

A CONSERVATIVE SLIDE LINE METHOD FOR CELL-CENTERED SEMI-LAGRANGIAN AND ALE SCHEMES IN 2D

SILVIA BERTOLUZZA¹, STÉPHANE DEL PINO² AND EMMANUEL LABOURASSE²

Abstract. In this paper, we propose a new cell-center method to treat sliding of compressible fluid domains. We describe at first the theoretical framework based on [S. Del Pino, *C. R. Acad. Sci. Paris, Ser. I* **348** (2010) 1027–1032]. We introduce the notion of slide lines thanks to a mortar-like approach. We propose and analyze a $\mathbb{P}_1 - \mathbb{P}_0$ discretization of the theoretical method. We also describe a simple ALE procedure that preserves the slide line Lagrangian so that no mixed-cells model is necessary. Finally we present a set of representative numerical tests.

Mathematics Subject Classification. 65Z05, 35L65, 65N08, 65N30, 76N15.

Received May 9, 2014. Revised February 15, 2015.

Published online January 14, 2016.

1. INTRODUCTION

Lagrangian calculation of compressible gas dynamics is a powerful tool, especially when dealing with unmixed multi-materials. Indeed, since Lagrangian methods naturally preserve contact discontinuities, the use of mixing models is not necessary. Thus, lots of effort has been put in deriving numerical methods for compressible gas flows using Lagrangian formulation. Let us cite a few of them.

Historically, von Neumann and Richtmyer proposed the first scheme able to deal with shock waves [29]. Then, based on their work, Wilkins published another foundation paper for Lagrangian methods [32], where among many other things, he stated the basis of staggered Lagrangian hydrodynamics schemes for multi-dimensional problems. From this point, many improvements to staggered schemes were proposed. We just want to emphasize the very important contribution of Caramana *et al.* who published in [11] a conservative Lagrangian staggered scheme for compressible gas flows in 2D. Shortly after, through the concept of nodal solvers, Després and Mazeran defined the Glace's scheme [17], a multidimensional extension of the acoustic Godunov solver. This was rapidly followed by the contribution of Maire [26], where they described the Eucclhyd scheme. It was originally designed to retrieve exactly the 1D acoustic Godunov solver for 1D flows on Cartesian grids (Glace being dependent of cell's aspect ratio). In the past years, the community of Lagrangian hydrocodes has been very active around these two new schemes. This paper focuses on yet another extension of cell-centered semi-Lagrangian schemes: taking into account sliding with these scheme family, preserving at most their good properties.

Keywords and phrases. Compressible gas, Lagrange, ALE, slide lines, finite-volumes, finite-elements, mortar.

¹ Istituto di Matematica Applicata e Tecnologie Informatiche del CNR, Pavia, Italy.

² CEA, DAM, DIF 91297, Arpajon France. stephane@delpino.fr; stephane.delpino@cea.fr

As we said above, one of the strengths of Lagrangian schemes is their automatic treatment of contact discontinuities since there are no mass fluxes between cells. So, at first sight they seem very well suited to compute the sliding of one material onto another. However, Lagrangian schemes compute a continuous grid velocity which makes it impossible to deal with sliding. To treat sliding of materials one on another, the classic approach consists in using separate meshes for each material so that the velocity field is discontinuous at interface, and modeling numerically the interaction of domains. This is in some sense close to domain decomposition techniques.

There have been many works to treat slide lines with staggered schemes since the initial treatment proposed in [32]. The recent work [10] treats the slide lines problem as boundary conditions. In [24], the approach is enriched by adding stabilization terms to make the method more robust. Unfortunately, even if the obtained scheme is conservative in each sub-domain (the underlying scheme is compatible-hydro [11]), conservation is lost at the slide line. The defect of conservation is used as an *a posteriori* indicator to assess the quality of the solution. Recently, contact or slide-line methods have been proposed for cell-centered Lagrangian schemes [13,28]. [13] is the only approach, to our knowledge, that treats slide-lines in a fully conservative way.

Obviously, the treatment of materials sliding can be also carried out by using Eulerian formulation. There are many possible approaches which have to deal with mixed cells. Indeed, in finite-volume approaches for instance, ρ , \mathbf{u} and E have one value per material in each cell (mixed-cells). For this kind of problems material should not melt so that people usually use interface reconstruction method to determine the volume fraction presence of each constituent, see for instance [4, 22, 33] or [18]. Sliding treatment in Eulerian formulation is an active research field see, as an example, the recent works on this topic for compressible multi-material flows [7] or [14]. While the first one is related to direct Eulerian treatment, the second uses a Lagrange-remap approach. Also, some work are focused on an ALE formulation of the sliding problem, see for instance [20].

More generally, contact, sliding or friction calculus is a field of much interests in computational mechanics. The number of approaches and publications related to this general topic is enormous and goes far beyond the compressible multi-materials community. We cannot present here all the different techniques that can be used, so we refer to the review of Bourago and Kukudzhanov [6] that addresses a variety of different works.

The method we present in this paper is a cell-centered Lagrangian technique to model slide lines between materials. We also present an ALE extension keeping the sliding interface Lagrangian. This allows us to avoid the use of mixing models.

In few words, the method is based on [15] where we defined an abstract formulation in which the grid velocity is solution of an abstract minimization problem. At this point it appears clearly that the sliding problem can be viewed as a constrained optimization problem. The constrain being nothing but imposing the continuity of the normal velocity, $[[\mathbf{u}]] \cdot \mathbf{n} = 0$, across the slide line.

Also, since the method presented in [15] computes the grid velocity as the solution of a variational problem (some kind of weighted L^2 -projection), it seemed natural to derive a mortar-like method [5] to impose weakly the continuity of $\mathbf{u} \cdot \mathbf{n}$ through the interface.

Before going into the details of the scheme construction, we want to emphasize the choice of presentation we made.

Even if our goal is to build a conservative method to simulate sliding of fluids using polygonal meshes, we will present an abstract framework that requires very weak hypothesis on cells geometry itself. So, large parts of this presentation will be focused on an abstract scheme and the real discretization will occur lately.

The $\mathbb{P}_1 - \mathbb{P}_0$ scheme we propose is a particular choice of discretization and quadrature formula. Many other schemes can be derived based on the abstract formulation and then probably better ones. In this sense, the method we propose is more general than the one described in [13].

Also, writing an abstract variational formulation, that can be rewritten as a minimization problem, to compute the grid velocity (see [15] or Sect. 2), makes it clear that mortar-like approaches (see for instance [3, 5, 9]) could be used to solve the slide lines problem, for compressible gas flows.

So, the paper articulates as follows. In a first part, we recall and precise the method that was shortly presented in [15]. After establishing the abstract formulation and its properties, we focus on a \mathbb{P}_1 discretization.

Then an abstract sliding method is derived. It is based on a mortar-like formulation and an adapted conservation constrain. Its generic properties are established.

Following that, we study the special case of a $\mathbb{P}_1 - \mathbb{P}_0$ discretization of the velocity problem. We analyze the problem and establish an inf-sup condition. This is very educative since it leads discretization conditions for the Lagrange multiplier's mesh. Also, we show that the solution of the linear system is continuous according to the data, which is a stability result for our method.

Then, we explain briefly a simple ALE method which keeps the sliding interface Lagrangian. Finally, we write the explicit in time first-order integration scheme that we used.

The last section is dedicated to numerical tests. We use two configurations for the Sod shock tube as a sanity check. Then we test Caramanna's piston problem [10]. We propose the next test to check the behavior of the method when the contact zone changes with time. Finally, we present the result we obtain on a challenging problem defined in [24].

In order to fix ideas, we present shortly the proposed method and justify some choices that we made.

1.1. Sketch of the method

We consider the practical $\mathbb{P}_1 - \mathbb{P}_0$ method that we analyze in Section 4. Each fluid domain Ω_i is defined by its own polygonal mesh \mathcal{M}_i . One denotes by Γ_i , the sliding boundary of \mathcal{M}_i . However, since Γ_i 's are different polygonal lines, one defines Γ – a “mean” mesh – that describes the slide line. The Lagrangian scheme has the following structure:

$$\begin{aligned} \forall i, \forall j \in \mathcal{M}_i, \quad V_j'(t) &= \sum_{l \in \mathcal{L}_j} \int_l \mathbf{u}_i^* \cdot \mathbf{n}_{jl}, & M_j \mathbf{u}_j'(t) &= - \sum_{\substack{l \in \mathcal{L}_j \\ l \not\subset \Gamma_i}} \int_l p_j^* \mathbf{n}_{jl} - \sum_{l \subset \Gamma_i} \int_l p_j^* \mathbf{n}_{il}, \\ M_j'(t) &= 0, & M_j E_j'(t) &= - \sum_{\substack{l \in \mathcal{L}_j \\ l \not\subset \Gamma_i}} \int_l p_j^* \mathbf{u}_i^* \cdot \mathbf{n}_{jl} - \sum_{l \subset \Gamma_i} \int_l p_j^* \mathbf{u}_i^* \cdot \mathbf{n}_{il}, \end{aligned}$$

where j designate a generic cell of one of the meshes, V_j is its volume, M_j its constant mass, \mathbf{u}_j its velocity and E_j denotes its total energy density. \mathcal{L}_j is the set of linear edges of the polygonal cell j and thus \mathbf{n}_{jl} is the unit normal to $l \in \mathcal{L}_j$ outgoing from j . Observe that for momentum and total energy balance equations, fluxes sum are treated differently on the edges that belong to Γ_i . On these sliding edges the unit normal that is used is \mathbf{n}_{il} which is the normal to Γ outgoing from domain Ω_i .

To complete the scheme definition, it remains to define the fluxes \mathbf{u}_i^* and p_j^* that are related by an acoustic Riemann invariant that is to be taken in the weak sens: $p_j^* - p_j + (\rho c)_j (\mathbf{u}_i^* - \mathbf{u}_j) \cdot \mathbf{n} = 0$, where $\mathbf{n} = \mathbf{n}_{jl}$ or \mathbf{n}_{il} according to the edge location. Observe that knowing \mathbf{u}_i^* defines the p_j^* 's.

Writing both the conservation and the sliding constrains, one obtains the following saddle point problem that is satisfied by the velocity of the edges \mathbf{u}_i^* : find $(\mathbf{u}_1^*, \mathbf{u}_2^*, \lambda) \in \mathcal{V}$

$$\begin{aligned} \forall (\mathbf{v}_1, \mathbf{v}_2, \mu) \in \mathcal{V}, \quad \sum_i a_i(\mathbf{u}_i^*, \mathbf{v}_i) + \sum_i b_i(\mathbf{v}_i, \lambda) &= \sum_i l_i(\mathbf{v}_i) \\ \text{and} \quad \sum_i b_i(\mathbf{u}_i^*, \mu) &= 0, \end{aligned}$$

where setting $A_{jl} = (\rho c)_j \mathbf{n}_{jl} \otimes \mathbf{n}_{jl}$ and $A_{il} = (\rho c)_j \mathbf{n}_{il} \otimes \mathbf{n}_{il}$, one has

$$\begin{aligned} a_i(\mathbf{u}, \mathbf{v}) &= \sum_{j \in \mathcal{M}_i} \sum_{\substack{l \in \mathcal{L}_j \\ l \not\subset \Gamma_i}} \int_l \mathbf{v}^T A_{jl} \mathbf{u} + \sum_{j \in \mathcal{M}_i} \sum_{\substack{l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \int_l \mathbf{v}^T A_{il} \mathbf{u}, \\ b_i(\mathbf{v}, \mu) &= \sum_{j \in \mathcal{M}_i} \sum_{\substack{l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \int_l \mathbf{v} \cdot \mathbf{n}_{il} \mu, \end{aligned}$$

and

$$l_i(\mathbf{v}) = \sum_{j \in \mathcal{M}_i} \sum_{\substack{l \in \mathcal{L}_j \\ l \not\subset \Gamma_i}} \left(\int_l p_j \mathbf{v} \cdot \mathbf{n}_{jl} + \int_l \mathbf{v}^T A_{jl} \mathbf{u}_j \right) + \sum_{j \in \mathcal{M}_i} \sum_{\substack{l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \left(\int_l p_j \mathbf{v} \cdot \mathbf{n}_{il} + \int_l \mathbf{v}^T A_{il} \mathbf{u}_j \right).$$

Actually, the practical scheme that is analyzed in Section 4 is defined by choosing $\mathcal{V} = \mathbf{P}_1(\mathcal{E}_1)^d \times \mathbf{P}_1(\mathcal{E}_2)^d \times \mathbf{P}_0(\Gamma)$: the velocities \mathbf{u}_i^* are continuous piecewise linear functions defined on the edges set \mathcal{E}_i of the mesh \mathcal{M}_i and λ is piecewise constant on the mesh Γ . At this point, the integrals on l can be computed either exactly or using quadrature formulas. Nevertheless, observe that the integrals that define $b_j(\cdot, \cdot)$ are more tricky to compute since \mathbf{v} and μ are not living on the same mesh. The particular treatment that we choose is detailed in Section 4.1. Finally, $(\mathbf{u}_1^*, \mathbf{u}_2^*, \lambda)$ is solution of the obtained linear system. This ends this short presentation of the scheme we present in this paper.

Remark 1.1. One could question why we choose a multiplier based formulation to impose the sliding constrain. Actually, we have also tested a penalty formulation, which writes

$$\forall(\mathbf{v}_1, \mathbf{v}_2), \quad \sum_i a_i(\mathbf{u}_i^*, \mathbf{v}_i) + \frac{1}{\epsilon} \int_{\Gamma} (\mathbf{u}_1^* - \mathbf{u}_2^*) \mathbf{n} \otimes \mathbf{n} (\mathbf{v}_1 - \mathbf{v}_2) = \sum_i l_i(\mathbf{v}_i).$$

We did not pursue with this formulation for two reasons. First, it is easy to check that it is not conservative. The conservation error grows with ϵ . Second, as it is often the case with penalty methods, the choice of ϵ might be tricky to made and one can see that very small ϵ can create numerical locking. Here, it means that one might impose $\mathbf{u}_1 = \mathbf{u}_2$ on Γ instead of $\mathbf{u}_1 \cdot \mathbf{n} = \mathbf{u}_2 \cdot \mathbf{n}$.

2. CONTINUOUS SCHEME WITHOUT SLIDE LINES

Let us recall Euler equations in semi-Lagrangian coordinates³ in their integral form. Obviously it is only written in a formal way, since the clean mathematical form are the PDEs stated in [17, 27], but it is commonly used since it is well suited to derive finite-volume methods.

Let $d = 2$ denote the space dimension. Let $\Omega(t) \in \mathbb{R}^d$ be an open connected set which represents the fluid domain at any time $t \in \mathbb{R}^+$. For any $\omega(t) \subset \Omega(t)$ moving at fluid's velocity \mathbf{u} , one can write the Euler's system of conservation laws

$$\frac{d}{dt} \int_{\omega(t)} 1 = \int_{\partial\omega(t)} \mathbf{u} \cdot \mathbf{n}, \quad \left(= \int_{\omega(t)} \nabla \cdot \mathbf{u} \right), \quad \text{volume conservation} \quad (2.1)$$

$$\frac{d}{dt} \int_{\omega(t)} \rho = 0, \quad \text{mass conservation} \quad (2.2)$$

$$\frac{d}{dt} \int_{\omega(t)} \rho \mathbf{u} = - \int_{\partial\omega(t)} p \mathbf{n}, \quad \left(= - \int_{\omega(t)} \nabla p \right), \quad \text{momentum conservation} \quad (2.3)$$

$$\frac{d}{dt} \int_{\omega(t)} \rho E = - \int_{\partial\omega(t)} p \mathbf{u} \cdot \mathbf{n}, \quad \left(= - \int_{\omega(t)} \nabla \cdot p \mathbf{u} \right), \quad \text{total energy conservation} \quad (2.4)$$

where ρ is the mass density, E is the specific total energy and $p = p(\rho, e)$ is the pressure. The specific internal energy is given by $e := E - \frac{1}{2} \mathbf{u} \cdot \mathbf{u}$. The unitary outward normal to $\partial\omega$ is denoted by \mathbf{n} . In the following, c will denote the sound speed.

Finally, we introduce some notations.

- Let \mathcal{M} be a conformal mesh of the connected set Ω at time t .

³*i.e.*: Lagrange plus update.

- Let us denote by j a generic cell of \mathcal{M} and by l a generic edge.
- We define \mathcal{L}_j , the set of the edges l of the cell j .
- Let us define its dual: \mathcal{J}_l the set of cells j that share l as an edge.

