



HAL
open science

A conservative pressure based solver with collocated variables on unstructured grids for two-fluid flows with phase change

Lei Zhang, Anela Kumbaro, Jean-Michel Ghidaglia

► **To cite this version:**

Lei Zhang, Anela Kumbaro, Jean-Michel Ghidaglia. A conservative pressure based solver with collocated variables on unstructured grids for two-fluid flows with phase change. *Journal of Computational Physics*, 2019, 390, pp.265 - 289. 10.1016/j.jcp.2019.04.007 . hal-03484472

HAL Id: hal-03484472

<https://hal.science/hal-03484472>

Submitted on 20 Dec 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.



Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License

A conservative pressure based solver with collocated variables on unstructured grids for two-fluid flows with phase change

Lei Zhang^{a,*}, Anela Kumbaro^b, Jean-Michel Ghidaglia^a

^a*CMLA, ENS Paris-Saclay, CNRS, Université Paris-Saclay, 94235 Cachan, France*

^b*CEA-Saclay DEN, DM2S, STMF, LMEC F-91191 Gif-sur-Yvette, France*

Abstract

Numerical simulation of two-phase flows based on the two-fluid six-equation model is the focus of this work. This approach is widely used in thermo-hydraulics for many industrial applications, particularly nuclear power plant safety analysis. A semi-implicit numerical method that has been successfully adopted in several industrial thermo-hydraulic codes, because of its efficiency and robustness, has been extended. In this paper, the governing equations are solved on unstructured grids with collocated variables to accommodate complicated geometries, and a deferred-correction method is used to deal with the non-orthogonality of unstructured grids. In addition, the numerical method is conservative, meaning that it is capable of capturing a shock wave exactly in single-phase flows, and ensuring the conservation of physically conservative quantities of a two-phase flow mixture. Numerical benchmarks and industrial test cases are performed in order to validate the numerical method and to evaluate its behavior with respect to different physical configurations.

Keywords: Two-phase flows, two-fluid six-equation model, pressure-based semi-implicit method, conservative scheme, unstructured grids

1. Introduction

Two-phase flows are encountered in many industrial fields, such as transport of gas or oil, treatment of waste water in chemical engineering, combustion of fuel in the automotive industry, etc. Understanding two-phase flow phenomenon is crucial for design, optimization, performance and safety analysis of industrial installations. This work is especially motivated by the needs of thermo-hydraulic calculations for nuclear reactors, including nuclear power plant safety analysis, and conception and optimization of nuclear thermo-hydraulic components. For example, a two-phase mixture will be ejected from the break when an accident occurs and there is a breach of the primary coolant system. It is thus essential to be able to predict the transient state of the fluid so that appropriate cooling of the

*Corresponding author

Email address: lzhang@cmla.ens-cachan.fr (Lei Zhang)

10 reactor core can be conducted.

Numerical simulation is becoming increasingly popular in the prediction of two-phase flows for complex industrial-type problems, due to increasing availability of computing power and resources, as well as the significant progress made in modeling the associated physical complexity. However, the modeling approach, that involves direct simulation of the instantaneous flow field by taking into account appropriate jump conditions across the interface between the two phases, has not yet been adapted for engineering analysis of complex multi-component flow fields [1, pp. 55-57]. The complexity of moving and deformable interfaces requires huge computing resources that exceed the computational capacity of today's computers. In addition, fine modeling of complex interfaces is normally not required in engineering problems. A good representation of the macroscopic aspects of fluid motion including accurate evaluation of the pressure is all that is needed. This leads to a second modeling approach called two-fluid modeling, which describes the flow at a larger scale than the interfaces and applies averaging procedures (e.g. in time [1], in space [2] or otherwise [3]), in order to derive macroscopic field and constitutive equations for a two-phase mixture. However, because the interface information is lost during averaging operations, it is important to model the mass, momentum and energy exchanges between the two phases at the interfaces. The validity and precision of two-fluid models largely depends on the description of these exchanges. According to various simplifying hypotheses, different mathematical two-fluid models [4] can be derived following this approach and have been used for industrial applications: homogeneous equilibrium model [5], drift-flux model [6], two-fluid six-equation model [7], multi-field models [8, 9, 10], etc. Here, we choose to work with the two-fluid six-equation model because it is the simplest model that takes into consideration both the mechanical and thermal disequilibrium between the two phases. In addition, this model is very close to multi-field models, thus generalization of the present work for more elaborated multi-field models should be straightforward. The two-fluid six-equation model adopted in this work is a single-pressure model with the two phasic pressures taken to be equal. However, two-pressure models also exist, such as the Baer-Nunziato model [11], which was initially developed for granular flow and was later extended to two-phase flows, e.g. in [12] using a pressure relaxation method and in the industrial RELAP-7 code [13].

A significant effort has been invested in the development of numerical methods for simulating fluid flows. In general, these methods can be classified into two main categories: pressure-based methods [6, 14] and density-based methods [15, 16, 17]. Density-based solvers experience stiffness problems as well as a loss of accuracy when approaching the low Mach number limit. Moreover, the time step is limited by the most rapidly propagating wave, which becomes restrictive for low Mach number flows. In two-phase flows, it is common to have a large range of Mach numbers due to the mixture of liquid and gas. Therefore, pressure-based methods, which normally perform well at low Mach numbers, are advantageous. Pressure is selected as a dependent variable in preference to density

45 because changes in pressure are significant at all speeds as opposed to variations in density, which become very small at low Mach numbers [18]. Pressure-based methods were originally developed for incompressible flows, such as the MAC method [19] and the SIMPLE algorithm [20]. Later these were extended to compressible flows, which yielded the ICE method [14] and methods in the SIMPLE family [21]. Based on the ICE method, improvements and adaptations were then made in [6] for
50 nuclear thermo-hydraulic applications, leading to the semi-implicit method. Exchange terms as well as the terms describing pressure wave propagation are determined implicitly, and the remaining terms describing fluid convection are calculated explicitly. This allows a larger stable time step compared to explicit methods, and the main feature of the method is the ability to reduce the discretized equations to a linear system for pressures at the new time-step. This semi-implicit method has been adopted by
55 several industrial codes (e.g. the RELAP5 code [22], the TRAC code [23]) for thermo-hydraulic analysis in nuclear reactors because of its efficiency and robustness. In addition, to enhance numerical stability, a fully implicit method [24] was also developed that treated all terms implicitly. Thus solution of a non-linear system of algebraic equations is needed at each time step. Furthermore, as a compromise between the more stable fully implicit method and the more computationally effective semi-implicit
60 method, some intermediary methods (e.g. the nearly implicit method [25], the SETS method [26]) were also developed. A general comparison between these numerical methods can be found in [27, 4, 28].

Indeed, the above-mentioned pressure-based methods are used by many industrial codes for thermo-hydraulic analysis in nuclear reactors, such as the CATHARE code [24] and the RELAP5 code [22]. These codes are well-known for their robustness and capacity to handle a variety of flow configurations.
65 However, they are limited when geometries become complicated because staggered grids are employed in order to prevent the well-known checker-board problem [29, pp. 196-200]. The checker-board problem occurs when a centered scheme is used for both pressure gradient and velocity at a cell face on collocated grids. On staggered grids vector components and scalar variables are stored at different locations that are shifted half a control volume in each coordinate direction. Implementation of a
70 staggered grid arrangement is thus tedious as different control volumes are used for the velocities in different directions and the scalar variables. Also, it becomes more complicate to treat boundary conditions and use a multigrid approach to accelerate convergence [30]. In addition, it is difficult for the staggered arrangement to be generalized for unstructured grids that simplify the description of complex geometries. In order to prevent the checker-board problem and reinforce numerical stability
75 for collocated grids, Rhie and Chow [31] proposed an interpolation method to compute the velocity at a cell face, where a damping term is added to the centered scheme. This technique eliminates the pressure-velocity decoupling and the checker-board problem in collocated arrangements. In [30], a comparison between staggered grids and collocated grids using Rhie-Chow interpolation demonstrated that collocated methods are equally efficient and reliable as staggered methods. Since, the interest in

80 use of collocated grids has grown, and indeed the Rhie-Chow interpolation method is now widely used in pressure-based methods [32, 33, 30]. Later, improvements were made to eliminate some defects of the standard Rhie-Chow interpolation and to extend its applicability. For example, the formulation was modified in [34] to alleviate the dependence of the convergent solution on the size of the time step, and was revised in [35] to treat discontinuous external forces. A generalized formulation can be found 85 in [36]. Adaptation of the interpolation method to unstructured grids was proposed in [32, 37, 38], where methods for the calculation of the pressure gradient at a cell face on unstructured grids were proposed based on discretization of the diffusion term [37, 39, 40].

The objective of this work is therefore to develop a pressure-based method for unstructured grids with collocated variables, which is capable of predicting complex two-phase flow phenomena. In 90 order to accommodate complex geometries using collocated grids, as in the nuclear thermo-hydraulic CUPID code [9, 41], the Rhie-Chow interpolation [31] method is employed to compute the velocity at cell faces so that the checker-boarder problem can be mitigated. Use is made of a modified version of Rhie-Chow interpolation [32, 37, 38] that is adapted for unstructured grids. This work is based on the semi-implicit method [6] due to its robustness and successful application in different physical 95 configurations. Linearizations are introduced in the semi-implicit discretization in order to simplify system resolution, meaning that the discretized equations can be reduced to a linear system for the new time-step pressure variables. Hence, iteration is not needed at each time step in the original scheme in contrast to methods from the SIMPLE family [18]. However, in order to obtain more accurate results on unstructured grids, a larger computational stencil for a cell than just that involving the nearest 100 neighboring cells is needed in the linear system for pressure. This affects storage requirements and increases the effort needed to solve the linear system. It is thus proposed that a deferred-correction type method [29, pp. 122-124] is used, which obtains solutions using an iterative procedure with a small computational stencil.

The semi-implicit method is not conservative for numerical convenience [6]. This is due to the 105 fact that non-conservative variables are used in the discretized equations and that linearization is introduced in order to obtain a linear system for the pressure variables. Nevertheless, conservation is an important property for a numerical scheme, which enables exact capturing of shock waves in single-phase flows [42, pp. 237-239][43]. Also, in two-phase flows this property ensures the exact conservation of mass, momentum and total energy of a mixture and can avoid reduction of time step 110 length and numerical errors produced in transient results [44]. Therefore, in this work a conservative method is developed that retains the robustness of the semi-implicit method. The conservative method involves the addition of a supplementary correction step to the semi-implicit method in order to make it conservative. In addition, this correction step is completely explicit, and is thus computationally effective.

115 Strategies for dealing with phase appearance and disappearance in two-phase flows are also pre-
sented in this work. These effects can be caused, for example, by direct heating of a liquid, phase
separation by gravity or geometry, etc. Numerical methods often encounter extra difficulties in cases
when a phase appears or disappears, including loss of void fraction positivity, certain terms becoming
singular, etc [45]. As in [46], a residual volume fraction is allowed for the disappearing phase, and the
120 void fraction is imposed in an interval predefined between 0 and 1. In addition, a formula is used to
redefine the associated volume fraction at a cell face when a phase disappears, such that the volume
fraction ratio of the disappearing phase at a cell face and the related cell center remains defined.
The performance of these strategies is demonstrated using several numerical test cases. Regarding
the equations of state (EOS) that are used to compute the state variables of each fluid, simplistic
125 approaches, such as ideal or stiffened gas EOS, are limited. This is especially true in cases that are
near saturation or where liquid compressibility is important [47]. Thus a tabulated EOS [47] based
on empirical correlations [48] is coupled with the numerical method in order to have more precise
physical representation of the fluid states, hence allowing accurate prediction of the transient behavior
of two-phase flows. Only simple models [10] for interfacial exchange terms are used here, and they
130 are combined with the tabulated EOS to obtain realistic results compared with experimental data.
Elaborate models that depend on given flow regimes are used in industrial codes [49, 22].

This paper is organized as follows. Section 2 introduces the two-fluid six-equation mathematical
model, together with equations of state and constitutive relations to close the mathematical system.
Section 3 contains a detailed description of the semi-implicit method generalized for collocated variables
135 on unstructured grids, as well as a conservative version of the numerical scheme. Section 4 presents
single-phase and two-phase numerical results to validate the numerical scheme and to illustrate its
properties with respect to different physical configurations.

2. Governing equations of the two-fluid six-equation model

A two-fluid six-equation model is adopted for the analysis of two-phase flows in order to take into
140 account both mechanical and thermal non-equilibrium [4, 1, 22]. This model can be obtained by time
and/or space averaging of local instantaneous equations for each phase [1], which requires modeling of
interfacial transfer terms in order to close the mathematical model.

2.1. Conservation equations

The continuity, momentum and energy equations for phase k ($k=v$ for vapor and $k=l$ for liquid) are respectively:

$$\frac{\partial}{\partial t}(\alpha_k \rho_k) + \nabla \cdot (\alpha_k \rho_k \mathbf{u}_k) = \Gamma_k, \quad (1)$$

$$\frac{\partial}{\partial t}(\alpha_k \rho_k \mathbf{u}_k) + \nabla \cdot (\alpha_k \rho_k \mathbf{u}_k \otimes \mathbf{u}_k) = \alpha_k \nabla P + \alpha_k \rho_k \mathbf{g} + \mathbf{M}_{ik} + \Gamma_k \mathbf{u}^i, \quad (2)$$

$$\frac{\partial}{\partial t}(\alpha_k \rho_k h_k) + \nabla \cdot (\alpha_k \rho_k e_k \mathbf{u}_k) = \alpha_k \frac{\partial}{\partial t} P - P \nabla \cdot (\alpha_k \mathbf{u}_k) + \Gamma_k h_{ik} + Q_{ik}. \quad (3)$$

where P is the thermodynamic pressure, \mathbf{g} is the gravitational acceleration, \mathbf{u}^i is the interfacial velocity, α_k , ρ_k , \mathbf{u}_k , e_k , h_k , are respectively the volume fraction, the density, the velocity, the specific internal energy and the specific enthalpy for phase k ; Γ_k , \mathbf{M}_{ik} , Q_{ik} , h_{ik} are respectively the interfacial mass transfer term due to phase change, the interfacial drag force, the interfacial energy transfer term and the specific enthalpy at the interface between the two phases. In addition, the thermodynamic relation $h_k = e_k + P/\rho_k$ is used, and the volume fractions of the two phases are constrained by $\alpha_v + \alpha_l = 1$. Instead of using specific internal energy, as in [9], specific enthalpy is used here due to the initial stages of development being made with a water table that had pressure and specific enthalpy as independent variables. This minor modification does not change the main features of the numerical method. One can remark that the time derivative term on the right hand side is $-P \frac{\partial \alpha_k}{\partial t}$ in the original energy equation involving the specific internal energy [24, 9]. This term physically represents the work that the interfacial pressure does on the interface between the two phases, because the term $\frac{\partial \alpha_k}{\partial t}$ is the time rate of change in the volume fraction due to the movement of the interface [50, pp. 216]. The use of $\frac{\partial}{\partial t}(\alpha_k \rho_k h_k)$ instead of $\frac{\partial}{\partial t}(\alpha_k \rho_k e_k)$ on the left hand side of energy equation (3) changes the time derivative term on the right hand side from $-P \frac{\partial \alpha_k}{\partial t}$ to $\alpha_k \frac{\partial}{\partial t} P$ using the thermodynamic relation $h_k = e_k + P/\rho_k$.

It is known that the single-pressure model given above is not unconditionally hyperbolic [4]. Actually, this model is hyperbolic if and only if one of the following conditions is satisfied [51, pp. 28][52]:

$$u_v = u_l, \text{ or } \alpha_v \rho_l = 0, \text{ or } \alpha_l \rho_v = 0, \text{ or } |u_v - u_l| \geq \sqrt{\gamma^2 ((\alpha_l \rho_v)^{1/3} + (\alpha_v \rho_l)^{1/3})^3}, \quad (4)$$

where $\gamma^2 = \frac{c_v^2 c_l^2}{\alpha_v \rho_l c_l^2 + \alpha_l \rho_v c_v^2}$ and c_k is the phasic sound speed. The hyperbolicity boundary $\sqrt{\gamma^2 ((\alpha_l \rho_v)^{1/3} + (\alpha_v \rho_l)^{1/3})^3}$ is of the order of the magnitude of the mixture sound speed, and is much larger than the range of relative velocities in common physical applications. Thus mathematically the single-pressure model presents an ill-posed initial value problem. Several authors have proposed adding some regularizing terms in order to make the system hyperbolic, for example adding a virtual mass term [53] to the momentum equation, or including an interfacial pressure default term [45] in the system. The latter is used in our studies, which involves the addition of $-(P - P_{\text{interface}}) \cdot \nabla \alpha_k$ to the right hand side of the

momentum equation [45]. The interfacial pressure default $P - P_{\text{interface}}$ characterizes the difference between the average bulk pressure and the interfacial pressure. For example, in the CATHARE code [49], the interfacial pressure default is given by:

$$P - P_{\text{interface}} = \delta \frac{\alpha_v \alpha_l \rho_v \rho_l}{\alpha_v \rho_l + \alpha_l \rho_v} |\mathbf{u}_v - \mathbf{u}_l|^2, \quad (5)$$

160 where the hyperbolicity of the system is ensured when the parameter $\delta > 1$ and $\frac{|\mathbf{u}_v - \mathbf{u}_l|^2}{c^2} \ll 1$, where c is a mean speed of sound in the mixture.

