicit refers to the fact that the advection term is still computed explicitly. In order to compute the diffusion term at time  $n + 1$  and at node  $\mathbf{x}_p$  with the condition at the interface given by Eq. (22), one needs to compute a ghost liquid temperature value  $T_{q_1}^{\text{liq},G}$  on node  $\mathbf{x}_{q_1}$ . Consequently, as for the velocity, we define two temperature fields  $T_{\text{liq}}$  and  $T_{\text{vap}}$  in order to ease the computation of ghost values by the GFM. The discretization of the laplacian of liquid temperature is given at node  $\mathbf{x}_p$  by

$$\nabla \cdot (\lambda_{\text{liq}} \nabla T_{\text{liq}})|_p = \frac{1}{V_p} \lambda_{\text{liq}} \frac{T_{q_1}^{\text{liq},G} - T_p^{\text{liq}}}{\|\Delta \mathbf{x}_{p,q_1}\|^2} \Delta \mathbf{x}_{p,q_1} \cdot \mathbf{A}_{p,q_1} + \frac{1}{V_p} \sum_{j=2}^5 \lambda_{\text{liq}} \frac{T_{q_j}^{\text{liq}} - T_p^{\text{liq}}}{\|\Delta \mathbf{x}_{p,q_j}\|^2} \Delta \mathbf{x}_{p,q_j} \cdot \mathbf{A}_{p,q_j}, \quad (62)$$

where the ghost value  $T_{q_1}^{\text{liq},G}$  is computed by linear extrapolation between the node  $\mathbf{x}_p$  and the interface whose temperature is set to  $T_{\text{sat}}$ , i.e.

$$T_{q_1}^{\text{liq},G} = T_p^{\text{liq}} + \frac{T_{\text{sat}} - T_p^{\text{liq}}}{\theta}. \quad (63)$$

The explicit advection term is computed using the standard second-order accurate finite-volume gradient operator for liquid nodes whose neighboring nodes are all located in the liquid phase. For liquid nodes which have a neighbor in the vapor phase, such as  $\mathbf{x}_p$ , the strategy previously presented, based on high-order differential operators from [3], for the computation of the temperature gradients used to define the mass transfer rate in Eq. (60), is reused to compute  $\nabla T|_p^{\text{liq},n}$ . Finally, in order to compute the temporal derivative, the high-order differential operators from [3] are used to extrapolate the temperature field across the interface. This provides nodes that are swept over by the interface between times  $n$  and  $n + 1$  with valid physical temperature values both at times  $n$  and  $n + 1$  [10].

The complete linear system used to solve Eq. (61) at node  $\mathbf{x}_p$  is given by

$$\begin{aligned} & \left( \frac{\rho_{\text{liq}} c_{p,\text{liq}}}{\Delta t} + \frac{1}{\mathcal{V}_p} \lambda_{\text{liq}} \frac{\Delta \mathbf{x}_{p,q_1}}{\theta \|\Delta \mathbf{x}_{p,q_1}\|^2} \cdot \mathbf{A}_{p,q_1} + \frac{1}{\mathcal{V}_p} \sum_{j=2}^5 \lambda_{\text{liq}} \frac{\Delta \mathbf{x}_{p,q_1}}{\|\Delta \mathbf{x}_{p,q_j}\|^2} \cdot \mathbf{A}_{p,q_j} \right) T_p^{\text{liq},n+1} \\ & \quad - \frac{1}{\mathcal{V}_p} \sum_{j=2}^5 \lambda_{\text{liq}} \frac{\Delta \mathbf{x}_{p,q_j}}{\|\Delta \mathbf{x}_{p,q_j}\|^2} \cdot \mathbf{A}_{p,q_j} T_{q_j}^{\text{liq},n+1} \quad (64) \\ & = \frac{\rho_{\text{liq}} c_{p,\text{liq}}}{\Delta t} T_p^{\text{liq},n} - \rho_{\text{liq}} c_{p,\text{liq}} \mathbf{u}_p^{\text{liq}} \cdot \nabla T_p^{\text{liq},n} + \frac{1}{\mathcal{V}_p} \lambda_{\text{liq}} \frac{\Delta \mathbf{x}_{p,q_1}}{\theta \|\Delta \mathbf{x}_{p,q_1}\|^2} \cdot \mathbf{A}_{p,q_1} T_{\text{sat}}, \end{aligned}$$

which is solved using the BiCGStab(2) linear solver [28]. While the relative distance  $\theta$  appears in both the left- and right-hand sides of Eq. (64) in denominators, our simulations did not require the use of a threshold on its value for stability. The same method is used to update the vapor temperature field.

#### 4.3. Remarks on the interface description

Once  $\mathbf{u}_{\text{vap}}$  and  $\dot{m}$  are known, Eq. (30) is used to advect the signed distance function  $\phi$  to the interface, in which the mass transfer rate is considered as a source term. As in [31], the convention stating that the interface is advected by the velocity of the enclosed phase is used. Since we focus on bubble growth, the liquid-vapor interface is then advected by the vapor velocity field. Figure 3 presents the actual full algorithm used in our solver.

Moreover, we emphasize here that  $\mathbf{u}_{\text{vap}}$  and  $(\dot{m}/\rho_{\text{vap}}) \mathbf{n}$  are two velocity vectors of different but dependent physical origins. The vapor velocity is due to the flow dynamics and the mass transfer rate is due to phase change. The vapor velocity field acts on the mass transfer rate by means of the vapor temperature field, and the mass transfer rate intervenes in the projection method used to update the vapor velocity field. As a result, these two velocity fields are coupled. This emphasizes the need of an overall accurate numerical method to simulate boiling flows since a small numerical error in the computation of one quantity can rapidly propagate to others, and eventually lead to an abnormal deformation of the interface.

## 5. Numerical results

Our numerical method is assessed on two benchmarks already used in Tanguy et al. [31] in the case of two-dimensional axisymmetric cartesian grids. These test-cases deal with bubble growth and give analytical expressions of the bubble radius at any time of the simulation. The first one consists in the growth of a bubble with a fixed mass transfer rate (the heat equation is not solved). The second one uses the evolution of the temperature field to define the mass transfer rate computed at the interface using Eq. (21), and therefore determine the time-dependent bubble radius.

### 5.1. Growth of a static bubble with a constant and uniform mass transfer rate

Our numerical method is evaluated on the case of a 2D static growing water bubble with a fixed mass transfer rate from [31]. The computational domain is a square of side length  $L = 8 \times 10^{-3}$  m. The initial bubble radius is  $R_0 = 1 \times 10^{-3}$  m and the imposed

