A one-mesh method for the cell-centered discretization of slide lines

Guillaume Clair, Bruno Després, Emmanuel Labourasse

A one-mesh method for the cell-centered discretization of slide lines

G. Clair[a,*], B. Despres[b], E. Labourasse[c]

[a] *Eurobios, 86 avenue Lénine, 94250 Gentilly, France*
[b] *UPMC Univ Paris 06, UMR 7598, Laboratoire J.-L. Lions, F-75005 Paris, France*
[c] *CEA, DAM, DIF, F-91297 Arpajon, France*

Abstract

A new method is described to handle slide lines in cell-centered Lagrangian schemes for the modeling of sliding problems between two fluids in the framework of compressible hydrodynamics. The method is an extension of the one proposed in the reference [1] and is conservative in momentum and total energy. Our method is based on the minimization of an objective function over a specific set that models the sliding constraint. We illustrate on several basic problems.

Keywords:
Slide lines, Constraints, Minimization method, Lagrangian Hydrodynamic, Compressible Gas dynamics

1. Introduction

Slide lines are particularly useful in Lagrangian schemes when large relative displacements occur between two distinct regions within a system (that may be a solid, a fluid or a mixture of both). They prevent the mesh tangling due to an extreme distortion which may cause the crash of the simulation. In Lagrangian formulation, the slide line is generally defined as a list of several nodes of the mesh whose motion is described by a specific method. In the literature, various methods have been proposed, most of them are reviewed in the recent survey of Burago [2]. Some of them started from the seminal work of Wilkins [3], see for instance the references [4, 5, 6, 7, 8] and the references therein. Such works are usually based on a master-slave procedure in which a set of nodes in the mesh associated with one side of the interface defines a line or surface (*master*) on which the nodes of the opposite side can slide (*slave*). This imposes to define arbitrarily a master side, which in most applications is naturally associated to the fluid with the highest inertia (that is the fluid with the highest density). The interaction of both sides of the slide line is then explicitly calculated, and their inter-penetration is prevented by an explicit *put-back-on* step [3].
Historically, most of the methods devoted to the treatment of slide lines were described for staggered schemes. Recently, new methods adapted to cell-centered Lagrangian schemes[9,

*corresponding author: G. Clair
Email address: gclair.recherche@gmail.com (G. Clair)

10, 11] have been published in the literature (readers can refer to [12, 13] for recent extensions of these schemes to slide lines). These methods exploit the advantage of using two discrete versions of the velocity in the mesh, the first one centered in each cell, which is the physical velocity computed from the law of conservation of the momentum, and the second one, staggered at the nodes and computed from a Riemann solver, which is used to ensure the geometric conservation law (GCL) condition and displace the nodes during the computation. Using two versions of the velocity therefore ensures the independence between any method influencing the motion of the nodes (such as a method treating slide lines) and the preservation of the momentum and total energy.
In this paper, we describe a new method to treat slide lines and adapted to cell-centered Lagrangian schemes. The method is an extension of the one proposed in our previous work [1]. In the latter, we reformulate the usual Riemann solvers to take into account any kind of constraints applying on a set of nodes in the mesh. The new formulation is based on the minimization of an objective function within a set of admissible velocities that models the constraints. Although only impact was studied in the previous reference, with the definition of convenient local constraints, we now evoke the possibility to extend the method to the treatment of the slide lines which now involves a global set of constraints. We expose in the first two parts of this paper the extension of our method to this context, using the numerical scheme GLACE [9], and show that the fundamental properties such as conservation of momentum and total energy are ensured. Then we apply our method to several numerical test cases which prove its efficiency and robustness.

2. Description of the method

2.1. Continuous formulation of the sliding problem

A schematics of the problem is presented on figure 1. Two fluids with different properties are in contact and separated by a contact interface Γ. The difference of velocity on each side of Γ involves a sliding, that we will always suppose pure (without any exchange of momentum throughout the slide line) in this work.

The properties of both fluids are described by the Euler equations by accounting the presence of the slide line. In the integral form, such equations are :

$$\begin{cases} \dfrac{d}{dt}\displaystyle\int_{V(t)} dV = \int_{V(t)} \nabla\cdot\mathbf{u}dV & \text{volume conservation,} \\ \dfrac{d}{dt}\displaystyle\int_{V(t)} \rho dV = 0 & \text{mass conservation,} \\ \dfrac{d}{dt}\displaystyle\int_{V(t)} \rho\mathbf{u}dV + \int_{V(t)} \nabla p dV = 0 & \text{momentum conservation,} \\ \dfrac{d}{dt}\displaystyle\int_{V(t)} \rho e dV + \int_{V(t)} \nabla\left(p\mathbf{u}\right) dV = 0 & \text{total energy conservation,} \end{cases} \tag{1}$$

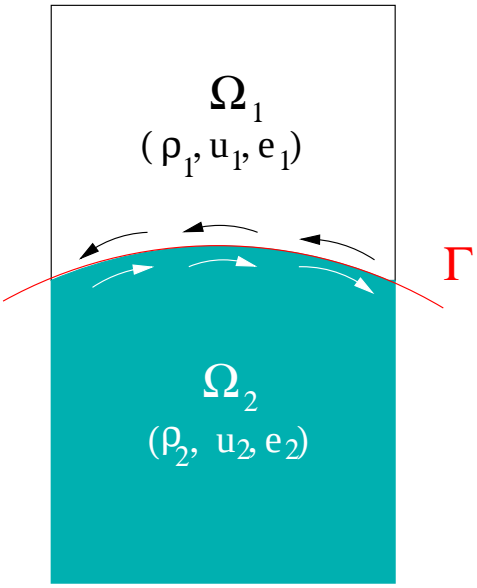

Figure 1: Simple schematics of a fluid sliding over another one. Γ is the slide line.

Here V(t) is a volume moving with the fluid and $\frac{d}{dt} = \frac{\partial}{\partial t} + \mathbf{u}\cdot\nabla$ is the material derivative. The physical variables are density ρ (and the specific volume $\tau = 1/\rho$), velocity $\mathbf{u}$, total energy e and pressure p. Across of the slide line, the normal component of the velocity must satisfy the sliding condition, that is:

$$\left(\mathbf{u}^+(\mathbf{x}(t),t) - \mathbf{u}^-(\mathbf{x}(t),t),\ \mathbf{n}(\mathbf{x}(t),t)\right) = 0 \quad \forall \mathbf{x} \in \Gamma \tag{2}$$

where the superscripts + and - denote values of a quantity on each side of the slide line. In this sense, $\mathbf{u}^+(\mathbf{x}(t),t)$ and $\mathbf{u}^-(\mathbf{x}(t),t)$ are the velocity vectors at the position $\mathbf{x}(t)$ and at time t on each side of Γ. The vector normal to Γ at the position $\mathbf{x}(t)$ is $\mathbf{n}(\mathbf{x}(t),t)$.