The two-fluid six-equation model (1)-(3), which takes into consideration both the mechanical and thermal disequilibrium between the two phases, is used in this work because of its wide application in industrial codes [24, 22]. Moreover, the present work can be directly generalized to more elaborate 165 multi-field models [9] thanks to the closeness of this model to multi-field models.

2.2. Equations of state (EOS) and constitutive relations

In order to close system (1)-(3), equations of state and constitutive relations for terms on the right-hand side of the governing equations should be specified.

170 As mentioned above, pressure and specific enthalpy are chosen as independent state variables, and all other thermodynamic variables except saturation properties are expressed as a function of these two variables. For example, the density $\rho_k = \rho_k(P, h_k)$. The saturation properties are represented as a function of pressure, e.g. the specific enthalpy of saturated vapor $h_v^{\text{sat}} = h_v^{\text{sat}}(P)$. Tabulated EOS are required to calculate accurately the dynamic behavior of two-phase flows, thus a water table 175 named Quicksteam [47] based on IAPWS97 [48] has been developed at the Centre de Mathématiques et de Leurs Applications (CMLA). The independent variables in IAPWS97 are the pressure P and the temperature T , and backward equations are available to compute the temperature T directly from the pressure P and the specific enthalpy h for liquid and gas without need for an iterative method. All thermodynamic variables can then be obtained using correlation formulas in IAPWS97 [48, 47] 180 as functions of P and T . However, for fluids in a metastable state, the backward equations used to calculate T from P and h are no longer valid. Thus for our application, the Newton-Raphson method is used to obtain the temperature T by solving $f(T) = \mathcal{H}(P, T) - h = 0$, where P , h and the function \mathcal{H} are given. A detailed explanation of the tabulated EOS and their integration in the simulation code is given in Appendix A.

Simple models for the interfacial exchange terms are now given. The aim of this initial work is not to obtain accurate results comparable with experimental data, but rather to show that realistic results can be obtained using the numerical method and to illustrate its behavior for different physical situations. Indeed, to allow for good comparison with experimental data, industrial codes use complicated

empirical relations for different flow regimes [22]. In order to conserve mass, momentum and energy at an interface separating the two phases, the following jump conditions should be satisfied:

$$\sum_{k=v,l} \Gamma_k = 0, \quad \sum_{k=v,l} \mathbf{M}_{ik} + \Gamma_k \mathbf{u}^i = 0, \quad \sum_{k=v,l} \Gamma_k h_{ik} + Q_{ik} = 0. \quad (6)$$

According to [54], an expression for the interfacial velocity is adopted here: $\mathbf{u}^i = \alpha_v \mathbf{u}_v + \alpha_l \mathbf{u}_l$, and the interfacial drag force is:

$$\mathbf{M}_{iv} = -\mathbf{M}_{iv} = -\frac{3}{8} \alpha_v \alpha_l (\alpha_v \rho_v + \alpha_l \rho_l) \frac{C_d}{R_b} |\mathbf{u}_r| \mathbf{u}_r, \quad (7)$$

where $\mathbf{u}_r = \mathbf{u}_v - \mathbf{u}_l$, C_d is the drag coefficient, and R_b is a mean particle radius. For example, R_b is the mean radius of bubbles in a bubbly flow (i.e., gas bubbles in liquid). As in [10], the energy transfer term is given by:

$$Q_{ik} = \frac{\alpha_v \alpha_l \rho_k (h_{ik} - h_k)}{\tau_{ik}}, \quad \text{for } k = v, l, \quad (8)$$

where τ_{ik} represents the relaxation time. For water and steam $\tau_{ik} = \tau$ is of the order of 10^{-3} s. Specific enthalpies at the interface between the two phases are saturation values: $h_{ik} = h_k^{\text{sat}}$, for $k = v, l$. From jump conditions (6), the following expression can be deduced:

$$\Gamma_v = -\Gamma_l = -\frac{Q_{iv} + Q_{il}}{h_{iv} - h_{il}} = -\frac{\alpha_v \alpha_l \rho_v (h_v^{\text{sat}} - h_v) + \alpha_v \alpha_l \rho_l (h_l^{\text{sat}} - h_l)}{\tau (h_v^{\text{sat}} - h_l^{\text{sat}})}, \quad (9)$$

so that the energy transfer terms are:

$$Q_{iv} + \Gamma_v h_{iv} = -(Q_{il} + \Gamma_l h_{il}) = -\frac{\alpha_v \alpha_l \rho_v (h_v^{\text{sat}} - h_v) \cdot h_l^{\text{sat}} + \alpha_v \alpha_l \rho_l (h_l^{\text{sat}} - h_l) \cdot h_v^{\text{sat}}}{\tau (h_v^{\text{sat}} - h_l^{\text{sat}})}. \quad (10)$$

185 3. Numerical scheme

In this section, the numerical scheme developed for this work is given in detail. The scheme is semi-implicit in time due to a large linear system for the mean pressure being solved in each cell at time $n + 1$. The scheme is built as follows:

- (i) First (Section 3.1.1), the momentum equations (11) are used to find the mean phasic velocities in each cell at time $n + 1$.
- (ii) Then (Section 3.1.2), using both the mass and energy equations, the scalar variables (void fraction, phasic specific enthalpy) can be determined once the new time-step phasic velocity at a cell face is obtained.
- (iii) Finally (Section 3.1.4), the aforementioned linear system for pressure variables at time $n + 1$ is derived using both the mass and energy equations, respectively (1) and (3), and the momentum equation (2).

Section 3.1 is devoted to the derivation of the scheme in the spirit of [6], [9] and [41]. This leads to a non conservative scheme that does not conserve total mass, total momentum and total energy of a two-phase mixture exactly, and as a result is not capable of correctly capturing shock waves in single-phase flows.. However, Section 3.2 presents the final version of the scheme that is conservative, and therefore captures shock waves correctly in single-phase flows, by ensuring the exact conservation of total mass, total momentum and total energy of a two-phase mixture.

3.1. Semi-implicit method

Since two-phase flow involves a wide range of Mach numbers, from supersonic to near incompressible, a method is sought that is not limited by the classical Courant-Friedrichs-Lewy stability condition $(u + c)\Delta t/\Delta x < 1$, where c is the sound speed, and $u > 0$ is the magnitude of the velocity. Thus, a semi-implicit method is developed in this work that is based on the method used in [6] for simulation of two-phase flow with the drift-flux model. In the semi-implicit method the terms responsible for pressure waves, including the pressure gradient in the momentum equation and the velocity at cell face in mass and energy equations, are determined implicitly. In addition, source terms, which can be stiff, are calculated implicitly. This ensures that the stability of the numerical method is independent of the sound speed, and the time step is limited by a CFL condition involving only the fluid velocity.

Readers can refer to [6, 9, 41] for extra information regarding the semi-implicit numerical method. Before introducing the numerical discretization, as in [6, 55], the momentum equations (2) are rewritten as follows:

$$\alpha_k \rho_k \frac{\partial \mathbf{u}_k}{\partial t} + \nabla \cdot (\alpha_k \rho_k \mathbf{u}_k \otimes \mathbf{u}_k) - \mathbf{u}_k \nabla \cdot (\alpha_k \rho_k \mathbf{u}_k) + \Gamma_k \mathbf{u}_k = -\alpha_k \nabla P + \alpha_k \rho_k \mathbf{g} + \mathbf{M}_{ik} + \Gamma_k \mathbf{u}^i. \quad (11)$$

This modification is reasonable as the semi-implicit method was originally developed for nuclear thermo-hydraulic applications. Rigorous conservation of momentum is not expected due to the existence of large sinks and sources (e.g., pumps, orifices) in nuclear hydraulic systems [6]. Nevertheless, a conservative scheme, which is capable of conserving exactly mass, momentum and total energy of a mixture, is developed in Section 3.2.

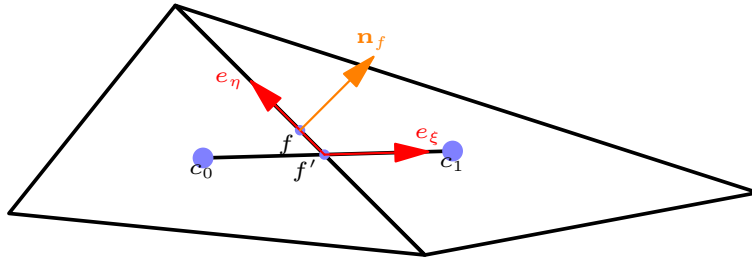


Figure 1: Illustration for an unstructured mesh.

Figure 1 shows a typical mesh used for the numerical method, where the control volume c_0 represents a triangle. The numerical method can also be applied to other mesh types, as long as they are conformal (i.e. identical location of mesh nodes on the faces between cells).

The semi-implicit scheme presented in this section is first-order in time and space, because Euler's method is employed for the temporal discretization and an upwind scheme is used for the spatial discretization.

3.1.1. Momentum equation

Integration of the momentum equation (11) over a control volume c_0 using the finite volume method leads to new time-step phasic velocities (i.e., the velocity of phase k at time t^{n+1}):

$$\begin{aligned} (\alpha_k \rho_k)_{c_0}^n \frac{(\mathbf{u}_k)_{c_0}^{n+1} - (\mathbf{u}_k)_{c_0}^n}{\Delta t} + \frac{1}{V_{c_0}} \sum_f (\alpha_k \rho_k \mathbf{u}_k)_f^n [(\mathbf{u}_k)_f^n \cdot \mathbf{S}_f] - \frac{(\mathbf{u}_k)_{c_0}^n}{V_{c_0}} \sum_f (\alpha_k \rho_k)_f^n [(\mathbf{u}_k)_f^n \cdot \mathbf{S}_f] + (\Gamma_k \mathbf{u}_k)_{c_0}^n \\ = -(\alpha_k)_{c_0}^n \nabla P_{c_0}^{n+1} + (\alpha_k \rho_k)_{c_0}^n \mathbf{g} + (\Gamma_k \mathbf{u}^i)_{c_0}^n + (\mathbf{M}_{ik})_{c_0}^{n+1}, \end{aligned} \quad (12)$$

where subscript $(\cdot)_f$ indicates the value at a cell face, and the area vector \mathbf{S}_f is defined as $\mathbf{S}_f = \mathbf{n}_f \cdot A_f$, i.e., the unit normal vector \mathbf{n}_f scaled by the area of the face f . The phasic velocity $(\mathbf{u}_k)_f^n$ at a cell face is determined using Rhie-Chow interpolation, which will be explained later (see equation (28)).

In the momentum discretization, additional to the pressure gradient term ∇P , the interfacial drag force \mathbf{M}_{ik} is also implicitly calculated, whereas all other terms are determined explicitly. In order to simplify the numerical method and avoid the necessity of an iterative procedure for phasic velocity time advance, the implicit interfacial term \mathbf{M}_{ik}^{n+1} is linearized with respect to its value at the previous step:

$$\mathbf{M}_{ik}^{n+1} \simeq \mathbf{M}_{ik}^n + \left(\frac{\partial \mathbf{M}_{ik}}{\partial \mathbf{u}_v} \right)^n \cdot (\mathbf{u}_v^{n+1} - \mathbf{u}_v^n) + \left(\frac{\partial \mathbf{M}_{ik}}{\partial \mathbf{u}_l} \right)^n \cdot (\mathbf{u}_l^{n+1} - \mathbf{u}_l^n). \quad (13)$$

Spatial links within one phase and between the two phases are considered in the linearization of \mathbf{M}_{ik}^{n+1} to enhance numerical stability. Thus we need to solve a linear system of dimension $2nd \times 2nd$ for each cell in order to obtain a relation between the new time phasic velocity and the new time pressure, where $nd \in \{1, 2, 3\}$ is the spatial dimension of the physical problem considered.

Substituting the linearization of \mathbf{M}_{ik}^{n+1} into the discretized momentum equation (12) and with some arrangements yields

$$d\mathbf{M} \cdot \begin{pmatrix} \mathbf{u}_v^{n+1} - \mathbf{u}_v^n \\ \mathbf{u}_l^{n+1} - \mathbf{u}_l^n \end{pmatrix}_{c_0} = \begin{pmatrix} \gamma'_v \\ \gamma'_l \end{pmatrix}_{c_0}^n - \Delta t \begin{pmatrix} \frac{\nabla P^{n+1}}{\rho_v^n} \\ \frac{\nabla P^{n+1}}{\rho_l^n} \end{pmatrix}_{c_0}, \quad (14)$$

where

$$\mathbf{dM} = \begin{pmatrix} \mathbf{I}^{nd} - \frac{1}{\alpha_v \rho_v} \frac{\partial \mathbf{M}_{iv}}{\partial \mathbf{u}_v} \Delta t & -\frac{1}{\alpha_v \rho_v} \frac{\partial \mathbf{M}_{iv}}{\partial \mathbf{u}_l} \Delta t \\ -\frac{1}{\alpha_l \rho_l} \frac{\partial \mathbf{M}_{il}}{\partial \mathbf{u}_v} \Delta t & \mathbf{I}^{nd} - \frac{1}{\alpha_l \rho_l} \frac{\partial \mathbf{M}_{il}}{\partial \mathbf{u}_l} \Delta t \end{pmatrix}_{c_0}^n, \quad (15)$$

$$\begin{aligned} (\gamma'_k)_{c_0}^n = \Delta t & \left\{ -\frac{\sum_f (\alpha_k \rho_k \mathbf{u}_k)_f^n [(\mathbf{u}_k)_f^n \cdot \mathbf{S}_f]}{(\alpha_k \rho_k)_{c_0}^n V_{c_0}} + \frac{(\mathbf{u}_k)_{c_0}^n \sum_f (\alpha_k \rho_k)_f^n [(\mathbf{u}_k)_f^n \cdot \mathbf{S}_f]}{(\alpha_k \rho_k)_{c_0}^n V_{c_0}} \right. \\ & \left. + \mathbf{g} - \frac{(\Gamma_k \mathbf{u}_k)_{c_0}^n}{(\alpha_k \rho_k)_{c_0}^n} + \frac{(\Gamma_k \mathbf{u}^i)_{c_0}^n}{(\alpha_k \rho_k)_{c_0}^n} + \frac{(\mathbf{M}_{ik})_{c_0}^n}{(\alpha_k \rho_k)_{c_0}^n} \right\}, \quad (16) \end{aligned}$$

and \mathbf{I}^{nd} is an identity matrix of dimension $nd \times nd$. Multiplying both sides of equation (14) by the inverse of matrix \mathbf{dM} leads to a new time-step phasic velocity:

$$(\mathbf{u}_k)_{c_0}^{n+1} = (\gamma_k)_{c_0}^n - (\boldsymbol{\beta}_k)_{c_0}^n \nabla P_{c_0}^{n+1}, \quad (17)$$

where $(\gamma_k)_{c_0}^n$ is obtained by

$$\begin{pmatrix} \gamma_v \\ \gamma_l \end{pmatrix}_{c_0}^n = \begin{pmatrix} \mathbf{u}_v \\ \mathbf{u}_l \end{pmatrix}_{c_0}^n + \mathbf{dM}^{-1} \begin{pmatrix} \gamma'_v \\ \gamma'_l \end{pmatrix}_{c_0}^n, \quad (18)$$

and $(\boldsymbol{\beta}_k)_{c_0}^n$ is a $nd \times nd$ matrix with elements given by (i-th row and j-th column):

$$(\boldsymbol{\beta}_v)_{ij} = \frac{\Delta t}{\rho_v} (\mathbf{dM}^{-1})_{ij} + \frac{\Delta t}{\rho_l} (\mathbf{dM}^{-1})_{ij'}, \quad (19)$$

$$(\boldsymbol{\beta}_l)_{ij} = \frac{\Delta t}{\rho_v} (\mathbf{dM}^{-1})_{i'j} + \frac{\Delta t}{\rho_l} (\mathbf{dM}^{-1})_{i'j'}, \quad (20)$$

where $i' = i + nd$, $j' = j + nd$.

235

At this stage, the new time-step phasic velocities cannot be obtained from equation (17) since the new time-step pressure is still unknown. In fact, this equation is only used when the new time-step pressure is available after the pressure equation (34) is solved, as explained in Section 3.1.4.