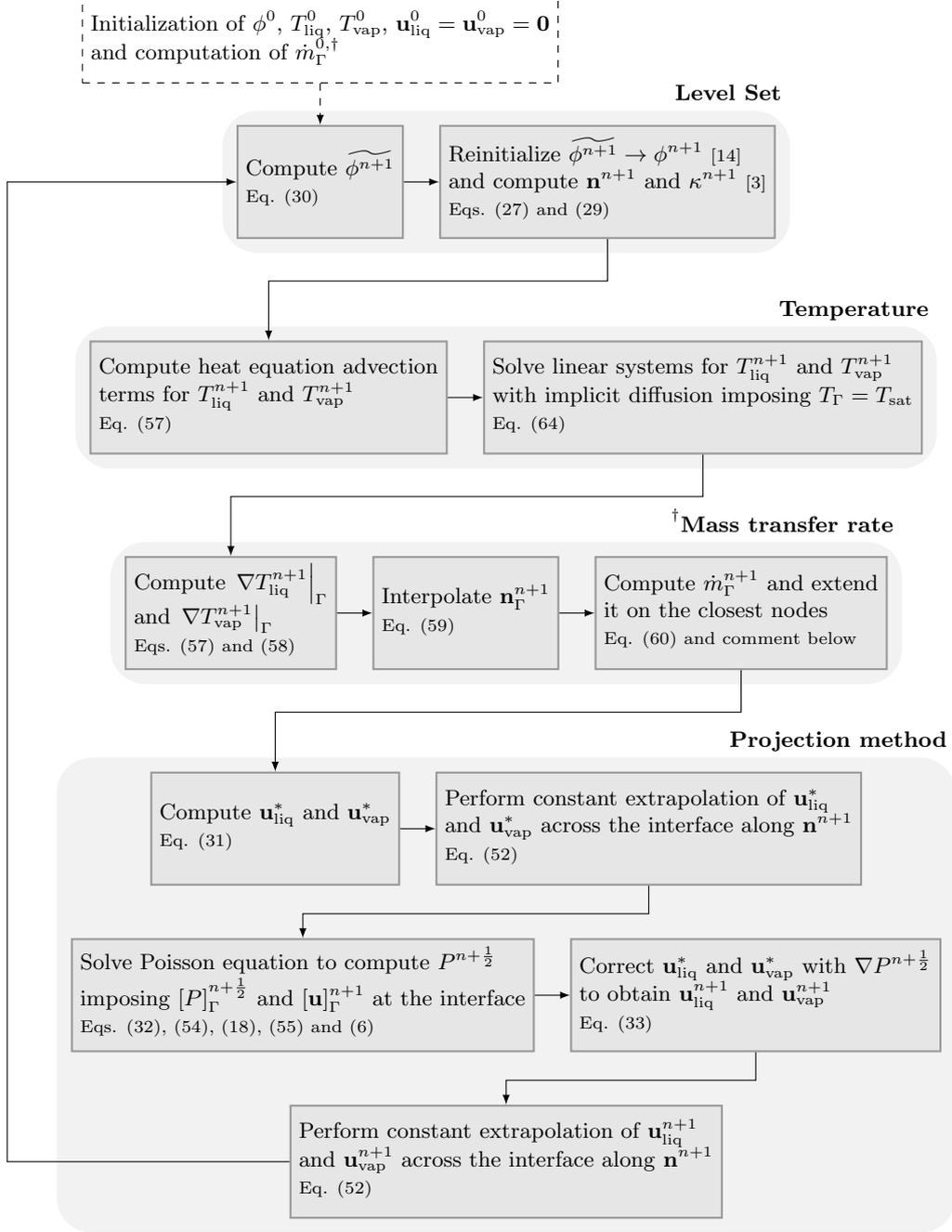


Figure 3: Algorithm used in our solver. The computations of the narrow band around the interface and of the relative distance  $\theta$  are omitted. They are actually performed at initialization, and twice in the temporal loop: once after the advection of the Level Set function and once after its reinitialization. Moreover, the computation of the time step and the exit condition of the temporal loop are also omitted.

$\rho_{\text{liq}}$	$1 \times 10^3$	$\text{kg m}^{-3}$
$\rho_{\text{vap}}$	1	$\text{kg m}^{-3}$
$\sigma$	$7 \times 10^{-2}$	$\text{N m}^{-1}$
$\mu_{\text{liq}}$	$1 \times 10^{-3}$	$\text{kg m}^{-1} \text{s}^{-1}$
$\mu_{\text{vap}}$	$1.78 \times 10^{-5}$	$\text{kg m}^{-1} \text{s}^{-1}$

Table 1: Physical properties used in Section 5.1.

mass transfer rate is  $\dot{m} = 1 \times 10^{-1} \text{ kg m}^{-2} \text{ s}^{-1}$ . The theoretical bubble radius  $R_{\text{th}}(t)$  is given at any time  $t \geq 0$  by

$$R_{\text{th}}(t) = R_0 + \frac{\dot{m}}{\rho_{\text{vap}}} t. \quad (65)$$

The simulations are performed until final time  $t_1 = 1 \times 10^{-2} \text{ s}$  needed for the bubble radius to double. The other physical parameters of interest are listed in Table 1. The computations have been performed on the four following grid cell sizes :  $4 \times 10^{-4} \text{ m}$ ,  $2 \times 10^{-4} \text{ m}$ ,  $1 \times 10^{-4} \text{ m}$ ,  $5 \times 10^{-5} \text{ m}$ , where the grid cell size is approximated on unstructured grids by a commercial meshing software. The coarsest grid used is shown in Fig. 4. In order to evaluate the accuracy of the method, relative errors on the bubble radius are

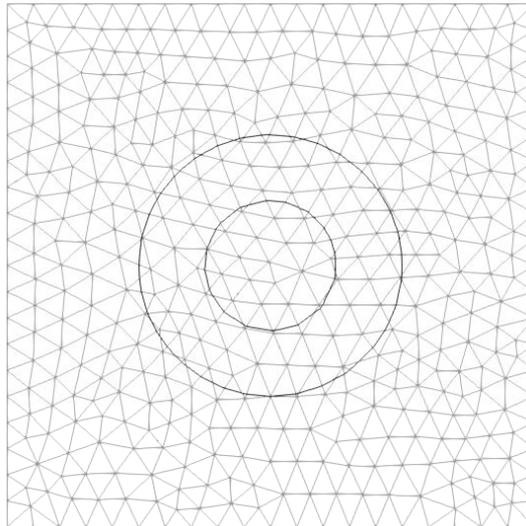


Figure 4: Coarsest grid used in two dimensions ( $\Delta x = 4 \times 10^{-4} \text{ m}$ ) for the simulation of the growth of a static bubble with a constant and uniform mass transfer rate. The small circle represents the initial interface of radius  $R_0 = 1 \times 10^{-3} \text{ m}$ , whereas the bigger circle represents the final theoretical interface (the radius is twice the initial one).

plotted at final time. The theoretical bubble radius at final time  $R_1 = R_{\text{th}}(t_1)$  is given by  $R_1 = 2 \times 10^{-3} \text{ m}$ . The considered relative errors,  $\xi_2(R)$  and  $\xi_\infty(R)$ , are respectively

based on Euclidean and infinity norms,

$$\xi_2(R) = \sqrt{\frac{\sum_{(\mathbf{x}_1, \mathbf{x}_2) \in \Lambda} (R(\mathbf{x}_\Gamma) - R_1)^2}{\sum_{(\mathbf{x}_1, \mathbf{x}_2) \in \Lambda} R_1^2}} \quad (66)$$

and

$$\xi_\infty(R) = \frac{1}{R_1} \max_{(\mathbf{x}_1, \mathbf{x}_2) \in \Lambda} |R(\mathbf{x}_\Gamma) - R_1|. \quad (67)$$

In these equations,  $\Lambda$  is the set of node pairs crossed by the interface,  $\mathbf{x}_\Gamma$  is the interface location interpolated on node pair  $(\mathbf{x}_1, \mathbf{x}_2) \in \Lambda$  by

$$\mathbf{x}_\Gamma = (1 - \theta_{\mathbf{x}_1, \mathbf{x}_2}) \mathbf{x}_1 + \theta_{\mathbf{x}_1, \mathbf{x}_2} \mathbf{x}_2, \quad (68)$$

with  $\theta_{\mathbf{x}_1, \mathbf{x}_2}$  the relative distance involved in Eq. (42), and  $R(\mathbf{x}_\Gamma)$  is given by

$$R(\mathbf{x}_\Gamma) = \|\mathbf{x}_\Gamma\|, \quad (69)$$

since the bubble is centered on the origin of the  $(\mathbf{e}_x, \mathbf{e}_y)$  frame. Figure 5a shows the interface location at initial and final times as well as the liquid and vapor velocity fields at final time, on the finest unstructured grid considered. One can see the excellent