In the following, we will consider that any edge of \mathcal{M} can be defined as $l = \mathbf{T}_l([0, 1])$ where \mathbf{T}_l is bijective and sufficiently regular, $\mathbf{T}_l \in C^1([0, 1])^d$ for instance.

The sliding method we propose in this paper relies strongly on the formulation that was shortly presented in [15]. For this reason, we recall here briefly its construction and its main properties.

We aim at writing a finite volume method, so we assume that in each cell j the velocity and pressure reconstructions $\mathbf{u}_j(\mathbf{x})$ and $p_j(\mathbf{x})$ are given at time t . Note that if $\mathbf{x} \notin j$, $\mathbf{u}_j(\mathbf{x}) = \mathbf{0}$ and $p_j(\mathbf{x}) = 0$.

Let us now introduce the unknowns \mathbf{u}^* , the edge velocity, and p_j^* the pressure imposed by cell j on its edges ($p_j^*(\mathbf{x}) = 0$ if $\mathbf{x} \notin \cup_{l \in \mathcal{L}_j} l$). The structure of the continuous scheme writes

$$\begin{aligned}
\forall j, \quad & \frac{d}{dt} \int_j 1 = \sum_{l \in \mathcal{L}_j} \int_l \mathbf{u}^* \cdot \mathbf{n}_{jl}, \\
& \frac{d}{dt} \int_j \rho = 0, \\
& \frac{d}{dt} \int_j \rho \mathbf{u} = - \sum_{l \in \mathcal{L}_j} \int_l p_j^* \mathbf{n}_{jl}, \\
& \frac{d}{dt} \int_j \rho E = - \sum_{l \in \mathcal{L}_j} \int_l p_j^* \mathbf{u}^* \cdot \mathbf{n}_{jl}. \tag{2.5}
\end{aligned}$$

Our aim is to compute \mathbf{u}^* and p_j^* 's to get a spatial discretization which is conservative and entropic.

Following [12, 17, 26], for instance, a local linearization of Euler's equations is performed by substituting acoustic Riemann invariants ($dp + \rho c \mathbf{d}\mathbf{u} \cdot \mathbf{n} = 0$) to the real ones. This approximation is commonly used and helpful to write entropic, continuous in time schemes.

We choose now to write the acoustic Riemann invariant at any point \mathbf{x} of each edge l of the mesh along $\mathbf{n}_{jl}(\mathbf{x})$, where $\mathbf{n}_{jl}(\mathbf{x})$ denotes the normal to l , outgoing from cell j :

$$\forall j, \forall l \in \mathcal{L}_j, \forall \mathbf{x} \in l, \quad p_j^*(\mathbf{x}) - p_j(\mathbf{x}) + (\rho c)_j (\mathbf{u}^*(\mathbf{x}) - \mathbf{u}_j(\mathbf{x})) \cdot \mathbf{n}_{jl}(\mathbf{x}) = 0. \tag{2.6}$$

We will now write a weak form of (2.6). Let us define $\mathcal{E} := \{\mathbf{x} \in \mathbb{R}^d / \exists l \text{ s.t. } \mathbf{x} \in l\}$ the one dimensional space of edges points. Also, we define $L_j^2(\mathcal{E}) := \{\phi \in L^2(\mathcal{E}), \text{ s.t. } \phi|_{\mathcal{E} \setminus \partial j} = 0\}$, the space of L^2 functions on the edges of cell j that vanishes on edges that do not belong to \mathcal{L}_j .

We assume $\mathbf{u}_j|_{\mathcal{E}} \in L_j^2(\mathcal{E})^d$, $p_j|_{\mathcal{E}} \in L_j^2(\mathcal{E})$, and $p_j^* \in L_j^2(\mathcal{E})$. Also, we assume $\mathbf{u}^* \in L^2(\mathcal{E})^d$ and that there exist two strictly positive real constants m and M such that $\forall j$, $0 < m < (\rho c)_j < M$.

Introducing matrices $A_{jl}(\mathbf{x}) := (\rho c)_j \mathbf{n}_{jl} \otimes \mathbf{n}_{jl}(\mathbf{x})$, which are positive and symmetric, but only of rank 1, we can write a weak form of (2.6) as

$$\forall j, \forall l \in \mathcal{L}_j, \forall \mathbf{v} \in L^2(\mathcal{E})^d, \quad \int_l p_j^* \mathbf{n}_{jl} \cdot \mathbf{v} + \int_l^t \mathbf{v} A_{jl} \mathbf{u}^* = \int_l p_j \mathbf{n}_{jl} \cdot \mathbf{v} + \int_l^t \mathbf{v} A_{jl} \mathbf{u}_j. \tag{2.7}$$

2.1. Variational problem

It is easy to check that the problem of finding $(\mathbf{u}^*, (p_j^*)_{j \geq 0})$ solution to (2.7) is under-determined: for any function \mathbf{u}^* in (2.6), p_j^* 's are defined. Thus, we need to add equations to close the system.

It allows us to impose a conservativity constraint which will be sufficient to compute \mathbf{u}^* and p_j^* 's:

$$\forall \mathbf{v} \in L^2(\mathcal{E})^d, \quad \sum_j \sum_{l \in \mathcal{L}_j} \int_l p_j^* \mathbf{n}_{jl} \cdot \mathbf{v} = 0. \tag{2.8}$$

Observe that (2.8) imposes weakly the equality of $(p_j^*)_{j \in \mathcal{J}_l}$ along the common edges l of cells j . So, to derive the continuous problem, it would be enough to consider only one function $p^* \in L^2(\mathcal{E})$. However, distinguishing $(p_j^*)_{j \geq 0}$ is necessary to obtain discrete schemes.

So, to compute grid velocity and fluxes involved in (2.5), it remains to solve problem (2.7)–(2.8). Substituting (2.7) in (2.8), one obtains the variational problem satisfied by \mathbf{u}^*

$$\forall \mathbf{v} \in L^2(\mathcal{E})^d, \quad \sum_j \sum_{l \in \mathcal{L}_j} \int_l {}^t \mathbf{v} A_{jl} \mathbf{u}^* = \sum_j \sum_{l \in \mathcal{L}_j} \int_l p_j \mathbf{n}_{jl} \cdot \mathbf{v} + \sum_j \sum_{l \in \mathcal{L}_j} \int_l {}^t \mathbf{v} A_{jl} \mathbf{u}_j. \quad (2.9)$$

This is a linear problem of the form, find $\mathbf{u}^* \in L^2(\mathcal{E})^d$, such that

$$\forall \mathbf{v} \in L^2(\mathcal{E})^d, \quad a(\mathbf{u}^*, \mathbf{v}) = l(\mathbf{v}),$$

where $l(\cdot)$ is a linear continuous form on $L^2(\mathcal{E})^d$ and $a(\cdot, \cdot)$ is a bilinear continuous and symmetric form on $L^2(\mathcal{E})^d \times L^2(\mathcal{E})^d$.

Remark 2.1 (uniqueness). It is easy to check that the kernel of $a(\cdot, \cdot)$ does not reduce to $\mathbf{0} \in L^2(\mathcal{E})^d$. Actually, the set of tangential velocities along edges is the kernel of $a(\cdot, \cdot)$.

The correct space to look for \mathbf{u}^* is not $L^2(\mathcal{E})^d$, but the space of normal velocities to the edge, $\mathcal{N} = \{\mathbf{w} \in L^2(\mathcal{E})^d, \text{ such that } \forall l, \mathbf{w}|_l = w_l \mathbf{n}_l, \text{ with } w_l \in L^2(l)\}$. This is expected since only normal velocity contributes to (2.5) for instance.

Since we assumed that there exists $m \in \mathbb{R}^+$ such that $\forall j, m < (\rho c)_j$, it is easy to show that $a(\cdot, \cdot)$ is coercive on \mathcal{N} . So, the problem is well-posed (using Lax–Milgram’s lemma) on this functional space.

However \mathcal{N} is more complex to approximate than $L^2(\mathcal{E})^d$. Thus we will continue with $L^2(\mathcal{E})^d$ since our purpose in this paper is to describe a slide lines method and not to deal with curved edges and non-linear edges velocities. Also, we recall that the final discrete problem is stable, that is, it has a unique solution that continuously depends on the data.

2.2. Properties

We now recall some properties of the abstract scheme that were established in [15].

Property 1. *The scheme is conservative in volume, mass, momentum and total energy.*

Property 2. *The continuous in time scheme is entropic if all cell-centered data are constants in each cell (i.e. if no reconstruction is performed).*

2.3. A \mathbb{P}_1 discretization of the continuous scheme without slide lines

In order to simplify the reading of this paper, and to fix some ideas, we will now describe a practical scheme that fits the abstract framework we just presented. As it is often the case for Galerkin methods, we choose a finite dimension subspace $\mathbb{V}_h \subset L^2(\mathcal{E})^d$ so that the discrete problem is: find $\mathbf{u}_h^* \in \mathbb{V}_h$ such that

$$\forall \mathbf{v}_h \in \mathbb{V}_h, \quad a(\mathbf{u}_h^*, \mathbf{v}_h) = l(\mathbf{v}_h). \quad (2.10)$$

In this paragraph, we give some details for the choice $\mathbb{V}_h := \mathbb{P}_1(\mathcal{E})^d$.

We make the common hypothesis that edges of the mesh are straight all along the simulation, so that each conformal transformation \mathbf{T}_l is affine. Also we want the mesh \mathcal{M} to remain conformal. So, we choose $\mathbf{u}_h^* \in \mathbb{P}_1(\mathcal{E})^d$, the space of continuous piecewise affine velocities on \mathcal{E} . Denoting \mathcal{F} the set of degrees of freedom (here located at the vertices of \mathcal{M}), we recall that

$$\mathbb{P}_1(\mathcal{E}) = \left\{ v \in C^0(\mathcal{E}), v = \sum_{\alpha \in \mathcal{F}} v_\alpha w_\alpha, \quad v_\alpha \in \mathbb{R} \right\},$$

where w_α is defined as $\forall \alpha, \beta \in \mathcal{F}, w_\alpha(\mathbf{x}_\beta) = \delta_{\alpha\beta}$ and $\forall \alpha \in \mathcal{F}, \forall l \in \mathcal{M}, w_\alpha|_l$ is affine.

2.3.1. Analysis of the \mathbb{P}_1 discretization

As shown in Section 2.1, $a(\cdot, \cdot)$ is not coercive on $L^2(\mathcal{E})^d$. However its coercivity can be established on $\mathbb{P}_1(\mathcal{E})^d$. This is the aim of this paragraph.

Here and in the following, to avoid the proliferation of constants, we will use the notation $x \lesssim y$ (resp. $x \gtrsim y$) to represent the inequality $x \leq Cy$ (resp. $x \geq cy$, with $C, c > 0$ independent of the mesh size). Writing $x \simeq y$ will signify that there exists two constants $C > c > 0$ such that $cx \leq y \leq Cx$. Writing $x \gtrsim 1$ (respectively $x \lesssim 1$ and $x \simeq 1$) will signify the existence of positive constants such that $x \geq c$ (resp. $x \leq C$ and $c \leq x \leq C$).

We recall that \mathcal{F} denotes the set of degrees of freedom of the space $\mathbb{P}_1(\mathcal{E})$, which can be identified with the set of vertices of \mathcal{E} . For all elements α of \mathcal{F} we denote by $\mathbf{x}_\alpha = (x_{\alpha,1}, x_{\alpha,2})$ the vector of coordinates of the corresponding vertex. For all $\alpha \in \mathcal{F}$ we denote by \mathcal{L}_α the set of edges that share \mathbf{x}_α as a vertex. We make the following assumptions on the mesh: there exist two angles $0 < \check{\vartheta} < \hat{\vartheta} < \pi$ such that for all $\alpha \in \mathcal{F}$ there exist two edges $l_1, l_2 \in \mathcal{L}_\alpha$ such that the angle $\vartheta(l_1, l_2)$ verifies $\check{\vartheta} \leq \vartheta(l_1, l_2) \leq \hat{\vartheta}$. Moreover there exists a constant τ such that for any $l_1, l_2 \in \mathcal{L}_\alpha$ we have $h_{l_1}/h_{l_2} < \tau$. This last assumption allows us to define a mesh-size parameter h_α associated with a vertex, defined, for instance as $h_\alpha = \min_{l \in \mathcal{L}_\alpha} h_l$.

Letting $\|\cdot\|_{0,\mathcal{E}} = (\int_{\mathcal{E}} |\cdot|^2)^{\frac{1}{2}}$ denote the norm of $L^2(\mathcal{E})^d$ for both $d = 1, 2$, $|\cdot|$ denoting respectively the absolute value and the euclidian norm, we have the following lemma.

Lemma 2.2. *There exist a constant σ , depending on $\check{\vartheta}$ and $\hat{\vartheta}$ as well as on ρc , such that for all $\mathbf{u}_h \in \mathbb{P}_1(\mathcal{E})^d$, it holds*

$$a(\mathbf{u}_h, \mathbf{u}_h) \geq \sigma \|\mathbf{u}_h\|_{0,\mathcal{E}}^2.$$

Proof. Let $\mathbf{u}_h = (u_h^1, u_h^2) \in \mathbb{P}_1(\mathcal{E})^d$. For each $l \in \mathcal{L} = \cup_j \mathcal{L}_j$ let $\mathbf{n}_l = (n_l^1, n_l^2)$ be a unit normal, pointing in one of the two possible directions. By the definition of a , we have

$$a(\mathbf{u}_h, \mathbf{u}_h) \geq 2m \sum_{l \in \mathcal{L}} \int_l |\mathbf{u}_h \cdot \mathbf{n}_l|^2,$$

where we recall that $\forall j, 0 < m < (\rho c)_j$.

Since $\mathbf{u}_h \in \mathbb{P}_1(\mathcal{E})^d$, the function $\mathbf{u}_h \cdot \mathbf{n}_l$ is linear on l . By standard results on linear functions, letting α_1^l and α_2^l denote the degrees of freedom corresponding to the two vertices of l , we have

$$\int_l |\mathbf{u}_h \cdot \mathbf{n}_l|^2 \geq Ch_l (|\mathbf{u}_h(\mathbf{x}_{\alpha_1^l}) \cdot \mathbf{n}_l|^2 + |\mathbf{u}_h(\mathbf{x}_{\alpha_2^l}) \cdot \mathbf{n}_l|^2).$$

Then we clearly have

$$a(\mathbf{u}_h, \mathbf{u}_h) \gtrsim \sum_{\alpha \in \mathcal{F}} \sum_{l \in \mathcal{L}_\alpha} h_l (|\mathbf{u}_h(\mathbf{x}_{\alpha_1^l}) \cdot \mathbf{n}_l|^2 + |\mathbf{u}_h(\mathbf{x}_{\alpha_2^l}) \cdot \mathbf{n}_l|^2). \quad (2.11)$$

Let us then concentrate on one degree of freedom α . Letting $l_1, l_2 \in \mathcal{L}_\alpha$ be the two edges given by the assumption on the mesh and $\mathbf{n}_{l_1} = (n_{l_1}^1, n_{l_1}^2)$ and $\mathbf{n}_{l_2} = (n_{l_2}^1, n_{l_2}^2)$ the corresponding normals, we have that $\mathbf{u}_h(\mathbf{x}_\alpha) = (u_h^1(\mathbf{x}_\alpha), u_h^2(\mathbf{x}_\alpha))$ verifies

$$\begin{pmatrix} n_{l_1}^1 & n_{l_1}^2 \\ n_{l_2}^1 & n_{l_2}^2 \end{pmatrix} \begin{pmatrix} u_h^1(\mathbf{x}_\alpha) \\ u_h^2(\mathbf{x}_\alpha) \end{pmatrix} = \begin{pmatrix} \mathbf{u}_h(\mathbf{x}_\alpha) \cdot \mathbf{n}_{l_1} \\ \mathbf{u}_h(\mathbf{x}_\alpha) \cdot \mathbf{n}_{l_2} \end{pmatrix}.$$

Observing that the determinant of the matrix on the left hand side is $n_{l_1}^1 n_{l_2}^2 - n_{l_1}^2 n_{l_2}^1 = \sin(\vartheta(l_1, l_2))$, it is easy to conclude that $|u_h^1(\mathbf{x}_\alpha)|^2 + |u_h^2(\mathbf{x}_\alpha)|^2 \lesssim \sum_{l \in \mathcal{L}_\alpha} (|\mathbf{u}_h(\mathbf{x}_{\alpha_1^l}) \cdot \mathbf{n}_l|^2 + |\mathbf{u}_h(\mathbf{x}_{\alpha_2^l}) \cdot \mathbf{n}_l|^2)$.