240

3.1.2. Mass and energy equations

Discretization of the mass and energy equations is now explained. The transported quantities are calculated explicitly, whereas the phasic velocities at a cell face and the source terms are determined implicitly, thus the discretized mass equation reads (adopting the notation $\Delta(\cdot) = (\cdot)^{n+1} - (\cdot)^n$):

$$\frac{(\rho_k)_{c_0}^n (\Delta \alpha_k)_{c_0} + (\alpha_k)_{c_0}^n (\Delta \rho_k)_{c_0}}{\Delta t} + \frac{\sum_f (\alpha_k \rho_k)_f^n [(\mathbf{u}_k)_f^{n+1} \cdot \mathbf{S}_f]}{V_{c_0}} = (\Gamma_k)_{c_0}^{n+1}. \quad (21)$$

Similarly, the discretized energy equation becomes:

$$\begin{aligned} & \frac{(\rho_k)_{c_0}^n (h_k)_{c_0}^n (\Delta\alpha_k)_{c_0} + (\alpha_k)_{c_0}^n (\rho_k)_{c_0}^n (\Delta h_k)_{c_0} + (\alpha_k)_{c_0}^n (h_k)_{c_0}^n (\Delta\rho_k)_{c_0}}{\Delta t} + \frac{\sum_f (\alpha_k \rho_k e_k)_f^n \left[(\mathbf{u}_k)_f^{n+1} \cdot \mathbf{S}_f \right]}{V_{c_0}} \\ & = (\alpha_k)_{c_0}^n \frac{\Delta P_{c_0}}{\Delta t} - P_{c_0}^n \frac{\sum_f (\alpha_k)_f^n \left[(\mathbf{u}_k)_f^{n+1} \cdot \mathbf{S}_f \right]}{V_{c_0}} + (Q_{ik} + \Gamma_k h_{ik})_{c_0}^{n+1}. \end{aligned} \quad (22)$$

In order to simplify the discretization, the implicit mass and energy transfers are linearized with respect to the independent state variables P and h_k :

$$q_k^{n+1} - q_k^n = \left(\frac{\partial q_k}{\partial P} \right)_{h_v, h_l} \Delta P + \left(\frac{\partial q_k}{\partial h_v} \right)_{P, h_l} \Delta h_v + \left(\frac{\partial q_k}{\partial h_l} \right)_{P, h_v} \Delta h_l, \quad (23)$$

where $q_k \in \{\Gamma_k, (Q_{ik} + \Gamma_k h_{ik})\}$. In addition, the density is also linearized around the independent state variables using the EOS:

$$\Delta\rho_k = \left(\frac{\partial\rho_k}{\partial P} \right)_{h_k} \Delta P + \left(\frac{\partial\rho_k}{\partial h_k} \right)_P \Delta h_k. \quad (24)$$

The method used to obtain partial derivatives $\left(\frac{\partial\rho_k}{\partial P} \right)_{h_k}$ and $\left(\frac{\partial\rho_k}{\partial h_k} \right)_P$ from the tabulated EOS Quicksteam data is explained in Appendix A. The four discretized equations can thus be put in matrix form:

$$(\mathcal{A})_{c_0}^n \cdot (\mathbf{x})_{c_0} = (\mathbf{s})_{c_0}^n - \sum_f (\mathbf{g})_f^n \left[(\mathbf{u}_v)_f^{n+1} \cdot \mathbf{S}_f \right] - \sum_f (\mathbf{l})_f^n \left[(\mathbf{u}_l)_f^{n+1} \cdot \mathbf{S}_f \right], \quad (25)$$

where \mathbf{x} is the vector of unknown variables, $\mathbf{x} = (\Delta h_v, \Delta h_l, \Delta\alpha_v, \Delta P)^T$, and \mathcal{A} is a 4×4 matrix. The expressions for $(\mathcal{A})_{c_0}^n$, $(\mathbf{g})_f^n$, $(\mathbf{l})_f^n$ and $(\mathbf{s})_{c_0}^n$ are as follows:

$$\begin{aligned} (\mathcal{A})_{c_0}^n &= \begin{pmatrix} \alpha_v \frac{\partial\rho_v}{\partial h_v} - \frac{\partial\Gamma_v}{\partial h_v} \Delta t & -\frac{\partial\Gamma_v}{\partial h_l} \Delta t & \rho_v & \alpha_v \frac{\partial\rho_v}{\partial P} - \frac{\partial\Gamma_v}{\partial P} \Delta t \\ -\frac{\partial\Gamma_l}{\partial h_v} \Delta t & \alpha_l \frac{\partial\rho_l}{\partial h_l} - \frac{\partial\Gamma_l}{\partial h_l} \Delta t & -\rho_l & \alpha_l \frac{\partial\rho_l}{\partial P} - \frac{\partial\Gamma_l}{\partial P} \Delta t \\ \alpha_v (\rho_v + h_v \frac{\partial\rho_v}{\partial h_v}) - \frac{\partial Q_v}{\partial h_v} \Delta t & -\frac{\partial Q_v}{\partial h_l} \Delta t & \rho_v h_v & \alpha_v (h_v \frac{\partial\rho_v}{\partial P} - 1) - \frac{\partial Q_v}{\partial P} \Delta t \\ -\frac{\partial Q_l}{\partial h_v} \Delta t & \alpha_l (\rho_l + h_l \frac{\partial\rho_l}{\partial h_l}) - \frac{\partial Q_l}{\partial h_l} \Delta t & -\rho_l h_l & \alpha_l (h_l \frac{\partial\rho_l}{\partial P} - 1) - \frac{\partial Q_l}{\partial P} \Delta t \end{pmatrix}_{c_0}^n, \\ (\mathbf{g})_f^n &= \frac{\Delta t}{V_{c_0}} \begin{pmatrix} (\alpha_v \rho_v)_f^n \\ 0 \\ (P)_{c_0}^n (\alpha_v)_f^n + (\alpha_v \rho_v e_v)_f^n \\ 0 \end{pmatrix}, \quad (\mathbf{l})_f^n = \frac{\Delta t}{V_{c_0}} \begin{pmatrix} 0 \\ (\alpha_l \rho_l)_f^n \\ 0 \\ (P)_{c_0}^n (\alpha_l)_f^n + (\alpha_l \rho_l e_l)_f^n \end{pmatrix}, \quad (\mathbf{s})_{c_0}^n = \Delta t \begin{pmatrix} \Gamma_v \\ \Gamma_l \\ Q_v \\ Q_l \end{pmatrix}_{c_0}^n, \end{aligned}$$

where $Q_k = Q_{ik} + \Gamma_k h_{ik}$.

Linearization of source terms. Here, detailed calculation of the linearization of source terms from Section 2.2 is given. This choice is simple, where the physical quantities are partially derived, is validated by numerical test cases. Using the source terms given by equations (9) and (10), we have

$$\Gamma_v^{n+1} - \Gamma_v^n = -\frac{(\alpha_v \alpha_l \rho_v)^n \frac{dh_v^{\text{sat}}}{dP} + (\alpha_v \alpha_l \rho_l)^n \frac{dh_l^{\text{sat}}}{dP}}{\tau(h_v^{\text{sat},n} - h_l^{\text{sat},n})} \Delta P + \frac{(\alpha_v \alpha_l \rho_v)^n}{\tau(h_v^{\text{sat},n} - h_l^{\text{sat},n})} \Delta h_v + \frac{(\alpha_v \alpha_l \rho_l)^n}{\tau(h_v^{\text{sat},n} - h_l^{\text{sat},n})} \Delta h_l. \quad (26)$$

$$(Q_{iv} + \Gamma_v h_{iv})^{n+1} = (Q_{iv} + \Gamma_v h_{iv})^n - \frac{(\alpha_v \alpha_l \rho_v)^n \frac{dh_v^{\text{sat}}}{dP} \cdot h_l^{\text{sat},n} + (\alpha_v \alpha_l \rho_l)^n \frac{dh_l^{\text{sat}}}{dP} \cdot h_v^{\text{sat},n}}{\tau(h_v^{\text{sat},n} - h_l^{\text{sat},n})} \Delta P + \frac{(\alpha_v \alpha_l \rho_v)^n \cdot h_l^{\text{sat},n}}{\tau(h_v^{\text{sat},n} - h_l^{\text{sat},n})} \Delta h_v + \frac{(\alpha_v \alpha_l \rho_l)^n \cdot h_v^{\text{sat},n}}{\tau(h_v^{\text{sat},n} - h_l^{\text{sat},n})} \Delta h_l. \quad (27)$$

3.1.3. Phasic velocity at a cell face

Phasic velocities at the cell faces are required in the discretized mass and energy equations (25) in order to find the unknown variables in vector \mathbf{x} . However, for our collocated grid arrangement the phasic velocities are stored at the cell centers, and the centered scheme that is used to calculate the cell face velocity using cell center velocity will produce the well-known checker-board problem [29, pp. 198]. In this work therefore, the checker-board problem is prevented by employing Rhie-Chow interpolation [31] to find the new time-step phasic velocity at cell face f (between cell c_0 and cell c_1):

$$(\mathbf{u}_k)_f^{n+1} = \frac{1}{2} [(\mathbf{u}_k)_{c_0}^{n+1} + (\mathbf{u}_k)_{c_1}^{n+1}] - (\beta_k)_f^n [(\nabla P)_f^{n+1} - \overline{(\nabla P)_f^{n+1}}], \quad (28)$$

where $(\beta_k)_f^n = \frac{1}{2} [(\beta_k)_{c_0}^n + (\beta_k)_{c_1}^n]$, $\overline{(\nabla P)_f}$ is taken to be the average of the derivatives for two adjacent cells. As in [32], use is made of transformation from physical coordinates (x, y) to computational coordinates (ξ, η) as shown in Figure 1, hence

$$(\nabla P)_f = \frac{\mathbf{n}_f}{\mathbf{n}_f \cdot \mathbf{e}_\xi} \left(\frac{\partial P}{\partial \xi} \right)_f + \overline{(\nabla P)_f} - \overline{(\nabla P)_f} \cdot \mathbf{e}_\xi \frac{\mathbf{n}_f}{\mathbf{n}_f \cdot \mathbf{e}_\xi}, \quad (29)$$

where \mathbf{n}_f is the unit normal vector to cell face f and \mathbf{e}_ξ is the unit vector pointing from the center of cell c_0 to the center of cell c_1 . Therefore, the new time-step phasic velocity at cell face f becomes:

$$(\mathbf{u}_k)_f^{n+1} = \frac{1}{2} [(\mathbf{u}_k)_{c_0}^{n+1} + (\mathbf{u}_k)_{c_1}^{n+1}] - (\beta_k)_f^n \left[\left(\frac{\partial P}{\partial \xi} \right)_f^{n+1} - \overline{(\nabla P)_f^{n+1}} \cdot \mathbf{e}_\xi \right] \frac{\mathbf{n}_f}{\mathbf{n}_f \cdot \mathbf{e}_\xi}. \quad (30)$$

The same Rhie-Chow interpolation method for unstructured grids is proposed in [36, 38]. **The**

245 **Rhie-Chow formula (30) is not directly used to obtain new time-step phasic velocities at a cell face, indeed a simpler formula (32) will be used for this purpose. An important point regarding equation (30) is that it is to be combined with the discretized momentum equation, the discretized mass and energy equations to finally deduce the new time-step pressure equation (34), as presented in Section 3.1.4.**

Only the basic Rhie-Chow interpolation (28) involving the pressure gradient is used in this work. However, more generalized interpolation, such as [36], should be employed in future, which includes contributions from the time step, treatment of discontinuous external forces, etc. In [8], the authors suggested a volume fraction gradient term be included in the Rhie-Chow formulation in order to avoid odd-even decoupling of the velocity and volume fraction fields.

3.1.4. System resolution

This section presents the derivation of a linear system for the pressure at time $n + 1$ from the previously-described discretized equations (17) and (25), and demonstrates the solution of the overall system.

Discretization of the mass and energy equations gives a relation between phasic velocities and pressure at the new time-step, as can be seen by inverting matrix \mathcal{A} in system (25). Denoting \mathbf{A}_4 as the fourth row vector of the matrix \mathcal{A} inverse, the last equation of the system (25) becomes:

$$(\Delta P)_{c_0} = \mathbf{A}_4 \cdot (\mathbf{s})_{c_0}^n - \mathbf{A}_4 \cdot \sum_f (\mathbf{g})_f^n \left[(\mathbf{u}_v)_f^{n+1} \cdot \mathbf{S}_f \right] - \mathbf{A}_4 \cdot \sum_f (\mathbf{l})_f^n \left[(\mathbf{u}_l)_f^{n+1} \cdot \mathbf{S}_f \right]. \quad (31)$$

Moreover, discretization of the momentum equation (17) and Rhie-Chow interpolation (30) yield two equations relating respectively the cell center velocity and the cell face velocity to pressure for the new time-step. The resolution strategy thus involves combining the three previously-mentioned equations in order to obtain an equation in which only the pressure is the unknown variable.

The above procedure is detailed as follows. First, expression (17) is substituted, which calculates the new time-step velocity at a cell center in Rhie-Chow formula (30), in addition assuming that $(\beta_k)_f^n \approx (\beta_k)_{c_0}^n \approx (\beta_k)_{c_1}^n$, we can obtain the phasic velocity at a cell face:

$$(\mathbf{u}_k)_f^{n+1} = \frac{1}{2} [(\gamma_k)_{c_0}^n + (\gamma_k)_{c_1}^n] - \frac{(\beta_k)_f^n \cdot \mathbf{n}_f}{\mathbf{n}_f \cdot \mathbf{e}_\xi} \frac{P_{c_1}^{n+1} - P_{c_0}^{n+1}}{\Delta \xi} + \mathbf{F}_k(P^{n+1}), \quad (32)$$

where $\Delta \xi$ is the distance between cells c_0 and c_1 , and the function $\mathbf{F}_k(P^{n+1})$ represents

$$\mathbf{F}_k(P^{n+1}) = -(\beta_k)_f^n \left[\overline{(\nabla P)_f^{n+1}} - \overline{(\nabla P)_f^{n+1}} \cdot \mathbf{e}_\xi \frac{\mathbf{n}_f}{\mathbf{n}_f \cdot \mathbf{e}_\xi} \right]. \quad (33)$$

The formulation $\mathbf{F}_k(P^{n+1})$ appears due to the fact that unstructured grids are considered, and this can be neglected for orthogonal grids [37]. By substituting equation (32), which relates the phasic velocity $(\mathbf{u}_k)_f^{n+1}$ to the new time-step pressure, into equation (31), which comes from the discretized mass and energy equations, a formula involving only the pressure can be obtained:

$$\begin{aligned} & (\Delta P)_{c_0} - \mathbf{A}_4 \cdot \sum_f \mathbf{C}_f (P_{c_1}^{n+1} - P_{c_0}^{n+1}) + \mathbf{A}_4 \cdot \sum_f \left[(\mathbf{g})_f^n F_v(P^{n+1}) + (\mathbf{l})_f^n F_l(P^{n+1}) \right] \\ & = \mathbf{A}_4 \cdot \left\{ (\mathbf{s})_{c_0}^n - \frac{1}{2} \sum_f \left[(\mathbf{g})_f^n [(\gamma_v)_{c_0}^n + (\gamma_v)_{c_1}^n] \cdot \mathbf{S}_f + (\mathbf{l})_f^n [(\gamma_l)_{c_0}^n + (\gamma_l)_{c_1}^n] \cdot \mathbf{S}_f \right] \right\}, \quad (34) \end{aligned}$$

where $\mathbf{C}_f = \frac{1}{\Delta\xi\mathbf{n}_f\cdot\mathbf{e}_\xi} \left\{ (\mathbf{g})_f^n \left[(\boldsymbol{\beta}_v)_f^n \cdot \mathbf{n}_f \cdot \mathbf{S}_f \right] + (\mathbf{l})_f^n \left[(\boldsymbol{\beta}_l)_f^n \cdot \mathbf{n}_f \cdot \mathbf{S}_f \right] \right\}$.

265

The system resolution procedure can be summarized as follows:

a) Equation (34) is applied to each computational cell in a grid containing N cells, which generates an $N \times N$ linear system for the pressure.

270

b) Once the new time-step pressure is solved using equation (32), the new time-step phasic velocity at a cell face can be obtained, and thus the right hand side of system (25) is known.

c) Finally, the new time-step independent variables contained in vector \mathbf{x} are obtained and the new time-step velocities are updated using equation (17).