2.2. Semi-discrete nodal solver

For sake of simplicity, we describe the semi-discrete in space algorithm in this section. In this work, we will perform a Euler scheme for the time integration, but more sophisticated schemes could be adapted without difficulty. The semi-discrete formulation of the system

(1) we use in this paper comes from the work of B. Després in the references [9, 14]:

$$M_j \tau_j'(t) = \sum_{r \in \mathcal{N}(j)} (\mathbf{C}_{j,r},\ \mathbf{u}_r) \tag{3}$$

$$M_j \mathbf{u}_j'(t) = - \sum_{r \in \mathcal{N}(j)} \mathbf{C}_{j,r} p_{j,r} \tag{4}$$

$$M_j e_j'(t) = - \sum_{r \in \mathcal{N}(j)} (\mathbf{C}_{j,r},\ \mathbf{u}_r)\, p_{j,r} \tag{5}$$

$$p_{j,r} = p_j + \rho_j c_j \left(\mathbf{u}_r - \mathbf{u}_j,\ \frac{\mathbf{C}_{j,r}}{|\mathbf{C}_{j,r}|} \right) \tag{6}$$

where $M_j(t)$, $\mathbf{u}_j(t)$ and $e_j(t)$ are respectively the mass, the velocity vector and the total energy within the j-th cell of the mesh. $\mathcal{N}(j)$ denotes the set of nodes belonging to the j-th cell. $\mathbf{C}_{j,r} = \nabla_{\mathbf{x}_r} V_j$ is a geometric vector depending on the volume V_j of the j-th cell. $p_{j,r}$ may be seen as the pressure that the j-th cell exerts on the r-th node (also called *nodal pressure*). In the GLACE scheme in particular, the nodal pressure is computed from a linearized Riemann-invariant relation in the direction of the $\mathbf{C}_{j,r}$, which ensures the entropy property. Such relation is given by (6).
In the GLACE scheme, two versions of the velocity variables are used: on one side, $\mathbf{u}_j$ is the velocity vector of the fluid at the center of the j-th cell which is commonly computed from the equation of conservation of momentum. On the other side, $\mathbf{u}_r$ is the velocity of the r-th node within the mesh, which is involved in the computation of the numerical fluxes in the equations of conservation of both mass and total energy. The latter velocity is computed at each node within the mesh from the following Riemann solver:

$$\forall r \in [1:N], \qquad \sum_{j \in \mathcal{C}(r)} \mathbf{C}_{j,r} p_{j,r} = 0$$

where $\mathcal{C}(r)$ denotes the set of cells which the r-th belongs to. The latter relation is equivalent to solve for each node the linear system:

$$\forall r \in [1:N], \qquad A_r \mathbf{u}_r = \mathbf{b}_r \tag{7}$$

A_r is a $d \times d$ symmetric positive-definite matrix (d is the dimension of the problem - in this paper, $d = 2$). Expressions of A_r and $\mathbf{b}_r$ may be found in the reference [9].

We have shown in our previous work that the use of two distinct velocities is very well adapted to account for any constraint applying on a list of nodes of the computational mesh [1]. Indeed, the Riemann solver may be reformulated as a minimization problem, where the function to minimize can be written as:

$$\begin{aligned} J : \mathbb{R}^{N \times d} &\to \mathbb{R} \\ \mathbf{U} &\to \frac{1}{2} (A\mathbf{U},\ \mathbf{U}) - (\mathbf{B},\ \mathbf{U}) \end{aligned} \tag{8}$$

where $\mathbf{U} = (\mathbf{u}_1,\ \mathbf{u}_2, \ldots,\ \mathbf{u}_N)^T$ and:

$$A = \begin{pmatrix} A_1 & & \ldots & & 0 \\ & \ddots & & & \\ \vdots & & A_r & & \vdots \\ & & & \ddots & \\ 0 & & \ldots & & A_N \end{pmatrix} \qquad \mathbf{B} = \begin{pmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_r \\ \vdots \\ \mathbf{b}_N \end{pmatrix}$$

Given that the matrices A_r are symmetric and positive definite for all r, A has the same property and J is therefore a strictly convex function.
The solution of any unconstrained problem may be seen as the minimum of the objective convex function J that lies in the set $\mathbb{R}^{N\times d}$. In the case of a constrained problem, a global minimum will be searched within a set of *admissible velocities* $\mathbb{K}$ that models all the constraints that apply on the computational mesh. If we suppose that K is a non empty, closed and convex set, this minimum is unique and will be denoted $\mathbf{U}_{\min}$, and we have:

$$\boxed{\mathbf{U}_{\min} = \underset{\mathbf{U}\in\mathbb{K}}{\operatorname{argmin}}\ J(\mathbf{U})} \tag{9}$$

The relation (9) will be referred from now as the *nodal solver*.

The semi-discrete formulation of the slide condition (2) will be expressed as:

$$\forall k \in \mathbb{I}, \qquad \left(\mathbf{u}_k^+(t) - \mathbf{u}_k^-(t),\ \mathbf{n}_k^{\mathrm{RES}}(t)\right) = 0 \tag{10}$$

and the expression of $\mathbb{K}$ in the case of a pure sliding is therefore:

$$\mathbb{K} = \{\mathbf{U} \in \mathbb{R}^{N\times d}, \quad \forall k \in I, \quad \left(\mathbf{u}_k^+(t) - \mathbf{u}_k^-(t),\ \mathbf{n}_k^{\mathrm{RES}}(t)\right) = 0\} \tag{11}$$

In the relation (11), $I \subset N$ denotes the set of indices of the nodes that belong to the slide line, $\mathbf{u}_k^+(t)$ and $\mathbf{u}_k^-(t)$ are the velocity vectors respectively on each side of Γ computed at the node k and time t, and $\mathbf{n}_k^{\mathrm{RES}}$ is a local normal defined at the node k at time t. Notations $\mathbf{u}_k^+(t)$ and $\mathbf{u}_k^-(t)$ may be quite ambiguous at first sight, since two distinct velocity vectors are defined for a single node. The definition of each term used in the relation (11) will be precisely given in the section below.

2.3. Formulation of the discrete velocities and normals along the slide line

We illustrate our purposes with the figure (2). Let Ω be the domain of computation. We respectively denote Ω_1 and Ω_2 the subset occupied by the fluids 1 and 2 at any time. Let us denote $\mathcal{M}_1$ (respectively $\mathcal{M}_2$) the part of the mesh used to discretize the domain Ω_1 (respectively the domain Ω_2). To clarify our work, on each figure gathering $\mathcal{M}_1$ and $\mathcal{M}_2$, the nodes belonging to $\mathcal{M}_1$ will be filled in black, while the nodes of $\mathcal{M}_2$ will be filled in white. With these notations, we have:

$$\mathcal{M} = \mathcal{M}_1 \cup \mathcal{M}_2 \tag{12}$$

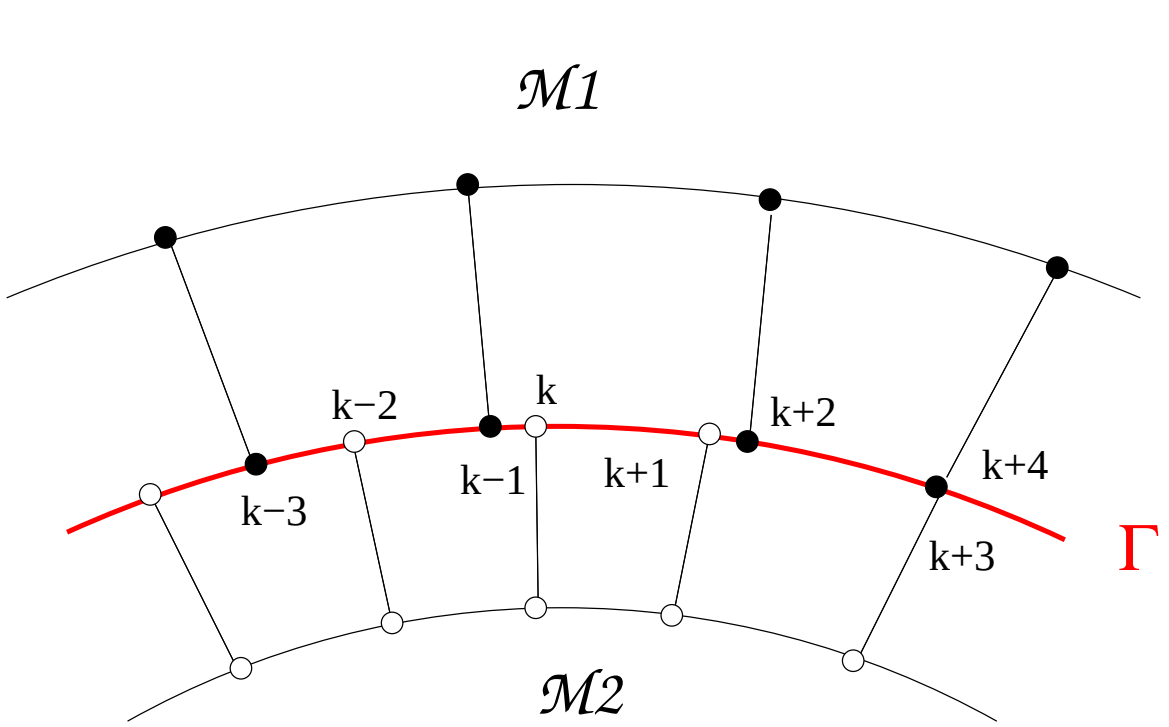

Figure 2: Example of meshes used for the computation

In our modeling, the slide line Γ is composed of several nodes that belong either to $\mathcal{M}_1$ or to $\mathcal{M}_2$. Thus we denote I_1 the set of indices of the nodes on Γ belonging to the mesh $\mathcal{M}_1$. Similarly, I_2 is the set of indices of the nodes on Γ belonging to the mesh $\mathcal{M}_2$. With theses notations:

$$I = I_1 \cup I_2 \tag{13}$$

On the figure (2), we may evidence two situations:

1. Nodes belonging to $\mathcal{M}_1$ may coincide with nodes belonging to $\mathcal{M}_2$. See for instance the nodes $k+3$ and $k+4$. Such nodes will be referred as coincident nodes.

2. In the general cases, nodes are not coincident, and such nodes will be referred as exceptional nodes [15]. Such denomination will be clearly explained in the section 2.6.

In this framework, we define in the next section the terms $\mathbf{u}_k^+(t)$ and $\mathbf{u}_k^-(t)$ for all $k \in I$.

2.3.1. Definition of $\mathbf{u}_k^+(t)$ and $\mathbf{u}_k^-(t)$

Let us consider the node k on the slide line on figure 2. The latter belongs to the mesh $\mathcal{M}_2$. Therefore, in the expression of the difference in velocity, one of the velocity vectors $\mathbf{u}_k^+(t)$ or $\mathbf{u}_k^-(t)$ will be set naturally to $\mathbf{u}_k$. For instance, we set $\mathbf{u}_k^+(t) = \mathbf{u}_k$.
We have then to define the velocity $\mathbf{u}_k^-(t)$ on the slide line Γ relatively to $\mathcal{M}_1$. For this purpose, we require the existence of a fictive node g (namely a *ghost node*), at the same position of the node k, but belonging to the mesh $\mathcal{M}_1$. Precisely, the node g belongs to the

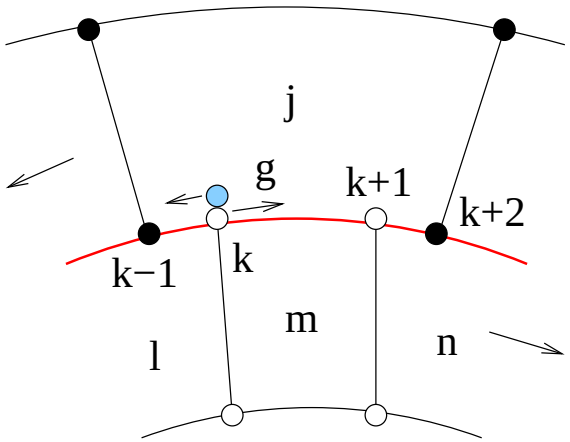

Figure 3: Introduction of a ghost node for the discrete formulation of the sliding condition

cell j. This is illustrated on the figure 3, which is a zoom of the figure 2 in the neighborhood of the node k. On this figure, we intentionally separate the position of nodes g and k to clarify our illustration, but readers have to keep in mind that they are coincident at the initial instant of the sliding. Moreover, arrows indicate the direction of motion for each mesh and for the nodes k and g. In this example, we suppose that $\mathcal{M}_1$ globally moves to the left (so does the node g), while $\mathcal{M}_2$ moves to the right (so does the node k). The velocities of nodes g and k must then satisfy the sliding condition (2). Denoting $\mathbf{u}_k^g(t)$ the velocity of the node g, we set $\mathbf{u}_k^-(t) = \mathbf{u}_k^g(t)$.

Following the work of A. Claisse [15] about the treatment of exceptional node in cell-centered Lagrangian schemes, we admit that the velocity of the ghost node $\mathbf{u}_k^g(t)$ is a combination of the velocity of its closest neighbours in the cell j, that are nodes $k-1$ and $k+2$ on the figures 2 and 3. Thus, we denote:

$$\mathbf{u}_k^g(t) = P_1^k(t)\mathbf{u}_{k-1}(t) + P_2^k(t)\mathbf{u}_{k+2}(t) \tag{14}$$

where $(P_1^k,\ P_2^k) \in \mathbb{R}^{2\times 2} \times \mathbb{R}^{2\times 2}$ and satisfy the relation:

$$\forall k \in I, \qquad P_1^k(t) + P_2^k(t) = I_2$$

I_2 is the identity matrix.