Recalling that for any $l_1, l_2 \in \mathcal{L}_\alpha$ the ratio between the lengths of l_1 and l_2 is bounded by a constant τ we immediately get that for any $l \in \mathcal{L}_\alpha$

$$h_\alpha |\mathbf{u}_h(\mathbf{x}_\alpha)|^2 \lesssim \sum_{l \in \mathcal{L}_\alpha} h_l (|\mathbf{u}_h(\mathbf{x}_{\alpha_1^l}) \cdot \mathbf{n}_l|^2 + |\mathbf{u}_h(\mathbf{x}_{\alpha_2^l}) \cdot \mathbf{n}_l|^2). \quad (2.12)$$

We now use once again the linearity of u_h^1 and u_h^2 . This allows us to write

$$\|\mathbf{u}_h\|_{0,\mathcal{E}}^2 = \sum_{l \in \mathcal{L}} \|\mathbf{u}\|_{0,l}^2 \lesssim \sum_{l \in \mathcal{L}} h_l (|\mathbf{u}_h(\mathbf{x}_{\alpha_1^l})|^2 + |\mathbf{u}_h(\mathbf{x}_{\alpha_2^l})|^2) \lesssim \sum_{\alpha \in \mathcal{F}} h_\alpha |\mathbf{u}_h(\mathbf{x}_\alpha)|^2. \quad (2.13)$$

In view of (2.11), combining (2.12) and (2.13) yields the thesis. \square

Proposition 2.3. *The problem (2.10) has a unique solution in $\mathbb{V}_h = \mathbb{P}_1(\mathcal{E})^d$.*

Proof. According to Lemma 2.2, for any valid mesh $a(\cdot, \cdot)$ is coercive on \mathbb{V}_h . Thus, all hypothesis of Lax–Milgram’s lemma are satisfied in this approximation space. \square

2.3.2. Properties of the \mathbb{P}_1 scheme

Property 3. *The discrete scheme is conservative in volume, mass, momentum and total energy. Moreover, the semi-discrete continuous in time scheme is entropic for piecewise constant data in cells.*

Proof. Since \mathbb{V}_h is a conformal subspace of $L^2(\mathcal{E})^d$, properties 1 and 2 hold. \square

Finally, we precise the form of the obtained linear system that one has to solve to compute the grid velocity. Standard calculations for Galerkin-like methods allow us to rewrite the problem (2.10) as

$$\forall \alpha \in \mathcal{F}, \quad \sum_{\beta \in \mathcal{F}} \left(\sum_j \sum_{l \in \mathcal{L}_j} A_{jl} \int_l w_\alpha w_\beta \right) \mathbf{u}_\beta = \sum_j \sum_{l \in \mathcal{L}_j} p_j \mathbf{n}_{jl} \int_l w_\alpha + \sum_j \sum_{l \in \mathcal{L}_j} A_{jl} \mathbf{u}_j \int_l w_\alpha,$$

where $\int_l w_\alpha = \mathbf{1}_l(\mathbf{x}_\alpha) \frac{|l|}{2}$, $\int_l w_\alpha w_\beta = \mathbf{1}_l(\mathbf{x}_\alpha) \mathbf{1}_l(\mathbf{x}_\beta) \left(\frac{|l|}{3} \delta_{\alpha,\beta} + \frac{|l|}{6} (1 - \delta_{\alpha,\beta}) \right)$ and $\mathbf{1}_l$ is the indicator function of edge l . One observes that the global matrix associated with this linear system is not $d \times d$ -bloc diagonal as it is the case in [17] or [26] because of $\int_l w_\alpha w_\beta$ terms. However it is sparse, symmetric and positive definite and should have good condition number since it is some kind of L^2 -projection matrix.

Observe that approaching $\int_l w_\alpha w_\beta \approx \mathbf{1}_l(\mathbf{x}_\alpha) \frac{|l|}{2} \delta_{\alpha,\beta}$ – trapezoid formula –, one retrieves the Eucclhyd’s scheme [26]. It is also possible to recover Glace [17] scheme using a \mathbb{P}_0 approximation on the edges dual-mesh (velocity is chosen constant on the set of half-edges connected to each vertex).

3. CONTINUOUS SCHEME WITH SLIDE LINES

In this paragraph, we will establish a slide lines framework for cell-centered semi-Lagrangian hydro-schemes. To do so, we will proceed formally, not taking care of the well-posedness of the abstract problems that we will introduce. We recall that our objective is to deal only with polygonal cells, but the abstract framework we will establish is a powerful tool for the understanding of the method as well as to derive new schemes.

In what follows, we restrict ourselves to the problem of two fluids sliding along each other. This simplifies notations without any *a priori* loss of generality. Let $i \in \{1, 2\}$ denote the domain number. A conformal mesh \mathcal{M}_i is associated to each domain Ω_i . Let Γ denote the slide line and let Γ_i be the set of edges of \mathcal{E}_i sliding on Γ . Let us define $\mathring{\mathcal{E}}_i = \mathcal{E}_i \setminus \Gamma_i$ the set of edges point that does not belong to the sliding interface – see Figure 1.

To model perfect sliding, one requires $\mathbf{u}^* \cdot \mathbf{n}$ to be continuous across Γ . Denoting by \mathbf{u}_i^* the grid velocity of domain i , a weak formulation of the constraint reads

$$\forall \mu \in L^2(\Gamma), \quad \int_\Gamma (\mathbf{u}_1^* - \mathbf{u}_2^*) \cdot \mathbf{n} \mu = 0.$$

This is a classical formulation that is clearly linked to [3, 5, 9], which studied numerical methods to ensure weak continuity of quantities for non-matching grids in the context of domain decomposition methods.

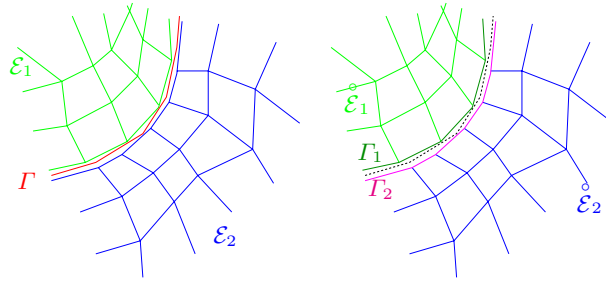


FIGURE 1. Configuration at slide line: straight edges example.

Denoting \mathbf{n} , the continuous unit normal to Γ , let us now introduce the constrained velocity space

$$\mathcal{C} := \left\{ (\mathbf{v}_1, \mathbf{v}_2) \in L^2(\mathcal{E}_1)^d \times L^2(\mathcal{E}_2)^d, \text{ such that } \forall \mu \in L^2(\Gamma), \int_{\Gamma} (\mathbf{v}_1 - \mathbf{v}_2) \cdot \mathbf{n} \mu = 0 \right\}.$$

\mathcal{C} is the space of admissible sliding grid velocities. To compute grids velocities in a similar way as presented in Section 2, we have to precise conservation constraints and choices for acoustic Riemann invariant directions at the interface Γ .

3.1. The choice of the normals

When dealing with polygonal meshes, the geometry of the sliding interface is not defined precisely: geometries of Γ and Γ_i 's may differ – see Figure 1 for instance.

This has many issues that will be discussed in this paper. An important one is its direct impact on the conservation constraint equations. In few words: it is necessary to consider the same normal for both domains at the interface for fluxes computation.

Building this method, it seems necessary to consider a common normal for flux calculation to achieve conservation of momentum. The choice we made is to consider the normal to the interface Γ .

In the following, we will consider the normal to Γ , normals to Γ_1 and Γ_2 will be useless for interface fluxes calculation. Let us define \mathbf{n}_{il} , the unit normal to Γ , viewed by the edge $l \subset \Gamma_i$ and outgoing from Ω_i . Observe that in the general case, \mathbf{n}_{il} is not normal to l .

3.2. Scheme structure

Since the edges treatment changes at the interface, we precise the differences in the scheme structure with regard to the non-sliding case (2.5). Indeed, one has to be careful on the treatment of Γ_i 's edges. We obtain the following scheme structure:

$$\forall i, \forall j \in \mathcal{M}_i, \quad \frac{d}{dt} \int_j 1 = \sum_{l \in \mathcal{L}_j} \int_l \mathbf{u}_i^* \cdot \mathbf{n}_{jl}, \quad (3.1)$$

$$\frac{d}{dt} \int_j \rho = 0, \quad (3.2)$$

$$\frac{d}{dt} \int_j \rho \mathbf{u} = - \sum_{\substack{l \in \mathcal{L}_j \\ l \not\subset \Gamma_i}} \int_l p_j^* \mathbf{n}_{jl} - \sum_{\substack{l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \int_l p_j^* \mathbf{n}_{il}, \quad (3.3)$$

$$\frac{d}{dt} \int_j \rho E = - \sum_{\substack{l \in \mathcal{L}_j \\ l \not\subset \Gamma_i}} \int_l p_j^* \mathbf{u}_i^* \cdot \mathbf{n}_{jl} - \sum_{\substack{l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \int_l p_j^* \mathbf{u}_i^* \cdot \mathbf{n}_{il}. \quad (3.4)$$

One notes in (3.3) and (3.4) the difference of edge treatments according to their nature (interface or inner edge). Also, since \mathbf{u}_i^* is the velocity of the edges of the mesh \mathcal{M}_i , volume variation is given *exactly* by (3.1). If one desires to update volume according to fluxes, one could alternatively use the same kind of flux decomposition as (3.3) or (3.4).

3.3. Conservation constraint

Conservation constraint for the sliding problem reads

$$\forall(\mathbf{v}_1, \mathbf{v}_2) \in \mathcal{C}, \quad \sum_i \sum_{\substack{j \in \mathcal{M}_i \\ l \in \mathcal{L}_j \\ l \not\subset \Gamma_i}} \int_l p_j^* \mathbf{v}_i \cdot \mathbf{n}_{jl} + \sum_i \sum_{\substack{j \in \mathcal{M}_i \\ l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \int_l p_j^* \mathbf{v}_i \cdot \mathbf{n}_{il} = 0. \quad (3.5)$$

The first term of the sum traduces the conservation for inner edges of each mesh while the second term applies for the exchanges between both domains.

One notices that again, relation (3.5) imposes the weak equality of pressures (p_j^* 's) along the inner edges and along the slide line. However, as it was the case for the non-sliding case, this artificial distinction is useful to establish discrete schemes.

3.4. Acoustic Riemann invariant relations

We must now precise the relations linking cell-centered quantities and edge values. Again, we use the acoustic Riemann invariant relations since it is a simple way to build entropic scheme in their continuous in time form.

We distinguish two cases: internal edges and interface edges. For internal edges, we write: $\forall i, \forall j \in \mathcal{M}_i, \forall l \in \mathcal{L}_j$ such that $l \not\subset \Gamma_i, \forall \mathbf{v}_j \in L^2(l)^d$,

$$\int_l p_j^* \mathbf{v}_i \cdot \mathbf{n}_{jl} + \int_l {}^t \mathbf{v}_i A_{jl} \mathbf{u}_i^* = \int_l p_j \mathbf{v}_i \cdot \mathbf{n}_{jl} + \int_l {}^t \mathbf{v}_i A_{jl} \mathbf{u}_j, \quad (3.6)$$

where $A_{jl} = (\rho c)_j \mathbf{n}_{jl} \otimes \mathbf{n}_{jl}$. And for interface edges, we have the slightly different expression due to the particular normal choice: \mathbf{n}_{il} . $\forall i, \forall j \in \mathcal{M}_i, \forall l \in \mathcal{L}_j$ such that $l \subset \Gamma_i, \forall \mathbf{v}_j \in L^2(l)^d$,

$$\int_l p_j^* \mathbf{v}_i \cdot \mathbf{n}_{il} + \int_l {}^t \mathbf{v}_i A_{il} \mathbf{u}_i^* = \int_l p_j \mathbf{v}_i \cdot \mathbf{n}_{il} + \int_l {}^t \mathbf{v}_i A_{il} \mathbf{u}_j, \quad (3.7)$$

where $A_{il} = (\rho c)_j \mathbf{n}_{il} \otimes \mathbf{n}_{il}$, the cell j satisfies $l = \partial j \cap \Gamma_i$.

3.5. Constrained problem

It is now sufficient to inject (3.6) and (3.7) in (3.5) to obtain the grid velocity problem: find $(\mathbf{u}_1^*, \mathbf{u}_2^*) \in \mathcal{C}$ such that $\forall(\mathbf{v}_1, \mathbf{v}_2) \in \mathcal{C}$,

$$\sum_i \sum_{\substack{j \in \mathcal{M}_i \\ l \in \mathcal{L}_j \\ l \not\subset \Gamma_i}} \int_l {}^t \mathbf{v}_i A_{jl} \mathbf{u}_i^* + \sum_i \sum_{\substack{j \in \mathcal{M}_i \\ l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \int_l {}^t \mathbf{v}_i A_{il} \mathbf{u}_i^* = \sum_i \sum_{\substack{j \in \mathcal{M}_i \\ l \in \mathcal{L}_j \\ l \not\subset \Gamma_i}} \left(\int_l p_j \mathbf{v}_i \cdot \mathbf{n}_{jl} + \int_l {}^t \mathbf{v}_i A_{jl} \mathbf{u}_j \right) \quad (3.8)$$

$$+ \sum_i \sum_{\substack{j \in \mathcal{M}_i \\ l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \left(\int_l p_j \mathbf{v}_i \cdot \mathbf{n}_{il} + \int_l {}^t \mathbf{v}_i A_{il} \mathbf{u}_j \right). \quad (3.9)$$

This problem in the constrained space has the canonical form, find $(\mathbf{u}_1^*, \mathbf{u}_2^*) \in \mathcal{C}$, such that

$$\forall(\mathbf{v}_1, \mathbf{v}_2) \in \mathcal{C}, \quad \sum_i a_i(\mathbf{u}_i^*, \mathbf{v}_i) = \sum_i l_i(\mathbf{v}_i), \quad (3.10)$$

where the forms a_i are bilinear and continuous and l_i are linear and continuous.

3.6. Saddle point problem

Let $\lambda \in L^2(\Gamma)$ a Lagrange multiplier. Let $\mathcal{V} := L^2(\mathcal{E}_1)^d \times L^2(\mathcal{E}_2)^d \times L^2(\Gamma)$. The problem, find $(\mathbf{u}_1^*, \mathbf{u}_2^*) \in \mathcal{C}$ solution of (3.10) is equivalent to find $(\mathbf{u}_1^*, \mathbf{u}_2^*, \lambda) \in \mathcal{V}$ solution of

$$\begin{aligned} \forall (\mathbf{v}_1, \mathbf{v}_2, \mu) \in \mathcal{V}, \quad \sum_i a_i(\mathbf{u}_i^*, \mathbf{v}_i) + \sum_i b_i(\mathbf{v}_i, \lambda) &= \sum_i l_i(\mathbf{v}_i) \\ \text{and} \quad \sum_i b_i(\mathbf{u}_i^*, \mu) &= 0, \end{aligned} \quad (3.11)$$

where the continuous bilinear form b_i is defined as

$$\begin{aligned} b_i : L^2(\mathcal{E}_i)^d \times L^2(\Gamma) &\mapsto \mathbb{R}, \\ (\mathbf{v}, \mu) \rightarrow b_i(\mathbf{v}, \mu) &:= \sum_{j \in \mathcal{M}_i} \sum_{\substack{l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \int_l \mathbf{v} \cdot \mathbf{n}_{il} \mu. \end{aligned} \quad (3.12)$$

Remark 3.1 (λ interpretation). Observe that, in (3.11), λ plays the role of an external mean pressure imposed by each domain on the other.

Remark 3.2. Similarly to the non-sliding case, a_i forms are not coercive on $L^2(\mathcal{E}_i)^d$. The set of tangential velocities along the edges is a_i 's kernel. However, choosing $\mathcal{V} := \mathcal{N}_1 \times \mathcal{N}_2 \times L^2(\Gamma)$, where \mathcal{N}_i are defined following Remark 2.1, one can easily show that (3.11) as a unique solution $(\mathbf{u}_1^*, \mathbf{u}_2^*, \lambda) \in \mathcal{V}$. Also, $(\mathbf{u}_1^*, \mathbf{u}_2^*)$ is the unique solution of (3.10) in $\mathcal{C} := \{(\mathbf{v}_1, \mathbf{v}_2) \in \mathcal{N}_1 \times \mathcal{N}_2, \text{ such that } , \forall \mu \in L^2(\Gamma), \int_\Gamma (\mathbf{v}_1 - \mathbf{v}_2) \cdot \mathbf{n} \mu = 0\}$.