275

For orthogonal grids, the calculation of the phasic velocity at a cell face involves only pressures at the center of the two cells neighboring the face (i.e. $\mathbf{F}_k(P^{n+1})$ disappears). However, for unstructured grids a large stencil can become necessary to compute more accurately this term, as indicated by the presence of the term $\mathbf{F}_k(P^{n+1})$ in the formula (32). On the one hand, we would like to keep the stencil as small as possible because the size of the computational stencil affects both the storage requirements and the effort needed to solve the system of linear equations. On the other hand, more accurate results are achieved when more cells than just the nearest neighbors are used to calculate the pressure gradient at a cell face (see equation (29)). In this work, it is proposed that a deferred-correction type method is used [29, pp. 122-124]. This consists of placing the term F_k in the second member of the system (34), and then an iterative procedure is used to approximate the solution. It should be noted that the matrix inverse remains constant throughout the iterative procedure.

280

285 3.1.5. More details

This section presents the calculation of several physical quantities that need to be specified in the resolution of the above-mentioned discretized equations, including physical quantities transported with the fluid motion at a cell face (i.e. $(\alpha_k\rho_k)_f^n$, $(\alpha_k\rho_k\mathbf{u}_k)_f^n$, $(\alpha_k)_f^n$ and $(\alpha_k\rho_k e_k)_f^n$) and the pressure gradient.

Transported quantities. The transported quantities, e.g., $(\alpha_k\rho_k)_f^n$, are determined using an upwind scheme that depends on the sign of the projected phasic velocity at a cell face [56]:

$$(\theta_k)_f^n = \begin{cases} (\theta_k)_{c_0}^n, & \text{if } (\mathbf{u}_k)_f^n \cdot \mathbf{S}_f \geq 0, \\ (\theta_k)_{c_1}^n, & \text{otherwise,} \end{cases} \quad (35)$$

290 where $(\theta_k)_f^n \in \{(\alpha_k\rho_k)_f^n, (\alpha_k)_f^n, (\alpha_k\rho_k e_k)_f^n\}$.

Transported quantity $(\alpha_k \rho_k \mathbf{u}_k)_f^n$. We now explain how the transported quantity $(\alpha_k \rho_k \mathbf{u}_k)_f^n$ is treated. It can be computed like other transported quantities by directly using an upwind scheme, as explained in the last paragraph. Furthermore, its calculation can be performed by multiplying the transported quantity $(\alpha_k \rho_k)_f^n$ by the phasic velocity at the cell face $(\mathbf{u}_k)_f^n$, where $(\alpha_k \rho_k)_f^n$ is calculated using an upwind scheme (see equation (35)), and the phasic velocity at the cell face $(\mathbf{u}_k)_f^n$ is determined using Rhie-Chow interpolation (see equation (28)). However, this second method leads to numerical instability, because the numerical diffusion term containing u'' is missing.

A one-dimensional problem with $u_k > 0$ is considered to illustrate this point. The spatial discretization of the convective term $\nabla \cdot (\alpha_k \rho_k \mathbf{u}_k \otimes \mathbf{u}_k) - \mathbf{u}_k \nabla \cdot (\alpha_k \rho_k \mathbf{u}_k) = \alpha_k \rho_k \nabla \mathbf{u}_k \cdot \mathbf{u}_k$ thus becomes

$$A = \frac{(\alpha_k \rho_k u_k)_{j+1/2}^n (u_k)_{j+1/2}^n - (\alpha_k \rho_k u_k)_{j-1/2}^n (u_k)_{j-1/2}^n}{\Delta x} - (u_k)_j^n \frac{(\alpha_k \rho_k)_{j+1/2}^n (u_k)_{j+1/2}^n - (\alpha_k \rho_k)_{j-1/2}^n (u_k)_{j-1/2}^n}{\Delta x}, \quad (36)$$

where j is the cell index, $j \pm 1/2$ indicates a cell face, Δx is the spacial discretization length scale.

In order to simplify A , we consider the one-dimensional case Rhie-Chow formulation (30):

$$\begin{aligned} (u_k)_{j+\frac{1}{2}}^{n+1} &= \frac{1}{2} [(u_k)_{j+1}^{n+1} + (u_k)_j^{n+1}] - (\beta_k)_{j+\frac{1}{2}}^n \left(\frac{P_{j+1}^{n+1} - P_j^{n+1}}{\Delta x} - \frac{1}{2} \frac{P_{j+2}^{n+1} - P_j^{n+1}}{2\Delta x} - \frac{1}{2} \frac{P_{j+1}^{n+1} - P_{j-1}^{n+1}}{2\Delta x} \right) \\ &= \frac{1}{2} [(u_k)_{j+1}^{n+1} + (u_k)_j^{n+1}] - \frac{1}{4} (\beta_k)_{j+\frac{1}{2}}^n [P'''(\Delta x)^2], \end{aligned} \quad (37)$$

where the last equality comes from Taylor expansion, and higher order terms are neglected. The variable β_k contains the coefficient Δt (see equations (19) and (20)), and we take $u_k \Delta t / \Delta x = O(1)$ because the CFL condition is $u_k \Delta t / \Delta x < 1$, hence

$$(u_k)_{j+\frac{1}{2}}^{n+1} = \frac{1}{2} [(u_k)_{j+1}^{n+1} + (u_k)_j^{n+1}] + O(\Delta x^3). \quad (38)$$

Therefore for transported quantity $(\alpha_k \rho_k u_k)_{j\pm 1/2}^n = (\alpha_k \rho_k)_{j\pm 1/2}^n \cdot (u_k)_{j\pm 1/2}^n$, using the upwind scheme (35) for $(\alpha_k \rho_k)_{j\pm 1/2}^n$ and equation (38) for $(u_k)_{j\pm 1/2}^n$ allows simplification of the convective term A as

$$A = \frac{\frac{1}{2} (\alpha_k \rho_k)_j^n (u_k)_{j+1/2}^n [(u_k)_{j+1}^n - (u_k)_j^n] + \frac{1}{2} (\alpha_k \rho_k)_{j-1}^n (u_k)_{j-1/2}^n [(u_k)_j^n - (u_k)_{j-1}^n]}{\Delta x} + O(\Delta x^2). \quad (39)$$

A Taylor expansion of the previous expression leads to

$$A = \alpha_0 \rho_0 u_0 u' + O(\Delta x^2), \quad (40)$$

where (α_0, ρ_0, u_0) is a reference state, and it can be observed that the numerical diffusion term containing $u'' \Delta x$ disappears. In order to handle this problem, it is proposed that numerical viscosity is added by modifying the calculation of $(\alpha_k \rho_k u_k)_{j\pm \frac{1}{2}}^n$:

$$(\alpha_k \rho_k u_k)_{l+\frac{1}{2}}^n = (\alpha_k \rho_k)_{l+\frac{1}{2}}^n \left\{ (u_k)_{l+\frac{1}{2}}^n + \delta [(u_k)_{l+1}^n - (u_k)_l^n] \right\}, \quad (41)$$

where $l \in \{j, j - 1\}$, and δ is a parameter to be determined. With this modification, the spatial discretization of the term $\nabla \cdot (\alpha_k \rho_k \mathbf{u}_k \otimes \mathbf{u}_k) - \mathbf{u}_k \nabla \cdot (\alpha_k \rho_k \mathbf{u}_k)$ becomes

$$A^* = \alpha_0 \rho_0 u_0 u' + \delta \alpha_0 \rho_0 u_0 u'' \Delta x + O(\Delta x^2). \quad (42)$$

300 By choosing $\delta = -1/2$ in this work, the numerical diffusion is the same as the case where the transported quantity $(\alpha_k \rho_k \mathbf{u}_k)_f^n$ is calculated directly by the upwind scheme (35).

Pressure gradient. When the new time-step pressure is obtained, its gradient, that is required in the expression (17), can be evaluated using the Green-Gauss reconstruction method [57, pp. 275]:

$$\nabla P_{c_0} = \frac{1}{V_{c_0}} \sum_f P_f \mathbf{S}_f, \quad (43)$$

where f represents all the faces of cell c_0 . The pressure at a cell face can be determined by

$$P_f = \frac{\omega_0 P_0 + \omega_1 P_1}{\omega_0 + \omega_1}, \quad (44)$$

where ω_0 and ω_1 are respectively the inverse distance between the center of cell c_0 and the center of face f and the distance between the center of cell c_1 and the center of face f . This relation only leads to a second-order approximation of P_f when the segment $c_0 \rightarrow c_1$ and face f intersection point coincides with the centroid of face f . This is not the case for a general unstructured grid, as shown in Figure 1, rather segment $c_0 \rightarrow c_1$ and face f intersect at point f' , that is different from the centroid of face f . Several methods to locate the intersection point and calculate the corresponding correction to find P_f are introduced in [57, pp. 275-280]. However, here we adopt a simpler method [33]. When the pressure gradient is obtained at each cell the following formula can be used to calculate P_f :

$$P_f = \frac{[P_{c_0} + \nabla P_{c_0} \cdot (\mathbf{c}_f - \mathbf{c}_0)] + [P_{c_1} + \nabla P_{c_1} \cdot (\mathbf{c}_f - \mathbf{c}_1)]}{2}. \quad (45)$$

This P_f is then used again in the Green-Gauss reconstruction (43) to improve the accuracy of the pressure gradient calculation. This suggests an iterative approach for computing successively improved approximations of the gradients. However, more iterations causes an increased effective stencil and can lead to oscillatory results. Therefore, only one or two iterations are typically used in practice. 305

An alternative method for calculation of the pressure gradient is the least squares approach [58]. From this, the formula (45) is used to obtain the pressure at cell faces, as required for the conservative method. These two approaches for calculation of the pressure gradient are used in this work, giving 310 equally satisfactory results.

3.2. Conservative semi-implicit method

In this section a conservative version of the semi-implicit method is developed. The non conservative nature of the semi-implicit method is due to the fact that non-conservative variables are used in the discretized equations and that linearization is introduced in order to obtain a linear system for the
315 pressure variables.

However as is commonly known, conservation is essential for a method to be able to exactly capture shock waves in single-phase flows [42, pp. 237-239]. For two-phase flows, it is desirable that the physical conservation of mass, momentum and total energy for the mixture should be respected by the numerical method. Indeed, non conservativeness can produce mass and energy errors for some classes
320 of transients during time advance, either resulting in automatic reduction of time step length, thus increased run time, or the growth of unacceptably large errors in transient results [44]. For example, in RELAP5 [22], at each iteration the density calculated by the semi-implicit numerical method is compared with a more conservative calculation, and if the difference is too large the time step length will be reduced.

Therefore, a new additional step which renders the numerical scheme conservative is introduced
325 here. This step is explicit in the sense that no linear system needs to be solved in this additional step, thus the computational cost is negligible. It should be made clear that the whole conservative scheme is still semi-implicit. As seen in the following presentation of the conservative scheme, the original non conservative pressure solver, presented in Section 3.1, is still required in order to calculate the new
330 time-step phasic velocity at a cell face. The whole conservative scheme is actually a combination of the original non conservative pressure solver and the additional step represented by equations (46)-(48), and therefore the scheme is semi-implicit.

The main idea of the conservative method is to consider the pressure solver presented above only as a means to calculate the new time-step phasic velocity at a cell face, and not as a final solution. Then the conservative variables are solved directly, as shown in equations (46)-(48), with starred terms $(\cdot)^*$ that are obtained by the non conservative pressure solver. Finally, the primitive variables α_k , \mathbf{u}_k , P and h_k are derived from the conservative variables. By construction, this method is conservative,

which ensures that mass, momentum, and total energy are conserved for the two-phase mixture.

$$\frac{(\alpha_k \rho_k)_{c_0}^{n+1} - (\alpha_k \rho_k)_{c_0}^n}{\Delta t} + \frac{\sum_f (\alpha_k \rho_k)_f^n [(\mathbf{u}_k)_f^{n+1} \cdot \mathbf{S}_f]}{V_{c_0}} = (\Gamma_k)_{c_0}^*, \quad (46)$$

$$\begin{aligned} \frac{(\alpha_k \rho_k \mathbf{u}_k)_{c_0}^{n+1} - (\alpha_k \rho_k \mathbf{u}_k)_{c_0}^n}{\Delta t} + \frac{\sum_f (\alpha_k \rho_k \mathbf{u}_k)_f^n [(\mathbf{u}_k)_f^{n+1} \cdot \mathbf{S}_f]}{V_{c_0}} + (\alpha_k)_{c_0}^n \frac{\sum_f P_f^* \mathbf{S}_f}{V_{c_0}} &= (\alpha_k \rho_k)_{c_0}^n \mathbf{g} + (\Gamma_k \mathbf{u}^i)_{c_0}^n \\ &+ (\mathbf{M}_{ik})_{c_0}^{n+1}, \end{aligned} \quad (47)$$

$$\begin{aligned} \frac{(\alpha_k \rho_k E_k)_{c_0}^{n+1} - (\alpha_k \rho_k E_k)_{c_0}^n}{\Delta t} + \frac{\sum_f [(\alpha_k \rho_k E_k)_f^n + (\alpha_k)_f^n P_f^*] [(\mathbf{u}_k)_f^{n+1} \cdot \mathbf{S}_f]}{V_{c_0}} + (P)_{c_0}^* \frac{(\alpha_k)_{c_0}^* - (\alpha_k)_{c_0}^n}{\Delta t} \\ = (\alpha_k \rho_k)_{c_0}^n \mathbf{g} \cdot (\mathbf{u}_k)_{c_0}^n + \left[\Gamma_k \left(h_{ik} + \frac{|\mathbf{u}^i|^2}{2} \right) + Q_{ik} \right]_{c_0}^* + (\mathbf{M}_{ik})_{c_0}^{n+1} \cdot (\mathbf{u}^i)_{c_0}^n. \end{aligned} \quad (48)$$

Similar to the non conservative semi-implicit method, it is necessary to linearize the implicit term $(\mathbf{M}_{ik})^{n+1}$ to avoid an iterative procedure. However, this linearization means that the numerical discretization is no longer conservative. For example, this can be seen in the momentum discretization, as the sum of the discretized momentum equations for the two fluids is not a consistent discretization for the two-phase mixture momentum equation in conservative form:

$$\sum_{k=v,l} \frac{\partial \alpha_k \rho_k \mathbf{u}_k}{\partial t} + \sum_{k=v,l} \nabla \cdot (\alpha_k \rho_k \mathbf{u}_k \otimes \mathbf{u}_k) = -\nabla P + \sum_{k=v,l} \alpha_k \rho_k \mathbf{g}. \quad (49)$$

A different set of equations, which are solved numerically, are used in order to preserve the conservativeness of the numerical method when interfacial terms are involved. The sum of the phasic momentum equation in conservative form (49) is discretized directly. In addition, the difference of the phasic momentum equation in the same form as the non conservative semi-implicit method is used:

$$\begin{aligned} \frac{\partial \mathbf{u}_v}{\partial t} - \frac{\partial \mathbf{u}_l}{\partial t} + \frac{\nabla \cdot (\alpha_v \rho_v \mathbf{u}_v \otimes \mathbf{u}_v)}{\alpha_v \rho_v} - \frac{\nabla \cdot (\alpha_l \rho_l \mathbf{u}_l \otimes \mathbf{u}_l)}{\alpha_l \rho_l} - \frac{\mathbf{u}_v \nabla \cdot (\alpha_v \rho_v \mathbf{u}_v)}{\alpha_v \rho_v} + \frac{\mathbf{u}_l \nabla \cdot (\alpha_l \rho_l \mathbf{u}_l)}{\alpha_l \rho_l} \\ + \frac{\mathbf{u}_v \Gamma_v}{\alpha_v \rho_v} - \frac{\mathbf{u}_l \Gamma_l}{\alpha_l \rho_l} = - \left(\frac{1}{\rho_v} - \frac{1}{\rho_l} \right) \nabla P + \frac{\mathbf{M}_{iv}}{\alpha_v \rho_v} - \frac{\mathbf{M}_{il}}{\alpha_l \rho_l} + \frac{\Gamma_v \mathbf{u}^i}{\alpha_v \rho_v} - \frac{\Gamma_l \mathbf{u}^i}{\alpha_l \rho_l}. \end{aligned} \quad (50)$$

Use of the two previous equations ensures that the momentum of the two-phase mixture is conserved numerically. A different approach is adopted for the energy equation. The new time-step energy is calculated directly using the discretized equation (48), by initially dropping the term $(\mathbf{M}_{ik})^{n+1} \cdot (\mathbf{u}^i)^n$. The energy term calculated this way is denoted by $(\widetilde{\alpha_k \rho_k E_k})^{n+1}$, as it is not the final solution. Then the new time-step energy term is finally corrected by

$$(\alpha_k \rho_k E_k)^{n+1} = (\widetilde{\alpha_k \rho_k E_k})^{n+1} + \Delta t [(\mathbf{M}_{ik})^{n+1} \cdot (\mathbf{u}^i)^n], \quad (51)$$

where $(\mathbf{M}_{ik})^{n+1}$ is already available from the resolution of the mass and momentum equations.