The general form of $P_1^k(t)$ and $P_2^k(t)$ may be found geometrically, see the reference [15]. The discrete formulation of the difference in velocity for the node k then writes:

$$\mathbf{u}_k^+(t) - \mathbf{u}_k^-(t) = \mathbf{u}_k(t) - \mathbf{u}_k^g(t) = \mathbf{u}_k(t) - \left(P_1^k(t)\mathbf{u}_{k-1}(t) + P_2^k(t)\mathbf{u}_{k+2}\right)$$

For all $k \in I$, the previous relation easily generalizes to:

$$\boxed{\forall t > 0, \qquad \forall k \in I, \qquad \mathbf{u}_k^+(t) - \mathbf{u}_k^-(t) = \mathbf{u}_k(t) - \left(P_1^k(t)\mathbf{u}_k^L(t) + P_2^k(t)\mathbf{u}_k^R(t)\right)} \tag{15}$$

$\mathbf{u}_k^L(t)$ and $\mathbf{u}_k^R(t)$ are defined as the closest neighbors of the ghost node introduced from the node k in the opposite mesh ('L' and R denotes Left and Right). The use of a ghost node finally becomes implicit for the discrete formulation of the difference in velocity.

Let us write the relation (15) for the nodes $k+1$ and $k+3$ on figure 2. For the node $k+1$, neighbors are the same as those of node k, but matrices P_1 and P_2 are different. We suppose that during the simulation, the node $k+3$ becomes coincident with the node $k+4$. Naturally, relation (15) leads to:

$$\begin{aligned} P_1^{k+3} &= P_1^{k+4} = I_2 \\ P_2^{k+3} &= P_2^{k+4} = 0 \\ \mathbf{u}_{k+3}^L &= \mathbf{u}_{k+3}^R = \mathbf{u}_{k+4} \\ \mathbf{u}_{k+4}^L &= \mathbf{u}_{k+4}^R = \mathbf{u}_{k+3} \end{aligned}$$

and so:

$$\mathbf{u}_{k+3}^+ - \mathbf{u}_{k+3}^- = \mathbf{u}_{k+4}^+ - \mathbf{u}_{k+4}^- = \mathbf{u}_{k+3} - \mathbf{u}_{k+4}$$

Let emphasis we choose a completely explicit time discretization for the constraint. It gives the discrete formulation of the relation (15):

$$\boxed{\forall n \in \mathbb{N}, \quad \forall k \in I, \quad \mathbf{u}_k^{+,n} - \mathbf{u}_k^{-,n} = \mathbf{u}_k^n - \left(P_1^{k,n} \mathbf{u}_k^{L,n} + P_2^{k,n} \mathbf{u}_k^{R,n} \right)} \tag{16}$$

In the next section, we present the method to compute the normal $\mathbf{n}_k^{\mathrm{RES}}(t)$.

2.3.2. Definition and computation of the normal $\boldsymbol{n}_k^{RES}(t)$

The discrete formulation of the normal n(x(t)) is arbitrary. Principally for a reason of symmetry, we propose the following formulation:

$$\boxed{\forall t > 0, \quad \forall k \in I, \quad \mathbf{n}_k^{\mathrm{RES}}(t) = \frac{1}{2} \left(\mathbf{n}_k^+(t) + \mathbf{n}_k^-(t) \right)} \tag{17}$$

where $\mathbf{n}_k^+(t)$ and $\mathbf{n}_k^-(t)$ are respectively the normals defined on each side of the slide line at the node k. Both terms have now to be defined.

Let us consider the two possible cases:

- if the node k is an exceptional node, we set $\mathbf{n}_k^+(t) = \mathbf{n}_k(t)$ and $\mathbf{n}_k^-(t) = \mathbf{n}_k^g(t)$. This choice is non restrictive and both normals can be permuted.
- if the node k coincides with a node $k+1$, then naturally $\mathbf{n}_k^+(t) = \mathbf{n}_k(t)$ and $\mathbf{n}_k^-(t) = \mathbf{n}_{k+1}(t)$. Again, indices can be permuted.

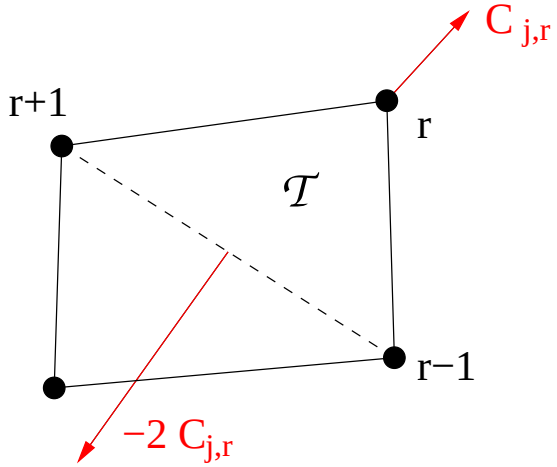

Figure 4: Graphical representation of the vector Cjr for a quadrangular cell j

Let us now compute the normals $\mathbf{n}_k(t)$ and $\mathbf{n}_k^g(t)$, $\forall k \in I$. Such computation is based on the definition and graphical representation of the vectors $\mathbf{C}_{j,r}$ used in the numerical fluxes. Our explanation refers to the figure 4. The expression of $\mathbf{C}_{j,r}$ at the r-th vertex in the j-th cell is:

$$\mathbf{C}_{j,r} = \frac{1}{2}\begin{pmatrix} y_{r+1} - y_{r-1} \\ x_{r-1} - x_{r+1} \end{pmatrix}$$

Thus, in the triangle $\mathcal{T}$, $\mathbf{C}_{j,r}$ represents half the opposite of a vector normal of the edge $\{r-1, r+1\}$, and the vector $\mathbf{C}_{j,r}/\left|\mathbf{C}_{j,r}\right|$ may be interpreted as a outward-pointing normal at the vertex r.

In the unconstrained version of the GLACE scheme [9], the following property holds for any node interior to the mesh:

$$\forall r \in \mathcal{M}/\partial\mathcal{M}, \qquad \sum_{j \in \mathcal{C}(r)} \mathbf{C}_{j,r} = 0$$

where $\partial\mathcal{M}$ denotes the boundary of the mesh $\mathcal{M}$. Such property ensures the preservation of the momentum. For a boundary node, the latter does not hold, and the resulting vector may be interpreting as a local outward-pointing normal.

Recall now a problem in which two distinct meshes are separated by a slide line. Any node that belongs to the slide line can be considered as a boundary node for its respective mesh. We refer to the figure 5, on which only one mesh has been drawn. At the node r, the sum:

$$\mathbf{n}_r(t) = \sum_{j \in \mathcal{C}(r)} \frac{\mathbf{C}_{j,r}}{\left|\mathbf{C}_{j,r}\right|}$$

represents a local outward-pointing normal. If we denote $\mathcal{T}$ the triangle whose vertices are $\{r-1,\ r,\ r+1\}$ on figure 5, then $\mathbf{n}_r(t)$ represents half the normal to the edge $\{r-1,\ r+1\}$.