3.7. Properties and remark

We now establish some properties of the continuous sliding method that will remain true at the discrete level. Also, we make some remarks on the lack of entropy production of the method at the sliding interface.

3.7.1. Conservation properties

We focus now on the conservation properties of the scheme (3.1)–(3.4) and (3.11).

Property 4. *The scheme (3.1)–(3.4) and (3.11) is conservative in mass, momentum and total energy.*

Proof. One proceeds the same way as for the case without slide lines (see Sect. 2.2).

- Mass: fluxes are null by construction.
- Momentum: one writes the sum of momentum variations

$$\sum_i \sum_{j \in \mathcal{M}_i} \frac{d}{dt} \int_j \rho \mathbf{u} = - \sum_i \sum_{j \in \mathcal{M}_i} \sum_{\substack{l \in \mathcal{L}_j \\ l \not\subset \Gamma_i}} \int_l p_j^* \mathbf{n}_{jl} - \sum_i \sum_{j \in \mathcal{M}_i} \sum_{l \in \mathcal{L}_j} \int_l p_j^* \mathbf{n}_{il}.$$

This sum is null since one can choose $\mathbf{v}_i = \mathbf{e}_k, \forall k \in \{1, \dots, d\}$, in (3.5).

- Total energy: the variation is given by

$$\sum_i \sum_{j \in \mathcal{M}_i} \frac{d}{dt} \int_j \rho E = - \sum_i \sum_{j \in \mathcal{M}_i} \sum_{\substack{l \in \mathcal{L}_j \\ l \not\subset \Gamma_i}} \int_l p_j^* \mathbf{u}_i^* \cdot \mathbf{n}_{jl} - \sum_i \sum_{j \in \mathcal{M}_i} \sum_{\substack{l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \int_l p_j^* \mathbf{u}_i^* \cdot \mathbf{n}_{il}.$$

This is obviously 0 since one can choose $(\mathbf{v}_1, \mathbf{v}_2) = (\mathbf{u}_1^*, \mathbf{u}_2^*)$ in (3.5). □

Remark 3.3 (Volume conservation). Summing upon all cells of all meshes the volume variation (3.1), elementary calculations lead to

$$\sum_i \sum_{j \in \mathcal{M}_i} \frac{d}{dt} \int_j 1 = \sum_i \sum_{j \in \mathcal{M}_i} \sum_{\substack{l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \int_l \mathbf{u}_i^* \cdot \mathbf{n}_{jl} = \sum_i \sum_{j \in \mathcal{M}_i} \sum_{\substack{l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \int_l \mathbf{u}_i^* \cdot (\mathbf{n}_{jl} - \mathbf{n}_{il}),$$

later term being obtained by choosing $\mu = 1$ in the second equation of (3.11). This shows as expected that the volume conservation error is located in the sliding interface and is strongly related to $\mathbf{n}_{jl} - \mathbf{n}_{il}$.

3.7.2. Galilean invariance

Let us now focus on this property, since it might not appear obvious that it is satisfied by the proposed sliding method.

Property 5. *The slide line method (3.1)–(3.4) and (3.11) is Galilean invariant.*

Proof. Let $\mathbf{u}_{\mathcal{G}} \in \mathbb{R}^d$, let $(\mathbf{u}_1^*, \mathbf{u}_2^*) \in \mathcal{C}$ be solution of (3.10). Then $(\mathbf{u}_1^* + \mathbf{u}_{\mathcal{G}}, \mathbf{u}_2^* + \mathbf{u}_{\mathcal{G}}) \in \mathcal{C}$ is solution of (3.10) when replacing all \mathbf{u}_j in l_i by $\mathbf{u}_j + \mathbf{u}_{\mathcal{G}}$ thanks to the linearity of the operators. Obviously, p_j^* 's remain unchanged due to (3.6) and (3.7).

So, the obtained momentum is the same in both frames. Also, total energy variation is changed by the expected inertial term

$$\mathbf{u}_{\mathcal{G}} \cdot \left(- \sum_{\substack{l \in \mathcal{L}_j \\ l \not\subset \Gamma_i}} \int_l p_j^* \mathbf{n}_{jl} - \sum_{\substack{l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \int_l p_j^* \mathbf{n}_{il} \right).$$

The proof is ended by noticing that since the meshes are displaced at velocities \mathbf{u}_i^* and $\mathbf{u}_i^* + \mathbf{u}_{\mathcal{G}}$ and since $\mathbf{u}_{\mathcal{G}}$ is constant, volumes remain the same in both case. The method being Lagrangian (M_j is constant), densities in both calculation are identical. \square

3.7.3. Entropy production

Actually, we cannot prove that the continuous in time scheme is entropic but we establish a local geometric-related indicator. Integrating the Gibbs's formula ($TdS = pd\tau + de$) in the generic cell j of mesh \mathcal{M}_i , one has

$$V_j \rho_j T_j \frac{d}{dt} S_j = \frac{d}{dt} \int_j \rho_j E_j - \mathbf{u}_j \cdot \frac{d}{dt} \int_j \rho_j \mathbf{u}_j + p_j \frac{d}{dt} \int_j 1.$$

Using (3.1)–(3.4) and the fact that \mathbf{u}_j and p_j are constant on l , one can write after simple calculations

$$V_j \rho_j T_j \frac{d}{dt} S_j = \sum_{\substack{l \in \mathcal{L}_j \\ l \not\subset \Gamma_i}} \int_l (p_{jl}^* (\mathbf{u}_j - \mathbf{u}_i^*) \cdot \mathbf{n}_{jl} + p_j \mathbf{u}_i^* \cdot \mathbf{n}_{jl}) + \sum_{\substack{l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \int_l (p_{jl}^* (\mathbf{u}_j - \mathbf{u}_i^*) \cdot \mathbf{n}_{il} + p_j \mathbf{u}_i^* \cdot \mathbf{n}_{jl}).$$

Now, choosing $\mathbf{v}_i = \mathbf{u}_j - \mathbf{u}_i^*$ in (3.6) and (3.7), we obtain after some rearrangements

$$\begin{aligned} V_j \rho_j T_j \frac{d}{dt} S_j &= \sum_{\substack{l \in \mathcal{L}_j \\ l \not\subset \Gamma_i}} \int_l^t (\mathbf{u}_j - \mathbf{u}_i^*) A_{jl} (\mathbf{u}_j - \mathbf{u}_i^*) + \sum_{\substack{l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \int_l^t (\mathbf{u}_j - \mathbf{u}_i^*) A_{il} (\mathbf{u}_j - \mathbf{u}_i^*) \\ &\quad + \sum_{\substack{l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \int_l p_j (\mathbf{u}_j - \mathbf{u}_i^*) \cdot (\mathbf{n}_{il} - \mathbf{n}_{jl}). \end{aligned}$$

Since A_{jl} and A_{il} are positive matrices, the first two terms of the sum are positive

$$\sum_{\substack{l \in \mathcal{L}_j \\ l \not\subset \Gamma_i}} \int_l^t (\mathbf{u}_j - \mathbf{u}_i^*) A_{jl} (\mathbf{u}_j - \mathbf{u}_i^*) + \sum_{\substack{l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \int_l^t (\mathbf{u}_j - \mathbf{u}_i^*) A_{il} (\mathbf{u}_j - \mathbf{u}_i^*) \geq 0.$$

An immediate consequence is that if the cell j has no edge carried by Γ_i then the scheme is entropic, since $\mathcal{L}_j \cap \Gamma_i = \emptyset$.

However if the cell touches the boundary, one gets the following necessary condition for local entropy production

$$p_j \sum_{\substack{l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \int_l (\mathbf{u}_i^* - \mathbf{u}_j) \cdot (\mathbf{n}_{il} - \mathbf{n}_{jl}) \leq \sum_{\substack{l \in \mathcal{L}_j \\ l \not\subset \Gamma_i}} \int_l^t (\mathbf{u}_j - \mathbf{u}_i^*) A_{jl} (\mathbf{u}_j - \mathbf{u}_i^*) + \sum_{\substack{l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \int_l^t (\mathbf{u}_j - \mathbf{u}_i^*) A_{il} (\mathbf{u}_j - \mathbf{u}_i^*).$$

Performing this analysis, we stated that the scheme is entropic for the whole set of cells but the ones connected to Γ_i 's. So, denoting by h the mesh size the volume where entropy production might not be positive is $O(h)$. So, one can hope that it converges to the physical solution as long as a whole shock does not remain located at the sliding interface. However, this lack of entropy can be disastrous with regard to the method stability. To address this problem, we will use subzone entropy stabilization [16].

4. SADDLE POINT PROBLEM DISCRETIZATION

As it was done for the non-sliding case, we use a conformal approximation subspace of finite dimension of $\mathcal{V} = L^2(\mathcal{E}_1)^d \times L^2(\mathcal{E}_2)^d \times L^2(\Gamma)$, in order to preserve scheme properties.

The first step in the discretization process consists in assuming that edges of the meshes \mathcal{M}_i are and remain straight all along the calculation. This naturally leads to discretize the grid velocity as edge supported $\mathbb{P}_1(\mathcal{E}_i)^d$ finite elements. This ensures continuity of each grid's displacement and edge velocities are affine.

The second step is the discretization of λ . We choose to impose two constraints. First, we assume that the interface between the two sliding fluids is continuous and piecewise linear. This implies that we have to deal with an additional 1d-mesh. Second, since λ can be reinterpreted as the mean pressure imposed by one domain on the other, no special regularity is expected at this point. This is why we take $\mathbb{P}_0(\Gamma)$ as the discretization space for the interface multiplier.

Finally, we have chosen $\mathbb{V}_h := \mathbb{P}_1(\mathcal{E}_1)^d \times \mathbb{P}_1(\mathcal{E}_2)^d \times \mathbb{P}_0(\Gamma)$ as a conformal approximation subspace of finite dimension of \mathcal{V} .

Remark 4.1 (Geometric approximation). Polygonal meshes description, for both fluids and the interface Γ lead to a second-order geometric error. Parabolic edges could lead to a third-order geometric error, but this is beyond the scope of this paper.

Property 6. *All the properties established in 3.7 remain true in \mathbb{V}_h .*

Proof. $\mathbb{V}_h \subset \mathcal{V}$. □

The discrete problem for the velocity calculation simply writes: find $(\mathbf{u}_{1h}^*, \mathbf{u}_{2h}^*, \lambda_h) \in \mathbb{V}_h$ such that

$$\begin{aligned} \forall (\mathbf{v}_{1h}, \mathbf{v}_{2h}, \mu_h) \in \mathbb{V}_h, \quad \sum_i a_i(\mathbf{u}_{ih}^*, \mathbf{v}_{ih}) + \sum_i b_i(\mathbf{v}_{ih}, \lambda_h) &= \sum_i l_i(\mathbf{v}_{ih}) \\ \text{and} \quad \sum_i b_i(\mathbf{u}_{ih}^*, \mu_h) &= 0. \end{aligned} \tag{4.1}$$

We now present the details of the calculation of the terms of the variational formula.

Contributions of non-sliding edges for \mathbb{P}_1 finite element approximations of a_i and l_i have already been studied in Section 2.3. So, we only focus on the contributions, to $a_i(\cdot, \cdot)$ and $l_i(\cdot)$, of hat-functions whose support intersect Γ_i . Also we examine $b_i(\cdot, \cdot)$'s expression for functions of $\mathbb{P}_1(\mathcal{E}_i)^d \times \mathbb{P}_0(\Gamma)$.

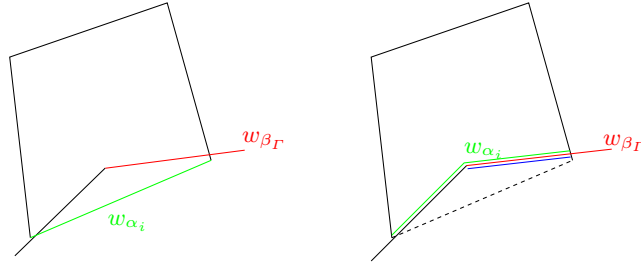


FIGURE 2. *Left:* In green we represent the support of the function w_{α_i} , the edge $l \in \Gamma_i$, in red, the support of the function w_{β_Γ} . *Right:* We represent how the support of w_{α_i} is viewed by the method. In blue the intersection of the two basis functions' support.

4.1. Slide line contributions

Denoting by \mathcal{F}_i the set of degrees of freedom of $\mathbb{P}_1(\mathcal{E}_i)$, we recall that

$$\forall \mathbf{v}_{ih} \in \mathbb{P}_1(\mathcal{E}_i)^d, \quad \mathbf{v}_{ih} = \sum_{\alpha_i \in \mathcal{F}_i} \mathbf{v}_{\alpha_i} w_{\alpha_i}, \quad \text{with } \mathbf{v}_{\alpha_i} \in \mathbb{R}^d,$$

where w_{α_i} is the continuous piecewise affine hat-function on \mathcal{E}_i such that $w_{\alpha_i}(\mathbf{x}_{\beta_i}) = \delta_{\alpha_i, \beta_i}$ (w_{α_i} being affine on any edge l of \mathcal{E}_i). Also, let \mathcal{F}_Γ be the set of degrees of freedom of $\mathbb{P}_0(\Gamma)$. One has

$$\forall \mu_h \in \mathbb{P}_0(\Gamma), \quad \mu_h = \sum_{\alpha_\Gamma \in \mathcal{F}_\Gamma} \mu_{\alpha_\Gamma} w_{\alpha_\Gamma}, \quad \text{with } \mu_{\alpha_\Gamma} \in \mathbb{R},$$

where denoting l_{β_Γ} a generic edge of Γ , one has $w_{\alpha_\Gamma} := \mathbf{1}_{l_{\alpha_\Gamma}}$.

We first consider the form $b_i(\cdot, \cdot)$ which contains all the difficulties.

$$\forall (\mathbf{v}_{ih}, \mu_h) \in \mathcal{C}, \quad b_i(\mathbf{v}_{ih}, \mu_h) = \sum_{\alpha_i \in \mathcal{F}_i} \sum_{\beta_\Gamma \in \mathcal{F}_\Gamma} \sum_{0 \leq p \leq d} \mu_{\beta_\Gamma} b_i(w_{\alpha_i} \mathbf{e}_p, w_{\beta_\Gamma}) (\mathbf{v}_{\alpha_i})_p.$$

So, the study of the slide line constraint is equivalent in studying the term $b_i(w_{\alpha_i} \mathbf{e}_p, w_{\beta_\Gamma})$.