3.3. Phase appearance and disappearance

335

In two-phase flows, phase appearance and disappearance can be encountered in many situations, e.g., phase appearance due to heating of the liquid (boiling), phase disappearance in phase separation

due to gravity, etc. Numerical difficulties may be observed when a phase appears or disappears [45], such as loss of void fraction positivity, convergence difficulties, etc.

Some strategies to deal with phase appearance and disappearance can be found in previous work. In the CATHARE code [46], the interfacial mass and energy transfer terms are conditioned so that the calculated void fraction does not exceed a predefined valid interval. Moreover, when a phase disappears, it is considered to have reached its saturation temperature and its velocity is assumed to be equal to the velocity of the other phase. In the MINCS code [59], a new parameter $\hat{\alpha}_k$ that satisfies $\hat{\alpha}_k/\alpha_k \rightarrow 0$ as $\alpha_k \rightarrow \alpha_{min}$ is defined, where α_{min} is a volume fraction threshold. The right hand sides of equations (1)-(3) are then expressed in terms of newly introduced $\hat{\alpha}_k$. This ensures that the division of the right hand sides of equations (1)-(3) by α_k will not cause problems when α_k is very small, as we have $\hat{\alpha}_k/\alpha_k \rightarrow 0$ in this situation. In [45], the authors presented some methods to deal with the loss of hyperbolicity of the two-fluid system and the lack of positivity for a density-based method.

Here, some basic ideas adopted in this work are presented to deal with phase appearance and disappearance. For the above numerical scheme, when void fraction α_v approaches 0 or 1, several terms in the numerical discretization will become singular. The performance of these strategies will be demonstrated through several numerical results in the next section.

Firstly, the void fraction is defined for the interval $[\alpha_{min}, \alpha_{max}]$, so if at the end of a step the calculated void fraction is out of this predefined interval, the value is corrected by, for example, imposing $\alpha_v = \alpha_{min}$ if $\alpha_v < \alpha_{min}$. Then in the momentum equation, it is required to divide both sides of equation (12) by $\alpha_k \rho_k$ to calculate the new time-step phasic velocity. For terms containing the coefficient $\alpha_k \rho_k$, like \mathbf{M}_{ik} defined by equation (7), $\alpha_k \rho_k$ can be directly eliminated, thus this division does not pose problem. However, for other terms, e.g., $\sum_f (\alpha_v \rho_v \mathbf{u}_v)_f^n [(\mathbf{u}_v)_f^n \cdot \mathbf{S}_f] / V_{c0}$ for the vapor when α_v approaches 0, two methods can be considered, the first is to divide these terms by $\max(\alpha_v, \alpha_{min}) \cdot \rho_v$ instead of $\alpha_v \rho_v$ to avoid division by an excessively small number; the second method is to redefine $(\alpha_v)_f$ in a manner that the division $(\alpha_v)_f / \alpha_v$ is well defined:

$$\frac{1}{(\alpha_v)_f} = \frac{1}{2} \cdot \left(\frac{1}{(\alpha_v)_K} + \frac{1}{(\alpha_v)_L} \right), \quad (52)$$

with K and L being the two neighboring cells of the considered face, so that $(\alpha_v)_f = \frac{2(\alpha_v)_K(\alpha_v)_L}{(\alpha_v)_K + (\alpha_v)_L}$, for $0 < (\alpha_v)_K, (\alpha_v)_L < \epsilon$ where ϵ is a small number, the division $(\alpha_v)_f / \alpha_v$ remains defined. Afterwards, the inversion of matrix \mathcal{A} is required in the step used to obtain the pressure equation (31). When source terms are not considered, the determinant of this matrix is:

$$\det(\mathcal{A}) = \alpha_v \alpha_l \left[\rho_v^2 \alpha_l \left(\frac{\partial \rho_l}{\partial h_l} + \rho_l \frac{\partial \rho_l}{\partial P} \right) + \rho_l^2 \alpha_v \left(\frac{\partial \rho_v}{\partial h_v} + \rho_v \frac{\partial \rho_v}{\partial P} \right) \right], \quad (53)$$

thus for void fraction α_v approaching 0 or 1, the matrix becomes singular. One can refer to [51, pp.

55-56] for the calculation of $\det(\mathcal{A})$ in the general case where source terms are included. The result is similar to (53), with the determinant also containing the coefficient $\alpha_v\alpha_l$. The terms inside the square brackets of equation (53) are evaluated for a simple case where the gas is considered to behave as an ideal gas and the liquid is incompressible:

$$P = (\gamma - 1)\rho_v e_v, \quad \frac{\partial \rho_l}{\partial P} = 0, \quad \frac{\partial \rho_l}{\partial h_l} = 0, \quad (54)$$

where γ is a constant coefficient for the gas. In this case the determinant of matrix \mathcal{A} is

$$\det(\mathcal{A}) = \alpha_v^2 \alpha_l \rho_l^2 \frac{\gamma}{(\gamma - 1)^2} \frac{P}{h_v^2}. \quad (55)$$

In this simple case, it is noticeable that for $\alpha_k \neq 0$ (coexistence of the two phases) the matrix \mathcal{A} remains invertible even though only one of the fluids is incompressible. Matrix \mathcal{A} needs to be inverted in equation (25) in order to find the unknown variables $(\mathbf{x})_{c_0}$

$$(\mathbf{x})_{c_0} = \frac{({}^t\mathcal{C})_{c_0}^n}{\det(\mathcal{A})} \cdot \left\{ (\mathbf{s})_{c_0}^n - \sum_f (\mathbf{g})_f^n [(\mathbf{u}_v)_f^{n+1} \cdot \mathbf{S}_f] - \sum_f (\mathbf{l})_f^n [(\mathbf{u}_l)_f^{n+1} \cdot \mathbf{S}_f] \right\}, \quad (56)$$

where \mathcal{C} is the cofactor matrix of \mathcal{A} satisfying $\mathcal{A}^{-1} = ({}^t\mathcal{C})/\det(\mathcal{A})$, and the elements of the cofactor matrix also contain the coefficients α_v and/or α_l . Consequently, like for the momentum equation, the coefficient $\alpha_v\alpha_l$ can be eliminated from both the determinant and the cofactor matrix when using equation (56) to solve for the unknown variables. Yet, for certain terms in the cofactor matrix only α_v or α_l is present. For example, the first row and column element C_{11} only contains the coefficient α_l (see [51, pp.56-58] for the complete calculation of the cofactor matrix), so as in the momentum equation, two methods can be used to calculate $C_{11}/\alpha_v\alpha_l$. Either the calculation $C_{11}/\max(\alpha_v, \alpha_{\min})\alpha_l$ is made instead of $C_{11}/\alpha_v\alpha_l$, or as C_{11} is to be multiplied by the corresponding element inside the square brackets of system (56), that is $\Delta t(\Gamma_v)_{c_0}^n + \Delta t \cdot (\alpha_v\rho_v)_f^n [(\mathbf{u}_v)_f^{n+1} \cdot \mathbf{S}_f]/V_{c_0}$, so we can rewrite

$$\frac{C_{11}}{(\alpha_v\alpha_l)_{c_0}^n} \cdot \left[\Delta t(\Gamma_v)_{c_0}^n + \frac{\Delta t}{V_{c_0}} (\alpha_v\rho_v)_f^n [(\mathbf{u}_v)_f^{n+1} \cdot \mathbf{S}_f] \right] = \frac{C_{11}}{(\alpha_l)_{c_0}^n} \cdot \frac{\left[\Delta t(\Gamma_v)_{c_0}^n + \frac{\Delta t}{V_{c_0}} (\alpha_v\rho_v)_f^n [(\mathbf{u}_v)_f^{n+1} \cdot \mathbf{S}_f] \right]}{(\alpha_v)_{c_0}^n}, \quad (57)$$

Since C_{11} contains the coefficient α_l , the division $\frac{C_{11}}{(\alpha_l)_{c_0}^n}$ can be made without any problem. As for the term $\frac{\left[\Delta t(\Gamma_v)_{c_0}^n + \frac{\Delta t}{V_{c_0}} (\alpha_v\rho_v)_f^n [(\mathbf{u}_v)_f^{n+1} \cdot \mathbf{S}_f] \right]}{(\alpha_v)_{c_0}^n}$, the source term typically contains the coefficient $\alpha_v\alpha_l$, and in addition equation (52) can be used to calculate $(\alpha_v)_f^n$ in order to deal with the division $\frac{(\alpha_v\rho_v)_f^n}{(\alpha_v)_{c_0}^n}$. In this way, matrix \mathcal{A} can be inverted, even during phase disappearance. Lastly, as in [46], source terms are conditioned such that the void fraction remains in the range $[\alpha_{\min}, \alpha_{\max}]$. For example, further condensation is not possible when the void fraction is too small.

360 **4. Numerical results**

The numerical method presented in Section 3 is based on a two-fluid six-equation model for simulating two-phase flows. It can be applied directly to simulation of single-phase flows using the Euler equations. Here, in addition to two-phase numerical results to illustrate the behaviour of the numerical method with respect to different physical configurations, single-phase results are also presented to show that the conservative method is capable of capturing shock waves exactly and dealing with low Mach number flows. Unless otherwise stated, the stiffened gas EOS is used for single-phase cases, while tabulated EOS is used for two-phase cases.

365 *4.1. Single-phase cases*

Riemann problems

370 Sod’s shock tube problem [54] can be performed experimentally by a sudden breakdown of a diaphragm (located at $x = 0.5$ m) in a 1-meter long tube separating two initial gas states at different pressures and densities (as in Table 1). An exact solution to the full system of one-dimensional Euler equations exists for this test case, as indicated by the dashed line in Figure 2 for the density and in Figure 3 for the velocity. The test problem is run using a uniform mesh of 800 cells. As represented in 375 Figures 2 and 3, the original scheme does not capture exactly the shock wave compared with the exact solution. Indeed, the intermediate state is not well calculated. Whereas the result obtained with the conservative scheme agrees well with the exact solution. Another Riemann problem, called the Lax shock tube [60], is performed with initial states given in Table 2. Again, the shock wave is captured exactly by the conservative scheme, which is not the case for non conservative scheme (see Figure 4, 380 with 400 cells in the two calculations). Many test cases have been performed in order to compare the two schemes. They show that the conservative scheme is equally as robust as the original scheme, but that it possesses the capacity to capture shock waves.

Table 1: Initial states for Sod’s shock tube problem.

	Left state	Right state
P [Pa]	1.	0.1
u [m/s]	0.	0.
h [J/kg]	3.5	2.8

Table 2: Initial states for Lax shock tube problem.

	Left state	Right state
P [Pa]	3.528	0.571
u [m/s]	0.698	0.
h [J/kg]	27.748	3.997

Bump in a channel

385 As already mentioned, pressure-based methods are intrinsically capable of dealing with low Mach number flows. Here a benchmark test is calculated using the pressure solver to demonstrate its capa-

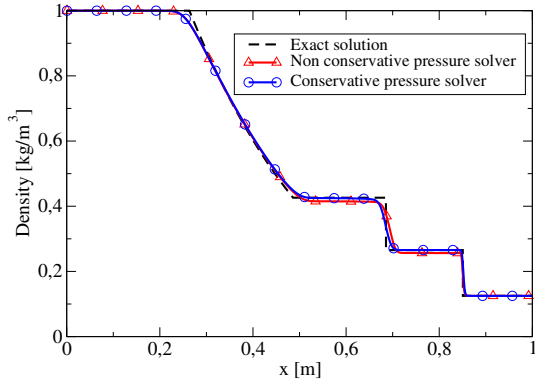


Figure 2: Sod's shock tube: density comparison at $t = 0.2$ s.

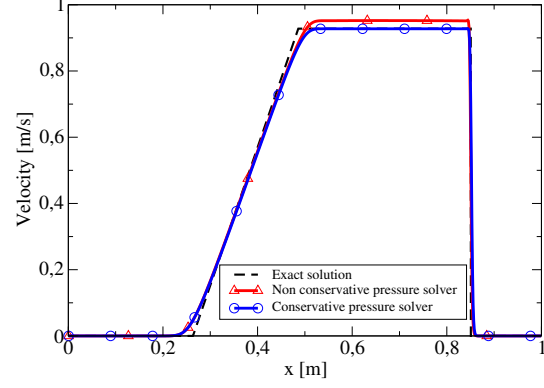


Figure 3: Sod's shock tube: velocity comparison at $t = 0.2$ s.

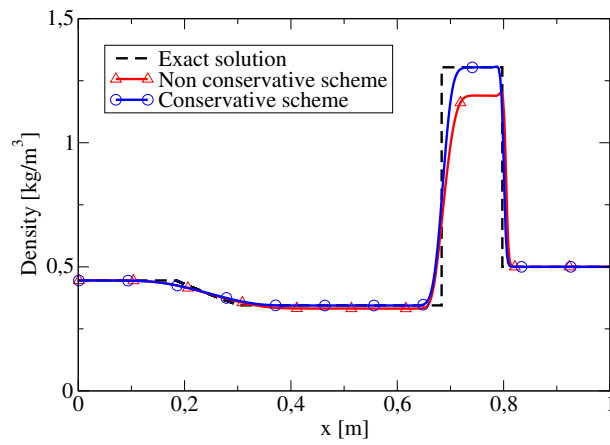


Figure 4: Lax shock tube: density comparison at $t = 0.12$ s.

bilities and to test whether the conservative version of the numerical scheme maintains the property of low Mach number treatment. An inviscid low Mach flow in a channel with a sinusoidal bump [61] (see Figure 5) is considered. The initial state for the fluid in the channel is given in Table 3, which corresponds to an initial Mach number of 0.01. Velocity and specific enthalpy are specified (values taken equal to the initial state) as inlet boundary conditions. The outlet boundary condition is imposed pressure: $P_{\text{exit}} = 10^5$ Pa. Slip conditions are applied to the upper and lower walls. Numerical results obtained using a structured mesh composed of 40×10 cells are shown in Figure 5. In contrast to density-based methods, which normally do not respond to the geometrical disturbance [61] at low Mach number, the pressure solver gives a symmetry of the Mach number contours with respect to the bump. In addition, the conservative version of the solver gives similar results to the original version.

We also evaluate the pressure variation with respect to the Mach number by imposing a varying fluid velocity at the inlet $u_{\text{inlet}} = 1, 0.1, 0.01, 0.001$, where the corresponding Mach numbers are respectively $M = 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}$. The pressure variations obtained with the four Mach numbers are respectively $9.220 \times 10^{-5}, 9.313 \times 10^{-7}, 9.428 \times 10^{-9}$ and 1.106×10^{-10} . As indicated in Figure 6, the pressure variation is of order $O(M^2)$, meaning good behavior at low Mach numbers. The pressure variation is defined by

$$P_{\text{var}} = (P_{\text{max}} - P_{\text{min}}) / P_{\text{max}}, \quad (58)$$

where P_{max} is the pressure maximum, and P_{min} is the pressure minimum.

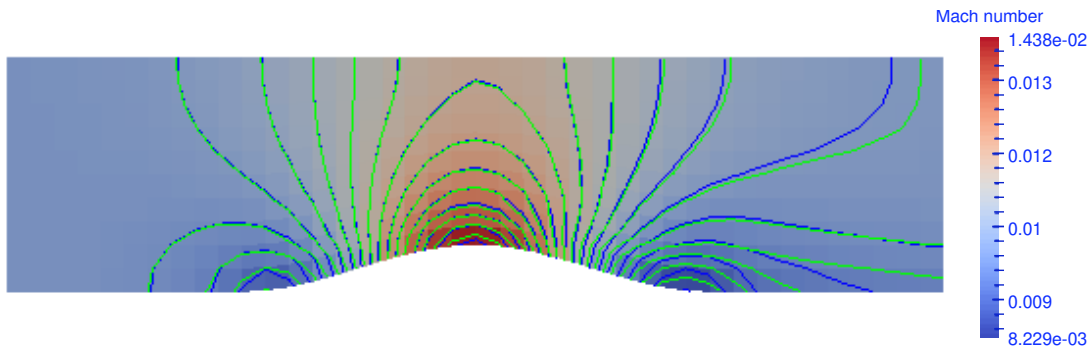


Figure 5: Bump in a channel: Comparison of Mach contour between conservative solver (green line) and non-conservative solver (blue line).

4.2. Two-phase cases

Among the two-phase test cases, only in Ransom’s water faucet test case it is the interfacial pressure default term (5) that makes the two-fluid six-equation model (1)-(3) hyperbolic. The objective is to illustrate the fact that with a refined mesh (less numerical diffusion), the interfacial pressure default term is necessary to stabilize the calculation.