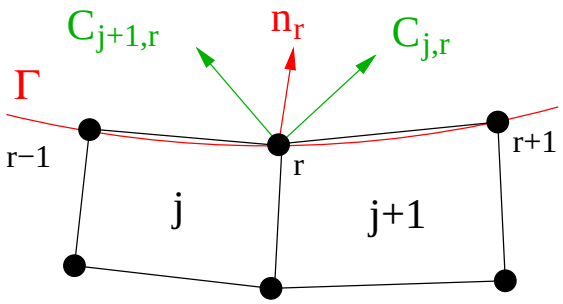

Figure 5: Graphical representation of the outward normal at any boundary node

Given that, we set:

$$\forall k \in I, \mathbf{n}_k(t) = \begin{cases} \dfrac{\sum\limits_{j \in \mathcal{C}(k)} \mathbf{C}_{j,k}}{\left| \sum\limits_{j \in \mathcal{C}(k)} \mathbf{C}_{j,k} \right|} & \text{if } k \in I_1 \\ -\dfrac{\sum\limits_{j \in \mathcal{C}(k)} \mathbf{C}_{j,k}}{\left| \sum\limits_{j \in \mathcal{C}(k)} \mathbf{C}_{j,k} \right|} & \text{if } k \in I_2 \end{cases} \tag{18}$$

The change in sign providing that k belongs either to I_1 or to I_2 is motivated by the fact that, in the case of two coincident nodes, both normals $\mathbf{n}_k^+$ and $\mathbf{n}_k^-$ have to point in the same direction. Otherwise, we might have $\mathbf{n}_k^+ = -\mathbf{n}_k^-$ and the resulting normal $\mathbf{n}_k$ becomes zero. Obviously, the choice of the direction of $\mathbf{n}_k^+$ and $\mathbf{n}_k^-$ is not restrictive (the signs in the relation (17) can be permuted).

Let us now compute the normal at the ghost node $\mathbf{n}_k^g(t)$, $\forall k \in I$. We have supposed that each ghost node closely depends on its neighbors in a specific cell. For instance, the ghost node on figure 3 depends on nodes $k-1$ and $k+1$ and belongs fictively to the cell j. In this sense, we may compute a local $\mathbf{C}_{j,g}$ for the ghost node, which is expressed as:

$$\mathbf{C}_{j,g} = \frac{1}{2} \begin{pmatrix} y_{k+1} - y_{k-1} \\ x_{k-1} - x_{k+1} \end{pmatrix}$$

The normal at the ghost node is then expressed as:

$$\mathbf{n}_k^g(t) = \begin{cases} -\dfrac{\mathbf{C}_{j,g}}{|\mathbf{C}_{j,g}|} & \text{if } k \in I_1 \\ \dfrac{\mathbf{C}_{j,g}}{|\mathbf{C}_{j,g}|} & \text{if } k \in I_2 \end{cases} \tag{19}$$

Again, the signs of $\mathbf{n}_k^g(t)$ depend on the ones in the expression of $\mathbf{n}_k(t)$ and have been chosen so that both normal must point in the same direction. If signs are permuted in the relation

(18), so are they in the relation (19).

We end this section by the fully discrete (in time) formulation of the relation (17):

$$\boxed{\forall n \in \mathbb{N}, \quad \forall k \in I, \quad \mathbf{n}_k^{\mathrm{RES},n} = \frac{1}{2}\left(\mathbf{n}_k^{+,n} + \mathbf{n}_k^{-,n}\right)} \tag{20}$$

for which all vectors $\mathbf{C}_{j,r}$ are evaluated at time t_n.

From relations (16) and (20), we give the discrete formulation of the set $\mathbb{K}$ in the next section.

2.4. Final expression and properties of $\mathbb{K}$

The full discrete expression of the set $\mathbb{K}$ is:

$$\boxed{\mathbb{K}_n = \{\mathbf{U} \in \mathbb{R}^{N\times d}, \quad \forall k \in I, \quad \left(\mathbf{u}_k^n - \left[P_1^{k,n}\mathbf{u}_k^{L,n} + P_2^{k,n}\mathbf{u}_k^{R,n}\right],\ \mathbf{n}_k^{\mathrm{RES},n}\right) = 0} \tag{21}$$

Note that the constraint may be rewritten as:

$$\forall k \in I, \quad {}^t\mathbf{n}_k^{\mathrm{RES},n}\mathbf{u}_k^n - {}^t\mathbf{n}_k^{\mathrm{RES},n}\left[P_1^{k,n}\mathbf{u}_k^{L,n} + P_2^{k,n}\mathbf{u}_k^{R,n}\right] = 0$$

If we denote k_L and k_R respectively the indices in the set I of the velocity vectors $\mathbf{u}_k^L$ and $\mathbf{u}_k^R$, then the previous relation can be reformulated in a matrix form $L\mathbf{U} = \mathbf{0}$. L is a $N \times 2N$ matrix which is defined as:

$$L = (l_{ij})_{\substack{1\le i\le N\\ 1\le j\le N}} = \begin{cases} {}^t\mathbf{n}_k^{\mathrm{RES},n}, & \text{if } j = k \\ -{}^t\mathbf{n}_k^{\mathrm{RES},n}P_1^k, & \text{if } j = k_L \\ -{}^t\mathbf{n}_k^{\mathrm{RES},n}P_2^k, & \text{if } j = k_R \end{cases} \tag{22}$$

Finally, we have:

$$\boxed{\mathbb{K}_n = \{\mathbf{U} \in \mathbb{R}^{N\times d}, L\mathbf{U} = \mathbf{0}\}} \tag{23}$$

Notice that all nodes of the sliding line Γ are coupled through this definition.

The discrete formulation (23) confers to $\mathbb{K}_n$ the following properties:

① $\mathbb{K}_n$ is non empty ($\mathbf{0} = (0, 0, \ldots, 0)^t \in \mathbb{K}_n$) and closed.

② $\mathbb{K}_n$ is convex.

These two properties guarantee the existence and the uniqueness (given that J is strictly convex) of the minimum $\mathbf{U}_{\min}$. In addition, $\mathbb{K}_n$ has the two important following properties:

③ Translations are elements of $\mathbb{K}_n$. A translation is an element that is written in the form $\mathbf{W_a}$, where $\mathbf{W_a} = (\mathbf{a}, \mathbf{a}, \ldots, \mathbf{a})^T$. Thus, elements in the form $\mathbf{U} + \mathbf{W_a}$ are in $\mathbb{K}_n$, which implies the preservation of the momentum [1].

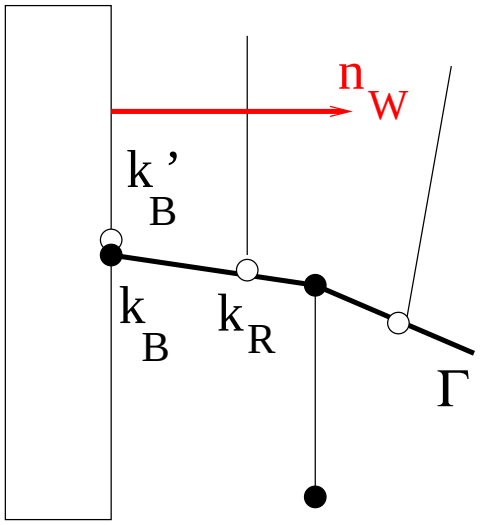

Figure 6: Schematics of boundary ndoes sliding on a wall. The normal to the wall is $\mathbf{n}_W$.