For convenience, we denote by \mathcal{L}_{α_i} the set of edges in Γ_i that contains the vertex \mathbf{x}_{α_i} . Using this notation, and the fact that only one cell j is connected to l , one has

$$b_i(w_{\alpha_i} \mathbf{e}_p, w_{\beta_\Gamma}) = \sum_{l \in \mathcal{L}_{\alpha_i}} \int_l w_{\alpha_i} \mathbf{e}_p \cdot \mathbf{n}_{il} w_{\beta_\Gamma}.$$

Now, we have to precise some notations that we intentionally kept vague, since its rigorous definition would have complexified the presentation. Actually, w_{α_i} and w_{β_Γ} may not live at the same location. So, in order to precise the integral's meaning, we introduce the twisted edge \hat{l} (green line on the right hand side of Fig. 2) and we denote by \hat{T} the continuous piecewise affine transformation of l to \hat{l} . We set

$$b_i(w_{\alpha_i} \mathbf{e}_p, w_{\beta_\Gamma}) = \sum_{l \in \mathcal{L}_{\alpha_i}} \int_{\hat{l} = \hat{T}(l)} w_{\alpha_i} \circ \hat{T}^{-1} \mathbf{e}_p \cdot \mathbf{n}_{il} w_{\beta_\Gamma}.$$

Since the choice of a \mathbb{P}_0 discretization implies $w_{\beta_\Gamma} \equiv \mathbf{1}_{l_{\beta_\Gamma}}$, and since \mathbf{n}_{il} is constant on l_{β_Γ} , one finally has

$$b_i(w_{\alpha_i} \mathbf{e}_p, w_{\beta_\Gamma}) = \sum_{l \in \mathcal{L}_{\alpha_i}} \mathbf{e}_p \cdot \mathbf{n}_{il}(l_{\beta_\Gamma}) \int_{\hat{l} \cap l_{\beta_\Gamma}} w_{\alpha_i} \circ \hat{T}^{-1}.$$

Performing the same analysis, $\forall \alpha_i, \beta_i \in \mathcal{F}_i$ such that $\mathbf{x}_{\alpha_i}, \mathbf{x}_{\beta_i} \in \Gamma_i$, one has for $1 \leq p, q \leq d$

$$a_i(w_{\beta_i} \mathbf{e}_q, w_{\alpha_i} \mathbf{e}_p) = \sum_{l \in \mathcal{L}_{\alpha_i} \cap \mathcal{L}_{\beta_i}} \sum_{l_\Gamma \subset \Gamma} {}^t \mathbf{e}_q A_{il} \mathbf{e}_p \int_{\hat{l} \cap l_\Gamma} w_{\alpha_i} \circ \hat{T}^{-1} w_{\beta_i} \circ \hat{T}^{-1},$$

since A_{il} is constant on l_Γ . Also, $\forall \alpha_i \in \mathcal{F}_i$ such that $\mathbf{x}_{\alpha_i} \in \Gamma_i$, if we denote by

$$l_i^\Gamma(\mathbf{v}) := \sum_{j \in \mathcal{M}_i} \sum_{\substack{l \in \mathcal{L}_j \\ l \subset \Gamma_i}} \left(\int_l {}^t \mathbf{v} A_{il} \mathbf{u}_j + \int_l p_j \mathbf{v} \cdot \mathbf{n}_{il} \right),$$

the slide line contribution to $l_i(\mathbf{v})$, one gets for all $1 \leq p \leq d$

$$l_i^\Gamma(w_{\alpha_i}) = \sum_{l \in \mathcal{L}_{\alpha_i}} \sum_{l_\Gamma \subset \Gamma} \left({}^t \mathbf{e}_p (A_{il} \mathbf{u}_j + p_j \mathbf{n}_{il}) \int_{\hat{l} \cap l_\Gamma} w_{\alpha_i} \circ \hat{T}^{-1} \right),$$

if we assumed that \mathbf{u}_j and p_j are constant on l , which is the case for first-order approximation. For higher-order schemes one must obviously integrate $\mathbf{u}_j(\mathbf{x})$ and $p_j(\mathbf{x})$ along l . At the end, the $\mathbb{P}_1 - \mathbb{P}_0$ discretization only requires the calculation of

$$\int_{\hat{l} \cap l_\Gamma} w_{\alpha_i} \circ \hat{T}^{-1} \quad \text{and} \quad \int_{\hat{l} \cap l_\Gamma} w_{\alpha_i} \circ \hat{T}^{-1} w_{\beta_i} \circ \hat{T}^{-1},$$

which can be computed exactly since functions are affine on the integration domain.

4.2. Analysis of the discrete saddle point problem

We start by observing that for $i = 1, 2$ the bilinear forms a_i are coercive, with respect to the $L^2(\mathcal{E}_i)$ norm, on the discrete space $\mathbb{P}_1(\mathcal{E}_i)^d$, the proof being the same as for Lemma 2.2. In order to analyze the discrete saddle point problem we then only need to prove a suitable inf-sup condition (see [8]). In order to do so we make a further (quite reasonable) assumption on Γ_i . If l_1 and l_2 are two edges of Γ_i sharing a vertex, then the angle $\vartheta(l_1, l_2)$ between the two is bounded away from 0 and 2π . Combining together the local affine transformations $\hat{T} : l \rightarrow \hat{l}$ defined on each edge of Γ_i , introduced in Section 4.1, we obtain two global continuous piecewise affine transformations $\hat{T}_i : \Gamma_i \rightarrow \Gamma$. We let $\mathbb{P}_{1i}(\Gamma) = \mathbb{P}_1(\mathcal{E}_i)|_{\Gamma_i} \circ \hat{T}_i^{-1}$ denote the \mathbb{P}_1 discretization on Γ induced by the discretization Γ_i through the transformation \hat{T}_i . We recall that the following property holds (see [1]): if the two discretizations $\mathbb{P}_0(\Gamma)$ and $\mathbb{P}_{1i}(\Gamma)$ are quasi-uniform, then there exists a constant γ such that if the mesh-size h_Γ of $\mathbb{P}_0(\Gamma)$ and h_i of $\mathbb{P}_{1i}(\Gamma)$ verify $h_i/h_\Gamma \leq \gamma$, then the following inf-sup condition holds:

$$\inf_{\lambda_h \in \mathbb{P}_0(\Gamma)} \sup_{\xi_h \in \mathbb{P}_{1i}(\Gamma)} \frac{\int_\Gamma \lambda_h \xi_h}{\|\lambda_h\|_{0,\Gamma} \|\xi_h\|_{0,\Gamma}} \gtrsim 1. \quad (4.2)$$

It is also possible to prove that a local version of this result holds. More precisely, letting \mathcal{L}_{Γ_i} and \mathcal{L}_Γ denote respectively the set of edges/elements of Γ_i and Γ , if for all edge $l \in \mathcal{L}_{\Gamma_i}$ we denote by $\hat{l} = \hat{T}_i(l)$ its image on Γ , we have the following result.

Lemma 4.2. *There exists $\gamma > 0$ such that if for each edge $l \in \mathcal{L}_{\Gamma_i}$ and $\tilde{l} \in \mathcal{L}_\Gamma$ such that $\hat{l} \cap \tilde{l} \neq \emptyset$ we have $h_{\tilde{l}}/h_{\hat{l}} < \gamma$ then (4.2) holds.*

Proof. For each $\tilde{l} = (a_{\tilde{l}}, b_{\tilde{l}})$ element of Γ let $\hat{x}_0(\tilde{l}) < \hat{x}_1(\tilde{l})$ be two points defined as

$$\hat{x}_0(\tilde{l}) = \hat{T}_i \left(\min_{l=(a,b) \in \mathcal{L}_{\Gamma_i} : \hat{l} \subseteq \tilde{l}} a \right), \quad \hat{x}_1(\tilde{l}) = \hat{T}_i \left(\max_{l=(a,b) \in \mathcal{L}_{\Gamma_i} : \hat{l} \subseteq \tilde{l}} b \right).$$

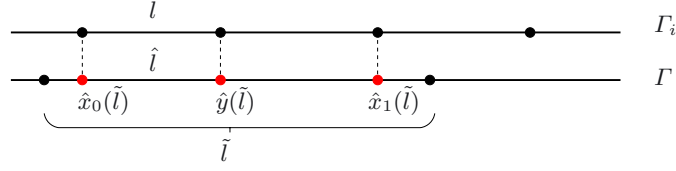


FIGURE 3. Representation of the notations used in Lemma 4.2, on a straight line.

By a simple geometric argument it is not difficult to see that if γ is small enough such two points exist and they verify $|\hat{x}_1(\tilde{l}) - \hat{x}_0(\tilde{l})| \gtrsim h_{\tilde{l}}$. Moreover, once again if γ is small enough, there exists at least a further vertex $\hat{y}(\tilde{l})$ of the grid $\hat{\mathcal{L}}_{\Gamma_i} = T(\mathcal{L}_{\Gamma_i})$ verifying $\hat{x}_0(\tilde{l}) < \hat{y}(\tilde{l}) < \hat{x}_1(\tilde{l})$. This configuration is depicted in Figure 3 for straight interfaces.

Let now $\lambda_h \in \mathbb{P}_0(\Gamma)$ be a piecewise constant function assuming the value $\lambda_{\tilde{l}}$ in \tilde{l} . Let $\hat{\xi}_h \in \mathbb{P}_{1i}(\Gamma)$ be defined as follows: for all $\tilde{l} \in \mathcal{L}_\Gamma$ $\hat{\xi}_h$ is linear in $(\hat{x}_0(\tilde{l}), \hat{y}(\tilde{l}))$ and in $(\hat{y}(\tilde{l}), \hat{x}_1(\tilde{l}))$, it assumes the values 0, $\lambda_{\tilde{l}}$ and 0 in $\hat{x}_0(\tilde{l})$, $\hat{y}(\tilde{l})$ and $\hat{x}_1(\tilde{l})$ respectively, and it vanishes outside $(\hat{x}_0(\tilde{l}), \hat{x}_1(\tilde{l}))$. It is immediate to check, by direct calculation, that $\|\hat{\xi}_h\|_{0,\tilde{l}}^2 \simeq h_{\tilde{l}}|\lambda_{\tilde{l}}|^2 \simeq \|\lambda_h\|_{0,\tilde{l}}^2$ and then $\|\xi_h\|_{0,\Gamma} \simeq \|\lambda_h\|_{0,\Gamma}$. It is also easy to check, once again by direct calculation, that $\int_{\tilde{l}} \lambda_h \hat{\xi}_h \simeq h_{\tilde{l}}|\lambda_{\tilde{l}}|^2$ and then $\int_\Gamma \lambda_h \hat{\xi}_h \simeq \|\lambda_h\|_{0,\Gamma}^2$. Then we have

$$\sup_{\xi_h \in \mathbb{P}_{1i}(\Gamma)} \frac{\int_\Gamma \lambda_h \xi_h}{\|\lambda_h\|_{0,\Gamma} \|\xi_h\|_{0,\Gamma}} \geq \frac{\int_\Gamma \lambda_h \hat{\xi}_h}{\|\lambda_h\|_{0,\Gamma} \|\hat{\xi}_h\|_{0,\Gamma}} \simeq 1.$$

The thesis follows thanks to the arbitrariness of λ_h . \square

Lemma 4.2 implies that, provided the discretization $\mathbb{P}_{1i}(\Gamma)$ is sufficiently finer than $\mathbb{P}_0(\Gamma)$, an inf-sup condition holds for our problem thanks to the following proposition.

Proposition 4.3. *Assume that*

$$\inf_{\lambda_h \in \mathbb{P}_0(\Gamma)} \sup_{\xi_h \in \mathbb{P}_{1i}(\Gamma)} \frac{\int_\Gamma \lambda_h \xi_h}{\|\lambda_h\|_{0,\Gamma} \|\xi_h\|_{0,\Gamma}} \gtrsim 1. \quad (4.3)$$

Then it holds that

$$\inf_{\lambda_h \in \mathbb{P}_0(\Gamma)} \sup_{\mathbf{v}_{ih} \in \mathbb{P}_1(\mathcal{E}_i)^d} \frac{b_i(\mathbf{v}_{ih}, \lambda_h)}{\|\mathbf{v}_{ih}\|_{0,\mathcal{E}_i} \|\lambda_h\|_{0,\Gamma}} \gtrsim 1. \quad (4.4)$$

Proof. Let $\lambda_h \in \mathbb{P}_0(\Gamma)$ and let $\xi_h \in \mathbb{P}_{1i}(\Gamma)$ such that

$$\frac{\int_\Gamma \lambda_h \xi_h}{\|\xi_h\|_{0,\Gamma}} \gtrsim \|\lambda_h\|_{0,\Gamma}.$$

Let \mathbf{v}_{ih} be defined as follows. For all \mathbf{x}_α vertex of Γ_i we let \mathbf{n}_α^+ and \mathbf{n}_α^- denote the two normals to the two edges of Γ_i that share \mathbf{x}_α as a vertex. Two cases can occur: $\mathbf{n}_\alpha^+ = \mathbf{n}_\alpha^- = \mathbf{n}_\alpha$ or not. In the first case we set $\mathbf{v}_{ih}(\mathbf{x}_\alpha) = \xi_h(T(\mathbf{x}_\alpha))\mathbf{n}_\alpha$. In the second case the two conditions

$$\mathbf{v}_{ih}(\mathbf{x}_\alpha) \cdot \mathbf{n}_\alpha^+ = \xi_h(T(\mathbf{x}_\alpha)), \quad \mathbf{v}_{ih}(\mathbf{x}_\alpha) \cdot \mathbf{n}_\alpha^- = \xi_h(T(\mathbf{x}_\alpha)),$$

uniquely determine $\mathbf{v}_{ih}(\mathbf{x}_\alpha)$. We complete the definition of \mathbf{v}_{ih} by setting it equals to $\mathbf{0}$ at all interior nodes of \mathcal{E}_i . It is not difficult to realize that $\|\mathbf{v}_{ih}\|_{0,\mathcal{E}_i} \simeq \|\xi_h \circ T\|_{0,\Gamma_i} \simeq \|\xi_h\|_{0,\Gamma}$. Then we can write

$$\frac{b_i(\mathbf{v}_{ih}, \lambda_h)}{\|\mathbf{v}_{ih}\|_{0,\mathcal{E}_i}} = \frac{\int_\Gamma \lambda_h [\mathbf{v}_{ih} \cdot \mathbf{n}_i] \circ T_i^{-1}}{\|\mathbf{v}_{ih}\|_{0,\mathcal{E}_i}} = \frac{\int_\Gamma \lambda_h \xi_h}{\|\xi_h\|_{0,\Gamma}} \gtrsim \|\lambda_h\|_{0,\Gamma}.$$

Then, for all $\lambda_h \in \mathbb{P}_0(\Gamma)$, $\sup_{\mathbf{v}_{ih} \in \mathbb{P}_1(\mathcal{E}_i)^d} \frac{b_i(\mathbf{v}_{ih}, \lambda_h)}{\|\mathbf{v}_{ih}\|_{0, \mathcal{E}_i} \|\lambda_h\|_{0, \Gamma}} \gtrsim 1$, which, thanks to the arbitrariness of $\lambda_h \in \mathbb{P}_0(\Gamma)$ implies the thesis. \square

Remark 4.4. Proposition 4.3 states that in order to verify the inf-sup condition (4.4) it is sufficient that $\mathbb{P}_{1i}(\Gamma)$ and $\mathbb{P}_0(\Gamma)$ satisfy the inf-sup condition (4.3). It is well known that if the mesh for $\mathbb{P}_{1i}(\Gamma)$ is coarser than the mesh for $\mathbb{P}_0(\Gamma)$, or if the two meshes coincide, than (4.2) does not hold. If we want (4.3) to hold, for two meshes with the same meshsize, we need to choose the mesh for $\mathbb{P}_0(\Gamma)$ as the dual mesh of $\mathbb{P}_{1i}(\Gamma)$, see [30].

Proposition 4.5. *If (4.3) holds for either $i = 1$ or $i = 2$ then*

$$\inf_{\lambda_h \in \mathbb{P}_0(\Gamma)} \sup_{(\mathbf{u}_{1h}, \mathbf{u}_{1h}) \in \mathbb{P}_1(\mathcal{E}_1)^d \times \mathbb{P}_1(\mathcal{E}_2)^d} \frac{\sum_i b_i(\mathbf{v}_{ih}, \lambda_h)}{\sum_i \|\mathbf{v}_{ih}\|_{0, \mathcal{E}_i} \|\lambda_h\|_{0, \Gamma}} \gtrsim 1. \quad (4.5)$$

Proof. To fix the ideas, let (4.3) hold for $i = 1$. We have

$$\begin{aligned} \inf_{\lambda_h \in \mathbb{P}_0(\Gamma)} \sup_{(\mathbf{u}_{1h}, \mathbf{u}_{1h}) \in \mathbb{P}_1(\mathcal{E}_1)^d \times \mathbb{P}_1(\mathcal{E}_2)^d} \frac{\sum_i b_i(\mathbf{v}_{ih}, \lambda_h)}{(\|\mathbf{v}_{1h}\|_{0, \mathcal{E}_1} + \|\mathbf{v}_{2h}\|_{0, \mathcal{E}_2}) \|\lambda_h\|_{0, \Gamma}} \\ \geq \inf_{\lambda_h \in \mathbb{P}_0(\Gamma)} \sup_{\mathbf{u}_{1h} \in \mathbb{P}_1(\mathcal{E}_1)^d} \frac{b_1(\mathbf{v}_{1h}, \lambda_h)}{(\|\mathbf{v}_{1h}\|_{0, \mathcal{E}_1} + \|\mathbf{0}\|_{0, \mathcal{E}_2}) \|\lambda_h\|_{0, \Gamma}} \gtrsim 1. \end{aligned}$$

which we obtained by bounding the supremum over $\mathbb{P}_1(\mathcal{E}_1)^d \times \mathbb{P}_1(\mathcal{E}_2)^d$ with the supremum over the subspace $\mathbb{P}_1(\mathcal{E}_1)^d \times \{\mathbf{0}\}$. \square

Remark 4.6. Lemma 4.2 requires the mesh size of Γ_i to be fine enough that the image through the mapping \hat{T}_i of at least two elements of the mesh of Γ_i is included in each element of the mesh of Γ . From the practical point of view it appears, as it is often the case with this kind of results, that this requirement is too strong, and, in the construction of the mesh for Γ , we will only make sure that it is strictly coarser than the image through \hat{T}_i of one of the Γ_i 's.