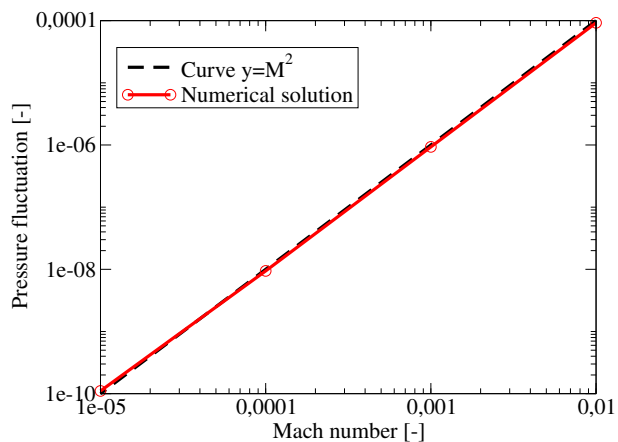


Figure 6: The pressure variation as a function of Mach number.

Table 3: Initial state for bump in a channel.

Physical quantity	Value
P [Pa]	10^5
u [m/s]	1.
h [J/kg]	$2.5 \cdot 10^4$

Table 4: Initial state for Ransom’s faucet.

Physical quantity	Value
α_v [-]	0.2
P [Pa]	10^5
u_v [m/s]	0.
u_l [m/s]	10.
h_v [J/kg]	3245940.
h_l [J/kg]	209283.

Relatively coarse meshes are used for other test cases, hence stability problems are eliminated by numerical diffusion. Coarse meshes are used here because we are in the validation phase of the numerical method, and the focus is on quick calculations to show that physically realistic results can
405 be obtained using the semi-implicit method.

Ransom’s water faucet

This is a non-stationary 1D test case proposed by Ransom [54], which initially consists of a water jet surrounded by stagnant gas contained within a cylindrical channel. The liquid is then accelerated under gravity, and because of mass conservation, this results in a narrowing of the liquid section.
410 The initial state for this test case is given in Table 4. The inlet boundary condition is velocities and enthalpies of both phases, and void fraction, all specified equal to initial state; at the outlet, the pressure is equal to 10^5 Pa.

An analytical expression for void fraction is available for this test, which consists of a discontinuity traveling from the inlet to the outlet, as represented by the dashed line in Figure 7. The numerical

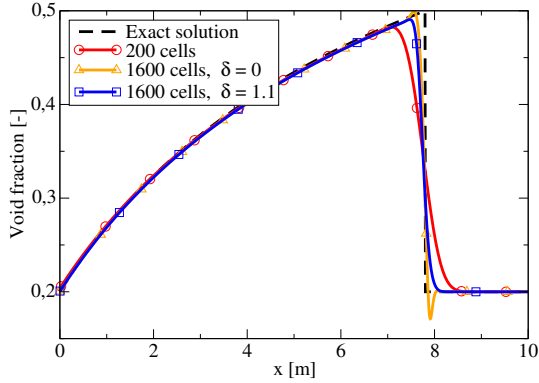


Figure 7: Comparison of void fraction between numerical solution and exact solution for Ransom's faucet at $t = 0.6$ s.

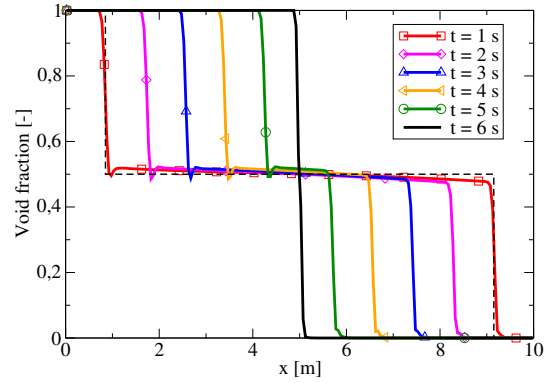


Figure 8: Sedimentation problem: void fraction profile at different instants.

415 results agree well with the exact solution, however for the refined mesh (light brown line), numerical instability is observed due to the non hyperbolicity of the two-fluid model. As expected, adding the interfacial pressure default given by equation (5) with $\delta = 1.1$ eliminates this instability (blue line).

Static sedimentation

This test case simulates the evolution of a mixture of liquid and gas in a vertical column [50, pp. 143-420 145], which validates the ability of the numerical method to cover the whole range of void fractions from 0 to 1. Initially the mixture is homogeneous and the two fluids are at rest. Wall boundary conditions are applied at both ends of the column. Under gravity, heavy liquid moves downward and light gas moves upward, meaning that two volume fraction waves are observed to propagate simultaneously from the bottom and top of the domain. Eventually a sharp interface separating the gas and liquid 425 is established. Figure 8 shows the void fraction profile at different instants, where a uniform mesh of 200 cells is used. The dashed line represents a theoretical result for the void fraction at $t = 1$ s using the assumption that the volumetric fluxes across the two volume fraction waves are compensated, as detailed in [50, pp. 143-145].

Boiling channel

430 This test case consists of two-phase flow in a uniformly heated one-dimensional channel. The heating along the channel results in vapor appearance. As in [45], there are two distinct cases: saturated case and subcooled case, with initial states given in Table 5. For the inlet boundary condition, velocities and enthalpies of both phases, and void fraction, are given equal to the initial state; at the outlet the pressure is fixed at $68.73 \cdot 10^5$ Pa.

435 The difference between these two cases is that in the first case, the liquid is initially saturated, so vapor appears immediately at the inlet; whereas in the second test case, the liquid is initially

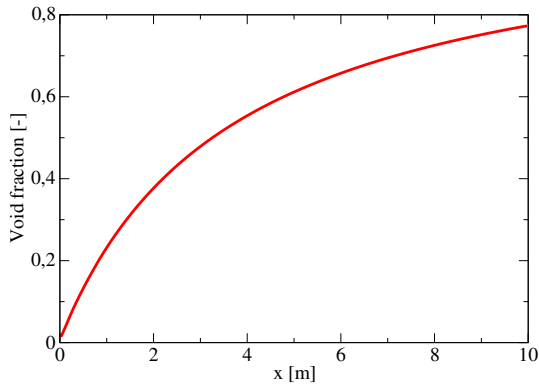


Figure 9: Saturated case for the boiling channel problem: void fraction profile at steady state.

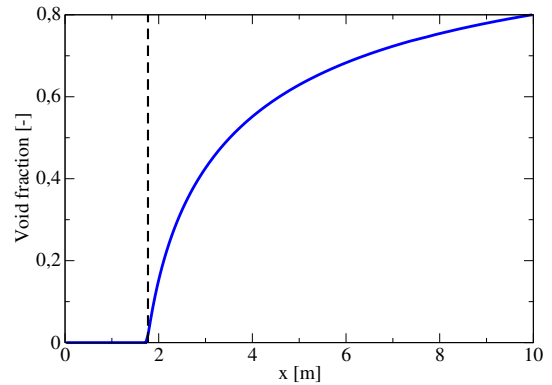


Figure 10: Saturated case for the boiling channel problem: void fraction profile at steady state.

subcooled, so it takes a finite distance for the liquid to reach saturation. A simple model from [45] is used for mass and energy exchanges between the two phases, which assumes that when $h_l < h_l^{\text{sat}}$ the heat flux applies only to the heating of the liquid and when $h_l \geq h_l^{\text{sat}}$ the heat flux leads only to evaporation, where h_l and h_l^{sat} are respectively the specific enthalpy of the liquid and the specific enthalpy of saturated liquid. Numerical results are illustrated in Figure 9 and Figure 10 (with 200 cells), with no particular problem observed at the boiling point (where vapor creation occurs) in the subcooled case. This boiling point can be obtained analytically by neglecting the internal energy and pressure variations along the channel [51, pp. 136], as indicated by the dashed line in Figure 10, which is well predicted by the numerical calculation. This test case demonstrates the ability of the numerical method to deal with phase appearance due to energy transfer.

Table 5: Initial states for boiling channel.

Physical quantity	Saturated case	Subcooled case
α_v [-]	10^{-5}	10^{-5}
P [Pa]	$68.73 \cdot 10^5$	$68.73 \cdot 10^5$
u_v [m/s]	0.78	0.78
u_l [m/s]	0.78	0.78
h_v [kJ/kg]	2774.	2774.
h_l [kJ/kg]	1260.	1220.

Tee-junction

Here, we consider two-phase flow in a T-junction composed of an inlet pipe, a run pipe and a branch pipe as illustrated in Figure 11. This test case, which is described in [17, 45], shows a dynamic

Table 6: Initial state for Tee-junction.

Physical quantity	Value
α_v [-]	0.45
P [Pa]	$150 \cdot 10^5$
u_v [m/s]	1.
u_l [m/s]	1.
h_v [kJ/kg]	2650.
h_l [kJ/kg]	1607.

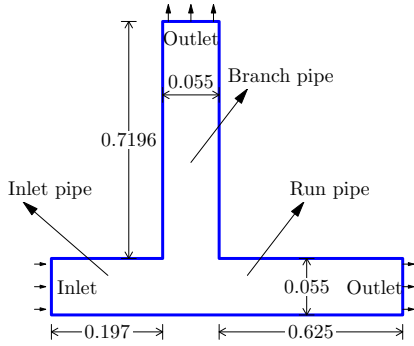
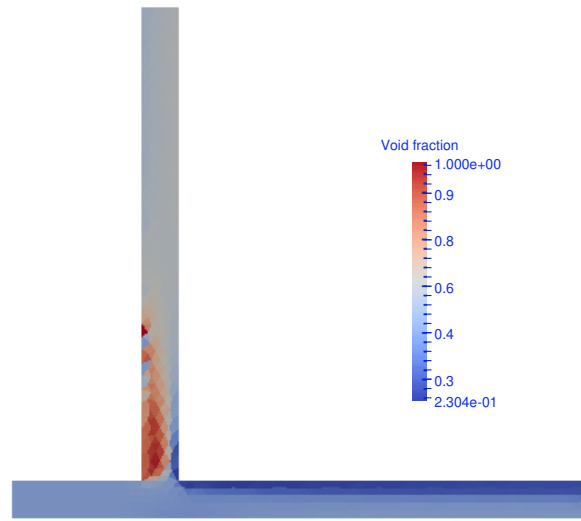


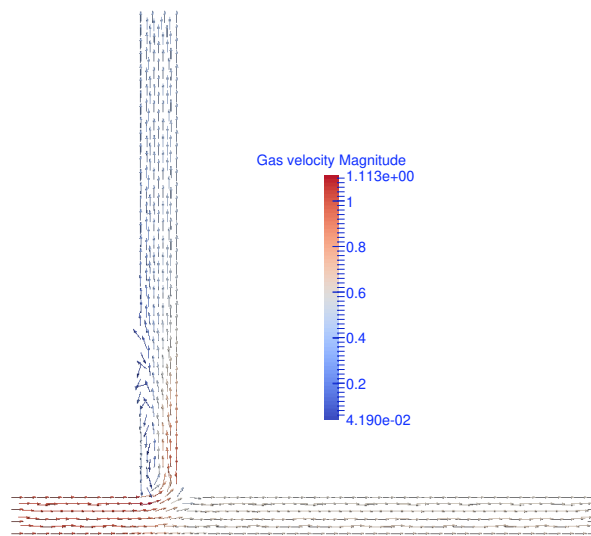
Figure 11: Geometry of the Tee-junction test case, dimensions in [m].

450 separation between liquid and gas caused by the geometry, as phase change is not taken into account in the calculation. The geometry of the test case is detailed in Figure 11, a mixture of gas and liquid enters the junction, and passes through both the run and branch pipes of the junction. The separation of the flow is controlled by imposing a constant pressure equal to the reference pressure (i.e., $150 \cdot 10^5$ Pa) at the outlet of the run pipe and a constant pressure that is lower than the reference pressure (i.e., 455 $149.998 \cdot 10^5$ Pa) at the outlet of the branch pipe. The pressure difference causes fluids to be pushed towards the branch pipe. The liquid is considered as incompressible with density $\rho_l = 10^3$ kg/m³, and the perfect gas equation of state with $\gamma_v = 1.4$ is used for the gas. The initial conditions are uniform in the domain (the velocity is in the axial direction of each tube), as presented in Table 6. The simulation is performed on an unstructured triangular mesh of 1005 cells with characteristic cell length equal to 460 0.02 m.

Due to the difference between liquid and gas density, and therefore the inertia, most of the liquid continues in the run pipe after the junction, whereas a large part of the gas enters the branch pipe, as shown in Figure 12a. In particular, the simulation predicts a recirculation of the flow at the inlet of the branch pipe, as shown in Figure 12b, and the gas is trapped therein. This test case shows the 465 capacity of the numerical method to treat a wide range of void fractions, and also behavior on an unstructured mesh. Since there is no experimental data for this test case, qualitative comparison can be made between the numerical results obtained here and those in [17, 45], where similar behavior has been observed.



(a) Void fraction.



(b) Gas velocity.

Figure 12: Tee Junction at steady state.

Water hammer

470 This test case consists of a large size tank with a pipeline connected at the bottom, as illustrated in
 Figure 13. Initially, water flows in the direction of the arrow with a uniform velocity. A sudden closure
 of the valve triggers a transient flow, and propagation of a pressure wave in the pipeline. Evaporation
 and condensation may be observed because of the pressure variation. Water hammer phenomena can
 damage nuclear installations, and thus cause problems in the primary and secondary cooling systems.
 475 Therefore, this test case is of interest in safety studies of nuclear installations [62].

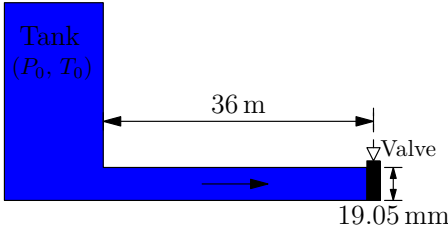


Figure 13: Illustration of water hammer problem.

Table 7: Initial states for water hammer.

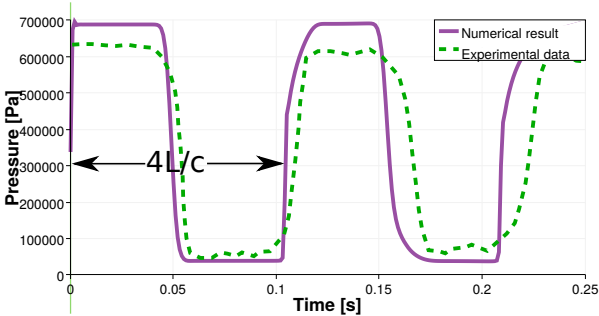
Physical quantity	Single-phase case	Two-phase case
α_v [-]	10^{-5}	10^{-5}
P [Pa]	$3.369 \cdot 10^5$	$10 \cdot 10^5$
u_v [m/s]	0.239	0.401
u_l [m/s]	0.239	0.401
h_v [kJ/kg]	2730.	2777.
h_l [kJ/kg]	100.5	688.8

As in [62], two cases are calculated with initial states given in Table 7: during the simulation of the single-phase case, pressure in the pipeline remains above the saturation pressure, and as a result no phase change occurs; by contrast in the two-phase case, the pressure tends to drop below the saturation pressure, so phase change is expected. Only the pipeline part of the installation is taken into consideration in the numerical simulation. A wall boundary condition is imposed at the valve, and the pressure and the enthalpies of the two phases are fixed (values taken from the initial states) at the junction between the tank and the pipeline. The numerical calculations are performed using a uniform mesh of 100 cells.

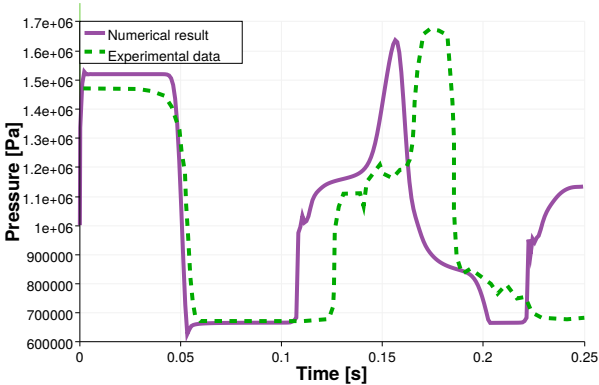
Figure 14a shows the time evolution of pressure calculated at the valve. When the valve is closed abruptly, the pressure at the valve increases immediately by $\Delta P = \rho c V_0$ [62], with ρ , c and V_0 being respectively liquid density, sound speed and initial liquid velocity. Thus, a pressure wave travels towards the tank with pressure equal to $P_0 + \Delta P$ and zero velocity fluid behind it. At time $t = L/c$ (where L is the length of the pipeline), the wave reaches the tank and is reflected back into the pipe, and the liquid velocity is reversed. This pressure wave travels back to the valve at $t = 2L/c$, and at that time the liquid velocity is equal to $-V_0$ in the whole domain. Afterwards, the pressure is reflected by the valve, and a sudden pressure drop can be observed at the valve, which corresponds to the pressure drop at about 0.05 s in Figure 14a. The physical behavior of the fluid after this can be analyzed in a similar manner, indeed a periodic phenomenon with a period of $4L/c$ for the pressure at the valve is present, as indicated in Figure 14a.