④ $\mathbb{K}_n$ is a cone, that is $\lambda\mathbf{U} \in \mathbb{K}_n,\ \forall\lambda > 0$ and $\forall\mathbf{U} \in \mathbb{K}_n$. This property naturally implies the preservation of total energy [1].

Now that the formulation of $\mathbb{K}_n$ has been clarified, an important remark about the conservation of volume will be proposed in the following section.

2.5. Treatment of the boundary nodes

There is no difference in treating nodes interior to the mesh and boundary nodes, except for boundary nodes on which may apply a wall condition. These ones must slide simultaneously over the slide line and the wall, so an additional sliding condition must be provided.

Let us consider the figure 6. We index the nodes sliding simultaneously over the slide line and the wall by k_B and k'_B. Again, we intentionally separate both nodes on the figure 6, but they remain coincident as long as they slide on the wall. Let us denote k_R the closest neighbor of both nodes in the sliding line. On the wall, nodes k_B and k'_B will remain coincident as long as they slide over the wall. They must satisfy the condition (10):

$$\left(\mathbf{u}_{k_B} - \mathbf{u}_{k'_B},\ \mathbf{n}_B\right) = 0 \tag{24}$$

where $\mathbf{n}_B$ is the normal of the edge defined by the pair of nodes $\{k_B,\ k_R\}$, that is $\mathbf{n}_B = (\mathbf{k}_B\mathbf{k}_R)^{\perp}$. We provide in the matrix L two additional conditions, that express the sliding over the wall:

$$\begin{cases} (\mathbf{u}_{k_B},\ \mathbf{n}_W) & = 0 \\ \left(\mathbf{u}_{k'_B},\ \mathbf{n}_W\right) & = 0 \end{cases} \tag{25}$$

where $\mathbf{n}_W$ is the normal at the wall.

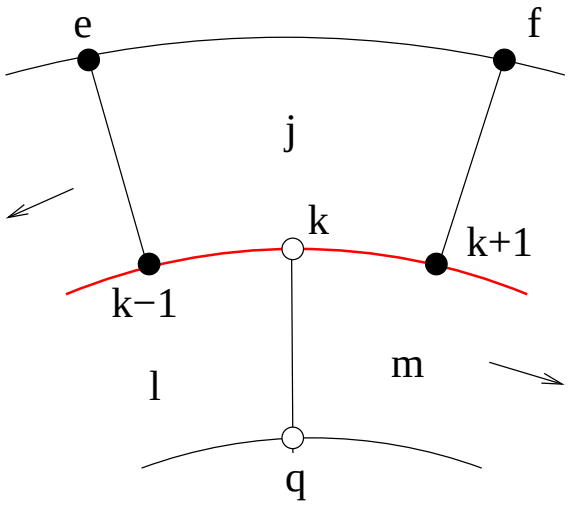

Figure 7: Illustration for the computation of the volume

2.6. Conservation of the volume

In the unconstrained version of the GLACE scheme [9], the computation of the volume of a cell with a specific geometry is expressed as:

$$\frac{d}{dt}V_j = \sum_{r\in\mathcal{N}(j)} \mathbf{C}_{j,r}\mathbf{u}_r \tag{26}$$

where the cardinal number of the set $\mathcal{N}(j)$ is equal to the number of nodes that define the cell. It is equal to 3 for a triangle cell and 4 for a quadrangle cell.

This relation is modified by the presence of a slide line. We use the figure 7 to illustrate our purposes. The volume of the cell j, which is a function of the nodes $\{k-1,\ k+1,\ e,\ f\}$ in the mesh $\mathcal{M}_1$, also depends on the exceptional node k. The latter must be therefore taken into account in the computation of its volume, which means that the cardinal number of the set $\mathcal{N}(j)$ is now 5. Then, we would write:

$$\begin{aligned}\frac{d}{dt}V_j &= \sum_{r\in\mathcal{N}(j)} \mathbf{C}_{j,r}\mathbf{u}_r + \mathbf{C}_{j,k} \\ &= \mathbf{C}_{j,k-1}\mathbf{u}_{k-1} + \mathbf{C}_{j,k+1}\mathbf{u}_{k+1} + \mathbf{C}_{j,e}\mathbf{u}_e + \mathbf{C}_{j,f}\mathbf{u}_f + \mathbf{C}_{j,k}\mathbf{u}_k\end{aligned} \tag{27}$$

Generally speaking, all exceptional nodes must be taken into account in the computation of the volume of the concerned cells in the opposite mesh. The term *exceptional* then refers to the nodes that do not clearly belong to a specific cell (itself belonging to the opposite mesh) but whose properties (the geometrical vector and the velocity) must be taken into account in the volume of such cell. In addition, we may name *exceptional cell* any cell whose volume depends on a single or several exceptional nodes.

A last remark ends this section: let us give a cell for which there is a single exceptional point. In the case of infinitely small timesteps (that is $\Delta t \to 0$), the ghost node and the exceptional node are nearly coincident and their velocity almost equal. In this case, we may reformulate the relation (27) by using the expression of the velocity of the ghost node (14):

$$\begin{aligned}
\frac{d}{dt}V_j &= \sum_{r\in\mathcal{N}(j)} \mathbf{C}_{j,r}\mathbf{u}_r \\
&= \mathbf{C}_{j,k-1}\mathbf{u}_{k-1} + \mathbf{C}_{j,k+1}\mathbf{u}_{k+1} + \mathbf{C}_{j,e}\mathbf{u}_e + \mathbf{C}_{j,f}\mathbf{u}_f + \mathbf{C}_{j,k}\mathbf{u}_k^g \\
&\approx \mathbf{C}_{j,k-1}\mathbf{u}_{k-1} + \mathbf{C}_{j,k+1}\mathbf{u}_{k+1} + \mathbf{C}_{j,e}\mathbf{u}_e + \mathbf{C}_{j,f}\mathbf{u}_f + \left({}^tP_1^k\mathbf{C}_{j,k}\mathbf{u}_{k-1} + {}^tP_2^k\mathbf{C}_{j,k}\mathbf{u}_{k+1}\right) \\
&\approx \left(\mathbf{C}_{j,k-1} + {}^tP_1^k\mathbf{C}_{j,k}\right)\mathbf{u}_{k-1} + \left(\mathbf{C}_{j,k+1} + {}^tP_2^k\mathbf{C}_{j,k}\right)\mathbf{u}_{k+1} + \mathbf{C}_{j,e}\mathbf{u}_e + \mathbf{C}_{j,f}\mathbf{u}_f
\end{aligned}$$

The two last relation say that the volume of any exceptional cell may be computed with the classical relation (26) but with a modification of the $\mathbf{C}_{j,r}$ vectors. These new geometrical vectors are labeled $\mathbf{D}_{j,r}$ and, with this notation, the computation of the volume, in the case of infinitely small timesteps, may be written as:

$$\frac{d}{dt}V_j \approx \sum_{r\in\mathcal{N}(j)} \mathbf{D}_{j,r}\mathbf{u}_r \tag{28}$$

These formulas can be used to analyze the variation of the volume of all cells.