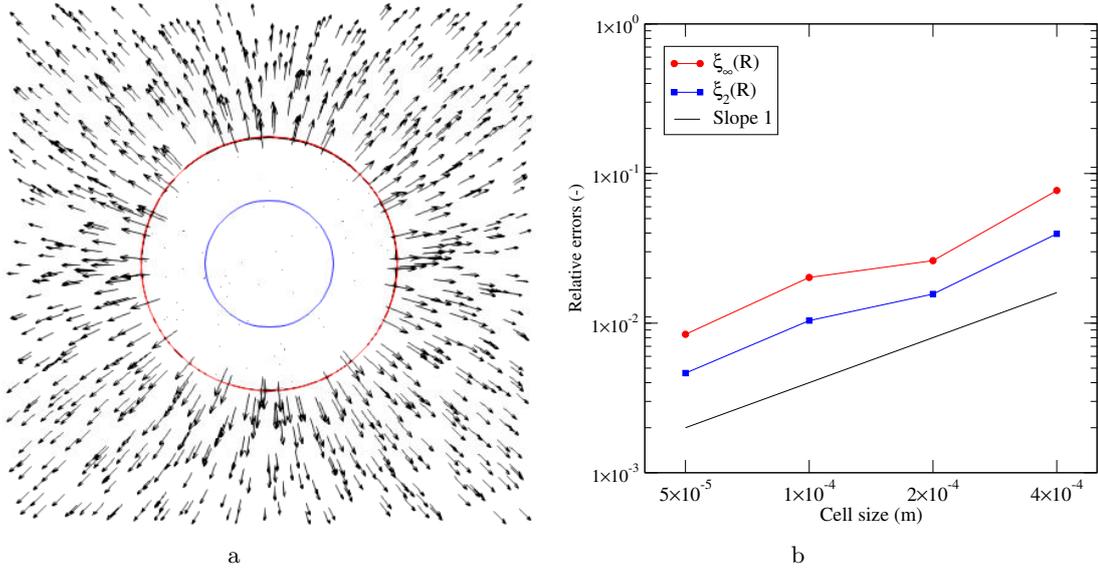


Figure 5: Two-dimensional simulations on triangular grids. On a, the interface location is shown at initial and final times on the finest grid considered ( $\Delta x = 5 \times 10^{-5}$  m). The initial interface is represented in blue, the computed interface at final time, in black, and the theoretical interface at final time, in red. The liquid and vapor velocity fields are also shown at final time. On b, the relative errors,  $\xi_2(R)$  and  $\xi_\infty(R)$ , with grid refinement are shown at final time.

agreement between the computed and theoretical interfaces at final time. As expected, the vapor is almost static while the liquid is ejected radially from the interface. Figure 5b

shows the relative errors,  $\xi_2(R)$  and  $\xi_\infty(R)$ , at final time. The method has a convergence rate close to one for the final bubble radius with respect to grid refinement for both relative errors,  $\xi_2(R)$  and  $\xi_\infty(R)$ . On the finest grid, the relative error  $\xi_\infty(R)$  is below 1%.

This test-case has also been performed on three-dimensional unstructured grids. Figures 6a and 6b show the interface at initial and final times in three dimensions on the finest grid considered. Figure 6c shows the corresponding relative errors,  $\xi_2(R)$  and  $\xi_\infty(R)$ , with grid refinement. In three dimensions, the method also has a convergence order close to one for the final bubble radius with respect to grid refinement for both relative errors,  $\xi_2(R)$  and  $\xi_\infty(R)$ . We emphasize here the excellent agreement between the theoretical and numerical results quantified through the stringent error based on the infinity norm,  $\xi_\infty(R)$ .

The proposed numerical method is then able to simulate phase change with high accuracy by means of a fixed mass transfer rate on two- and three-dimensional unstructured grids. Indeed, the method has demonstrated its ability to accurately model the interface motion and capture the interface location when the mass transfer rate is uniform and constant. The ability to accurately compute the interface location on unstructured grids opens the path to numerical simulations of liquid-vapor phase change on complex geometries.

## 5.2. Growth of a static bubble in a superheated liquid

The next step is the validation of the coupling between the Navier-Stokes equation and the heat equation by means of the mass transfer rate computed using Eq. (56) and the methodology thereafter. This test-case consists in a vapor bubble surrounded by a superheated liquid [31]. The temperature field is initialized in the vapor phase at saturation temperature, and is initialized in the liquid phase by means of a radially symmetric profile whose analytical formulation has been derived in three dimensions by Scriven [27]. Recently we have derived an analytical formulation in two dimensions for the same problem [26]. To the best of our knowledge, such two-dimensional derivation did not exist in the literature. The mathematical demonstrations of both three- and two-dimensional analytical formulations are detailed in [26] and recalled below. The temperature gradient is non-zero in the liquid side of the interface, leading to a non-zero mass transfer rate at the interface. Due to the initially stationary bubble and to the symmetry of the problem, the liquid-vapor interface motion is then only due to phase change. The gravitational forces being neglected, the interface is expected to remain spherical in three dimensions and circular in two dimensions for the whole temporal evolution. The duration of the simulation is equal to the theoretical duration needed for the bubble radius to double.

### 5.2.1. Two-dimensional case

In order to avoid the influence of anisotropic control volumes, we first present the results obtained on two-dimensional grids built only with equilateral triangles, leading to regular hexagonal control volumes. The computational domain is a rhombus of side length  $L = 2.4 \times 10^{-2}$  m of which a portion is shown in Fig. 7. The grids used are listed in Table 2. The initial bubble radius is  $R_0 = 1 \times 10^{-3}$  m and the simulation is performed for the physical time prescribed by the analytical solution for doubling the radius. The

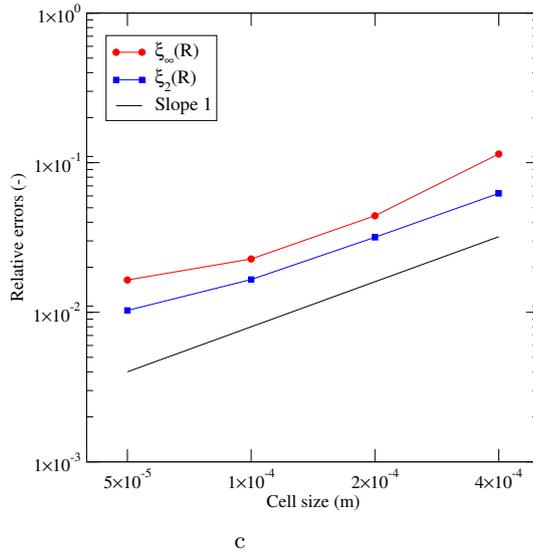
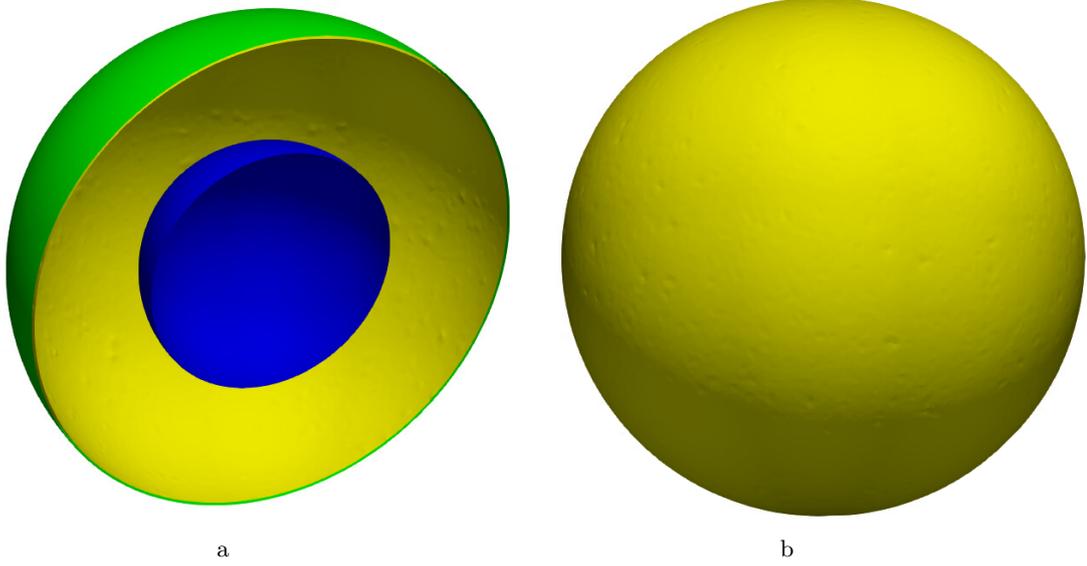