The pressure evolution at the valve for the two-phase flow case is illustrated in Figure 14b. In contrast to the single-phase case, at $t = 2L/c$, the pressure at the valve drops to P_{sat} (saturation pressure), which leads to evaporation, as shown in Figure 15. The peak at about $t = 0.16$ s is due to an interaction between the pressure wave and the vapor cavity created by the evaporation process. A detailed physical interpretation of the two cases can be found in [62]. The simplified exchange models presented in Section 2.2 are used to perform this test case, and the obtained numerical results agree

well with those found with the CATHARE code and from experiments [62]. The numerical results can be improved by taking into account the elasticity of the pipeline [62] and by using more elaborate exchange terms. This test shows that the numerical method is capable of computing the fast transients found in pressure wave propagation.



(a) Single-phase case.



(b) Two-phase case.

Figure 14: Water hammer problem: pressure at the valve vs time.

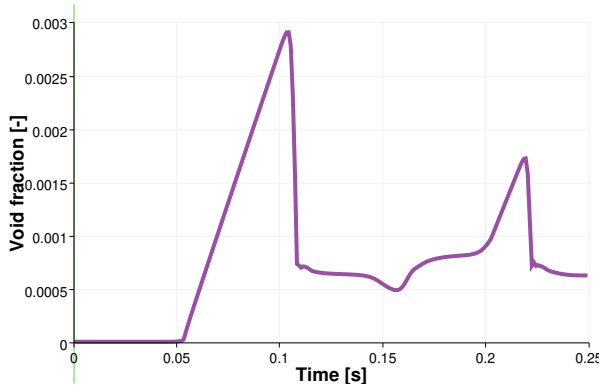


Figure 15: Water hammer problem: void fraction at the valve vs time for two-phase case.

This test case consists of a rapid depressurization of a horizontal tube by opening one end of the tube [54, 17]. It aims to simulate the behavior of the two-phase flow when a breach appears in the primary circuit in a pressurized water reactor (PWR). Prediction of the flow due to this breach is essential so that a suitable solution to deal with this problem can be considered [7]. The geometry for this test case is a 4.389-m long tube with an internal diameter of 0.1 m, as shown in Figure 16. Experimental data exists for pressure at the 6 measurement points P_i , $i = 1, \dots, 6$, and also for the void fraction at measurement point α , shown in Figure 16. The tube is initially filled with degassed water (initial void fraction equal to 10^{-5}) beneath the saturation temperature and at rest, at a pressure of 150 bar and a temperature of 300°C . One end of the tube opens in an estimated time period of 0.001 s and the pressure at this end drops from 150 bar to 1 bar (atmospheric pressure). As a result of this opening, rapid depressurization takes place throughout the tube, which results in almost total evaporation of the liquid (known as flashing) and the propagation of a pressure wave in the water-vapor mixture. Liquid and steam are ejected from the tube at very high velocities.

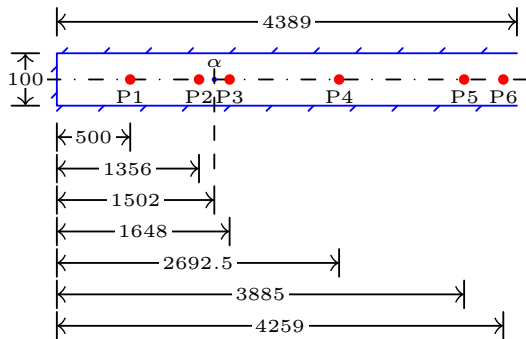


Figure 16: Geometry of fast depressurization test case, all dimensions in [mm].

For fluid in the metastable zones, the approach described in [54] is used to extrapolate the thermodynamic quantities. Figures 17 and 18 show a comparison between the numerical results (with 100 cells) and the experimental data at measurement points P_1 and P_2 . It can be noted that the numerical solution reproduces the physical phenomenon well, however the accuracy of the numerical results remains to be improved. Since we have used an extrapolation for fluids in metastable zones, the accuracy of the numerical results can be improved if use the actual equations of state in these zones. In addition, a better choice of drag force may also lead to results closer to the experimental data. Here we also compare the numerical solution with a reference solution obtained using the VFFC scheme [16]. The VFFC scheme is a density-based method, which calculates explicitly all the characteristic waves of the system. Indeed, the two solutions obtained using the same drag force (equation 7), and mass and energy transfer models (equations 9 and 10) agree well.

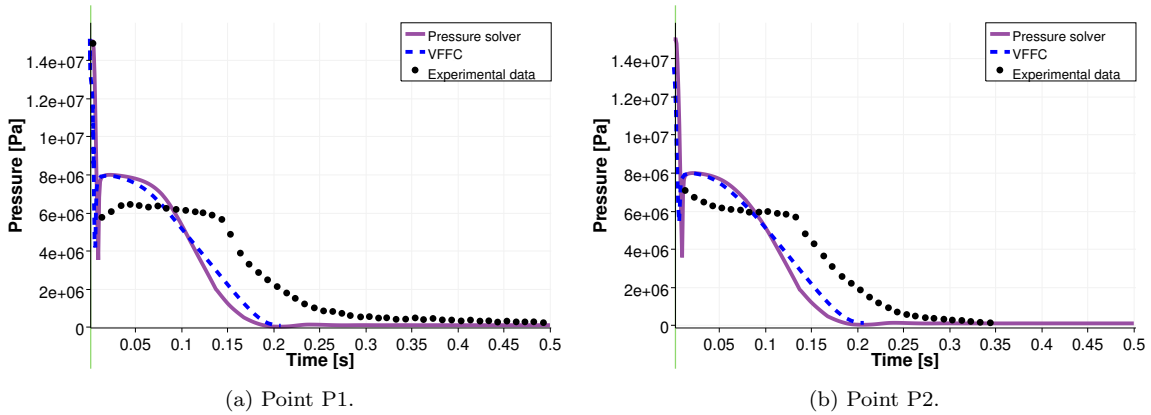


Figure 17: Fast depressurization: temporal evolution of the pressure at measurement points P_1 and P_2 .

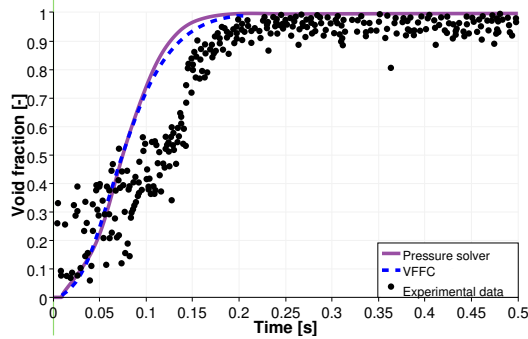


Figure 18: Fast depressurization: temporal evolution of the void fraction at measurement point α .

530 **5. Conclusion**

In this paper, a two-fluid six-equation model, with the addition of an interfacial pressure default term to make the model hyperbolic, is adopted for simulation of two-phase flows. The study extends the semi-implicit scheme originally used on structured grids with staggered variables to unstructured grids with collocated variables, in order to accommodate complex geometries. As in the nuclear component CUPID code, Rhie-Chow interpolation is used to calculate the velocity at a cell face in order to mitigate the checker-board problem. In the resulting pressure equations, terms accounting for the grid non-orthogonality are calculated explicitly using a deferred-correction method, which keeps the computational stencil small and allows accurate results to be obtained. The semi-implicit method is made conservative with a supplementary cost effective explicit correction step. Single-phase numerical results show that the conservative scheme is equally robust and able to deal with low Mach number flows as the original semi-implicit scheme, and more importantly, is capable of capturing exactly shock waves. Two-phase benchmark tests validate the numerical method for different physical

configurations and indicate that the method is able to handle a large range of void fractions, including phase appearance and disappearance. The numerical method is coupled with a tabulated EOS library
545 named Quicksteam in order to simulate industrial tests. Numerical results comparable to experimental data are obtained using a simplified model for mass and energy exchanges between the two phases.

Acknowledgements

The authors would like to thank Christophe Labourdette for providing the Quicksteam library and valuable discussions on the adaptation of the EOS library Quicksteam to our current studies and one
550 of the Referees for many comments that allowed us to improve the paper. This work was carried out during the Ph.D. studies of the first author under a doctoral contract with École Normale Supérieure Paris-Saclay.

Appendix A. Tabulated EOS

Tabulated EOS are required in order to predict accurately the dynamics of two-phase flows. Thus
555 based on the industrial formulation IAPWS [48], a water table named Quicksteam [47] was developed at the Centre de Mathématiques et de Leurs Applications (CMLA), ENS Paris-Saclay. In this section, the IAPWS formulation and its integration within our simulation code is explained.

The IAPWS formulation divides the temperature (T)-pressure (P) plane into five regions, as shown in Figure A.19, with the following range of validity:

$$\begin{aligned} 273.15 \text{ K} \leq T \leq 1073.15 \text{ K}, & \quad p \leq 100 \text{ MPa}, \\ 1073.15 \text{ K} \leq T \leq 2273.15 \text{ K}, & \quad p \leq 50 \text{ MPa}. \end{aligned}$$

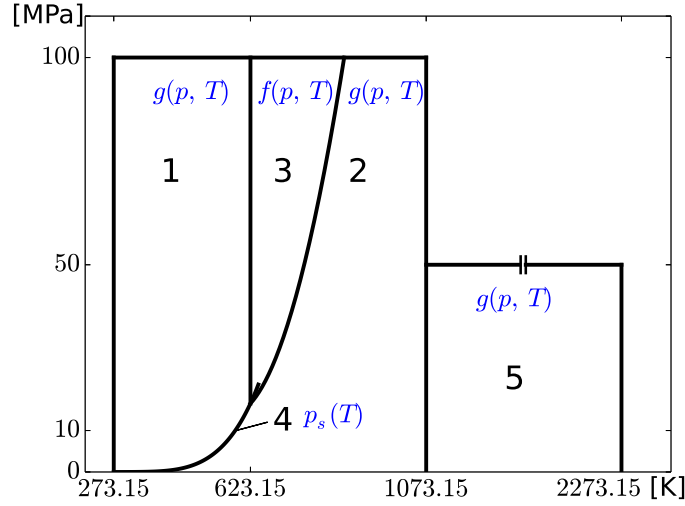


Figure A.19: Regions of IAPWS.

Region 3 concerns supercritical fluids, and region 5 is for high temperature applications. The regions of interest in our applications, as well as those used by Quicksteam [47], are region 1 (for liquid), region 2 for gas and region 4 for the saturation state. In regions 1 and 2, a polynomial function of temperature and pressure based on correlations of empirical data is used to correlate Gibbs free energy. For example, in region 1 the Gibbs free energy is expressed as

$$g(p, T) = RT \sum_{i=1}^{34} n_i \left(7.1 - \frac{p}{p^*} \right)^{I_i} \left(\frac{T^*}{T} - 1.222 \right)^{J_i}, \quad (\text{A.1})$$

where R is the specific gas constant of ordinary water, while the two reference constants p^* and T^* , the coefficients n_i and exponents I_i and J_i are calibrated according to experimental data [48]. A similar polynomial function exists for region 2.

Thus all thermodynamic properties can also be expressed as polynomial functions of temperature and pressure, which can be obtained analytically using the following classical relations based on the

specific Gibbs free energy (g):

$$\text{Specific volume:} \quad v = \left(\frac{\partial g}{\partial P} \right)_T, \quad (\text{A.2})$$

$$\text{Density:} \quad \rho = \frac{1}{v} = \left[\left(\frac{\partial g}{\partial P} \right)_T \right]^{-1}, \quad (\text{A.3})$$

$$\text{Specific enthalpy:} \quad h = g - T \left(\frac{\partial g}{\partial T} \right)_P, \quad (\text{A.4})$$

$$\text{Isobaric cubic expansion coefficient:} \quad \alpha_V = \frac{1}{v} \left(\frac{\partial v}{\partial T} \right)_P = \frac{1}{v} \left(\frac{\partial^2 g}{\partial T \partial P} \right), \quad (\text{A.5})$$

$$\text{Isothermal compressibility:} \quad k_T = -\frac{1}{v} \left(\frac{\partial v}{\partial P} \right)_T = -\frac{1}{v} \left(\frac{\partial^2 g}{\partial P^2} \right)_T, \quad (\text{A.6})$$

$$\text{Specific isobaric heat capacity:} \quad c_p = (\partial h / \partial T)_P = -T \left(\frac{\partial^2 g}{\partial T^2} \right)_P, \quad (\text{A.7})$$

Moreover, as mentioned in Section 3.1.2, the derivatives $(\frac{\partial \rho}{\partial h})_P$ and $(\frac{\partial \rho}{\partial P})_h$ are required in the discretized mass and energy equations of the numerical method. The independent variables in these two derivatives are the pressure P and the specific enthalpy h . However, only derivatives as function of (P, T) are directly available using the specific Gibbs free energy (g). Therefore, one needs to express these two derivatives $(\frac{\partial \rho}{\partial h})_P$ and $(\frac{\partial \rho}{\partial P})_h$ by other derivatives that are functions of (P, T) :

$$\left(\frac{\partial \rho}{\partial h} \right)_P = \frac{\left(\frac{\partial \rho}{\partial T} \right)_P}{\left(\frac{\partial h}{\partial T} \right)_P}, \quad (\text{A.8})$$

$$\left(\frac{\partial \rho}{\partial P} \right)_h = -\frac{\left(\frac{\partial \rho}{\partial T} \right)_P \left(\frac{\partial h}{\partial P} \right)_T - \left(\frac{\partial \rho}{\partial P} \right)_T \left(\frac{\partial h}{\partial T} \right)_P}{\left(\frac{\partial h}{\partial T} \right)_P}. \quad (\text{A.9})$$

The above relations can be obtained using the chain rule used for computing the derivative of the composition of two functions [51, pp. 195-196]. Finally, using the thermodynamic properties defined in (A.2)-(A.7) we obtain

$$\left(\frac{\partial \rho}{\partial h} \right)_P = -\frac{\alpha_V}{v c_p}, \quad (\text{A.10})$$

$$\left(\frac{\partial \rho}{\partial P} \right)_h = \frac{\alpha_V (1 - T \alpha_V) + \frac{k_T}{v} c_p}{c_p}. \quad (\text{A.11})$$

It is now explained how the Quicksteam water table is integrated into the simulation code. Two main issues need to be treated; firstly the choice of independent variables (P, h) in our code being different to (P, T) used in the IAPWS standard, and secondly the handling of fluids in a metastable state.

565 Handling of the liquid phase is shown here, while the treatment for the gas phase can be obtained similarly. The saturation temperature T_{sat} can first be calculated from the pressure using the equation $\mathcal{T}_{\text{sat}}(P)$ for region 4, and accordingly the saturation enthalpy $h_l^{\text{sat}}(P, T_{\text{sat}})$. Then, a determination of

whether the liquid is in a metastable state can be made by comparing the calculated enthalpy h_l to the saturation enthalpy h_l^{sat} . If the liquid is in a stable state, meaning that $h_l < h_l^{\text{sat}}$, the existing backward equation in IAPWS can be used directly to deduce the temperature T from (p, h_l) without using an iterative process. Once the temperature is obtained, the above-mentioned thermodynamic relations can be applied afterwards.

However, the backward equation is no longer valid if the liquid is in a metastable state ($h_l > h_l^{\text{sat}}$). An iterative process is therefore needed to determine the temperature from the pressure and the specific enthalpy by solving

$$f(T_l) = \mathcal{H}(P, T_l) - h_l = 0, \quad (\text{A.12})$$

where P , h_l and the function $\mathcal{H}(P, T_l)$ are given. The Newton-Raphson method applied to equation (A.12) gives

$$\begin{aligned} T_l^{k+1} &= T_l^k - \frac{f(T_l^k)}{f'(T_l^k)} \\ &= T_l^k - \frac{\mathcal{H}(P, T_l^k) - h_l}{c_p}, \end{aligned} \quad (\text{A.13})$$

which allows the liquid temperature to be obtained iteratively, and we can use $T_l^0 = T_{\text{sat}}$ to initiate the iterative process. In practice, for the test cases presented in this work using tabulated EOS, only two or three iterations are required to find a convergent solution for temperature with a relative difference of 10^{-3} between two successive iterations. Similar to the stable case, all other thermodynamic relations that are polynomial functions of pressure and temperature can be applied when the liquid temperature is known.