Theorem 4.7. *Under the assumptions of Proposition 4.5, the discrete saddle point problem (4.1) admits a unique solution $(\mathbf{u}_{1h}^*, \mathbf{u}_{2h}^*, \lambda_h)$ which verifies*

$$\|\mathbf{u}_{1h}^*\|_{0, \mathcal{E}_1} + \|\mathbf{u}_{2h}^*\|_{0, \mathcal{E}_2} + \|\lambda_h\|_{0, \Gamma} \lesssim \sum_i \|l_i\|_{L^2(\mathcal{E}_i)'} \quad (4.6)$$

Moreover the following error bound hold:

$$\sum_i \|\mathbf{u}_i^* \cdot \mathbf{n}_{\mathcal{E}_i} - \mathbf{u}_{ih}^* \cdot \mathbf{n}_{\mathcal{E}_i}\|_{L^2(\mathcal{E}_i)} \lesssim \sum_i \inf_{\mathbf{v}_{ih} \in \mathbb{P}_1(\mathcal{E}_i)} \|\mathbf{u}_{\mathcal{E}_i}^* \cdot \mathbf{n}_i - \mathbf{v}_{ih}^* \cdot \mathbf{n}_{\mathcal{E}_i}\|_{L^2(\mathcal{E}_i)} + \inf_{\mu_h \in \mathbb{P}_0(\Gamma)} \|\lambda - \mu_h\|_{L^2(\Gamma)}. \quad (4.7)$$

where $\mathbf{u}_i^* \cdot \mathbf{n}_{\mathcal{E}_i}$, $i = 1, 2$ and λ are the (uniquely determined) normal components to the grid of velocity and multiplier respectively, of any of the solutions of the continuous saddle point problem.

Proof. The existence, uniqueness and stability estimate for the discrete solution are obtained by standard techniques as a consequence of coercivity and of the validity of the inf-sup condition. As far as the error estimate is concerned we observe that, letting

$$\ker B_h = \left\{ (\mathbf{v}_{1h}, \mathbf{v}_{2h}) : \sum_i b_i(\mathbf{v}_{ih}, \mu_h) = 0, \forall \mu_h \in \mathbb{P}_0(\Gamma) \right\} \subset \mathbb{P}_1(\mathcal{E}_1)^d \times \mathbb{P}_1(\mathcal{E}_2)^d,$$

by proceeding exactly as in the proof of (Prop. 2.4 in [8]), it is easy to prove that

$$\sum_i \|\mathbf{u}_i^* \cdot \mathbf{n}_{\mathcal{E}_i} - \mathbf{u}_{ih}^* \cdot \mathbf{n}_{\mathcal{E}_i}\|_{L^2(\mathcal{E}_i)} \lesssim \inf_{(\mathbf{v}_{1h}, \mathbf{v}_{1h}) \in \ker B_h} \sum_i \|\mathbf{u}_i^* \cdot \mathbf{n}_{\mathcal{E}_i} - \mathbf{v}_{ih}^* \cdot \mathbf{n}_{\mathcal{E}_i}\|_{L^2(\mathcal{E}_i)} + \inf_{\mu_h \in \mathbb{P}_0(\Gamma)} \|\lambda - \mu_h\|_{L^2(\Gamma)}.$$

It remains to prove that

$$\inf_{(\mathbf{v}_{1h}, \mathbf{v}_{1h}) \in \ker B_h} \sum_i \|\mathbf{u}_i^* \cdot \mathbf{n}_{\mathcal{E}_i} - \mathbf{v}_{ih}^* \cdot \mathbf{n}_{\mathcal{E}_i}\|_{L^2(\mathcal{E}_i)} \lesssim \sum_i \inf_{\mathbf{v}_{ih} \in \mathbb{P}_1(\mathcal{E}_i)^d} \|\mathbf{u}_i^* \cdot \mathbf{n}_{\mathcal{E}_i} - \mathbf{v}_{ih}^* \cdot \mathbf{n}_{\mathcal{E}_i}\|_{L^2(\mathcal{E}_i)}.$$

Thanks to the inf-sup condition this can also be proven by proceeding as in the proof of (Prop. 2.5 in [8]). \square

Remark 4.8. Coherently with Remark 3.2, the Theorem 4.7 only deals with the error on the normal components of the velocity.

Remark 4.9. The inequality (4.6) of Theorem 4.7 states the stability of the method at each time step. Also, inequality (4.7) demonstrates the optimality of the solution for a given mesh. These results make us confident in the quality of this sliding method.

4.3. Linear system and quadrature choices

The grid velocity $U = \begin{pmatrix} U_1 \\ U_2 \end{pmatrix}$ is solution of the linear system

$$\begin{pmatrix} A_1 & 0 & {}^t B_1 \\ 0 & A_2 & {}^t B_2 \\ B_1 & B_2 & 0 \end{pmatrix} \begin{pmatrix} U_1 \\ U_2 \\ \Lambda \end{pmatrix} = \begin{pmatrix} L_1 \\ L_2 \\ 0 \end{pmatrix}, \quad (4.8)$$

where the sub-vectors are defined as $U_i := (\mathbf{u}_{\alpha_i}^*)_{\alpha_i \in \mathcal{F}_i}$ and $\Lambda := (\lambda_{\alpha_\Gamma})_{\alpha_\Gamma \in \mathcal{F}_\Gamma}$.

The matrices A_i are sparse since direct interactions only occur between nodes connected by one edge. The matrices B_i are also sparse since λ 's interactions with \mathbf{u}_{ih}^* are local.

Observe that the matrix is symmetric (since A_i are symmetric), but it is not positive, as usual when dealing with constrained linear problems in saddle point form. Many methods can be chosen to compute the solution of (4.8).

Uzawa's algorithm or its improvements (see [21] for instance) is a good candidate to solve the problem. However, it is important to emphasize that all the conservation properties of the scheme are obtained if and only if the system is solved exactly! Simple Uzawa strategies may converge very slowly since they are very sensitive to the condition number of the matrix $BA^{-1}{}^t B$.

Our aim is not to find the more efficient way to solve (4.8), but we want to compute the solution up to the machine precision to illustrate the good properties of the method. So we used a direct solver for the saddle-point problem: SuperLU [25].

Remark 4.10. Observe that the use of a direct solver is not an issue. As discussed bellow, in practice one can reduce the linear system to local problems for interior nodes and a linear system coupling nodes of the slide lines. The linear system is small since one reduces its dimension to a 1D problem and so its resolution is cheap.

In order to show the flexibility of the method, we want to emphasize that one does not require to replace his whole favorite cell-centred hydro-scheme (Glace [17] or Eucclhyd [26]) and does not need to solve the fully coupled velocity system for slide lines calculations.

As precised in Section 2.3, one can change locally the quadrature formulae without any loss of the properties as long as this local quadrature formula is used at every step of the method. For the numerical tests that are presented in this paper we did the following choice.

- Integrals on Γ (or Γ_i 's) are computed exactly.
- Integrals on $\mathring{\mathcal{E}}_i = \mathcal{E}_i \setminus \Gamma_i$ are computed using trapezoid formula approximation.

Doing so, the linear system recasts to

$$D_{\mathcal{E}_i}^\circ U_{\mathcal{E}_i}^\circ = L_{\mathcal{E}_i}^\circ, \quad (4.9)$$

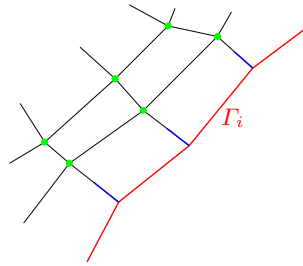


FIGURE 4. Example of solver placement for the domain Ω_i . In red: the continuous scheme on the slide line with exact integration which contributes to (4.10). In blue: Eucclhyd scheme contribution to the linear system (4.10) arising using trapezoid formula. In green: classic nodal-solver, Glace or Eucclhyd for instance.

for the inner degrees of freedom of domain i , and

$$\begin{pmatrix} A_{\Gamma_1} & 0 & {}^t B_{\Gamma_1} \\ 0 & A_{\Gamma_2} & {}^t B_{\Gamma_2} \\ B_{\Gamma_1} & B_{\Gamma_2} & 0 \end{pmatrix} \begin{pmatrix} U_{\Gamma_1} \\ U_{\Gamma_2} \\ \Lambda \end{pmatrix} = \begin{pmatrix} L_{\Gamma_1} \\ L_{\Gamma_2} \\ 0 \end{pmatrix}, \tag{4.10}$$

for the interface (coupled) degrees of freedom, which is much smaller than (4.8).

Also, as shown at the end of Section 2.3, the nodal-solver corresponding to (4.9) is Eucclhyd. Since D_{ε_i} is $d \times d$ -bloc diagonal, the problem is local at each node. One can eventually replace these local Eucclhyd nodal-solvers by Glace for instance. The chosen strategy is illustrated by Figure 4.

4.4. An interface meshing strategy

We now describe the simple method we used to mesh Γ . Basically, the approach consists in two main steps.

- First, we compute a geometric mesh. It consists in computing a coarse grid for which normal changes occur rarely with regard to the fluids meshes. This allows a fine and oscillatory mesh to slide along a coarse one.
- The second step consists in refining the geometrical mesh enough to get local precision (λ acting locally as the pressure imposed by one fluid on the other, mesh must be fine enough to avoid interpenetration), but still fulfilling Theorem 4.7’s hypothesis to preserve stability.

4.4.1. Proposed algorithm

- (1) The idea is to browse both Γ_i ’s at once in order to determine the geometrical (normal) changes.
 - (a) The vertices of the Γ_i ’s are sorted using their curvilinear distance to the starting point.
 - (b) One browses entirely the longest of the current edges of the Γ_i ’s. A position corresponding to the extremity of this edge is determined on the other boundary. Then one chooses as a next geometric vertex, the middle of the next longest edge on the Γ_i ’s. One repeats this step until the whole interface has been browsed. Figure 5a illustrates the procedure.
- (2) The three interfaces are browsed at once. While browsing one edge e_Γ of Γ , if we cross more than two edges of Γ_i then e_Γ is divided by adding a vertex which corresponds to the middle of the following edge of Γ_i , see Figure 5b.

Let us emphasis that for Lagrangian calculations, one has no direct control on Γ_i ’s evolution. These are driven by the fluids motion while they slide along Γ . Practically, it appeared that step (4.4.1) of the algorithm, which reconstructs in some way the geometry of the sliding interface, is critical with regard to the quality of the calculation. This reconstruction step has been established empirically and could be largely improved.

In the following section, we propose a simple ALE which allows more control on Γ_i ’s evolution.

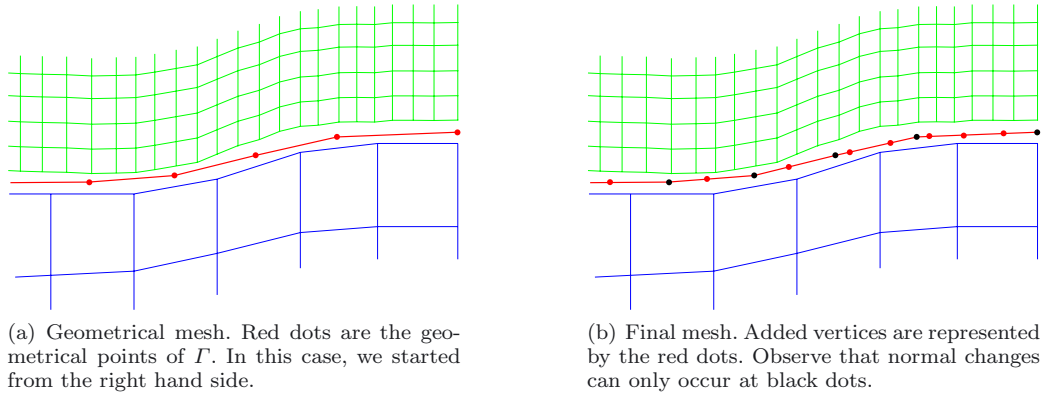


FIGURE 5. Construction of the interface mesh.

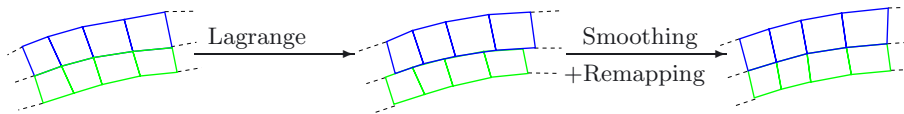


FIGURE 6. The simple ALE procedure decomposes in two steps. A Lagrangian step where each mesh is displaced at its own velocity. Then, a smoothing step where the whole mesh is remapped conforming.

5. A SIMPLE ALE TREATMENT

Use of rezoning is also a practical way to avoid difficulties due to non-matching sliding lines. Sketch of the idea can be found for instance in [4].

The principle of the method is very simple. Let us assume the grid to be conforming at the beginning of a time step. We call $\mathbf{x}_{i,j}^n \in \mathbb{R}^d$ the coordinates of the i th vertex of Γ_j , so that at this time $\mathbf{x}_{i,1}^n = \mathbf{x}_{i,2}^n, \forall i$. We apply the sliding algorithm previously described. At the end of this stage, the grid is no more conforming $\mathbf{x}_{i,1}^* \neq \mathbf{x}_{i,2}^*$. We then impose the positions of each $\mathbf{x}_{i,1}^{n+1}$ to be equal to these of $\mathbf{x}_{i,2}^{n+1}$ by computing

$$\mathbf{x}_{i,1}^{n+1} = \mathbf{x}_{i,2}^{n+1} = \theta \mathbf{x}_{i,1}^* + (1 - \theta) \mathbf{x}_{i,2}^*, \tag{5.1}$$

with $\theta \in [0, 1]$. A classical algorithm (Tipton-like [31] in our case) is used to displace other vertices, in order to smooth the grid.

All the conservative quantities are then remapped on this new grid, thanks to a Benson-like algorithm [4] that ensures the conservation of the mass, momentum and total energy. We assume that this stage preserves the position of the interfaces $\Gamma_1 \equiv \Gamma_2$. Consequently, no fluxes are computed through the slide line.

6. NUMERICAL TESTS

To illustrate the behavior of the method, we perform several representative tests.

The tests are run using an first-order Euler explicit time integration that is build the same way as classical Glace or Eucclhyd schemes [17, 26].

As it is common for Lagrangian schemes, we use a stabilization ingredient. This is even more necessary when dealing with slide lines. For instance, Caramana [10] uses a strong hourglass control (with a merit factor of 1)

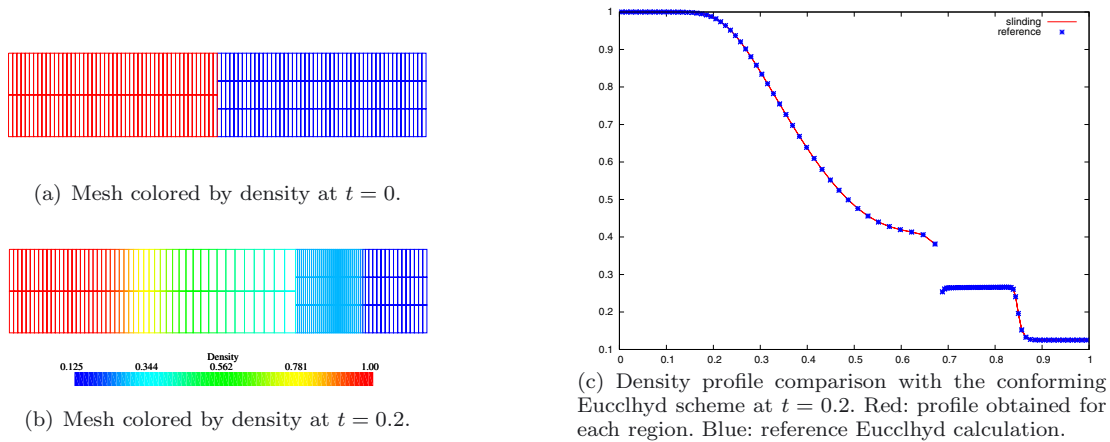


FIGURE 7. Sod orthogonal interface.

close to slide line (refer also to [2]). Kutchařík *et al.* [24] use the same strategy and add surface tension at the slide line. Here, we use subzone entropy [16] which is build to preserve the conservation and consistency properties of cell-centered Lagrangian schemes. It is easy to adapt formulas (27) and (28) pages 6567–6568 of [16] to our method and see that properties are also preserved.

