European Journal of Mechanics B/Fluids 28 (2009) 475–485

Contents lists available at ScienceDirect

European Journal of Mechanics B/Fluids

www.elsevier.com/locate/ejmflu

A totally Eulerian finite volume solver for multi-material fluid flows

J.-P. Braeunig [a], B. Desjardins [b], J.-M. Ghidaglia [c,*]

[a] CEA/DIF Bruyères-le-Châtel, 91297 Arpajon Cedex, France
[b] ENS Paris, D.M.A., 45 rue d'Ulm, 75230 Paris cedex 05, France
[c] CMLA, ENS Cachan and CNRS, UniverSud and LRC MESO, ENS Cachan and CEA/DIF, 61 avenue du Président Wilson, F-94235 Cachan Cedex

ARTICLE INFO

Article history:
Received 28 April 2008
Received in revised form 19 March 2009
Accepted 19 March 2009
Available online 26 March 2009

Keywords:
Compressible hydrodynamics
Finite volume method
Pure Eulerian method
Interface capturing
Sliding
Euler equation

ABSTRACT

The purpose of this work is to present a new numerical scheme for multi-material fluid flows in dimension $d \geqslant 1$. It is a totally Eulerian conservative scheme that allows to compute sharp interfaces between non-miscible fluids. The underlying flux scheme in single material cells is the so-called FVCF scheme, whereas interface reconstruction and directional splitting is used in multi-material cells. One of the novelty of our approach is the introduction of the concept of "condensate" which allows to handle mixed cells containing two or more materials. Moreover, it has been designed to allow free sliding of materials on each others, thanks to a material volume centered computation of variables in mixed cells.

1. Introduction

The numerical simulation of fluid material interfaces encompasses a wide range of numerical methods, depending on the various physical situations, in particular the relevant space and time scales involved. A diffuse interface, with mixing of materials at the molecular scale, can be treated for instance with diffusion Anderson et al. [2]. If the mixing is driven by hydrodynamic instabilities in the fully turbulent regime, then it may be treated with turbulence models in a statistical approach as described for instance by Launder and Spalding in [12]. In the case when the diffusion scale between materials can be neglected with respect to macroscopic hydrodynamic structures, then interface motion may be represented with sharp interfaces as introduced for instance by Noh and Woodward [14] or Youngs [18] for compressible flows, or Zaleski et al. [11] for incompressible flows. In this case, contact properties between materials may be modelled either by exact sliding, no slip condition, or friction.

The physical assumptions of this work are the following: the multi-material fluid flow is assumed to be compressible, laminar, subject to large and transient deformations. The fluid model addressed here is the compressible Euler equations, in a flow regime such that molecular viscosity within materials is neglected: materials are considered as immiscible and separated by sharp interfaces, with perfect sliding of one material on each others. Each material is characterized by its own equation of state. For instance, this set of assumptions is well adapted to simulate the interaction between a shock wave propagating in the air and a water droplet, as described in Takayama [10].

The compressible Euler equations in dimension of space d can be written in a conservative form as follows:

$$\begin{cases} \partial_t \rho + \operatorname{div}(\rho u) = 0, \\ \partial_t (\rho u) + \operatorname{div}(\rho u \otimes u) + \nabla p = 0, \\ \partial_t (\rho E) + \operatorname{div}\big((\rho E + p)u\big) = 0, \end{cases} \tag{1}$$

where ρ denotes the density, $u \in \mathbb{R}^d$ the velocity field, e the specific internal energy, p the pressure, and $E = e + |u|^2/2$ the specific total energy. An equation of state of the form $EOS(\rho, e, p) = 0$ is provided in order to close the system.

Multi-material fluid flows computation may be treated with many different numerical strategies. Lagrangian methods are very natural to capture interface motion and contact between materials. Material volumes are meshed and nodes at the boundary of each material represent the interfaces. The Lagrangian evolution of these nodes naturally defines the interface motion. One can consider each meshed material as one independent block, interacting with the others through their boundaries. Most of Lagrangian schemes preserve mass conservation on each block, therefore on each material. Conservations of momentum and total energy are seldom ensured in a strict sense whenever staggered grids for velocity and pressure are used. Unfortunately, accuracy and robustness of Lagrangian schemes are limited by large mesh distortions, for instance in the case of vortical flows as it is shown by Aymard et al. in [3].

* Corresponding author.
E-mail address: ejmb@ghidaglia.net (J.-M. Ghidaglia).

0997-7546/$ – see front matter
doi:10.1016/j.euromechflu.2009.03.003

Eulerian methods can be very accurate on shock or rarefaction waves with high space discretization order. A great variety of schemes for single fluid flow computation exists among three main families: finite element methods, finite differences, and finite volume methods. The finite element method can be very accurate by using high order base functions, but it is in general difficult to handle complex physics, strong shock waves and it does not ensure generally local conservation of variables for multi-material fluid flows. Moreover upwinding, which turns out to be a key tool for the simulation of flows dominated by convection, is rather difficult to achieve. The formalism of finite volume methods is closer to the mechanical viewpoint, very generic for different types of physical applications. Thus it might be easier to add physical models as surface tension or turbulent diffusion for instance. The discretization order is limited, but this method is very accurate for hydrodynamic shock waves, because of the similarity between numerical treatment and mechanics.

The extension of Eulerian schemes to multi-material fluid flows can be obtained by various techniques. One is to introduce the mass fraction c_α of material α and to let it evolve according to material velocity. The cell is called a pure cell if a material α satisfies $c_\alpha = 1$ and is called a mixed cell if $c_\alpha \in]0,1[$. Pure cells filled by material α are calculated in the same manner as for the single fluid method. Mixed cells evolution is computed using a mixing equation of state that takes into account material mass fractions. The drawback here is the interface numerical diffusion, which prevents sharp interface capturing. However, very accurate methods exist that limit this diffusion, see for instance Després and Lagoutière [5]. In another family of methods, called *Level Set Methods* [17], a signed distance function ϕ is defined instead of mass fractions, advected by the material velocity. The materials position is determined according to the sign of this distance function. For instance, negative values are associated with material 1, positive values to material 2 and isovalue 0 corresponds to the interface position in the domain. Mixed cells are defined by the set where the function ϕ vanishes. This method gives smooth curves of a sharp interface between materials, dealing with complex or singular geometry. Nevertheless, the interface is sharp, but not the quantities that are averaged in the mixed cells to write scheme fluxes, for instance. Thus variables conservation is not guaranteed without specific corrections and spurious oscillations may appear, as discussed in [1] and [16].

Quantities sharpness and conservation at interfaces may be obtained using a subgrid interface reconstruction. In mixed cells, the interface is approximated by straight lines by most authors, but sometimes by more complex curves separating materials, or more complex theory by Shashkov et al. [6] for instance. A famous method using sharp interface reconstruction is the Lagrange-Remap Finite Volume scheme, developed by Noh and Woodward [14] and improved by D.L. Youngs [18], belonging to the family of so-called Volume of Fluid (VOF) methods. The first step is Lagrangian, which means that mesh nodes move with material velocity. The second one is a remapping of Lagrangian cells onto the original Eulerian mesh, by exchanging volume fluxes between cells corresponding to the Lagrangian motion of cell edges. Interface location in a mixed cell is determined by partial volumes of the materials and by the interface normal vector calculated using volume fractions in neighboring cells. Thus the ratio of each material in volume fluxes is deduced from geometrical considerations. Some methods with the same kind of operator splitting are used for incompressible multi-material fluid flows as for instance in Zaleski et al. [11]. These methods provide sharp interfaces for materials and discontinuous quantities in mixed cells, allowing large deformations and transient flows. In this context, the drawback of these methods is the limited accuracy of the underlying single phase scheme due to diffusion occurring in the remap step. Moreover, extended physics at material interfaces such as sliding effects, is not possible.

The FVCF method (Finite Volume with Characteristic Flux) has been introduced by Ghidaglia et al. [7,8] for simulating single phase compressible flows. The method described in this paper, so called NIP method (Natural Interface Positioning), is an extension to multi-material fluid flows of the FVCF method. It is a cell centered totally Eulerian scheme, in which material interfaces are represented by a discontinuous piecewise linear curve. An original treatment for interface evolution is proposed on Cartesian structured meshes which is locally conservative in mass, momentum and total energy and allow the materials to slip on each others. Discrete conservation laws are written on partial volumes as well as on pure cells, considering the interface in the cell as a moving boundary with zero diffusion between materials. A specific data structure named "condensate" is introduced in order to write the finite volume scheme even when the considered volume is made of moving boundaries, i.e. interfaces. This treatment includes an explicit computation of pressure and velocity at interfaces.

Section 2 begins with some definitions used in this work and the driving ideas motivating this original method. Then the "condensate" definition is given and its construction is described, first in $1D$ and afterwards in higher dimensions. The way we write the scheme in "condensates" is then described, followed with some remarks about pressure evolution in mixed cells and a way to control it. In Section 3 are shown $2D$ results illustrating the capability of the method to deal with perfect sliding, high pressure ratios and high density ratios.