References

- [1] M. Ishii, T. Hibiki, Thermo-fluid dynamics of two-phase flow, Springer, 2011.
- [2] J.-M. Delhayé, Équations aux moyennes spatiales et aux moyennes statistiques des écoulements diphasiques, Communication présentée au Comité technique de la société Hydrotechnique de France.
- [3] D. A. Drew, S. L. Passman, Theory of multicomponent fluids, Springer, 1998.
- [4] H. B. Stewart, B. Wendroff, Two-phase flow: models and methods, Journal of Computational Physics 56 (3) (1984) 363–409. doi:[http://dx.doi.org/10.1016/0021-9991\(84\)90103-7](http://dx.doi.org/10.1016/0021-9991(84)90103-7).
- [5] F. De Vuyst, J.-M. Ghidaglia, G. Le Coq, On the numerical simulation of multiphase water flows with changes of phase and strong gradients using the homogeneous equilibrium model, International Journal of Finite Volumes 2 (1) (2005) 1–36.

- 590 [6] D. R. Liles, W. H. Reed, A semi-implicit method for two-phase fluid dynamics, *Journal of Computational Physics* 26 (3) (1978) 390–407. doi:[http://dx.doi.org/10.1016/0021-9991\(78\)90077-3](http://dx.doi.org/10.1016/0021-9991(78)90077-3).
- [7] H. Städtke, G. Franchello, B. Worth, U. Graf, P. Romstedt, A. Kumbaro, J. García-Cascales, H. Paillère, H. Deconinck, M. Ricchiuto, B. Smith, F. De Cachard, E. Toro, E. Romenski, 595 S. Mimouni, Advanced three-dimensional two-phase flow simulation tools for application to reactor safety (ASTAR), *Nuclear Engineering and Design* 235 (2-4) (2005) 379–400. doi:<http://dx.doi.org/10.1016/j.nucengdes.2004.08.052>.
- [8] R. F. Kunz, B. W. Siebert, W. Kevin Cope, N. F. Foster, S. P. Antal, S. M. Ettore, A coupled phasic exchange algorithm for three-dimensional multi-field analysis of heated flows with 600 mass transfer, *Computers & Fluids* 27 (7) (1998) 741–768. doi:[http://dx.doi.org/10.1016/S0045-7930\(97\)00064-9](http://dx.doi.org/10.1016/S0045-7930(97)00064-9).
- [9] J. J. Jeong, H. Y. Yoon, H. K. Cho, J. Kim, I. K. Park, A semi-implicit numerical scheme for a transient two-fluid three-field model on an unstructured grid, *International Communications in Heat and Mass Transfer* 35 (5) (2008) 597–605. doi:<http://dx.doi.org/10.1016/j.icheatmasstransfer.2008.01.008>. 605
- [10] S. Faure, J.-M. Ghidaglia, Violent flows in aqueous foams I: physical and numerical models, *European Journal of Mechanics - B/Fluids* 30 (4) (2011) 341–359. doi:<http://dx.doi.org/10.1016/j.euromechflu.2011.03.003>.
- [11] M. R. Baer, J. W. Nunziato, A two-phase mixture theory for the deflagration-to-detonation transition (ddt) in reactive granular materials, *International Journal of Multiphase Flow* 12 (6) (1986) 861–889. doi:[http://dx.doi.org/10.1016/0301-9322\(86\)90033-9](http://dx.doi.org/10.1016/0301-9322(86)90033-9). 610
- [12] R. Saurel, R. Abgrall, A multiphase Godunov method for compressible multiuid and multiphase flows, *Journal of Computational Physics* 150 (2) (1999) 425–467. doi:<http://dx.doi.org/10.1006/jcph.1999.6187>.
- 615 [13] R. A. Berry, J. W. Peterson, H. Zhang, R. C. Martineau, H. Zhao, L. Zou, D. Andrs, RELAP-7 Theory Manual (2014).
- [14] F. H. Harlow, A. A. Amsden, A numerical fluid dynamics calculation method for all flow speeds, *Journal of Computational Physics* 8 (2) (1971) 197–213. doi:[http://dx.doi.org/10.1016/0021-9991\(71\)90002-7](http://dx.doi.org/10.1016/0021-9991(71)90002-7).

- 620 [15] I. Toumi, A. Kumbaro, An approximate linearized Riemann solver for a two-fluid model, *Journal of Computational Physics* 124 (2) (1996) 286–300. doi:<http://dx.doi.org/10.1006/jcph.1996.0060>.
- [16] J.-M. Ghidaglia, A. Kumbaro, G. L. Coq, On the numerical solution to two fluid models via a cell centered finite volume method, *European Journal of Mechanics - B/Fluids* 20 (6) (2001) 841–867. doi:[http://dx.doi.org/10.1016/S0997-7546\(01\)01150-5](http://dx.doi.org/10.1016/S0997-7546(01)01150-5).
625
- [17] A. Kumbaro, Simplified eigenstructure decomposition solver for the simulation of two-phase flow systems, *Computers & Fluids* 64 (2012) 19–33. doi:<http://dx.doi.org/10.1016/j.compfluid.2012.04.025>.
- [18] F. Moukalled, M. Darwish, B. Sekar, A pressure-based algorithm for multi-phase flow at all speeds, *Journal of Computational Physics* 190 (2) (2003) 550–571. doi:[http://dx.doi.org/10.1016/S0021-9991\(03\)00297-3](http://dx.doi.org/10.1016/S0021-9991(03)00297-3).
630
- [19] F. H. Harlow, J. E. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface, *The Physics of Fluids* 8 (12) (1965) 2182–2189. doi:<http://dx.doi.org/10.1063/1.1761178>.
- 635 [20] A. V. Patankar, D. B. Spalding, A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows, *International Journal of Heat and Mass Transfer* 15 (10) (1972) 1787–1806. doi:[http://dx.doi.org/10.1016/0017-9310\(72\)90054-3](http://dx.doi.org/10.1016/0017-9310(72)90054-3).
- [21] K. C. Karki, S. V. Patankar, Pressure based calculation procedure for viscous flows at all speeds in arbitrary configurations, *AIAA Journal* 27 (9) (1989) 1167–1174. doi:<http://dx.doi.org/10.2514/3.10242>.
640
- [22] RELAP5/MOD3.3 Code Manual Volume I: Code Structure, System Models, and Solution Methods (2001).
- [23] M. J. Thurgood, J. M. Kelly, T. E. Guidotti, R. J. Kohrt, K. R. Crowell, COBRA/TRAC - A thermal-hydraulics code for transient analysis of nuclear reactor vessels and primary coolant systems (1983).
645
- [24] F. Barre, M. Parent, B. Brun, Advanced numerical methods for thermalhydraulics, *Nuclear Engineering and Design* 145 (1-2) (1993) 147–158. doi:[http://dx.doi.org/10.1016/0029-5493\(93\)90064-G](http://dx.doi.org/10.1016/0029-5493(93)90064-G).

- [25] J. A. Trapp, R. A. Riemke, A nearly-implicit hydrodynamic numerical scheme for two-phase flows, *Journal of Computational Physics* 66 (1) (1986) 62–82. doi:[http://dx.doi.org/10.1016/0021-9991\(86\)90054-9](http://dx.doi.org/10.1016/0021-9991(86)90054-9).
650
- [26] J. H. Mahaffy, A stability-enhancing two-step method for fluid flow calculations, *Journal of Computational Physics* 46 (3) (1982) 329–341. doi:[http://dx.doi.org/10.1016/0021-9991\(82\)90019-5](http://dx.doi.org/10.1016/0021-9991(82)90019-5).
- [27] M. A. Holmes, Stability of finite difference approximation of two fluid, two phase flow equations, Ph.D. thesis, North Carolina State University (1995).
655
- [28] D. Bestion, System code models and capabilities, in: THICKET, 2008.
- [29] J. H. Ferziger, M. Peric, Computational methods for fluid dynamics, Springer, 2002.
- [30] M. Peric, R. Kessler, G. Scheuerer, Comparison of finite-volume numerical methods with staggered and colocated grids, *Computers & Fluids* 16 (4) (1988) 389–403. doi:[http://dx.doi.org/10.1016/0045-7930\(88\)90024-2](http://dx.doi.org/10.1016/0045-7930(88)90024-2).
660
- [31] C. M. Rhie, W. L. Chow, Numerical study of the turbulent flow past an airfoil with trailing edge separation, *AIAA Journal* 21 (11) (1983) 1525–1532. doi:<http://dx.doi.org/10.2514/3.8284>.
- [32] S. R. Mathur, J. Y. Murthy, A pressure-based method for unstructured meshes, *Numerical Heat Transfer, Part B: Fundamentals* 31 (2) (1997) 195–215. doi:<http://dx.doi.org/10.1080/10407799708915105>.
665
- [33] S. Muzaferija, D. Gosman, Finite-volume CFD procedure and adaptive error control strategy for grids of arbitrary topology, *Journal of Computational Physics* 138 (2) (1997) 766–787. doi:<http://dx.doi.org/10.1006/jcph.1997.5853>.
- [34] S. K. Choi, Note on the use of momentum interpolation method for unsteady flows, *Numerical Heat Transfer, Part A: Applications* 36 (5) (1999) 545–550. doi:<http://dx.doi.org/10.1080/104077899274679>.
670
- [35] J. Mencinger, I. Zun, On the finite volume discretization of discontinuous body force field on collocated grid: application to VOF method, *Journal of Computational Physics* 221 (2) (2007) 524–538. doi:<http://dx.doi.org/10.1016/j.jcp.2006.06.021>.
675
- [36] S. Zhang, X. Zhao, S. Bayyuk, Generalized formulations for the Rhie-Chow interpolation, *Journal of Computational Physics* 258 (2014) 880–914. doi:<http://dx.doi.org/10.1016/j.jcp.2013.11.006>.

- [37] I. Demirdžić, S. Muzaferija, Numerical method for coupled fluid flow, heat transfer and stress analysis using unstructured moving meshes with cells of arbitrary topology, *Computer Methods in Applied Mechanics and Engineering* 125 (1-4) (1995) 235–255. doi:[http://dx.doi.org/10.1016/0045-7825\(95\)00800-G](http://dx.doi.org/10.1016/0045-7825(95)00800-G).
- [38] F. Denner, B. G. M. van Wachem, Fully-coupled balanced-force VOF framework for arbitrary meshes with least-squares curvature evaluation from volume fractions, *Numerical Heat Transfer, Part B: Fundamentals* 65 (3) (2014) 218–255. doi:<http://dx.doi.org/10.1080/10407790.2013.849996>.
- [39] I. Demirdžić, On the discretization of the diffusion term in finite-volume continuum mechanics, *Numerical Heat Transfer, Part B: Fundamentals* 68 (1) (2015) 1–10. doi:<http://dx.doi.org/10.1080/10407790.2014.985992>.
- [40] C.-N. Xiao, F. Denner, B. G. M. van Wachem, Fully-coupled pressure-based finite-volume framework for the simulation of fluid flows at all speeds in complex geometries, *Journal of Computational Physics* 346 (2017) 91–130. doi:<http://dx.doi.org/10.1016/j.jcp.2017.06.009>.
- [41] J. J. Jeong, H. Y. Yoon, I. K. Park, H. K. Cho, H. D. Lee, Development and preliminary assessment of a three-dimensional thermal hydraulics code, CUPID, *Nuclear Engineering and Technology* 42 (3) (2010) 279–296. doi:<http://dx.doi.org/10.5516/NET.2010.42.3.279>.
- [42] R. J. Leveque, *Finite volume methods for hyperbolic problems*, Cambridge University Press, 2002.
- [43] D. R. van der Heul, C. Vuik, P. Wesseling, A conservative pressure-correction method for flow at all speeds, *Computers & Fluids* 32 (8) (2003) 1113–1132. doi:[http://dx.doi.org/10.1016/S0045-7930\(02\)00086-5](http://dx.doi.org/10.1016/S0045-7930(02)00086-5).
- [44] Z. Fu, F. Aydogan, R. J. Wagner, More conservative governing equations in RELAP5: Derivation of equations, *Annals of Nuclear Energy* 85 (2015) 523–531. doi:<http://dx.doi.org/10.1016/j.anucene.2015.05.034>.
- [45] F. Cordier, P. Degond, A. Kumbaro, Phase appearance or disappearance in two-phase flows, *Journal of Scientific Computing* 58 (1) (2014) 115–148. doi:<http://dx.doi.org/10.1007/s10915-013-9725-9>.
- [46] D. Bestion, The phase appearance and disappearance in the CATHARE code, in: *International Workshop on Trends in Numerical and Physical Modeling for Industrial Multiphase Flows*, Cargèse, France, 2000.

- [47] C. Labourdette, J.-M. Ghidaglia, J. A. Redford, S. Faure, Accurate state variables for fluid flow simulation using Quicksteam and Quickmethane, *European Journal of Mechanics - B/Fluids* 65 (2017) 132–140. doi:<http://dx.doi.org/10.1016/j.euromechflu.2017.03.003>.
- [48] W. Wagner, J. R. Cooper, A. Dittmann, J. Kijima, H.-J. Kretzschmar, A. Kruse, R. Mares, K. Oguchi, H. Sato, I. Stocker, O. Sifner, Y. Takaishi, I. Tanishita, J. Trubenbach, T. Willkommen, The IAPWS industrial formulation 1997 for the thermodynamic properties of water and steam, *Journal of Engineering for Gas Turbines and Power* 122 (1) (2000) 150–184. doi:<http://dx.doi.org/10.1115/1.483186>.
- [49] D. Bestion, The physical closure laws in the CATHARE code, *Nuclear Engineering and Design* 124 (3) (1990) 229–245. doi:[http://dx.doi.org/10.1016/0029-5493\(90\)90294-8](http://dx.doi.org/10.1016/0029-5493(90)90294-8).
- [50] H. Städtke, *Gas dynamic aspects of two-phase flow*, Wiley-VCH, 2006.
- [51] L. Zhang, Modélisation, analyse et simulation d’écoulements en thermohydraulique par modèles 6 équations, Ph.D. thesis, ENS Paris-Saclay (2017).
- [52] V. H. Ransom, D. L. Hicks, Hyperbolic two-pressure models for two-phase flow, *Journal of Computational Physics* 53 (1984) 124–151.
- [53] D. Drew, L. Cheng, J. R. T. Lahey, The analysis of virtual mass effects in two-phase flow, *International Journal of Multiphase Flow* 5 (4) (1979) 233–242. doi:[http://dx.doi.org/10.1016/0301-9322\(79\)90023-5](http://dx.doi.org/10.1016/0301-9322(79)90023-5).
- [54] J.-M. Rovarch, *Solveurs tridimensionnels pour les écoulements diphasiques avec transfert d’énergie*, Ph.D. thesis, ENS de Cachan (2006).
- [55] I. K. Park, H. K. Cho, H. Y. Yoon, J. J. Jeong, Numerical effects of the semi-conservative form of momentum equations for multi-dimensional two-phase flows, *Nuclear Engineering and Design* 239 (11) (2009) 2365–2371. doi:<http://dx.doi.org/10.1016/j.nucengdes.2009.06.011>.
- [56] B. Larroutourou, How to preserve the mass fractions positivity when computing compressible multi-component flows, *Journal of Computational Physics* 95 (1) (1991) 59–84. doi:[http://dx.doi.org/10.1016/0021-9991\(91\)90253-H](http://dx.doi.org/10.1016/0021-9991(91)90253-H).
- [57] F. Moukalled, L. Mangani, M. Darwish, *The finite volume method in computational fluid dynamics – an advanced introduction with OpenFOAM and Matlab*, Springer, 2016.
- [58] M. A. Christon, The consistency of pressure-gradient approximations used in multi-dimensional shock hydrodynamics, *International Journal for Numerical Methods in Fluids* 64 (1) (2010) 71–97. doi:<http://dx.doi.org/10.1002/flid.2143>.

- 740 [59] M. Hirano, Al., Application of MINCS code to numerical benchmark problems for two-fluid model, Japanese J. Multiphase Flow 1 (1) (1987) 79–95. doi:<http://dx.doi.org/10.3811/jjmf.1.79>.
- [60] F. Cordier, P. Degond, A. Kumbaro, An asymptotic-preserving all-speed scheme for the Euler and Navier-Stokes equations, Journal of Computational Physics 231 (17) (2012) 5685–5704. doi:<http://dx.doi.org/10.1016/j.jcp.2012.04.025>.
- 745 [61] S. Clerc, Numerical simulation of the homogeneous equilibrium model for two-phase flows, Journal of Computational Physics 161 (1) (2000) 354–375. doi:<http://dx.doi.org/10.1006/jcph.2000.6515>.
- [62] G. Serre, D. Bestion, Two-phase water-hammer simulation with the CATHARE code, in: 9th International Conference on Nuclear Engineering, France, 2001.