Figure 6: Three-dimensional extension of the simulations illustrated in Fig. 5 to tetrahedral grids. On a, for the finest grid ( $\Delta x = 5 \times 10^{-5}$  m), the initial interface is shown in blue, the theoretical and computed interfaces are shown at final time in green and yellow, respectively. On b, on the same grid, the complete computed interface is shown at final time. On c, the relative errors on the bubble radius,  $\xi_2(R)$  and  $\xi_{\infty}(R)$ , with grid refinement are shown at final time.

theoretical liquid temperature field is given at any time  $t$  by [26]

$$\frac{T(s) - T_{\infty}}{T_{\text{sat}} - T_{\infty}} = \exp(\beta^2 - s^2), \quad (70)$$

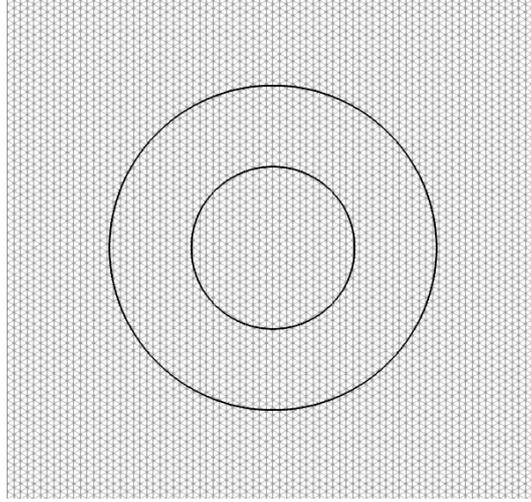


Figure 7: Coarsest two-dimensional grid ( $\Delta x = 9.375 \times 10^{-5}$  m) composed only of equilateral triangles used in our simulations. The small circle represents the initial interface of radius  $R_0 = 1 \times 10^{-3}$  m, whereas the bigger circle represents the final theoretical interface (the radius is twice the initial one).

Characteristic grid cell size $\Delta x$ (m)	Number of cells $N$ (-)
$9.375 \times 10^{-5}$	14 450
$4.688 \times 10^{-5}$	57 800
$2.344 \times 10^{-5}$	231 200

Table 2: Two-dimensional triangular grids.

where  $s$  is the reduced variable defined as

$$s = \frac{r}{2\sqrt{\alpha_{\text{liq}}t}}, \quad (71)$$

$r = (x^2 + y^2)^{\frac{1}{2}}$  is the radial variable,  $\alpha_{\text{liq}} = \lambda_{\text{liq}}/(\rho_{\text{liq}}c_{p,\text{liq}})$  is the thermal diffusivity of the liquid phase,  $T_\infty$  is the temperature in the liquid phase when  $r \rightarrow \infty$ , and  $\beta$  is the growth rate defined as

$$\beta = \left(1 - \frac{\rho_{\text{vap}}}{\rho_{\text{liq}}}\right)^{-\frac{1}{2}}. \quad (72)$$

Equation (70) is used to initialize the liquid temperature field. The theoretical bubble radius  $R_{\text{th}}(t)$  is given, for some initial  $t_0 > 0$  which will be computed later, at any time  $t \geq t_0$  by

$$R_{\text{th}}(t) = 2\beta\sqrt{\alpha_{\text{liq}}t}. \quad (73)$$

The physical properties of interest corresponding to a water vapor bubble in liquid water are listed in Table 3 where the heat capacity at constant pressure of the liquid phase has been decreased by two orders of magnitude with respect to the value used in Tanguy

$\rho_{\text{liq}}$	958	$\text{kg m}^{-3}$
$\rho_{\text{vap}}$	0.59	$\text{kg m}^{-3}$
$\sigma$	$5.9 \times 10^{-2}$	$\text{N m}^{-1}$
$\mu_{\text{liq}}$	$2.82 \times 10^{-4}$	$\text{kg m}^{-1} \text{s}^{-1}$
$\mu_{\text{vap}}$	$1.23 \times 10^{-6}$	$\text{kg m}^{-1} \text{s}^{-1}$
$\lambda_{\text{liq}}$	0.6	$\text{W m}^{-1} \text{K}^{-1}$
$\lambda_{\text{vap}}$	0.026	$\text{W m}^{-1} \text{K}^{-1}$
$c_{\text{p,liq}}$	42.16	$\text{J kg}^{-1} \text{K}^{-1}$
$c_{\text{p,vap}}$	2034	$\text{J kg}^{-1} \text{K}^{-1}$
$L_v$	$2.257 \times 10^6$	$\text{J kg}^{-1}$
$T_{\text{sat}}$	373	K

Table 3: Physical properties used in Section 5.2. Only  $c_{\text{p,liq}}$  differs (by two orders of magnitude) from the value used in Tanguy et al. [31].

et al. [31] (see discussion in Section 5.2.2). Figure 8 shows the computed interface at final time and the final temperature, velocity and pressure fields for the finest grid. One can notice the high accuracy on the bubble shape at final time : the bubble is perfectly circular and its radius is in excellent agreement with the theoretical one. Figure 9 shows the relative errors  $\xi_2(R)$  and  $\xi_\infty(R)$ .

### 5.2.2. Three-dimensional case

Our numerical tests are now extended to three dimensions using tetrahedral unstructured grids. Note that this test-case is similar to the test-cases performed by Tanguy et al. [31] on two-dimensional axisymmetric cartesian grids. One noticeable difficulty arising with such grids is the anisotropy of control volumes. The computational domain is a cube of side length  $L = 1.2 \times 10^{-2}$  m. The initial bubble radius is  $R_0 = 1 \times 10^{-3}$  m and the simulation is performed for the physical time prescribed by the analytical solution for doubling the radius. The liquid temperature field is initialized by [27, 31, 26]

$$\frac{T(s) - T_\infty}{T_{\text{sat}} - T_\infty} = \frac{2\beta^3}{\text{Ja}} \exp(\beta^2 + 2\epsilon\beta^2) \int_s^{+\infty} x^{-2} \exp(-x^2 - 2\epsilon\beta^3 x^{-1}) dx, \quad (74)$$

where Ja is the Jakob number defined as

$$\text{Ja} = \frac{\rho_{\text{liq}} c_{\text{p,liq}} (T_\infty - T_{\text{sat}})}{\rho_{\text{vap}} L_v}, \quad (75)$$

$\beta$  is implicitly defined as

$$\text{Ja} = 2\beta^3 \exp(\beta^2 + 2\epsilon\beta^2) \int_\beta^{+\infty} x^{-2} \exp(-x^2 - 2\epsilon\beta^3 x^{-1}) dx, \quad (76)$$

and  $\epsilon = 1 - \rho_{\text{vap}}/\rho_{\text{liq}}$ . The theoretical bubble radius is given by Eq. (73). In Tanguy et al. [31], the simulations are performed for Jakob numbers ranging from 3 to 10. The authors have simulated the test-case using different numerical methods. We chose to compare our method to the variant giving the smallest final relative errors on the bubble radius for the highest tested Jakob numbers, since increasing the Jakob number decreases