2. The FVCF-NIP method

Let us define pure and mixed cells as follows: a cell C of volume Vol_C may contain n_m materials, each of them filling a partial volume Vol_C^k with

$$\sum_{k=1}^{n_m} Vol_C^k = Vol_C.$$

The system of conservation laws (1) can be written in a generic conservative form: let $V = (\rho, \rho u, \rho E)^t$ be the unknown vector of conservative variables and the flux F be a matrix valued function defined as:

$$F\colon \mathbb{R}^{d+2} \longrightarrow \mathbb{R}^{d+2} \times \mathbb{R}^d$$

$$V \longmapsto F(V).$$

For all direction $n \in \mathbb{R}^d$, $F(V)\cdot n$ is given in terms of V by:

$$F(V)\cdot n = \big(\rho(u\cdot n), \rho u(u\cdot n) + pn, (\rho E + p)(u\cdot n)\big). \tag{2}$$

The compressible Euler equations (1) can then be rewritten as follows:

$$\partial_t V + \operatorname{div} F(V) = 0. \tag{3}$$

A centered variable vector $V_k = (\rho_k, \rho_k u_k, \rho_k E_k)^t$ and an equation of state $EOS_k(\rho_k, e_k, p_k) = 0$ are also associated with each material $k \leqslant n_m$. We shall say that the cell C is a *pure cell* if $n_m = 1$, and a *mixed cell* otherwise.

The main idea is to let interfaces evolve through a directional splitting scheme, without loosing the accuracy of the fully Eulerian scheme in the bulk of materials. Of course, this scheme is restricted to structured Cartesian meshes. The NIP interface capturing method uses $2D/3D$ information of partial volumes in each direction, allowing for material sliding. The method preserves local conservation of each component of the variable vector $V = (\rho, \rho u, \rho E)^t$ by writing a conservative scheme of these variables

even on partial volumes. Considering a conservation equation of the form:

$$\partial_t v + \partial_x f = 0,$$

the conservative form for a scheme is defined here in $1D$ as follows:

$$\frac{v_i^{n+1} - v_i^n}{dt} + \frac{f_{i,i+1}^{n,n+1} - f_{i-1,i}^{n,n+1}}{\Delta x} = 0,$$

where $dt = t^{n+1} - t^n$ is the time step, Δx the space step, v_i^n variable v value in cell i a time t^n, $f_{i,i+1}^{n,n+1}$ the flux of variable v from cell i to cell $i+1$ between time t^n and t^{n+1}.

Conservative form of the scheme is not only necessary to ensure conservation of variables at the discrete level, but it is also necessary to compute the right weak solution of the partial differential equations system when discontinuities are propagating in the flow. In particular, velocity of these waves cannot properly be computed when using non-conservative schemes, see LeVeque [13].

2.1. Single fluid scheme

The interface capturing method NIP uses a directional splitting on Cartesian structured mesh. The method is thus detailed for only one generic direction denoted by x. In d dimensions of space, the algorithm described in direction x has to be replicated d times, one for each direction. However, this directional splitting does not modify at all the underlying single fluid scheme FVCF for pure cells. In $2D$:

- variables and interface positions at t^{nx} are calculated from those at t^n by the x direction step,
- variables and interfaces positions at t^{n+1} are calculated from those at t^{nx} by the y direction step.

$$\begin{aligned} Vol_i \frac{V_i^{nx} - V_i^n}{dt} + A_x\left(\phi_\ell^n + \phi_r^n\right) &= 0, \\ Vol_i \frac{V_i^{n+1} - V_i^{nx}}{dt} + A_y\left(\phi_d^n + \phi_u^n\right) &= 0, \end{aligned} \tag{4}$$

with time step dt, the cell volume Vol_i, the cell face area A_x and A_y respectively normal to x and y directions, up, down, right and left direction fluxes ϕ_u^n, ϕ_d^n, ϕ_r^n, ϕ_ℓ^n calculated with respect of the outgoing normal direction n_d of cell face Γ_d in direction d using variables at time t^n, i.e.

$$\phi_d^n = \frac{1}{A_d} \int_{\Gamma_d} \left(F(V^n) \cdot n_d\right) dS.$$

Of course, results obtained for V^{n+1} by using two steps (4) and by adding fluxes in all directions in one step are strictly identical, when fluxes are calculated using variables at time t^n.

2.2. Multi-fluid scheme

The multi-material extension proposed here is to write the finite volume scheme (4) on each partial volume in a mixed cell, in order to obtain a conservative scheme. Two difficulties appear when dealing with partial volumes:

- the above scheme (4) is constrained with the same CFL condition as the single fluid scheme FVCF:

  $$dt < \min_i \left(\frac{Vol_i}{A(\|u_i\| + c_i)} \right)$$

  with u_i the velocity vector and c_i the sound speed in cell C_i. Here, control volumes Vol_i are the partial volumes in mixed cells that may be arbitrarily small, and then lead to tiny time steps;
- when an interface moves from a cell C_1 to a neighboring cell C_2, the scheme cannot directly be written on each material volume, because one material is disappearing in cell C_1 and appearing in cell C_2.

These difficulties are artificially induced by the interface motion in an Eulerian mesh. As an example, let us consider a system composed of two identical materials, same variable values and same equations of state: the interface is obviously artificial. We want this two material system to have the same behavior as the identical one material system, in particular no restriction on the time step due to mixed cells.

Remark 1. In this method, the time step is calculated only considering pure cells. We do not want small volumes in mixed cells to slowdown dramatically the computation.

Our method consists in removing cell edges when the cell contains an interface. Therefore each partial volume is merged with

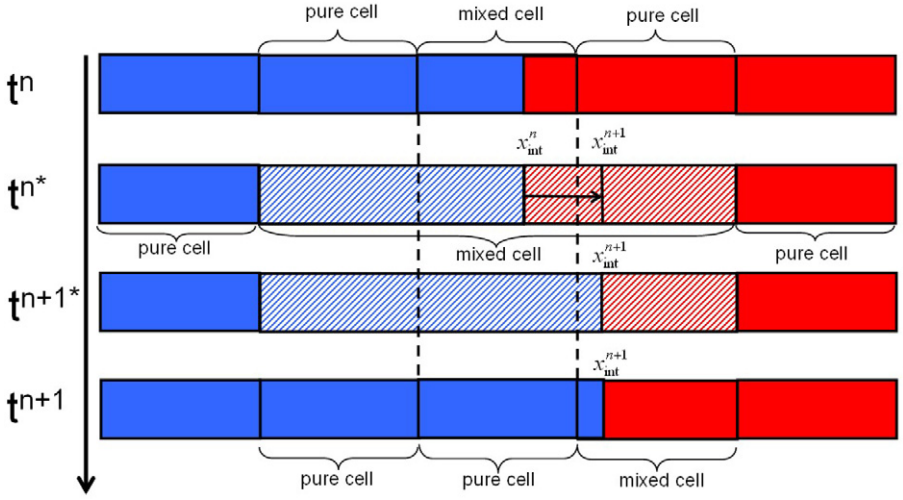

Fig. 1. Evolution of an interface through a cell face.

the neighboring pure cells filled with the same material, see Fig. 1. Variables in these enlarged partial volumes are obtained by writing the conservation laws:

$$\begin{aligned}
\overline{Vol_1} &= Vol_1 + Vol_{pure\ 1},\\
\overline{Vol_2} &= Vol_2 + Vol_{pure\ 2},\\
\overline{V}_1 &= \frac{Vol_1 V_1 + Vol_{pure\ 1} V_{pure\ 1}}{\overline{Vol_1}},\\
\overline{V}_2 &= \frac{Vol_2 V_2 + Vol_{pure\ 2} V_{pure\ 2}}{\overline{Vol_2}}.
\end{aligned} \tag{5}$$

The $1D$ three cell system made of one mixed cell between two pure cells, is associated with its left and right single fluid fluxes ϕ_ℓ and ϕ_r. Internal cell edges are forgotten, considering only enlarged volumes $\overline{Vol_1}$ and $\overline{Vol_2}$ and averaged variables $\overline{V}_1$ and $\overline{V}_2$, separated by an interface. This system is a data structure called condensate that will be defined more generally in Section 2.2.1.

The CFL condition is then taken as its value calculated on pure cells, because waves that were evolving in mixed cells are diffused by the conservative average in the *condensate*. We assume that velocity values stay close to those in pure cells. Thus interfaces should not intersect a cell edge during the time step, thanks to the CFL constraint on time step in pure cells. The new values of condensate volumes and variables are then remapped on the Eulerian mesh. The interface might have moved from one cell to another during the *condensate* evolution.