If not precised, tests run Eucclhyd’s scheme as the nodal solver for more coherence (*i.e.* it is the natural scheme that is obtained when using trapezoid formula on non sliding edges). Also, we recall that the arising linear system is solved by using SuperLU [25]: an exact resolution is required to get the properties of the scheme, especially conservation.

6.1. Sod shock tube

We use this case as a sanity check for the sliding method. We recall the case configuration. We choose $\Omega =]0, 1[\times]0, 0.2[$ as the computational domain. Initial data is

$$\mathbf{u} = \mathbf{0}, \quad \rho = \begin{cases} 1 & \text{if } x < \frac{1}{2}, \\ 0.125 & \text{elsewhere,} \end{cases} \quad p = \begin{cases} 1 & \text{if } x < \frac{1}{2}, \\ 0.1 & \text{elsewhere.} \end{cases}$$

The fluid follows a perfect gas law, adiabatic constant is $\gamma = 1.4$.

We run two different configurations to compute Sod shock tube approximation.

6.1.1. Orthogonal interface

First, the sliding interface is oriented orthogonally to the flow.

The mesh is the union of a uniform 50×2 cells set for the initially heavy region and 50×3 cells for the light zone, see Figure 7a. So, that it is non-conforming at the contact discontinuity all along the calculation.

At mesh non-conformity, we impose $[[\mathbf{u}]] \cdot \left(\frac{1}{0}\right) = 0$ using presented sliding method. For nodes that are not located in the sliding interface, we use the Eucclhyd’s scheme.

Observe that for this calculation we do not use subzone entropy stabilization: it is not necessary to preserve symmetries.

Since the solution of this problem is 1D, we expect our algorithm to compute the same solution as the Eucclhyd’s scheme on a conforming grid (we recall that Eucclhyd is not sensitive to the aspect ratio of cells for 1D problems).

Figure 7b illustrates that the solution remains uni-dimensional. In this particular case, since Γ and Γ_i coincide, \mathbf{n}_{il} and \mathbf{n}_{jl} normals are the same. So, the scheme is entropic and conservative in volume.

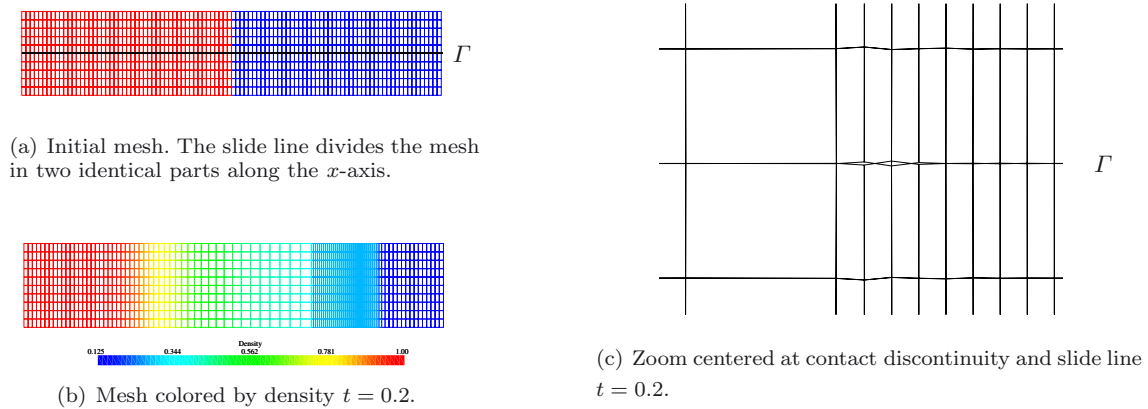


FIGURE 8. Sod orthogonal interface.

Moreover, on Figure 7c, we compare the density profile to the one obtained on a 100×2 grid using the Eucclhyd's scheme. Profiles match perfectly, confirming that in this case the method acts as a generalization of the Eucclhyd's scheme for non-matching grids.

6.1.2. Aligned interface

As a second sanity check test, we use the configuration depicted on Figure 8a. The computational domain $\Omega =]0, 1[\times]0, 0.2[$ is divided into two regions delimited by $y = 0.1$. Each subdomain is meshed using 100×5 cells. The slide line method is applied on $y = 0.1$ boundaries. Again, no stabilization is used to run this test.

The results are presented in Figure 8b. At first sight, one could think that the numerical solution is 1D. However, a zoom is performed on the meshes, centered on the contact discontinuity and the slide line. One observes little oscillations illustrating that the obtained result is not 1D. Let us emphasize however, that the oscillations remain small during the whole calculation.

Actually, this was expected: according to its discretization, λ has almost no chance to be able to compensate exactly the pressure variation along the x -axis. In the previous case, pressure at the sliding interface is constant at all time, so that it is exactly represented in $\mathbb{P}_0(\Gamma)$.

Nevertheless, on the one hand, the error is small on this particular test and on the other hand, quality of the solution could probably be improved in computing a better mesh for Γ . Such a good quality of the solution was not what we first expected for this particular test.

6.2. Caramana's piston problem

This test case has been defined in [10]. The computational domain is $\Omega =]0, 1[\times]0, \frac{1}{2}[$. Initially, the fluid state is defined as

$$\mathbf{u} = \mathbf{0}, \quad \rho = \begin{cases} 1 & \text{if } y < \frac{1}{4}, \\ 10 & \text{elsewhere,} \end{cases} \quad p = \begin{cases} 20 & \text{in }]0.95, 1[\times]0, \frac{1}{4}[, \\ \frac{2}{3} \times 10^{-8} & \text{elsewhere.} \end{cases}$$

The fluid follows a perfect gas law with an adiabatic constant $\gamma = \frac{5}{3}$. Note that, as it is done in [24], $p = 0$ is replaced by $p = \frac{2}{3} \times 10^{-8}$ for numerical purpose. Symmetric boundary conditions are imposed on the whole $\partial\Omega$.

This initial data generates a shock wave due to the high pressure zone that travels in the domain. Combined with the density jump at the interface, this leads to sliding along this discontinuity line.

For this simulation we take two different uniform meshes of Ω . The first one is made of 100×50 square cells as in [10, 24] (see Fig. 9a). The second one is its two-by-two refined mesh: 200×100 . For each mesh, three strategies are used:

- Lagrangian with subzone entropy stabilization [16] (coefficient being set to 10^{-2} for cells connected to Γ_i 's and to 10^{-3} for the others);
- Lagrangian without any stabilization;
- ALE as described in Section 5, without any stabilization during the Lagrangian step. Fifty iterations of a Jun-like [23] algorithm are used to rezone inner nodes of each mesh. A limited second-order fluxing-based advection process is used to remap conservative quantities.

The results we obtain on the 100×50 mesh are presented in Figures 9b–9c. The pressure scale is fixed to $[0, 1.8]$ to allow better comparison. At first we emphasize that the use of the subzone entropy stabilization does not seem to deteriorate the position of the sliding interface since the results in Figures 9b and 9c look quite similar. However, one might notice that in the case of stabilized calculation the light domain mesh is less distorted at the sliding interface. Also, the ALE simulation (Fig. 9d) provides a close result attesting the correct behaviour of the approach, even if we over-smoothed meshes. The only disadvantage of such a strategy is the excess of numerical dissipation which is classical and not due to the sliding method.

Using the 200×100 mesh allows us to advance one step further in the test case analysis. The results are presented in Figures 9e–9g. We performed these tests in order to illustrate two facts. The first one is that the Lagrangian stabilized and the ALE simulation show some kind of qualitative convergence, which is a good point for the method. The second point is related to the Lagrangian calculation without stabilization: one sees on Figure 9f the failure of the computation – approximation is presented at time $t = 0.285$ while δt is decreasing to 0 to satisfy CFL constrain. Indeed, it fails since some interface cells of the light domain get tangled. However, looking carefully at the tangled zone (zoom in Fig. 9f), one notices that the interface is still well described.

Of course, there is no analytical solution to this problem, but one can predict some quantitative values of the density as long as the rarefaction wave that follows the strong shock do not interact with it. Indeed, for 1d-shocks, the post-shock density should satisfy $\rho_{\text{post}} = \frac{\gamma+1}{\gamma-1}\rho_{\text{pre}}$ (see for instance [34]), that is a maximum of 4 in the light zone and 40 in the heavy part, ρ_{pre} and ρ_{post} denoting respectively the pre- and post-shock densities. On Figure 9h, one observes that post-shock densities are particularly accurate, especially in the light domain, where the shockwave is almost planar.

Finally, one should notice that as it is often the case for gas-gas contacts the interface is unstable (one can easily check that initially $\nabla \rho \cdot \nabla p < 0$ on $]0.95, 1[\times \{\frac{1}{4}\}$). One could argue that the subzone entropy treatment produces enough numerical dissipation to keep artificially the interface “smooth”. However, as it is shown in [16], using this stabilization technique does not affect the consistency of the overall scheme with Euler’s equations. Due to this interface instability, this is a discriminant test since more realistic flows such elasto-plastic flows are generally more stable and should be simpler in that regard.

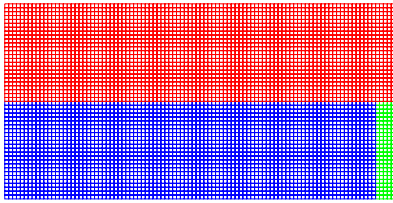
6.3. Sliding obstacle test

We imagined this test case to present the ability of the approach to deal with non-conforming geometrical slide lines, *i.e.* sets of sliding edges might change during the calculation. Obviously, there are no theoretical obstruction to this kind of situation, the difficulty being the calculation of Γ and Γ_i 's sliding parts.

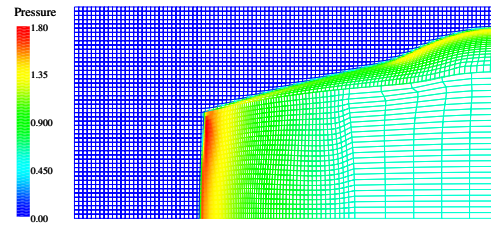
For this kind of simulation (non-conforming slide line geometries), we add the following ingredient to the algorithm. Assume for instance, that an extremity of Γ_1 projects inside Γ_2 , then we actually displace the vertex by projecting it on Γ_2 . This is mandatory to avoid excess of interpenetration at extremities.

At time $t = 0$, the initial configuration is given by $\Omega = \Omega_1 \cup \Omega_2$, where $\Omega_1 =]-\frac{1}{4}, \frac{1}{2}[\times]\frac{1}{4}, \frac{3}{2}[$ and $\Omega_2 =]-\frac{1}{2}, 0[\times]1, \frac{1}{2}[$. The state of the two fluids is given by

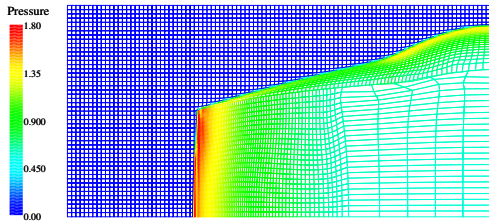
$$\rho = \begin{cases} 1 & \text{in } \Omega_1, \\ 2 & \text{in } \Omega_2, \end{cases} \quad \mathbf{u} = \begin{cases} (2, -1) & \text{in } \Omega_1, \\ \mathbf{0} & \text{in } \Omega_2, \end{cases} \quad \text{and} \quad p = 0 \text{ in } \Omega.$$



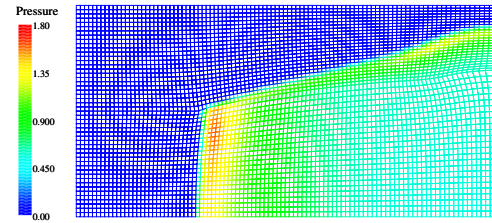
(a) Initial 100×50 mesh.



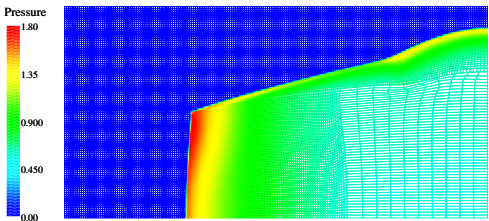
(b) Pressure field - Time $t = 0.3$. Mesh 100×50 . Lagrangian simulation with stabilization.



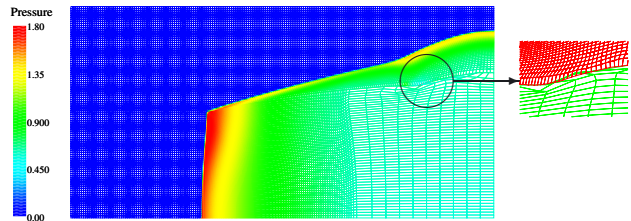
(c) Pressure field - Time $t = 0.3$. Mesh 100×50 . Lagrangian simulation without any stabilization.



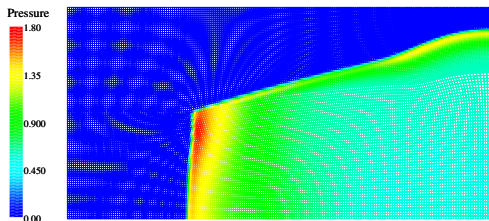
(d) Pressure field - Time $t = 0.3$. Mesh 100×50 . ALE simulation without any stabilization.



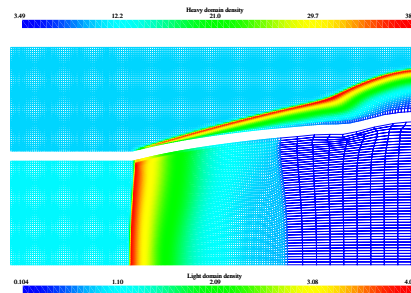
(e) Pressure field - Time $t = 0.3$. Mesh 200×100 . Lagrangian simulation with stabilization.



(f) Pressure field - Time $t = 0.285$. Mesh 200×100 . Lagrangian simulation without any stabilization.



(g) Pressure field - Time $t = 0.3$. Mesh 200×100 . ALE simulation without stabilization.



(h) Density in each domain - Time $t = 0.3$. Mesh 200×100 . Lagrangian simulation with stabilization.

FIGURE 9. Caramana test case.

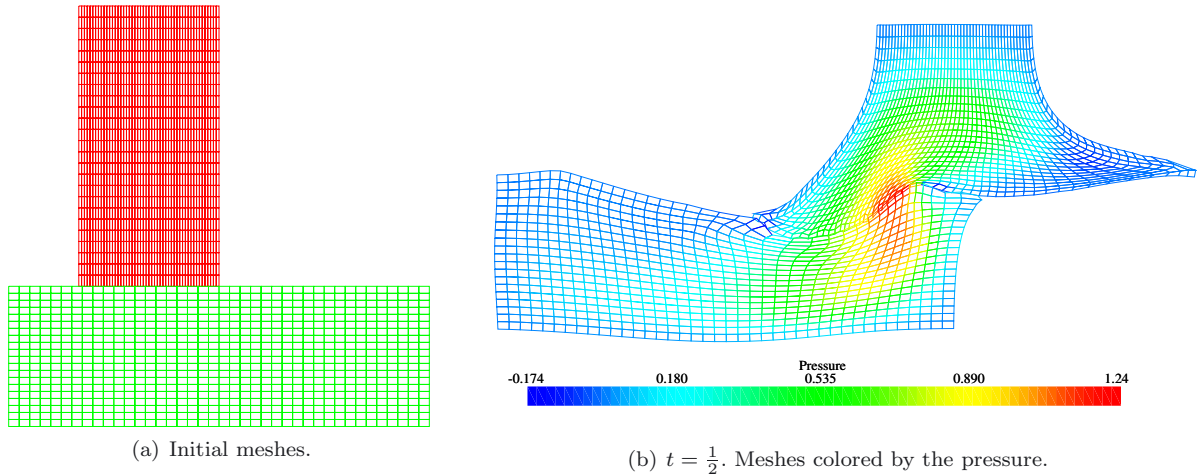


FIGURE 10. Obstacle test.