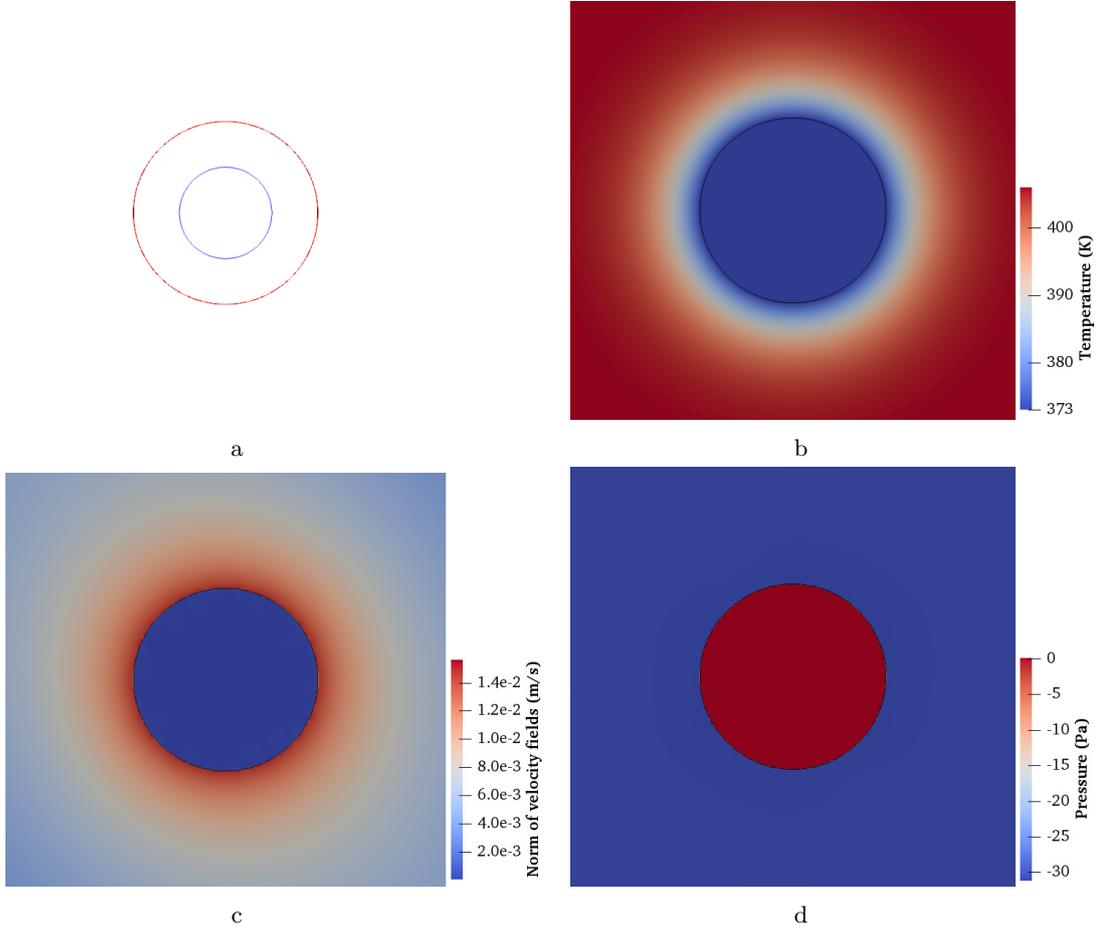


Figure 8: Results in two dimensions on the finest triangular grid used ( $\Delta x = 2.344 \times 10^{-5}$  m). On a, the initial interface is shown in blue, the final computed interface, in black, and the final theoretical interface in red. The final liquid and vapor temperature fields are shown on b, the final velocity fields are shown on c and the final pressure field is shown on d.

the width of the thermal boundary layer at the interface, thus making the mass transfer rate more challenging to compute on a given grid [31]. We performed our simulations with Jakob numbers of 3, 5, 7 and 9. Table 4 reproduces the errors given for these Jakob numbers in Table 6 of [31]. We performed our simulations on the three three-dimensional unstructured tetrahedral grids listed in Table 5. In order to speed up computations, the characteristic cell sizes of these grids are multiplied by a factor 1.6 with respect to the values used in [31] and listed in Table 4. Moreover, the liquid heat capacity at constant pressure is decreased by two orders of magnitude with respect to the value used in [31] (see Table 3), with the goal of decreasing the computing time. Indeed, Eq. (75) shows that if  $c_{p,\text{liq}}$  decreases, then  $T_\infty$  increases (all other parameters being unchanged). As a result, the liquid temperature field, given at time  $t$  by Eq. (74), has a steeper initial slope close to the interface. Consequently, the mass transfer rate  $\dot{m}$  and so the interface

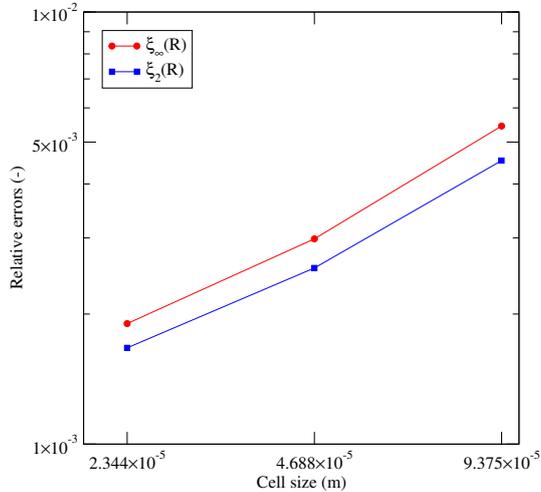


Figure 9: Relative errors on the final bubble radius,  $\xi_2(R)$  and  $\xi_{\infty}(R)$ , for the two-dimensional simulations on unstructured grids illustrated in Fig. 8.

$\Delta x$ (m)	Ja=3	Ja=5	Ja=7	Ja=9
$9.375 \times 10^{-5}$	9.3%	21.0%	30.5%	36.6%
$4.688 \times 10^{-5}$	1.4%	2.8%	7.9%	14.4%
$2.344 \times 10^{-5}$	1.0%	1.0%	0.5%	1.0%

Table 4: Relative errors on the bubble radius at final time for different Jakob numbers on three two-dimensional axi-symmetric cartesian grids extracted from Table 6 of Tanguy et al. [31]. The cell sizes in the first column have been computed from the number of cells and the size of the domain given by the authors.

velocity due to phase change (right-hand side of Eq. (30)) are then increased, leading to faster simulations. Table 6 shows  $\xi_{\infty}(R)$  on the three grids and for the four Jakob numbers considered. The relative error  $\xi_{\infty}(R)$  decreases with grid refinement for all Jakob numbers except for  $Ja = 3$  on the finest grid. One can indeed observe an error increase for the smallest Jakob number on the finest grid. This behavior can be related to the results of Tanguy et al. [31] listed in Table 4 in which the error on the bubble radius reaches a minimum value on the finest grid and does not clearly converge with grid refinement below this minimum value. Also,  $\xi_{\infty}(R)$  increases with the Jakob number. As stated above and in Tanguy et al. [31], increasing the Jakob number reduces, by Eq. (74), the width of the thermal boundary layer at the interface. Consequently, the mass transfer rate and the interface velocity due to phase change are computed with less accuracy since a lower number of grid nodes are located in the thermal boundary layer at every iteration. Then, the bubble radius, measured at final time from the signed distance function advected by the interface velocity due to phase change, is also less accurate. The convergence of the bubble radius with grid refinement and decrease of the Jakob number has been obtained by Tanguy et al. [31] on two-dimensional axisymmetric cartesian grids, as summarized in Table 4. In the present work, this convergence is extended to

$\Delta x$ (m)	Number of cells $N$ (-)
$1.5 \times 10^{-4}$	554 800
$7.5 \times 10^{-5}$	1 933 621
$3.75 \times 10^{-5}$	9 186 699

Table 5: Three-dimensional tetrahedral grids. The cell sizes correspond to the ones listed in Table 4 multiplied by 1.6. The refined area around the interface is shown in Fig. 10.

$\Delta x$ (m)	Ja=3	Ja=5	Ja=7	Ja=9
$1.5 \times 10^{-4}$	6.8%	20%	34%	47%
$7.5 \times 10^{-5}$	0.8%	1.7%	7.9%	16%
$3.75 \times 10^{-5}$	1.7%	0.6%	1.0%	1.9%

Table 6: Relative error on the bubble radius at final time,  $\xi_\infty(R)$ , for different unstructured grids and Jakob numbers in three dimensions.

three-dimensional unstructured grids with comparable accuracy: in our simulations, the bubble radius at final time is generally slightly more accurate for low Jakob numbers (3 and 5) and slightly less accurate for high Jakob numbers (7 and 9) than the one obtained by Tanguy et al. [31]. Since our grid cells are larger than the ones used in Tanguy et al. [31] by a factor 1.6, and since our grids are three-dimensional, which implies that the sizes of the tetrahedra mentioned in Table 5 are only approximated by the meshing software, further comparison would not be relevant. Table 6 is then considered as the validation of our solver for the growth of static bubbles.