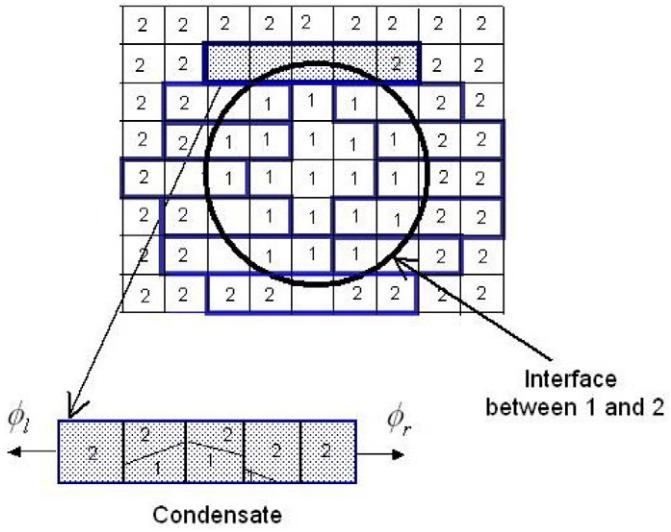

Fig. 2. Extraction of neighboring mixed cells from the grid to become a *condensate* during x direction step.

2.2.1. Definition of a condensate

The multi-material treatment in $2D/3D$ with interface reconstruction on an Eulerian mesh requires to take into account three main constraints:

- to write conservation laws in a robust way without any restriction on the time step from mixed cells,
- to allow interface motion from one cell to another,
- to allow two or more neighboring mixed cells.

The method described here introduces a $1D$ data structure named *condensate* (see Fig. 2).

Definition. A condensate is a $1D$ data structure constituted of nc layers of different materials, separated by nc-1 Lagrangian interfaces. The boundaries of the condensate are Eulerian edges, where a flux go through, see Fig. 3 third line. Each layer is associated with layer centered variables and each interface is associated with a $2D/3D$ normal vector.

Actually, this numerical strategy consists in condensating neighboring mixed cells in one direction of the Cartesian mesh, in which interfaces between materials are considered $1D$, namely they are considered vertical in x direction step. They move independently from the Eulerian mesh, see Fig. 3.

A *condensate* then contains layers of successive different materials that are separated by interfaces and the thickness of these layers is calculated by volume conservation. The ordering of material layers is known by the $2D/3D$ description at the previous time step. It is determined by volume fractions in neighboring cells, as it is described in Section 2.2.3. Their evolution is calculated in a Lagrangian point of view and the scheme is written as described in Section 2.3.1. Obviously, layers can be as thin as partial volumes are small. Once quantities and interface positions inside the *condensate* are known at time t^{n+1}, they are remapped on the original Eulerian mesh, see Fig. 3.

2.2.2. Construction of a condensate

The mesh is scanned line by line in x direction step and a *condensate* is created if a mixed cell is detected. The number of successive mixed cells is then determined, and they are associated

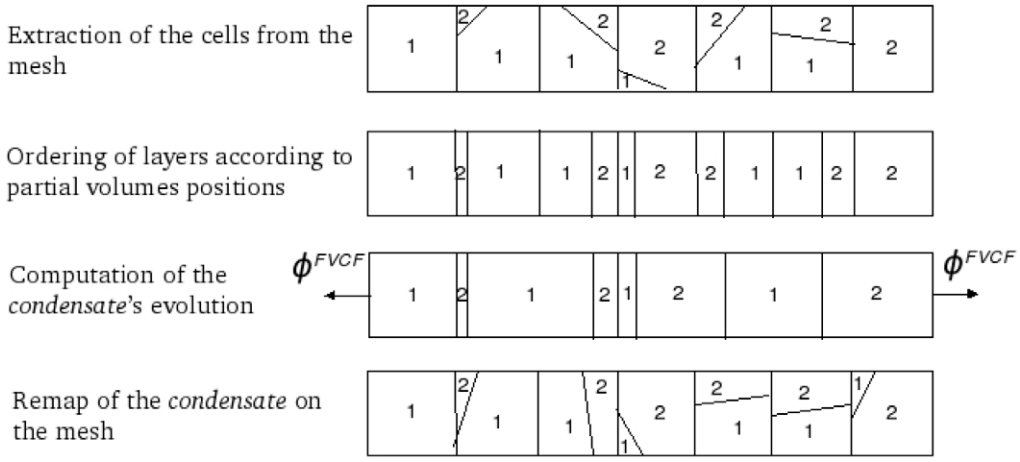

Fig. 3. Treatment of neighboring mixed cells by using a *condensate*.

with the previous and next pure cells of these mixed cells. The evolution of this set of cells is calculated independently of the Eulerian mesh as a *condensate*. The *condensate* is described by the following variables:

- each volume in these mixed and pure cells becomes a layer of the *condensate*, separated by a vertical interface, in x direction step, which the abscissa is determined by volume conservation. Materials ordering in each direction is known in each mixed cell. If two successive layers are filled of the same material, then these two are merged in one layer and variables are averaged in a conservative way. Thus the *condensate* is constituted of nc layers of different successive materials,
- variables such as volume, density, velocity, internal energy, pressure are volume centered for partial volumes in mixed cells, thus they are known for each layer,
- border interfaces abscissas are known for each layer by construction,
- the interface normal vector is calculated in $2D/3D$ and is associated with each $1D$ interface of the *condensate*. If a $1D$ interface of the *condensate* is created from a cell edge and not from an interface in a mixed cell, then the $2D/3D$ associated normal vector is naturally taken as the cell edge normal vector,
- the *condensate* description is completed by two outgoing fluxes, on the left and right side. These fluxes are single fluid FVCF fluxes calculated before the *condensate* generation. At this stage, information related to the Eulerian mesh are no longer necessary to update *condensate* variables values at time t^{nx}.

2.2.3. Remapping of a condensate

Condensate variables calculated during x direction step are remapped, that is to say variables in cells at time t^n that were extracted from the Eulerian mesh to constitute the *condensate* are updated with values calculated in the condensate at time t^{nx}, end time of the x direction step. Interface positions in the *condensate* give the nature of cells, mixed or pure, at time t^{nx}. Partial volumes in each mixed cell is determined according to $1D$ interface positions at time t^{nx}. The ordering of materials in the mixed cell is obvious in x direction by construction of the *condensate* successive layers. The ordering in y direction could have changed during x direction step. In a mixed cell C, it is given at t^{nx} by volume fractions in neighboring cells, C_{up} up and C_{down} down in y direction:

- if the volume fraction of material 1 in cell C_{up} is larger than the one in C_{down}, then the material 1 is on the upper part of the mixed cell,
- if the volume fraction of material 1 in cell C_{down} is larger than the one in C_{up}, then the material 1 is on the lower part of the mixed cell.

2.2.4. Interface reconstruction in $2D/3D$

At time t^{nx}, mixed cells, partial volumes and materials ordering in each direction are known. The $2D/3D$ normal is calculated with *Youngs* formulation [18]. It is based on an approximation of the volume fraction gradient ∇f in mixed cells, which gives the normal direction in cell i:

$$n_i = -\frac{\nabla f_i}{\|\nabla f_i\|}, \tag{6}$$

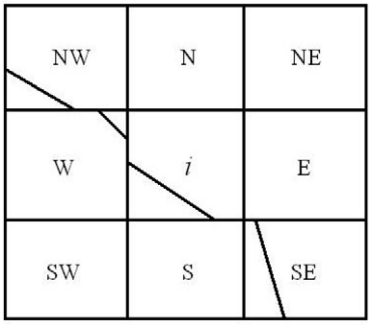

Fig. 4. Neighborhood of cell i in $2D$.

with $\nabla f_i = (\frac{\partial f_i}{\partial x}, \frac{\partial f_i}{\partial y})^t$ the $2D/3D$ volume fraction gradient (see Fig. 4). It is approximated by (we give the formulas in 2D, 3D extensions are straightforward):

$$\partial_x f_i = \frac{f_E^* - f_W^*}{2\Delta x} \quad \text{and} \quad \partial_y f_i = \frac{f_N^* - f_S^*}{2\Delta y}, \tag{7}$$

where Δx and Δy respectively denote the x and y space steps, and

$$f_E^* = \frac{f_{NE} + 2f_E + f_{SE}}{4}, \qquad f_W^* = \frac{f_{NW} + 2f_W + f_{SW}}{4},$$
$$f_N^* = \frac{f_{NW} + 2f_N + f_{NE}}{4}, \qquad f_S^* = \frac{f_{SW} + 2f_S + f_{SE}}{4}. \tag{8}$$

2.3. Numerical scheme in the condensate

In this section, the way we compute the evolution of Eulerian variables in a condensate is described. The scheme is taking into account interface motion and Eulerian fluxes at the condensate boundaries. Moreover, fluxes through moving interfaces are written in such a way that the perfect sliding condition between materials is settled.