3. Nodal solver

The nodal solver is given by the relation (9), which we remind here:

$$\mathbf{U}_{\min} = \underset{\mathbf{U}\in\mathbb{K}}{\operatorname{argmin}}\ J(\mathbf{U}) \tag{29}$$

The problem of minimization under constraints of an objective function, in our case the function J, is a subject that has been largely described in the literature [16, 17]. The standard procedure of resolution is to introduce the Lagrangian $\mathcal{L}$ related to (9), which is defined as:

$$\mathcal{L}(\mathbf{U},\ \lambda) = J(\mathbf{U}) + \boldsymbol{\lambda}\cdot F(\mathbf{U})$$

where $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_M)^T$ are the M Lagrange multipliers which referred to the sliding conditions applying on M nodes in the mesh. In the case of a pure sliding, M is equal to the cardinal number of I (see (13)). The function $\mathbf{F}(\mathbf{U}) = L\mathbf{U}$ models the sliding condition applying on each node that belongs to Γ.

The problem of a pure sliding is an equality-constrained problem, for which finding $\mathbf{U}_{\min}$ is equivalent to find the extremal point of the Lagrangian:

$$\nabla\mathcal{L} = 0 \begin{cases} A\mathbf{U} - \mathbf{B} + {}^tL\boldsymbol{\lambda} = 0 \\ L\mathbf{U} = 0 \end{cases} \tag{30}$$

From the first equation, $\mathbf{U}$ may be expressed as a function of $\boldsymbol{\lambda}$:

$$\mathbf{U} = A^{-1}\left(\mathbf{B} - {}^{t}L\boldsymbol{\lambda}\right) \tag{31}$$

which is then multiplied by L on the left:

$$0 = LA^{-1}\left(\mathbf{B} - {}^{t}L\boldsymbol{\lambda}\right)$$

$\boldsymbol{\lambda}$ may be now determine as the solution of the system:

$$\left(LA^{-1}\,{}^{t}L\right)\boldsymbol{\lambda} = LA^{-1}\mathbf{B} \tag{32}$$

Notice that the $N \times N$ matrix $LA^{-1}\,{}^{t}L$ is symmetric and non-negative. In our computation, we use the conjugate gradient algorithm [16] to solve the equation (32). We denote such solution $\boldsymbol{\lambda}_{\min}$. Finally, we deduce the solution $\mathbf{U}_{\min}$ from the relation (31). Note that we do not have any certainty about the positivity of the matrix $LA^{-1}\,{}^{t}L$, so that the solution of the equation (32) may be non unique. It may be then written as:

$$\boldsymbol{\lambda} = \boldsymbol{\lambda}_{\min} + \boldsymbol{\lambda}_0 \tag{33}$$

where $\boldsymbol{\lambda}_0 \in \mathrm{Ker}(LA^{-1}\,{}^{t}L)$ is arbitrary.

Proposition 3.1. *Ker*$(LA^{-1}\,{}^{t}L) =$ *Ker*$({}^{t}L)$

Proof. Obviously, $\mathrm{Ker}({}^{t}L) \subset \mathrm{Ker}(LA^{-1}\,{}^{t}L)$. The matrix $LA^{-1}\,{}^{t}L$ represents an endomorphism and is symmetric, so that we have the relations $\mathrm{Ker}(LA^{-1}\,{}^{t}L) = (\mathrm{span}(LA^{-1}\,{}^{t}L))^{\perp} \subset \mathrm{span}(L) = \mathrm{Ker}({}^{t}L)$. □

This means that the exact solution of the equation (32) is not required to determine the unique solution $\mathbf{U}_{\min}$ but may be found up to an element of $\mathrm{Ker}({}^{t}L)$.

3.1. Building the slide line

Our method requires an additional procedure that we name *the building procedure*. The objective is to design the neighbors of each node, which in consequence allows to compute the normal $\mathbf{n}^{\mathrm{RES}}$, the matrices P_1 and P_2 , and in a more general way the matrix L. The method we describe here is inspired from the seminal work of Wilkins [3]. To determine the neighbors of the node k (see figure 8), we successively consider each pair of consecutive nodes of the opposite mesh belonging to the slide line. Each pair belongs to a specific cell. For instance, the pair $\{k_J,\ k_L\}$ belongs to the cell $\mathcal{B}$, while the pair $\{k_L,\ k_R\}$ belongs to the cell $\mathcal{C}$. The procedure consists in computing the areas of two triangles involving the node k, the pair of nodes of the opposite mesh and the center of the cell that owns the pair of nodes. For instance, we consider the pair $\{k_J,\ k_L\}$ and compute the areas of the triangles $\{k,\ \mathbf{B},\ k_J\}$ and $\{k,\ \mathbf{B},\ kL\}$, where $\mathbf{B}$ is the center of the cell $\mathbf{B}$. We find that both areas are of the same sign, so that the node k is not exceptional for the cell $\mathbf{B}$. Now we consider the pair $\{k_L,\ k_R\}$. For this pair, the areas of the triangles $\{k,\ \mathbf{C},\ k_L\}$ and$\{k,\ \mathbf{C},\ k_R\}$ have opposite signs. In that latter case, it means that the node k is exceptional to the cell $\mathcal{C}$, so that the neighbors of the node k are nodes k_L and k_R.

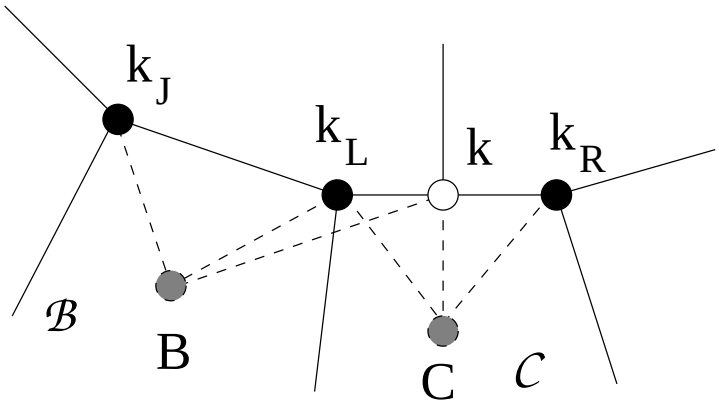

Figure 8: Illustration of the procedure to build the slide line

3.2. Condition on coincident nodes

When two nodes get closer and are expected to become coincident, we may observe that both nodes freeze on the slide line and move together as if there was a single node. Theoretically, the problem may be explained from the relation (10). When two nodes coincide, the normal defined for each node must be rigorously equal. This is given by the relation (6) and from the fundamental property of the GLACE scheme, that is:

$$\sum_{j \in \mathcal{C}(r)} \mathbf{C}_{j,r} = 0$$

It may arise that both normals may not be equal however. The most common reason is that a small interpenetration of meshes $\mathcal{M}_1$ and $\mathcal{M}_2$ occurs, or a small void separates the two nodes. In such cases, let us write the sliding condition (2) for each node, that we denote k and l:

$$\begin{gathered} \left(\mathbf{u}_k - \mathbf{u}_l,\ \mathbf{n}_k^{\mathrm{RES}}\right) = 0 \\ \left(\mathbf{u}_k - \mathbf{u}_l,\ \mathbf{n}_l^{\mathrm{RES}}\right) = 0 \\ \mathbf{n}_k^{\mathrm{RES}} \neq \mathbf{n}_l^{\mathrm{RES}} \end{gathered}$$