For a Jakob number of 5, Eq. (76) gives  $\beta = 5.304$ , Eq. (73) gives the initial time  $t_0 = 5.98 \times 10^{-4}$  s and the final time  $t_1 = 2.39 \times 10^{-3}$  s and Eq. (75) gives  $T_\infty = 538$  K. For illustrative purposes, Fig. 10 shows a portion of the coarsest grid used to validate this test-case in three dimensions. Figure 11 shows the initial interface and the final computed and theoretical interfaces for the finest grid, as well as the relative errors on the final bubble radius,  $\xi_2(R)$  and  $\xi_\infty(R)$ , for the three grids used. The final bubble radius converges with grid refinement for both relative errors,  $\xi_2(R)$  and  $\xi_\infty(R)$ . We emphasize here the excellent agreement between the theoretical and numerical results for the finest grid, whereas, for  $\xi_\infty(R)$ , one single inaccurate nodal value can lead to a high relative error. This is a very stringent criterion constituting a severe test for the numerical accuracy. On the finest grid, the relative error  $\xi_\infty(R)$  is equal to 0.6%, as reported in Table 6. Figure 12 shows the radial temperature at initialization and final time. As expected, the liquid temperature is very close to  $T_\infty$  in all the liquid phase, except in a region close to the interface, the thermal boundary layer, where a steep gradient responsible for the interface movement is observed. In this area, the liquid temperature decreases from  $T_\infty = 538$  K in the liquid phase to  $T_{\text{sat}} = 373$  K at the interface, showing that the immersed Dirichlet boundary condition given by Eq. (22) is satisfied throughout the simulation. Indeed, the use of an implicit algorithm to impose saturation temperature at the subgrid position of the interface does not allow for a direct verification of the respect of such constraint. Figure 13 shows the liquid temperature and velocity fields at final time on the finest grid. One can see the small thickness of the

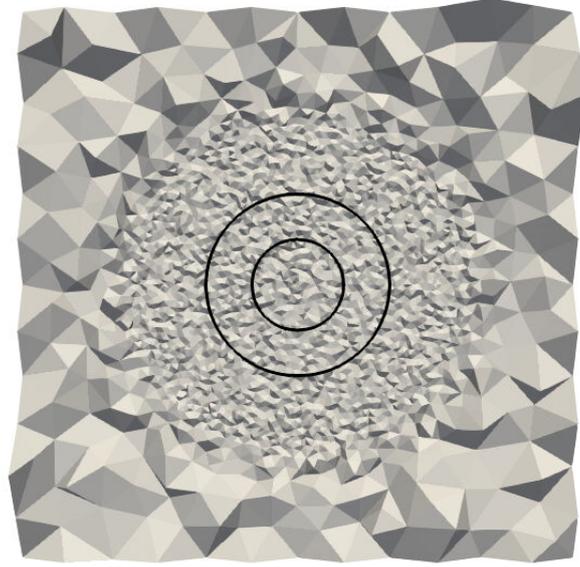


Figure 10: Refined three-dimensional cubic grid used in our simulations. The small circle represents the initial interface location of radius  $R_0 = 1 \times 10^{-3}$  m, the bigger circle represents the final theoretical interface location (the radius is twice the initial one). The grid is refined only in an area englobing a narrow band around the interface in order to speed up computations. This grid has a characteristic cell size around the interface of  $\Delta x = 1.5 \times 10^{-4}$  m and counts 554 800 grid cells. This is the coarsest grid used in our 3D tests.

boundary layer, compared to the bubble diameter, which requires a fine enough grid to be captured. As expected, the liquid is ejected from the interface and the liquid velocity field is thus aligned with the interface normal vector. The velocity jump at the interface due to phase change is responsible for the liquid motion which can exit the domain thanks to the outlet boundary conditions used on all six faces of the cubic domain.

Our solver is able to maintain the spherical shape of the bubble throughout the simulation as expected from the theoretical solution. Since this test-case is very severe relatively to the accuracy of the interface capturing or tracking method used, especially on unstructured grids, we believe that the present results have a strong interest for further simulations of two-phase flows with phase change on unstructured grids.

## 6. Conclusion and perspectives

In this work, a numerical method for simulations of two-phase flows with phase change due to heat transfer (boiling) on unstructured grids has been developed within the YALES2 code and assessed in two and three dimensions. The method has first been validated on the test-case of a static growing bubble with a fixed uniform and constant mass transfer rate, in two- and three-dimensional unstructured grids. Compared to the analytical solution, the bubble radius at final time is first-order accurate with grid refinement in the  $L^\infty$  sense. The ability to accurately compute the interface location on unstructured grids opens the path to numerical simulations of liquid-vapor phase change

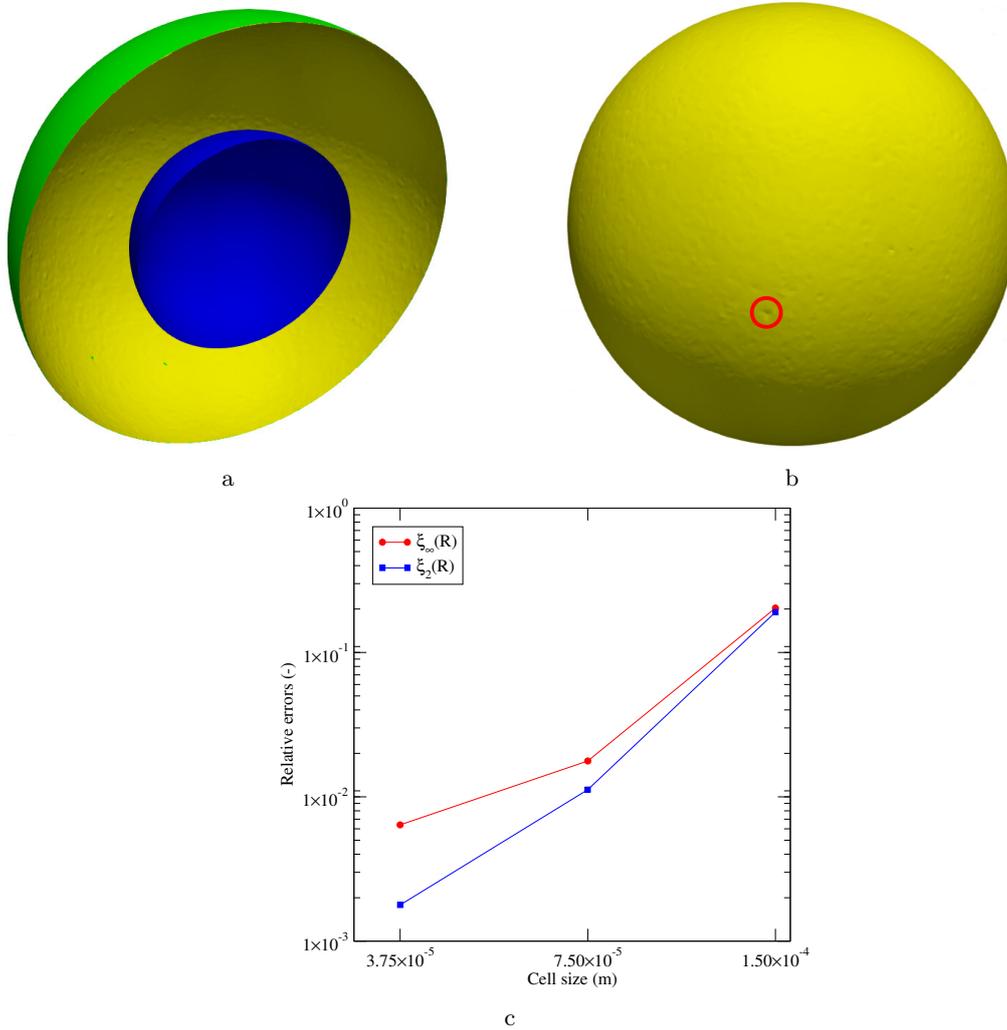


Figure 11: Three-dimensional extension of the simulations illustrated in Fig. 8 to tetrahedral grids. On a, for the finest grid ( $\Delta x = 3.75 \times 10^{-5}$  m), the initial interface is shown in blue, the theoretical and computed interfaces are shown at final time in green and yellow, respectively. On b, on the same grid, the complete computed interface is shown at final time. On c, the relative errors on the bubble radius,  $\xi_2(R)$  and  $\xi_\infty(R)$ , with grid refinement are shown at final time. Local errors in the interface position, as shown in red on b, strongly affect the value of  $\xi_\infty(R)$  whereas their effect on  $\xi_2(R)$  is more limited, as shown in c.