2.3.1. 1D integration of the system

First, we consider the following $1D$ system of conservation laws where, in this paragraph, x represents the normal direction n:

$$\partial_t V + \partial_x F(V) = 0. \tag{9}$$

- Let be $F(V)\cdot n$ the flux in direction n of conservative variables in $1D$:

$$F(V)\cdot n = \big(\rho(u\cdot n), \rho u(u\cdot n) + pn, (\rho E + p)(u\cdot n)\big) = V(u\cdot n) + pN \tag{10}$$

  with $V = (\rho, \rho u, \rho E)^t$ and $N = (0, n, u\cdot n)$.
- Let A be the transverse section area and Vol the volume of a cell in $1D$.
- For given materials $i = 1, 2$, let ρ_i be the density, u_i the velocity vector, e_i the specific internal energy, p_i the pressure, $E_i = e_i + (u_i)^2/2$ the total specific energy, and V_i the variable vector associated with material i in the partial volume Vol_i inside a mixed cell. Of course, one has $Vol_1 + Vol_2 = Vol$.

Let $\Omega(t)$ be a volume and its surface boundary $\Gamma(t) = \Gamma_{u_{int}=0} \cup \Gamma_{u_{int}\neq 0}$, with $\Gamma_{u_{int}=0}$ time independent edges and $\Gamma_{u_{int}\neq 0}$ moving edges with local velocity u_{int} on $\Gamma_{u_{int}\neq 0}$ depending on time (see Fig. 5). Then, system (9) is integrated over $\Omega(t)$, taking into account volume evolution with time:

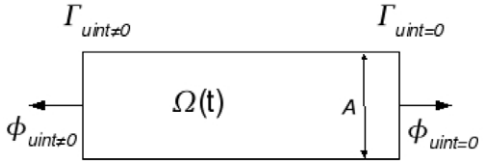

Fig. 5. 1D integration of a condensate layer.

$$\int_{\Omega(t)} \left(\partial_t V + \partial_x F(V)\right) d\tau = \frac{d}{dt}\int_{\Omega(t)} V\, d\tau - \int_{\Gamma_{u_{int}\neq0}} V(u_{int}\cdot n_{int})\, ds + \int_{\Gamma_{u_{int}\neq0}} F(V)(e_x\cdot n_{int})\, ds + \int_{\Gamma_{u_{int}=0}} F(V)(e_x\cdot n_{int})\, ds. \quad (11)$$

Moreover, using the notation $N(u_{int}, n_{int}) = (0, n_{int}, u_{int}\cdot n_{int})^t$ and Eq. (10):

$$\int_{\Gamma_{u_{int}\neq0}} F(V)\cdot n\, ds = \int_{\Gamma_{u_{int}\neq0}} V(u_{int}\cdot n_{int})\, ds + \int_{\Gamma_{u_{int}\neq0}} p_{int} N(u_{int}, n_{int})\, ds. \quad (12)$$

Finally simplifying the common advection terms in (11) and (12), the integrated system reads:

$$\frac{d}{dt}\int_{\Omega(t)} V\, d\tau + \int_{\Gamma_{u_{int}\neq0}} p_{int} N(u_{int}, n_{int})\, ds + \int_{\Gamma_{u_{int}=0}} F(V)\cdot n\, ds = 0. \quad (13)$$

The finite volume scheme then reads as:

$$\frac{|\Omega^{n+1}|V^{n+1} - |\Omega^n|V^n}{dt} + A\phi_{u_{int}=0} + A p_{int} N(u_{int}, n_{int}) = 0 \quad (14)$$

with

$$V^n = \frac{1}{|\Omega^n|}\int_{\Omega^n} V^n d\tau$$

the average value in Ω^n at time t^n, dt the time step, A the transverse section of the $1D$ cell (or face area in $2D/3D$ in the direction n),

$$\phi_{u_{int}=0} = \frac{1}{A}\int_{\Gamma} F\cdot n\, ds$$

the flux through the edge Γ given by the single fluid scheme. The flux through a moving interface is $\phi_{int} = p_{int}N(u_{int}, n_{int})$. The determination of interface pressure p_{int} and velocity u_{int} will be detailed in Section 2.3.3.

2.3.2. Evolution in a condensate

The evolution of *condensate* variables is $1D$, but using $2D/3D$ information. Three types of layers exist in the *condensate*:

- the first layer ($k = 1$) has an interface on the left side that does not move and where a single fluid outgoing flux is imposed. The interface on the right side is moving,
- internal layers ($1 < k < nc$) have moving interfaces on the left and right sides,
- the last layer ($k = nc$) has an interface on the right side that does not move and where a single fluid outgoing flux is imposed. The interface on the left side is moving (see Fig. 6).

Let us describe the calculation of *condensate* variables at time t^{nx} from those at time t^n in x direction fluid. Superscript $n+1$ denotes generically the result at the end of each directional step:

- first layer $k = 1$:
$$\frac{Vol_1^{n+1}V_1^{n+1} - Vol_1^n V_1^n}{dt} + A\left(\phi_\ell + \phi_{int}^{1,2}\right) = 0, \quad (15)$$
- internal layers $1 < k < nc$:
$$\frac{Vol_k^{n+1}V_k^{n+1} - Vol_k^n V_k^n}{dt} + A\left(\phi_{int}^{k-1,k} + \phi_{int}^{k,k+1}\right) = 0, \quad (16)$$
- last layer $k = nc$:
$$\frac{Vol_{nc}^{n+1}V_{nc}^{n+1} - Vol_{nc}^n V_{nc}^n}{dt} + A\left(\phi_{int}^{nc-1,nc} + \phi_r\right) = 0, \quad (17)$$

where Vol_k^n denotes the volume, $V_k^n = (\rho_k^n, \rho_k^n u_k^n, \rho_k^n E_k^n)^t$ the variable vector in layer k at time t^n obtained in Section 2.2.2 by conservative averaging of merged layer variables, see Eqs. (5). Fluxes ϕ_ℓ and ϕ_r are the prescribed outgoing fluxes at the *condensate* boundaries, $\phi_{int}^{i,j} = p_{int}^{i,j}(0, n_{1D}^{i,j}, u_{int}^{i,j}\cdot n_{1D}^{i,j})^t$ the flux through the moving interface between layers i and j, with $n_{1D}^{i,j}$ the x axis unit vector in x direction step, from layer i to layer j. In next sections, detailed calculations are provided for each variable.

Calculation of layer masses Masses m^{n+1} at time t^{n+1} are known independently of layers volumes. They only depend on mass fluxes at the left and the right boundary of the *condensate*:

$$\begin{cases} m_1^{n+1} = m_1^n - dt\, A\phi_\ell(1), \\ m_k^{n+1} = m_k^n, \\ m_{nc}^{n+1} = m_{nc}^n - dt\, A\phi_r(1) \end{cases} \quad (18)$$

with ϕ the FVCF numerical flux, whose components correspond to $(\rho u, \rho u\otimes u + p, (\rho E + p)u)$. These masses are positive if the CFL condition is fulfilled. This is satisfied because the first and last layer contain at least a pure cell.

Calculation of layer volumes Layer volume evolution in the *condensate* is given by interface velocities:

$$\begin{cases} Vol_1^{n+1} = Vol_1^n + dt\, A\left(u_{int,x}^{1,2}\right), \\ Vol_k^{n+1} = Vol_k^n + dt\, A\left(u_{int,x}^{k,k+1} - u_{int,x}^{k-1,k}\right), \\ Vol_{nc}^{n+1} = Vol_{nc}^n + dt\, A\left(-u_{int,x}^{nc-1,nc}\right). \end{cases} \quad (19)$$

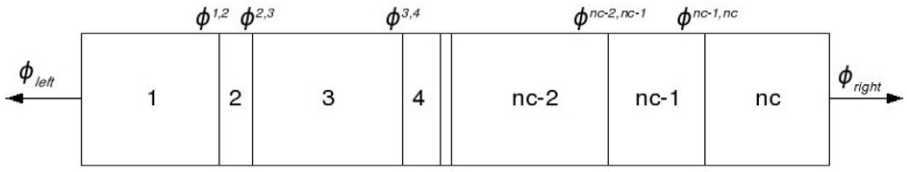

Fig. 6. Numbering of layers and fluxes in a condensate.