We choose a stiffened gas law that sets $p(\rho, \epsilon) = (\gamma - 1)\rho\epsilon - \gamma p_0$, with $\gamma = \frac{5}{3}$ and $p_0 = 1$. This allows us to impose a null external pressure on $\partial\Omega$. The simulation is run up to $t = \frac{1}{2}$ so that Ω_1 flows outside of Ω_2 .

For this computation, we use the following options. Ω_1 is divided into 50×25 rectangular cells while the mesh for Ω_2 is a regular 40×20 grid. For inner nodes (not connected to Γ_i 's we use a first-order Glace's scheme. Also, the subzone entropy parameter is set to 10^{-2} for Γ_i 's nodes and to 10^{-3} for other ones. This is the tuning we use for classical (non-sliding) simulations.

Figure 10b presents the position of the meshes at the final time $t = \frac{1}{2}$. Meshes are colored by pressure. It illustrates that the method was able to treat the changes of sliding area and is quite robust, since deformations are significant.

One may observe oscillations at the sliding interface. These could be troubleshooting for the stability of the calculation, but their effect is not huge since the calculation, on this particular configuration, runs up to time 1.1, after which time step goes down.

6.4. Bullet in channel

We conclude this section by presenting the results we obtain on a very challenging test. This test case is defined in [24] but unfortunately as most of the cases presented here has no reference solution. So, we give some qualitative results comparing Lagrangian simulations and the ALE approach we propose. The test is defined as follows.

The computational domain $\Omega =]0, 2[\times]0, 10[$ is partitioned into four rectangular parts, Ω_{ct} , Ω_b , Ω_{cb} and Ω_r , as depicted on Figure 11.

The entire domain is occupied by a fluid that follows a perfect gas law of adiabatic constant $\gamma = \frac{5}{3}$. Its initial state is set as

$$\rho = \begin{cases} 10 & \text{in } \Omega_r, \\ 1 & \text{in } \Omega_b, \\ 0.1 & \text{in } \Omega_{ct} \cup \Omega_{cb}, \end{cases} \quad \mathbf{u} = \begin{cases} \mathbf{0} & \text{in } \Omega_r \cup \Omega_{ct} \cup \Omega_{cb}, \\ (0, -7) & \text{in } \Omega_b, \end{cases} \quad p = 0 \text{ in } \Omega.$$

Following [24] we present the results of computation at time $t = 1.17$.

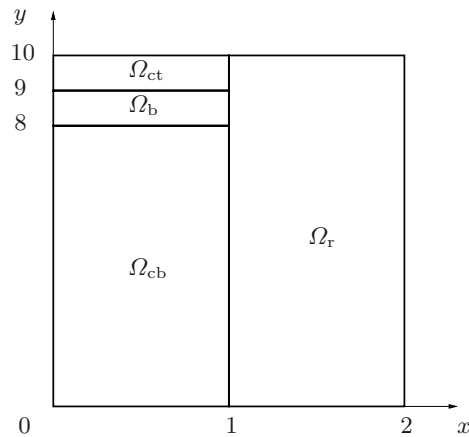


FIGURE 11. Bullet in the channel initial geometry. The computational domain is stretched along the x -axis to ease representation.

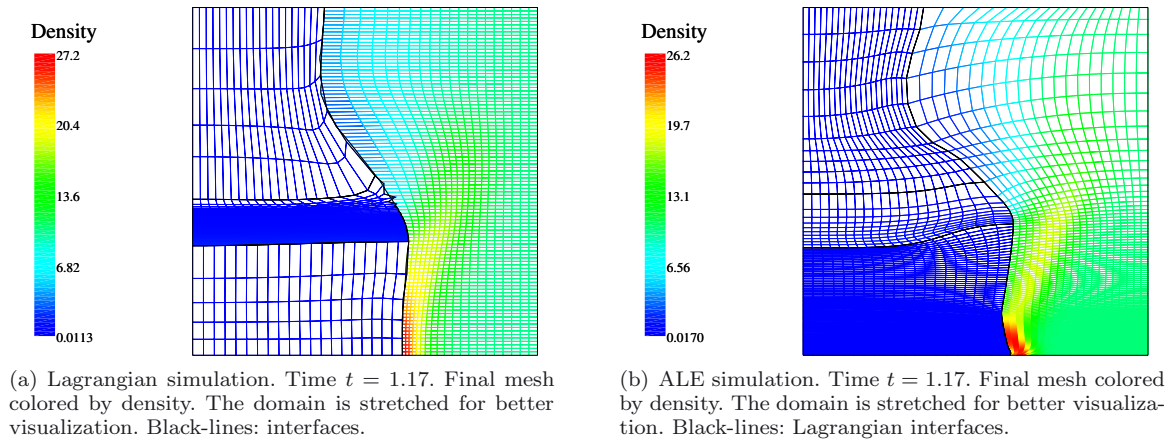


FIGURE 12. Bullet in channel.

We run this test using two strategies.

Lagrangian: Each subdomain is meshed as follows. Ω_r : 20×100 , Ω_{ct} : 20×5 , Ω_b : 20×90 and Ω_{cb} : 20×5 . The subzone entropy parameter is set to 0.5 in the whole domain.

ALE: The domain is meshed uniformly using 40×100 cells. Subzone entropy is set to 0.1 on the slide line and to 0 elsewhere. All interfaces are kept Lagrangian as well as Ω_b itself. The Lagrange-advection process we use is the one described in section 5 choosing θ in (5.1) so that Ω_b 's sliding nodes are relocated to Ω_{ct} , Ω_b or Ω_{cb} 's sliding nodes which are kept Lagrangian.

With the exception of Ω_b and the sliding interfaces, the grids are smoothed using five iterations of the method proposed by Escobar *et al.* [19] after each Lagrangian step.

The results of both simulations are respectively presented in Figures 12a and 12b. One can check that both results are quite similar according to the differences of mesh strategy: Ω_b 's final interfaces are located around the same places, which indicates that flows behaviors are comparable. Also, the geometry of the interface between Ω_r and $\Omega_{ct} \cup \Omega_{cb}$ seems more precisely described with the ALE strategy which was expected since Ω_{ct} and Ω_{cb} 's meshes are very coarse in the Lagrangian case. Also, the density distributions look quite similar.

One can notice oscillations on the top of the interface between Ω_b and Ω_r in the Lagrangian case. While, these are not small, they are amplified by the scaling that is used in Figure 12a and are not killing the calculus.

From our point of view, this very challenging case is a robustness test. Achieving the Lagrangian simulation requires an excessive use of subzone entropy, which is not satisfactory since numerical viscosity might be too important. However, let us recall that even with such an important coefficient, the scheme remains consistent and conservative and the comparison with the ALE calculation tends to show that the solution is acceptable.

7. CONCLUSIONS AND PERSPECTIVES

In this paper, we present a new framework to model slide lines for semi-Lagrangian calculations.

The abstract scheme we developed is defined under very weak hypothesis on grid velocities and multiplier's regularity. Indeed, we only assume $(\mathbf{u}_1^*, \mathbf{u}_2^*, \lambda) \in \mathcal{V} = L^2(\mathcal{E}_1)^d \times L^2(\mathcal{E}_2)^d \times L^2(\Gamma)$ to show conservation of mass, momentum and total energy. Also, we showed that the abstract scheme is Galilean invariant and that the entropy production error and volume conservation error are located in the slide line.

Actually, the method is built following a mortar-like [5] approach based on the scheme we defined in [15], where the grid velocity is solution of a minimization problem (a weighted L^2 -projection). Section 2 recalls the method presented in [15] and precises some points that were not given in the short note.

The sliding method we defined in this paper summarizes as a minimization problem for the grids velocity calculation under the constrain of the continuity of the normal velocity.

Thanks to the abstractness of the sliding framework, any conformal approximation $\mathcal{V}_h \subset \mathcal{V}$ will inherit the properties, established in Section 3, provided the arising linear system is solved exactly. This is a strength of the approach that allows the construction of many different practical schemes. For instance, one could consider $\mathbb{P}_1 - \mathbb{P}_1$ approximation.

Here, as a first guess, we focused to the $\mathbb{P}_1 - \mathbb{P}_0$ case. The reason for this choice was that since λ acts as a pressure, we had no reason to impose any regularity. Others choices might lead to supplementary properties that remain to be analyzed.

We analyzed the method, showing that under the constraint that Γ 's mesh is locally coarser than the finest of Γ_i 's meshes, the linear system, arising from the $\mathbb{P}_1 - \mathbb{P}_0$ discretization of the saddle point problem, has a unique solution. Also, under the same hypothesis, we showed a stability and convergence result.

We presented some numerical tests that illustrate that the approach is fine, Lagrangian and ALE calculations comparing quite well. This is good news since many improvements can clearly be made to the method, as we discuss bellow. Also, these first tests (especially in Sect. 6.4) indicate that other stabilization methods should improve the approach. Good candidates can be found in [24] for instance. Observe that our ALE procedure looks more robust, which was expected. We believe that more realistic calculations would rather involve elasto-plastic materials, that should be more stiff and lead to more stable flows, rather than focusing on gas/gas sliding, which is the hardest case.

Finally we emphasize that the abstract formulation allows to envisage higher-order discretization of the geometry at the interface. For instance one could use parabolic edges in order to reach a third-order geometric error. In this particular case for instance, one would need a \mathbb{P}_2 discretization for the grid velocity. Then many possible choices would be possible for multiplier's space as soon as it satisfies a discrete inf-sup condition (\mathbb{P}_1 , discontinuous- \mathbb{P}_1 , \mathbb{P}_2, \dots). Also, in that case, a mesh generation strategy for Γ is to be defined. Moreover, in order to maintain third-order geometry, we believe that the whole scheme must be third-order and even if [15] seems to be a first step, it remains to be done in the non-sliding case.

REFERENCES

- [1] I. Babuška, The finite element method with Lagrangian multipliers. *Numer. Math.* **20** (1973) 179–192.
- [2] A.L. Bauer, D.E. Burton, E.J. Caramana, R. Loubère, M.J. Shashkov and P.P. Whalen, The internal consistency, stability, and accuracy of the discrete, compatible formulation of Lagrangian hydrodynamics. *J. Comput. Phys.* **218** (2006) 572–593.
- [3] F. Ben Belgacem, The mortar finite element method with Lagrange multipliers. *Numer. Math.* **84** (1999) 173–199.

- [4] D.J. Benson, Computational methods in Lagrangian and Eulerian hydrocodes. *Comput. Meth. Appl. Mech. Engrg.* **99** (1992) 235–394.
- [5] C. Bernardi, Y. Maday and A.T. Patera, A New Nonconforming Approach to Domain Decomposition: The Mortar Element Method. *Nonlin. Partial Differ. Equ. Appl.* Edited by H. Brezis and J. L. Lions. Pitman, New York (1994) 13–51.
- [6] N.G. Bourago and V.N. Kukudzhanov, A Review of Contact Algorithms. The Institute for Problems in Mechanics of RAS. *Izv. RAN, MTT Translation into english* (2005) 45–87.
- [7] J.P. Braeunig, B. Desjardin and J.M. Ghidaglia, A totally Eulerian finite volume solver for multi-material fluid flows. *Eur. J. Mech. B/Fluids* **28** (2009) 475–485.
- [8] F. Brezzi and M. Fortin, *Mixed and Hybrid Finite Element Methods*. Springer–Verlag, New York (1991).
- [9] F. Brezzi and L.D. Marini, Macro Hybrid Elements and Domain Decomposition Methods. In Vol. 89 of *Optimisation et Contrôle, Meeting in honour of J. Céa*, edited by J.D. et al. CÉPADUÈS–Edition, Toulouse (1993) (1992).
- [10] E.J. Caramana, The implementation of slide lines as a combined force and velocity boundary condition. *J. Comput. Phys.* **228** (2009) 3911–3916.
- [11] E.J. Caramana, D.E. Burton, M.J. Shashkov and P.P. Whalen, The construction of compatible hydrodynamics algorithms utilizing conservation of total energy. *J. Comput. Phys.* **146** (1998) 227–262.
- [12] G. Carré, S. Del Pino, B. Després and E. Labourasse, A cell-centered Lagrangian hydrodynamics scheme in arbitrary dimension. *J. Comput. Phys.* **228** (2009) 5160–5183.
- [13] G. Clair, B. Després and E. Labourasse, A one-mesh method for the cell-centered discretization of sliding. *Comput. Meth. Appl. Mech. Engrg.* **269** (2014) 315–333.
- [14] A. Claisse, P. Rouzier and J.M. Ghidaglia, A 2D Sliding Algorithm for Eulerian Multimaterial Simulations. In *ECCOMAS 6th European Congress on Computational Methods in Applied Sciences and Engineering* (2012).
- [15] S. Del Pino, A curvilinear finite-volume method to solve compressible gas dynamics in semi-Lagrangian coordinates. *C. R. Acad. Sci. Paris, Ser. I* **348** (2010) 1027–1032.
- [16] B. Després and E. Labourasse, Stabilization of cell-centered compressible Lagrangian methods using subzonal entropy. *J. Comput. Phys.* **231** (2012) 6559–6595.
- [17] B. Després and C. Mazeran, Lagrangian gas dynamics in two dimensions and Lagrangian systems. *Arch. Rational Mech. Anal.* **178** (2005) 327–372.
- [18] V. Dyadeshko and M. Shashkov, Reconstruction of multi-material interfaces from moment data. *J. Comput. Phys.* **227** 5361–5384 (2008)
- [19] J.M. Escobar, E. Rodríguez, R. Motenegro and G.M. Montero, G.Y.J., Simultaneous untangling and smoothing of tetrahedral meshes. *Comput. Meth. Appl. Mech. Engrg.* **192** (2003) 2775–2787.
- [20] G. Folzan, *Modélisation multi-matériaux multi-vitesse en dynamique rapide*. Under the direction of P. Le Tallec and J.-P. Perlat (in french). Ph.D. thesis, École Polytechnique (2013).
- [21] G.H. Golub and C.F. Van Loan, *Matrix Computations*, 3rd edition. John Hopkins University Press (1996).
- [22] C.W. Hirt and B.D. Nichols, Volume Of Fluid (VOF) method for the dynamics of free boundaries. *J. Comput. Phys.* **39** (1981) 201–225.
- [23] B.I. Jun, A modified equipotential method for grid relaxation. Tech. Rep. UCRL-JC-138277. Lawrence Livermore National Laboratory (2000)
- [24] M. Kucharik, R. Loubère, L. Bednárík and R. Liska, Enhancement of Lagrangian Slide Lines as a Combined for and Velocity Boundary Condition. *Comput. Fluids* (2012).
- [25] X.S. Li, An Overview of SuperLU: Algorithms, Implementation and User Interface. In Vol. 31 (2005) 302–325.
- [26] P.H. Maire, R. Abgrall, J. Breil and J. Ovardia, A cell-centered Lagrangian scheme for two-dimensional compressible flow problems. *SIAM J. Sci. Comput.* **29** (2007) 1781–1824.
- [27] C. Mazeran, *Sur la structure mathématique et l’approximation numérique de l’hydrodynamique Lagrangienne bidimensionnelle*. Under the direction of B. Després (in french). Ph.D. thesis, Université Bordeaux I (2007).
- [28] N.R. Morgan, M.A. Kenamond, D.E. Burton, T.C. Carney and D.J. Ingraham, An approach for treating contact surfaces in Lagrangian cell-centered hydrodynamics. *J. Comput. Phys.* **250** (2013) 527–554.
<http://dx.doi.org/10.1016/j.jcp.2013.05.015>
<http://www.sciencedirect.com/science/article/pii/S002199911300346X>
- [29] J. von Neumann and R.D. Richtmyer, A method for the calculation of hydrodynamics shocks. *J. Appl. Phys.* **21** (1950) 232–237.
- [30] O. Steinbach, On a generalized L_2 projection and some related stability estimates in Sobolev spaces. *Numer. Math.* **90** (2002) 775–786.
- [31] R. Tipton, Grid optimization by equipotential relaxation. Unpublished manuscript (1990).
- [32] M.L. Wilkins, Calculation of Elastic-Plastic Flow. In Vol. 3 of *Meth. Comput. Phys.* Academic Press (1964) 211–263.
- [33] D.L. Youngs, Time dependent Multi-Material Flow with Large Fluid Distortion. In *Numer. Methods Fluid Dyn.* Edited by K.W. Morton, M.J. Baines (1982) 273–285
- [34] Y.B. Zel’dovich and Y.P. Raizer, *Physics of Shock Waves and High-Temperature Hydrodynamic Phenomena*. Vol. 1. Academic Press, New York and London (1966).