on complex geometries. The method has then been extended to take into account the computation of the mass transfer rate at the interface from the thermal fluxes on both sides of the interface. To this purpose, the heat equation is solved in both phases and two temperature fields are used to take into account the discontinuity of the heat flux at the interface. An immersed Dirichlet boundary condition is imposed at the interface in order to ensure that boiling always occurs at saturation temperature. The mass transfer

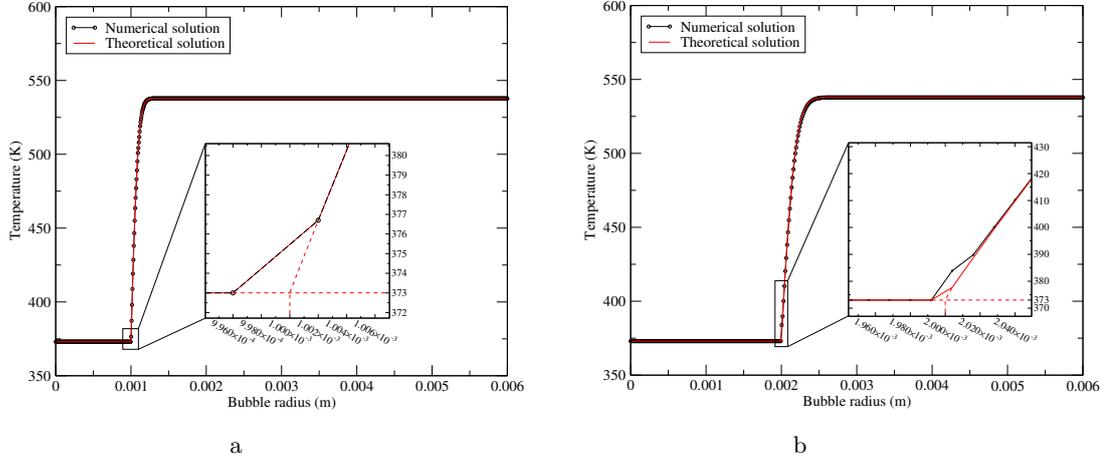


Figure 12: Radial profile of the three-dimensional temperature field shown at a initialization ( $R_0 = 1 \times 10^{-3}$  m) and b final time ( $R_1 = 2 \times 10^{-3}$  m) for  $\Delta x = 3.75 \times 10^{-5}$  m.

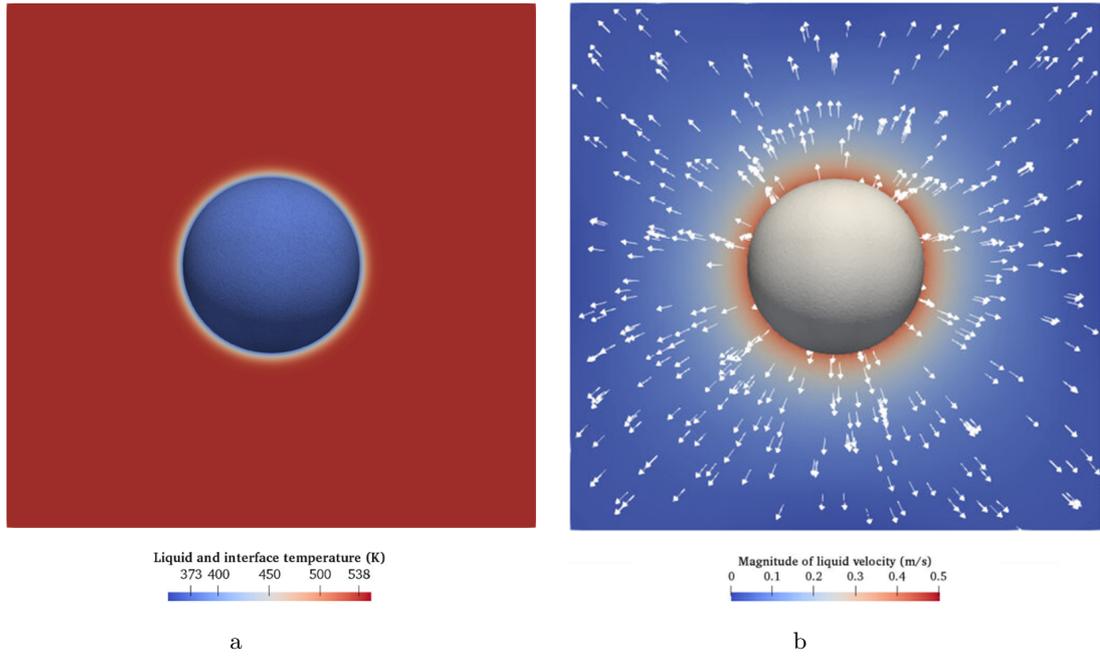


Figure 13: Results in three dimensions on the finest tetrahedral grid used ( $R_1/\Delta x = 53$ ). The three-dimensional interface is represented together with a two-dimensional slice of the computational domain. On a, the liquid temperature field is shown at final time. On b, the liquid velocity field is shown at final time. These fields correspond to the simulation illustrated in Fig. 11b.

rate is computed using a new framework to extrapolate the liquid temperature gradient from the liquid phase to the interface, and the vapor temperature gradient from the

vapor phase to the interface, in order to compute the mass transfer rate precisely at the interface. The overall implementation has been validated against the test-case of a bubble at rest growing in a superheated liquid, in two and three dimensions. Excellent agreement with the theoretical bubble radius has been obtained at final time on two- and three-dimensional unstructured grids. Moreover, the bubble radius converges with grid refinement in all cases. In three dimensions, the relative  $L^\infty$ -norm of the error on the bubble radius is below 2% on the finest grid considered. These results demonstrate the ability of our numerical method to accurately simulate two-phase flows with phase change where the mass transfer rate is computed from the thermal fluxes at the interface, thus taking a step towards realistic boiling simulations in industrial context. To the best of our knowledge, this work has provided the first numerical simulations of bubble growth in three dimensions with a bubble radius converging with grid refinement in the  $L^\infty$ -norm sense. The  $L^\infty$  convergence of the radius implies that the position of the bubble surface converges to the exact solution at every surface point. It is then considered as an important ingredient for predictive numerical simulations.

The natural extension of this work is the simulation of nucleate boiling, a mode of heat transfer widely used in industrial applications, occurring when a liquid is in contact with a solid whose temperature is above the liquid boiling point, leading to the formation of vapor bubbles on the solid surface. Numerical simulations of nucleate boiling require a methodology to take the motion of the contact line (where the solid, liquid and vapor phases meet) into account, as well as the contact angle existing between the solid surface and the liquid-vapor interface. The numerical simulation of nucleate boiling on unstructured grids is currently under development as an extension of this work. The inclusion of the contact line and contact angle in our method is expected to enable predictive three-dimensional direct numerical simulations of nucleate boiling at the bubble scale in the coming years.