Calculation of layer velocities and total energies Let us introduce some notations associated with each layer k:

$$\theta_k = \frac{m_k^n}{m_k^{n+1}}, \quad \text{and} \quad \kappa_k = \frac{Vol_k^n}{dt\,A}. \tag{20}$$

Extracting equations for the velocity u_k in x direction, for the velocity v_k in y direction and for the total energy E_k in Eqs. (15), (16) and (17), we obtain:

$$\begin{cases} u_1^{n+1} = \theta_1\left(u_1^n - \dfrac{p_{int,x}^{1,2} + \phi_\ell(2)}{\rho_1^n \kappa_1}\right), \\ u_k^{n+1} = u_k^n - \dfrac{p_{int,x}^{k,k+1} - p_{int,x}^{k-1,k}}{\rho_k^n \kappa_k}, \\ u_{nc}^{n+1} = \theta_{nc}\left(u_{nc}^n - \dfrac{\phi_r(2) - p_{int,x}^{nc,nc-1}}{\rho_{nc}^n \kappa_{nc}}\right), \end{cases} \tag{21}$$

$$\begin{cases} v_1^{n+1} = \theta_1\left(v_1^n - \dfrac{\phi_\ell(3)}{\rho_1^n \kappa_1}\right), \\ v_k^{n+1} = v_k^n, \\ v_{nc}^{n+1} = \theta_{nc}\left(v_{nc}^n - \dfrac{\phi_r(3)}{\rho_{nc}^n \kappa_{nc}}\right), \end{cases} \tag{22}$$

$$\begin{cases} E_1^{n+1} = \theta_1\left(E_1^n - \dfrac{p_{int,x}^{1,2} u_{int,x}^{1,2} + \phi_\ell(4)}{\rho_1^n \kappa_1}\right), \\ E_k^{n+1} = E_k^n - \dfrac{p_{int,x}^{k,k+1} u_{int,x}^{k,k+1} - p_{int,x}^{k-1,k} u_{int,x}^{k-1,k}}{\rho_k^n \kappa_k}, \\ E_{nc}^{n+1} = \theta_{nc}\left(E_{nc}^n - \dfrac{\phi_r(4) - p_{int,x}^{nc,nc-1} u_{int,x}^{nc-1,nc}}{\rho_{nc}^n \kappa_{nc}}\right). \end{cases} \tag{23}$$

2.3.3. Calculation of interface pressure and velocity

Interface pressure and velocity in $1D$ In a consistent way with respect to the Eulerian scheme FVCF [7,8], the local Riemann invariants, here $\pi^+ = p + \rho c u$ and $\pi^- = p - \rho c u$, are advected with the sound speed $\pm c$:

$$\partial_t \pi^+ + c\partial_x \pi^+ = 0, \quad \text{and} \quad \partial_t \pi^- - c\partial_x \pi^- = 0. \tag{24}$$

As for the single fluid scheme *FVCF*, advection velocities in the numerical scheme are linearized around interface values. Then the equations are solved following characteristic curves. Here, the advection velocity c (resp. $-c$) is sound velocity that is always positive (resp. negative). Therefore, upwinding leads to:

$$\begin{aligned} \pi_{int}^+ = \pi_\ell^+ \quad &\Leftrightarrow \quad p_{int} + \rho_{int} c_{int} u_{int} = p_\ell + \rho_\ell c_\ell u_\ell, \\ \pi_{int}^- = \pi_r^- \quad &\Leftrightarrow \quad p_{int} - \rho_{int} c_{int} u_{int} = p_r - \rho_r c_r u_r. \end{aligned} \tag{25}$$

Values of p_{int} and u_{int} are then given by:

$$\begin{cases} p_{int} = \dfrac{\rho_r c_r p_\ell + \rho_\ell c_\ell p_r}{\rho_\ell c_\ell + \rho_r c_r} + \rho_\ell c_\ell \rho_r c_r \dfrac{u_\ell - u_r}{\rho_\ell c_\ell + \rho_r c_r}, \\ u_{int} = \dfrac{\rho_\ell c_\ell u_\ell + \rho_r c_r u_r}{\rho_\ell c_\ell + \rho_r c_r} + \dfrac{p_\ell - p_r}{\rho_\ell c_\ell + \rho_r c_r}. \end{cases} \tag{26}$$

These expressions are consistent with Godunov's acoustic solver [9], [15].

As in the Eulerian case, the time step dt is given by the CFL stability condition for this Lagrangian scheme:

$$dt < \min_i\left(\frac{Vol_i}{Ac_i}\right) \tag{27}$$

with Vol the volume and A the transverse section of the cell.

Proposition 2. *Let us set the following approximation for sound speeds*:

$$\tilde{c}_i = \min(c_i, \kappa_i)$$

in cell i *and* $\kappa_i = Vol_i^n/(dt\,A)$. *Eqs.* (26) *for* p_{int} *and* u_{int} *are modified into*:

$$\begin{cases} p_{int} = \dfrac{\rho_r \tilde{c}_r p_\ell + \rho_\ell \tilde{c}_\ell p_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} + \rho_\ell \tilde{c}_\ell \rho_r \tilde{c}_r \dfrac{u_\ell - u_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r}, \\ u_{int} = \dfrac{\rho_\ell \tilde{c}_\ell u_\ell + \rho_r \tilde{c}_r u_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} + \dfrac{p_\ell - p_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r}. \end{cases} \tag{28}$$

Let us consider the case of a layer i *between two moving interfaces where the fluxes are computed using formulae* (28).

The layer evolution between t^n *and* t^{n+1} *is "entropic" in the sense of the following discrete entropy inequality:*

$$T_i^n\left(s_i^{n+1} - s_i^n\right) = e_i^{n+1} - e_i^n + p_i^n\left(\frac{1}{\rho_i^{n+1}} - \frac{1}{\rho_i^n}\right) \geqslant 0 \tag{29}$$

with the temperature T_i^n, *the density* ρ_i^n, *the internal energy* e_i^n, *the pressure* p_i^n, *the entropy* s_i^n *in cell* i *at time* t^n.

Proof can be found in Braeunig PhD report [4].

Interface pressure and velocity in $2D/3D$ Interface pressure and velocity in $1D$ are given in Section 2.3.3:

$$\begin{cases} p_{int}^{\ell,r} = \dfrac{\rho_r \tilde{c}_r p_\ell + \rho_\ell \tilde{c}_\ell p_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} + \rho_\ell \tilde{c}_\ell \rho_r \tilde{c}_r \dfrac{u_\ell - u_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r}, \\ u_{int}^{\ell,r} = \dfrac{\rho_\ell \tilde{c}_\ell u_\ell + \rho_r \tilde{c}_r u_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} + \dfrac{p_\ell - p_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r}, \end{cases} \tag{30}$$

with $\tilde{c}_i = \min(c_i, \kappa_i)$ in layer i and $\kappa_i = \frac{Vol_i^n}{dt\,A}$. The $2D/3D$ interface is in general not parallel with mesh directions. $1D$ formulae are then written in the $2D/3D$ direction of the interface normal vector $n^{\ell,r}$ as follows:

$$n^{\ell,r} \cdot n_{1D}^{\ell,r} \geqslant 0, \tag{31}$$

with $n_{1D}^{\ell,r}$ the x axis unit vector in x direction step, from layer ℓ to layer r.

In the next definition, perfect sliding of materials is addressed.

Definition. In $2D/3D$, when contact condition between two materials is perfect sliding, the pressure gradient $\nabla p_{int}^{\ell,r}$ and the velocity $u_{int}^{\ell,r}$ at the interface have non-zero normal component and zero tangential components.

Therefore, we define the pressure gradient and the velocity at the interface as follows using results of Section 2.3.3:

$$\begin{cases} \nabla p_{int}^{\ell,r} = \left(p_{int}^{\ell r} - \dfrac{\rho_r \tilde{c}_r p_\ell + \rho_\ell \tilde{c}_\ell p_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r}\right) n^{\ell r}, \\ u_{int}^{\ell r} = \left(u_{int}^{\ell r} \cdot n^{\ell r}\right) n^{\ell r}. \end{cases} \tag{32}$$

Then $1D$ formulae (28) can be rewritten along the interface normal vector as follows:

$$\begin{cases} \nabla p_{int}^{\ell r} \cdot n^{\ell r} = \rho_\ell \tilde{c}_\ell \rho_r \tilde{c}_r \dfrac{u_\ell - u_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} \cdot n^{\ell r}, \\ u_{int}^{\ell r} \cdot n^{\ell r} = \dfrac{\rho_\ell \tilde{c}_\ell u_\ell + \rho_r \tilde{c}_r u_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} \cdot n^{\ell r} + \dfrac{p_\ell - p_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} \end{cases} \tag{33}$$

with $u_{int}^{\ell r}$ the $2D/3D$ interface velocity, $n^{\ell r} = (n_x^{\ell r}, n_y^{\ell r}, n_z^{\ell r})^t$ the $2D/3D$ interface normal unit vector, p^ℓ and p^r pressures, u^ℓ and u^r velocity vectors in left and right layers.

This result is fully $2D/3D$. The numerical strategy for the multi-dimensional approach consists in writing the scheme in such a

way that the interface evolves direction after direction. Therefore, in x direction step, interface motion is $1D$ only along x axis. The $2D/3D$ interface pressure gradient $\nabla p_{int}^{\ell r}$ and velocity $u_{int}^{\ell r}$ are projected onto x axis to obtain the interface pressure and velocity in x direction step:

$$\begin{cases} p_{int,x}^{\ell r} = \dfrac{\rho_r \tilde{c}_r p_\ell + \rho_\ell \tilde{c}_\ell p_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} + \left(\rho_\ell \tilde{c}_\ell \rho_r \tilde{c}_r \dfrac{u_\ell - u_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} \cdot n^{\ell r} \right) n_x^{\ell r}, \\ u_{int,x}^{\ell r} = \left(\dfrac{\rho_\ell \tilde{c}_\ell u_\ell + \rho_r \tilde{c}_r u_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} \cdot n^{\ell r} + \dfrac{p_\ell - p_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} \right) n_x^{\ell r}. \end{cases} \tag{34}$$

Unfortunately, these formulae do not satisfy contact discontinuity conditions for the velocity. Considering an uniform velocity field u_0 along the x axis and an uniform field of pressure p_0, interface velocity and pressure are:

$$\begin{cases} p_{int,x}^{\ell r} = p_0, \\ u_{int,x}^{\ell r} = u_0 \left(n_x^{\ell r} \right)^2. \end{cases} \tag{35}$$

To fix this problem, we will use an alternative formulation for the $1D$ step interface motion, which preserves contact discontinuity and provides the same result, considering all steps of the directional splitting.

Proposition 3. *Let us assume that the normal unit vector at the interface does not change during the time step dt. Therefore interfaces at time t^n and t^{n+1} are parallel. Let us consider a point M_0 on the interface at time t^n. The position of point M_1 on the interface at time t^{n+1} is obtained using (34) as follows*:

$$\overrightarrow{M_0 M_1} = dt \left(\left(\frac{\rho_\ell \tilde{c}_\ell u_\ell + \rho_r \tilde{c}_r u_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} \cdot n^{\ell r} \right) n^{\ell r} + \frac{p_\ell - p_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} n^{\ell r} \right). \tag{36}$$

Let us consider point M_2 defined by:

$$\overrightarrow{M_0 M_2} = dt \left(\frac{\rho_\ell \tilde{c}_\ell u_\ell + \rho_r \tilde{c}_r u_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} + \frac{p_\ell - p_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} n^{\ell r} \right). \tag{37}$$

For each point M_0 on the interface at time t^n, the line defined by M_1 and the normal vector n and the line defined by M_2 and the normal vector n are the same line defining the interface at time t^{n+1}. M_1 and M_2 have the same projection on the normal direction. Therefore, interface position at time t^{n+1} obtained using $1D$ steps is the same as if using the following $1D$ interface pressure and velocity formulation in the generic x direction step:

$$\begin{cases} p_{int,x}^{\ell r} = \dfrac{\rho_r \tilde{c}_r p_\ell + \rho_\ell \tilde{c}_\ell p_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} + \left(\rho_\ell \tilde{c}_\ell \rho_r \tilde{c}_r \dfrac{u_\ell - u_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} \cdot n^{\ell r} \right) n_x^{\ell r}, \\ u_{int,x}^{\ell r} = \dfrac{\rho_\ell \tilde{c}_\ell u_{\ell,x} + \rho_r \tilde{c}_r u_{r,x}}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} + \dfrac{p_\ell - p_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} n_x^{\ell r}. \end{cases} \tag{38}$$

Proof. Since normal unit vectors $n^{\ell r}$ at the interface are the same at time t^n and t^{n+1}, we will show that vector $\overrightarrow{M_1 M_2}$ is orthogonal to vector $n^{\ell r}$. Thus M_1 and M_2 are on the same line (see Fig. 7).

$$\begin{aligned} \overrightarrow{M_1 M_2} \cdot n^{\ell r} &= -\overrightarrow{M_0 M_1} \cdot n^{\ell r} + \overrightarrow{M_0 M_2} \cdot n^{\ell r} \\ &= -\left(\frac{\rho_\ell \tilde{c}_\ell u_\ell + \rho_r \tilde{c}_r u_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} \cdot n^{\ell r} \right) n^{\ell r} \cdot n^{\ell r} \\ &\quad - \frac{p_\ell - p_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} n^{\ell r} \cdot n^{\ell r} \\ &\quad + \frac{\rho_\ell \tilde{c}_\ell u_\ell + \rho_r \tilde{c}_r u_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} \cdot n^{\ell r} + \frac{p_\ell - p_r}{\rho_\ell \tilde{c}_\ell + \rho_r \tilde{c}_r} n^{\ell r} \cdot n^{\ell r} \\ &= 0 \end{aligned}$$

because $\|n^{\ell r}\| = 1$.

Therefore, one can use formulation (38) that preserves contact discontinuity in $1D$ steps, instead of formulation (34) which does not. □

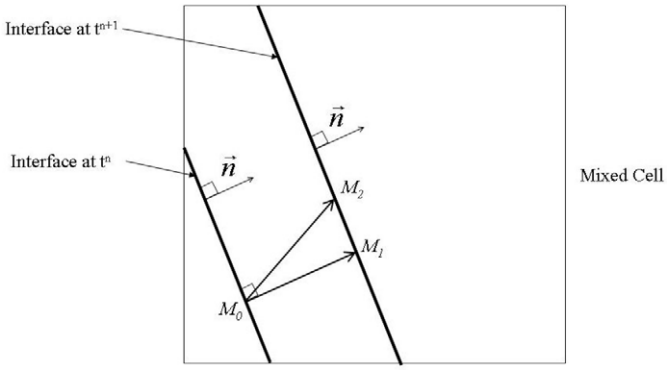

Fig. 7. Equivalence between formulations (34) and (38) for interface velocity.

2.4. Control of pressure evolution

Pressure is a quantity which has to be particularly controlled. Experimentally, we observe that when pressure has monotonic variations, other quantities such as density, velocity or internal energy are also monotonic. When using an equation of state of the form $p = P(\rho, e)$, with density ρ and specific internal energy e, no evolution equation for p is available. Pressure evolution in time is then controlled through the evolution of density ρ and specific entropy s. Let us introduce the Grüneisen coefficient $\Gamma = \frac{1}{\rho}(\frac{\partial p}{\partial e})_\rho$ and the sound speed $c^2 = (\frac{\partial p}{\partial \rho})_s$.

The variation of p as a function of variations of ρ and s reads:

$$\frac{dp}{p} = \left(\frac{\rho \Gamma}{p} \right) T \, ds + \left(\frac{\rho c^2}{p} \right) \frac{d\rho}{\rho}. \tag{39}$$

2.4.1. Discrete control of pressure evolution

At the discrete level, the only variables of the scheme that can be modified without loosing conservation of layers Eulerian quantities are interfaces pressure and velocity. The purpose of this section is to find constraints on these variables that ensure a "reasonable" variation of p during one time step. It is defined as follows:

$$\left| \frac{p^{n+1} - p^n}{p^n} \right| \leqslant 2\varepsilon \tag{40}$$

with $\varepsilon = 0.1$ for instance. This approach is very usual in compressible hydrodynamics computation, in particular to limit density's evolution in such a way that pressure evolution is controlled. In other words, it may be interpreted as a control of the compression rate from one time step to the next one.

For computational efficiency, the discrete control of pressure p between t^n and t^{n+1} will be achieved with explicit values at time t^n, using expression (39):

$$\begin{aligned} \frac{p^{n+1} - p^n}{p^n} \approx{}& \left(\frac{\rho^n \Gamma^n}{p^n} \right) T^n \left(s^{n+1} - s^n \right) \\ &+ \left(\frac{\rho^n (c^n)^2}{p^n} \right) \frac{\rho^{n+1} - \rho^n}{\rho^n}. \end{aligned} \tag{41}$$

We will exhibit the dependency of each term of this equation to interfaces pressure and velocity and control each one.

In the next sections, we consider the case of a layer i between two moving interfaces, $+$ denotes the right one and $-$ denotes the left one.

2.4.2. Control of density evolution

Let us control density evolution in Eq. (39) as follows:

$$\left| \left(\frac{\rho^n (c^n)^2}{p^n} \right) \frac{\rho^{n+1} - \rho^n}{\rho^n} \right| \leqslant \varepsilon. \tag{42}$$

Density can be expressed as follows:

$$m_i^{n+1} = m_i^n,$$
$$\rho_i^{n+1} = m_i^{n+1}/Vol_i^{n+1},$$
$$Vol_i^{n+1} = Vol_i^n\left(1 + \frac{\Delta(u)_i}{\kappa_i}\right) \tag{43}$$

with m_i the mass of layer i, $\Delta(u)_i = u^+ - u^-$, $\kappa_i = Vol_i^n/dt\,A$.

The constraint (42) then writes:

$$\left|\frac{-\frac{\Delta(u)_i}{\kappa_i}}{1+\frac{\Delta(u)_i}{\kappa_i}}\right| \leqslant \frac{\varepsilon p_i^n}{\rho_i^n (c_i^n)^2}.$$

Assuming, according to relations (43), that $|\frac{\Delta(u)_i}{\kappa_i}| < \varepsilon < 1$ to ensure the positivity of layers volumes, we obtain the following constraint:

$$\left|\frac{\Delta(u)_i}{\kappa_i}\right| \leqslant \frac{\varepsilon p_i^n}{\rho_i^n (c_i^n)^2}. \tag{44}$$

When this constraint is satisfied, density variation is controlled as defined in (42).

We have assumed that $|\frac{\Delta(u)_i}{\kappa_i}| < 1$ to ensure positivity of layers volume. This hypothesis is ensured for perfect gases with $\gamma_i^n > 1$, thanks to constraint (44) with $p_i^n/\rho_i^n (c_i^n)^2 = 1/\gamma_i^n < 1$. For general equations of state, we have to verify that $p_i^n/\rho_i^n (c_i^n)^2 < 1$. If this is not satisfied, we have to choose ε in such a way that $\frac{\varepsilon p_i^n}{\rho_i^n (c_i^n)^2} < 1$.