### Acknowledgments

This work has been supported by the LabEx Tec21 (Investissements d’Avenir-Grant Agreement No. ANR-11-LABX-0030). The authors would like to thank Patrick Bégou for support on algorithms implementation, Geoffroy Vaudor and Romain Janodet for their help on the use of the geometrical approach for the reinitialization of the Level Set function, Bruno Voisin for his help on some mathematical derivations, and Savinien Pertant for various fruitful discussions on the numerical method and its implementation.

## References

- [1] T. D. Aslam. A partial differential equation approach to multidimensional extrapolation. *J. Comput. Phys.*, 193(1):349 – 355, 2004.
- [2] P. Bénard, A. Viré, V. Moureau, G. Lartigue, L. Beaudet, P. Deglaire, and L. Briceux. Large-eddy simulation of wind turbines wakes including geometrical effects. *Computers & Fluids*, 173:133–139, 2018.
- [3] M. Bernard, G. Lartigue, G. Balarac, V. Moureau, and G. Puigt. A framework to perform high-order deconvolution for finite-volume method on simplicial meshes. *Int. J. Numer. Methods Fluids*, 92(11):1551–1583, 2020.
- [4] J. U. Brackbill, D. B. Kothe, and C. Zemach. A continuum method for modeling surface tension. *J. Comput. Phys.*, 100(2):335 – 354, 1992.
- [5] C. Chnafa, S. Mendez, and F. Nicoud. Image-based large-eddy simulation in a realistic left heart. *Computers & Fluids*, 94:173–187, 2014.
- [6] A. J. Chorin. Numerical solution of the Navier-Stokes equations. *Math. Comp.*, 22:745–762, 1968.
- [7] A. Dalmon, K. Kentheswaran, G. Mialhe, B. Lalanne, and S. Tanguy. Fluids-membrane interaction with a full eulerian approach based on the level set method. *J. Comput. Phys.*, 406:109171, 2020.
- [8] O. Desjardins, V. Moureau, and H. Pitsch. An accurate conservative level set/ghost fluid method for simulating turbulent atomization. *J. Comput. Phys.*, 227(18):8395 – 8416, 2008.
- [9] R. P. Fedkiw, T. Aslam, B. Merriman, and S. Osher. A non-oscillatory eulerian approach to interfaces in multimaterial flows (the ghost fluid method). *J. Comput. Phys.*, 152(2):457 – 492, 1999.
- [10] F. Gibou, L. Chen, D. Nguyen, and S. Banerjee. A level set based sharp interface method for the multiphase incompressible Navier-Stokes equations with phase change. *J. Comput. Phys.*, 222(2):536 – 555, 2007.
- [11] R. Goldman. Curvature formulas for implicit curves and surfaces. *Comput. Aided Geom. D.*, 22(7):632 – 658, 2005. Geometric Modelling and Differential Geometry.
- [12] L. Guédot, G. Lartigue, and V. Moureau. Design of implicit high-order filters on unstructured grids for the identification of large-scale features in large-eddy simulation and application to a swirl burner. *Phys. Fluids*, 27(4):045107–30, 2015.
- [13] C. W. Hirt and B. D. Nichols. Volume of fluid (VOF) method for the dynamics of free boundaries. *J. Comput. Phys.*, 39(1):201 – 225, 1981.
- [14] R. Janodet, G. Vaudor, G. Lartigue, P. Bénard, V. Moureau, and R. Mercier. An unstructured conservative level-set algorithm coupled with dynamic mesh adaptation for the computation of liquid-gas flows. In *29th European Conference on Liquid Atomization and Spray Systems (ILASS Europe)*, Paris, France, September 2019.
- [15] D. Juric and G. Tryggvason. A front-tracking method for dendritic solidification. *J. Comput. Phys.*, 123(1):127 – 148, 1996.
- [16] D. Juric and G. Tryggvason. Computations of boiling flows. *Int. J. Multiph. Flow*, 24(3):387 – 410, 1998.
- [17] J. Kim and P. Moin. Application of a fractional-step method to incompressible Navier-Stokes equations. *J. Comput. Phys.*, 59(2):308 – 323, 1985.
- [18] B. Lalanne, L. R. Villegas, S. Tanguy, and F. Risso. On the computation of viscous terms for incompressible two-phase flows with level set/ghost fluid method. *J. Comput. Phys.*, 301:289 – 307, 2015.
- [19] M. Malandain. *Simulation massivement parallèle des écoulements turbulents à faible nombre de Mach*. PhD thesis, Institut National des Sciences Appliquées de Rouen, 2013.
- [20] M. Malandain, N. Maheu, and V. Moureau. Optimization of the deflated conjugate gradient algorithm for the solving of elliptic equations on massively parallel machines. *J. Comput. Phys.*, 238:32 – 47, 2013.
- [21] H. Montazeri, S. H. Zandavi, and A. Bazylak. Sharp interface models for two-phase flows: Insights towards new approaches. *Comput. Methods Appl. Mech. Eng.*, 322:238 – 261, 2017.
- [22] V. Moureau, P. Domingo, and L. Vervisch. Design of a massively parallel cfd code for complex geometries. *CR Mécanique*, 339(2):141 – 148, 2011. High Performance Computing.
- [23] V. Moureau, P. Domingo, and L. Vervisch. From large-eddy simulation to direct numerical simulation of a lean premixed swirl flame: Filtered laminar flame-pdf modeling. *Combust. Flame*, 158(7):1340 – 1357, 2011.
- [24] D. Q. Nguyen, R. P. Fedkiw, and M. Kang. A boundary condition capturing method for incompressible flame discontinuities. *J. Comput. Phys.*, 172(1):71 – 98, 2001.

- [25] S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *J. Comput. Phys.*, 79(1):12 – 49, 1988.
- [26] G. Sahut. *Simulation numérique de l'ébullition sur maillages non structurés*. PhD thesis, 2019.
- [27] L. E. Scriven. On the dynamics of phase growth. *Chem. Eng. Sci.*, 10(1):1 – 13, 1959.
- [28] G. L. G. Sleijpen and D. R. Fokkema. BiCGStab( $\ell$ ) for linear equations involving unsymmetric matrices with complex spectrum. *Electron. Trans. Numer. Anal.*, 1:11–32, 1993.
- [29] G. Son and V. K. Dhir. Numerical Simulation of Film Boiling Near Critical Pressures With a Level Set Method. *J. Heat Transfer*, 120(1):183–192, 02 1998.
- [30] S. Tanguy, T. Ménard, and A. Berlemont. A level set method for vaporizing two-phase flows. *J. Comput. Phys.*, 221(2):837 – 853, 2007.
- [31] S. Tanguy, M. Sagan, B. Lalanne, F. Couderc, and C. Colin. Benchmarks and numerical methods for the simulation of boiling flows. *J. Comput. Phys.*, 264:1–22, 2014.
- [32] G. Tryggvason, B. Bunner, A. Esmaceli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nas, and Y.-J. Jan. A front-tracking method for the computations of multiphase flow. *J. Comput. Phys.*, 169(2):708 – 759, 2001.
- [33] G. Tryggvason, R. Scardovelli, and S. Zaleski. *Direct Numerical Simulations of Gas-Liquid Multiphase Flows*. Cambridge University Press, 2011.
- [34] S. O. Unverdi and G. Tryggvason. A front-tracking method for viscous, incompressible, multi-fluid flows. *J. Comput. Phys.*, 100(1):25 – 37, 1992.
- [35] S. W. J. Welch. Local simulation of two-phase flows including interface tracking with mass transfer. *J. Comput. Phys.*, 121(1):142 – 154, 1995.
- [36] S. W. J. Welch and J. Wilson. A volume of fluid based method for fluid flows with phase change. *J. Comput. Phys.*, 160(2):662 – 682, 2000.
- [37] S. Wilhelm, G. Balarac, O. Métais, and C. Ségoufin. Analysis of head losses in a turbine draft tube by means of 3D unsteady simulations. *Flow Turbul. Combustion*, 97(4):1255–1280, 2016.