2.4.3. Control of entropy evolution

Next, let us control entropy evolution in Eq. (39) as follows:

$$\left|\left(\frac{\rho_i^n \Gamma_i^n}{p_i^n}\right) T_i^n \left(s_i^{n+1} - s_i^n\right)\right| \leqslant \varepsilon. \tag{45}$$

Considering the discrete expression for entropy's evolution (29) defined in Proposition 2 and expressions of quantities at time t^{n+1} in Section 2.3.2, we obtain:

$$T^n(s^{n+1} - s^n) = -\frac{\Delta(pu)_i - p_i^n \Delta(u)_i - u_i^n \Delta(p)_i}{\rho_i^n \kappa_i} - \frac{1}{2}\left(\frac{\Delta(p)_i}{\rho_i^n \kappa_i}\right)^2$$

and thus (45) is equivalent to:

$$\left(\frac{\rho_i^n \Gamma_i^n}{p_i^n}\right)\left|-\frac{\Delta(pu)_i - p_i^n \Delta(u)_i - u_i^n \Delta(p)_i}{\rho_i^n \kappa_i} - \frac{1}{2}\left(\frac{\Delta(p)_i}{\rho_i^n \kappa_i}\right)^2\right| \leqslant \varepsilon \tag{46}$$

with the operator $\Delta(x)_i = x_i^+ - x_i^-$ and $\kappa_i = Vol_i^n/dt\,A$.

Let us define a velocity with an expression close from sound speed for perfect gases:

$$\left(\tilde{c}_i^n\right)^2 = \frac{p_i^n}{\rho_i^n \Gamma_i^n}. \tag{47}$$

With this notation, (46) rewrites as:

$$\left|-\frac{\Delta(pu)_i - p_i^n \Delta(u)_i - u_i^n \Delta(p)_i}{\rho_i^n \kappa_i (\tilde{c}_i^n)^2} - \frac{1}{2}\left(\frac{\Delta(p)_i}{\rho_i^n \kappa_i \tilde{c}_i^n}\right)^2\right| \leqslant \varepsilon. \tag{48}$$

Both terms in this equation are of second order whenever $\frac{\Delta(p)_i}{\rho_i^n \kappa_i \tilde{c}_i^n}$ and $\frac{\Delta(u)_i}{\tilde{c}_i^n}$ are small. In particular, the first term is of second order:

$$\Delta(pu)_i - p_i^n \Delta(u)_i - u_i^n \Delta(p)_i = \left(u_i^- - u_i^n\right)\Delta(p)_i + \left(p_i^- - p_i^n\right)\Delta(u)_i + \Delta(p)_i \Delta(u)_i, \tag{49}$$

considering that $|u_i^- - u_i^n| \approx |\Delta(u)_i|$ and $|p_i^- - p_i^n| \approx |\Delta(p)_i|$ when pressure and velocity are monotonic.

The second term $(\frac{\Delta(p)_i}{\rho_i^n \kappa_i \tilde{c}_i^n})^2$ is obviously of second order in $\frac{\Delta(p)_i}{\rho_i^n \kappa_i \tilde{c}_i^n}$.

Nevertheless, when the Lagrangian CFL condition (27) in the layer i is no longer fulfilled, i.e. $\kappa_i = Vol_i^n/dt\,A \leqslant (c_i^n)$, the second term dominates because it is divided by $(\kappa_i)^2$. Moreover, this term leads to a non-entropic behavior, because it has always a negative sign in the entropy equation. Therefore, it is the one we choose to control by a constraint on $\Delta(p)_i$ which reads:

$$\left|\Delta(p)_i\right| \leqslant \rho_i^n \kappa_i \sqrt{\frac{2\varepsilon p_i^n}{\rho_i^n \Gamma_i^n}}. \tag{50}$$

An algorithm can be built to modify interface pressures and velocities in order to satisfy constraints (44) and (50), see Braeunig PhD thesis [4].

3. Results

The method described so far has been developed in the context of an industrial code and has been challenged against real fluid flows. However these cases cannot be presented here and we shall provide hereafter typical benchmark cases which demonstrate the properties achieved by our method. Other numerical results can be found in Braeunig [4].

3.1. Perfect sliding

We consider two materials separated by an oblique interface. Material 1 on the left-hand side and material 2 on the right-hand side are perfect gases of coefficient $\gamma = 1.4$. Material 1 initial state is ($\rho = 1.4$, $p = 0.25$, $u_x = 0.4$, $u_y = 1$). Material 2 initial state is ($\rho = 2.8$, $p = 0.25$, $u_x = -0.4$, $u_y = -1$). Initial velocities are parallel to the interface for both materials, but with opposite sign. For this normalized benchmark, units are dropped. Final time is 0.25. The domain is 1 long and 1 wide. The mesh is made of 50×50 cells. See Fig. 8.

3.2. Shock wave interaction with a bubble of air in water

We consider a half cylindrical bubble of air initially sticking to the wall inside water. A shock wave is propagating in wa-

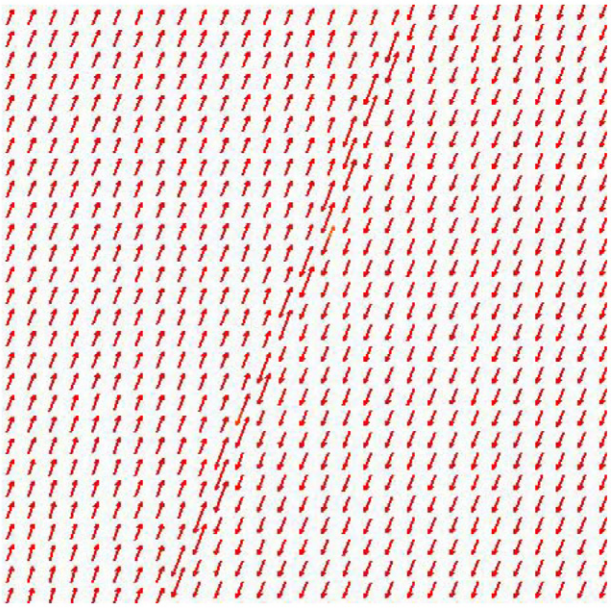

Fig. 8. Perfect sliding, zoom at the velocity field, at final time $t = 0.25$. The vertical relative displacement of materials at $t = 0.25$ is $(u_1 - u_2)_y t = 0.5$, thus 25 cells.

ter from left to right. Air is supposed to follow a perfect gas type ($p = (\gamma_a - 1)\rho e$ with $\gamma_a = 1.4$) equation of state and water a stiffened gas law type ($p = (N-1)\rho e - \Pi$ with $N = 7$ and $\Pi = 21 \times 10^8$ Pa). Initial state for air is set to ($\rho_a = 1$ kg/m^3, $p_a = 10^5$ Pa, $u_a = 0$ m/s), for water to ($\rho_0 = 1000$ kg/m^3, $p_0 = 10^5$ Pa, $u_0 = 0$ m/s), and for shocked water to ($\rho = 1322.05$ kg/m^3, $p = 200\,000 \times 10^5$ Pa, $u = 2206.68$ m/s). Mach number is around $M = 6$. The domain extension is 10 cm long and 5 cm wide. The bubble radius is set to 2 cm. The mesh is made of 200×100 cells (see Figs. 9, 10).

Remark 4. This is a case where taking account of material interface sliding and keeping physical variables associated with each material in mixed cells seems important. In this computation, any variable average between materials such as water and air leads to a strong error in terms of physical modelling. This is mainly due to strong differences as far as density, sound speed and compressibility properties are concerned.

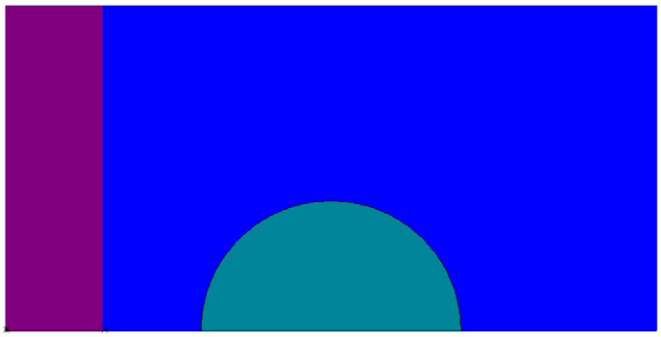

Fig. 9. Initial geometry, a shock wave is propagating in water from left to right. An air bubble sticks to the wall.

3.3. Spike of water in air

We consider a sinusoidal shape interface between water and air. A shock wave in water is propagating from right to left. The initial state for air is ($\rho_a = 1$ kg/m^3, $p_a = 10^5$ Pa, $u_a = 966.253$ m/s); for water ($\rho_0 = 1000$ kg/m^3, $p_0 = 10^5$ Pa, $u_0 = 966.253$ m/s); for shocked water ($\rho = 1285.71$ kg/m^3, $p = 42\,015 \times 10^5$ Pa, $u = 0$ m/s). Mach number is around $M = 3$. The domain is 39.6 cm long and 3.6 cm wide. Only half of the geometry is computed. The mesh is made of 270×30 cells with geometric progression in the grid, with aspect ratios preserving square cells on the spike (see Figs. 11, 12).

Fig. 11. Initial geometry, a shock wave is propagating in water from right to left. A sinusoidal shape interface between air on the left and water on the right.

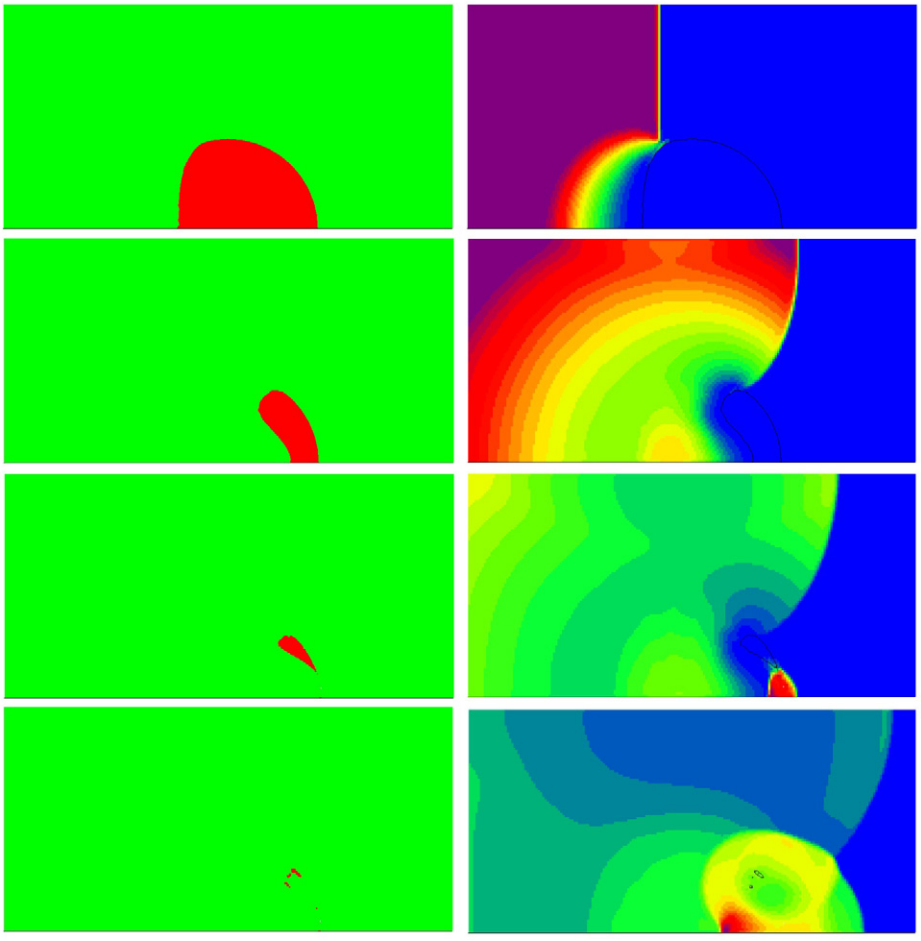

Fig. 10. Shock–bubble interaction, left column geometry, right column pressure, at times 3, 6.4, 7.4, 9 s.

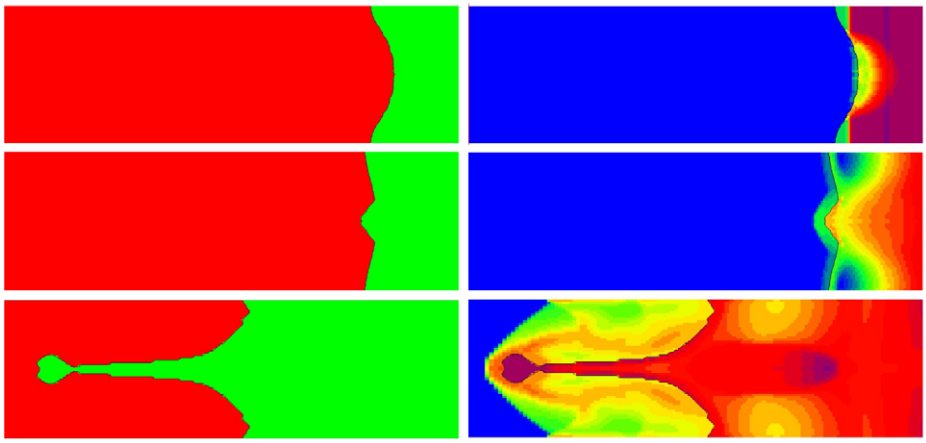

Fig. 12. Spike of water in air, left column geometry, right column sound speed, at times 3, 7 and 44 µs.

Remark 5. This case involving Richtmyer–Meshkov instability is a severe test for the method robustness. Indeed, while the spike is getting bigger, a very strong rarefaction wave develops in water. In air around the spike, an aerodynamical flow is developing, with a steady shock behind the spike head (see Fig. 12).

4. Conclusion

This method has been built step by step using severe benchmarks with very strong shocks, high sound speed ratios, high density ratios, high compressibility ratios and different equations of state for the fluids. This paper describes the current result of the numerical studies we made to improve robustness. Of course, some points have to be further investigated. First concerning accuracy of the interface reconstruction considered yet to be of order one, that we will extend to order two by improvement of the remapping step of condensates. Second to investigate the possibility of free sliding of the materials in an Eulerian method with quantitative benchmarks and compare to other existing methods.

References

[1] R. Abgrall, How to prevent pressure oscillations in multicomponent flow calculations: a quasi conservative approach, J. Comput. Phys. 125 (1996) 150–160.
[2] D.M. Anderson, G.B. McFadden, A.A. Wheeler, A diffuse-interface methods in fluid mechanics, Ann. Rev. Fluid Mech. 30 (1998) 139–165.
[3] C. Aymard, J. Flament, J.-P. Perlat, ALE with mixed elements, in: Proceedings of Workshop Numerical Methods for Multi-material Fluid Flows, Prague, 2007, http://www-troja.fjfi.cvut.cz/~multimat07/prezentace.html.
[4] J.-P. Braeunig, Sur la simulation d'écoulements multi-matériaux par une méthode eulérienne directe avec capture d'interfaces en dimensions 1, 2 et 3 (technical part in English), Thèse de Doctorat N^o 2007/85, Ecole Normale Supérieure de Cachan, 2007.
[5] B. Després, E. Labourasse, F. Lagoutière, The Vofire method for multicomponent flows on unstructured meshes, J.-L. Lions Laboratory preprint R07052, 2007.
[6] V. Dyadechko, M. Shashkov, Moment of fluid interface reconstruction, Los Alamos report LA-UR-05-7571, January 25, 2006.
[7] J.-M. Ghidaglia, A. Kumbaro, G. Le Coq, Une méthode volumes finis à flux caractéristiques pour la résolution numérique des systèmes hyperboliques de lois de conservation, C. R. Acad. Sc. Paris 322 (I) (1996) 981–988.
[8] J.-M. Ghidaglia, A. Kumbaro, G. Le Coq, On the numerical solution to two fluid models via a cell centered finite volume method, Eur. J. Mech. B/Fluids 20 (6) (2001) 841–867.
[9] S.K. Godunov, A difference method for the numerical computation of discontinuous solutions of the equations of fluid dynamics, Math. Sbornik 47 (1959) 271–290.
[10] D. Igra, K. Takayama, A high resolution upwind scheme for multi-component flows, International journal for numerical methods in fluids, Int. J. Numer. Methods Fluids (ISSN 0271-2091) 38 (10) (2002) 985–1007.
[11] B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, G. Zanetti, Modelling merging and fragmentation in multifluid flows with SURFER, J. Comput. Phys. 113 (1) (1994) 134–147.
[12] B.E. Launder, D.B. Spalding, Mathematical Models of Turbulence, Academic Press, 1972.
[13] R.J. LeVeque, Numerical Methods for Conservation Laws, Birkhäuser, 1990.
[14] W.F. Noh, P. Woodward, SLIC (Simple Line Interface Calculation), Lectures Notes in Physics, vol. 59, Springer, Berlin, 1976.
[15] R.D. Richtmyer, K.W. Morton, Difference Methods for Initial-Value Problems, John Wiley, 1967.
[16] J. Sethian, P. Smereka, Level set methods for fluid interfaces, Ann. Rev. Fluid Mech. 35 (2003) 341–372.
[17] M. Sussman, P. Smereka, S. Osher, A level set approach for computing solutions to incompressible two-phase flow, J. Comput. Phys. 114 (1994) 146–159.
[18] D.L. Youngs, Time-dependent multi-material flow with large fluid distortion, in: K.W. Morton, M.J. Baines (Eds.), Numerical Methods for Fluid Dynamics, 1982, pp. 273–285.