Both conditions means that the vector $\mathbf{u}_k - \mathbf{u}_l$ must be simultaneously orthogonal to both $\mathbf{n}_k^{\mathrm{RES}}$ and $\mathbf{n}_l^{\mathrm{RES}}$. In our minimization procedure, the unique solution is obviously $\mathbf{u}_k - \mathbf{u}_l = 0$. This implies that the nodes freeze and moves with a single velocity. At the theoretical level, this is most probably due to a hidden inf-sup condition that we do not analyze in this work. Instead we rely on the following procedure which reveals very efficient. Indeed, we impose that the normals of two nodes getting closer are strictly equal when the distance between them becomes smaller than a certain value ϵ. Such condition is theoretically expressed as:

$$\forall (k,l) \in I_1 \times I_2, |\mathbf{x}_k - \mathbf{x}_l| \leq \epsilon \Rightarrow \mathbf{n}_k^{\mathrm{RES}} = \mathbf{n}_l^{\mathrm{RES}} \tag{34}$$

By setting $\epsilon = 10^{-5}$ in our numerical tests, we have observed that such small even arbitrarily choice is enough to prevent any freezing pathologies.

3.3. CFL condition

Our method does not change the usual CFL condition used in the GLACE scheme, which is written in a general form [9, 11]:

$$\max_j \left(\frac{c_j}{\Delta x_j} \right) \Delta t \leq 1 \tag{35}$$

The definition of Δx_j is not obvious for an unstructured mesh. In our case, we use the following relation:

$$\Delta x_j = \frac{1}{V_j} \sum_{r \in \mathcal{N}(j)} \sqrt{\mathbf{C}_{j,r} \cdot \mathbf{C}_{j,r}} \tag{36}$$

In the 2D formalism, Δx_j is the ratio between the perimeter of the j-th cell divided by its surface.

Note that it is also possible to evaluate the time step restriction from the volume variation as described in [18].

3.4. Sketch of the algorithm

To conclude the numerical part, we summarize the steps required to update the physical quantities from time t^n to time t^{n+1}.

1. Build the slide line corresponding to time t^n, using the algorithm described in section 3.1.
2. Compute the geometrical features (normals) using equation (20).
3. Calculate the matrices $P_1^{k,n}$ and $P_2^{k,n}$ of equation (16) for all nodes of the slide line.
4. Express the matrix A and L, and the vector $\mathbf{B}$ of the system (30) and solve the linear system (32). Deduce the nodal velocity $\mathbf{U}_{\min}$ solution of the constrained problem.
5. Calculate the nodal pressures $p_{j,r}$ using equation (6).
6. Update the cell-centered velocities and specific total energies using equations 4 and 5.
7. Move the nodes thanks to the nodal velocities, compute the new volumes of the cells and deduce the new density.
8. Update the cell-centered pressure and sound speed with the equations of state.
9. Compute the new timestep and go back to step 1.

Note that step 5 to 9 are common with the classical solver described in [9, 14].

4. Numerical examples

The sliding algorithm is conservative in momentum and total energy as will be evidenced for several numerical examples. We think that no other published method exhibits this property.

4.1. Sliding between two mobiles

The most simple case is a pure sliding between two mobiles without any deformation of the

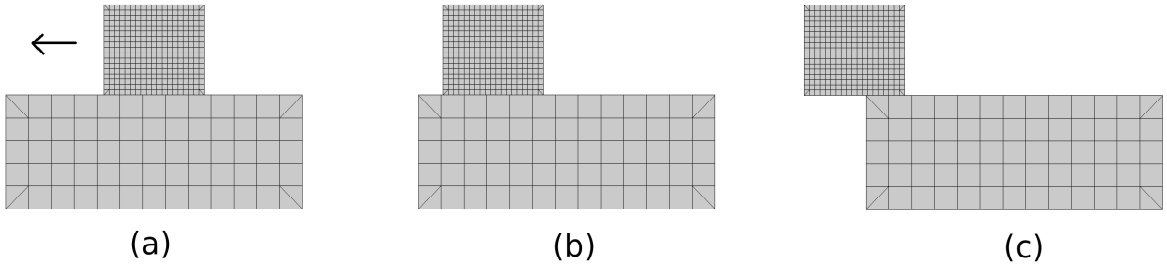

Figure 9: Pure sliding of a mobile over another one, without any deformation of the slide line. Results are given for three computational times: (a) t = 0, (b) t = 0.3 and (c) t = 0.7

slide line. It is primarily designed to check whether the procedure of treatment of the exceptional nodes in the mesh works well. The results proposed on the figure (9) illustrate the situation. We consider two mobiles in contact, the mobile beneath (that we denote by Ω_1) is fixed and the mobile above (denoted by Ω_2) moves to the left with a constant velocity on the general form $\mathbf{u}_0 = (u, 0)^T$. The results are given for $u = -1$. Let us denote $\mathcal{M}_1$ the mesh discretizing the mobile beneath and $\mathcal{M}_2$ the one discretizing the mobile above. Dimensions of Ω_1 is $L \times l = 0.13 \times 0.04$ and the mesh $\mathcal{M}_1$ is composed of square cells of side 0.01. Ω_2 have dimensions $L \times l = 4.10^{-2} \times 4.10^{-2}$ and $\mathcal{M}_2$ is composed of square cells of side $2,5.10^{-3}$.

On the figure (9), the slide line is set at the upper boundary of the mesh $\mathcal{M}_1$ (and the lower boundary of $\mathcal{M}_2$). Such test case simply shows the good behaviour of our method in the most basic case, that is when the slide line remains fixed during the computation. As expected, the simulation is stable and the good behaviour of the model is observed.

4.2. The Sod test case

We present the well-know Sod test case [19] in which an artificial slide line has been introduced. Such test case attempts to show that the presence of the slide does not disturb the symmetry, the stability and the convergence of the numerical solution. Let the domain of computation be a rectangle with dimensions $(L \times l : 1 \times 0.1)$. we first set the initial discontinuity at $x = 0.5$, so that the slide line is initially orthogonal to the flow. The mesh $\mathcal{M}_1$ at the left-hand side of the discontinuity is composed of square cells of side 5×10^{-3}, while

the mesh $\mathcal{M}_2$ at the right-hand side is composed with square cells of side 10^{-2}. Initially, some nodes coincide on the slide lines. Physical quantities of the left-hand side are initialized with $(\rho,\ u_X,\ u_Y, p)^T = (1, 0, 0, 1)^T$, while those at the right-hand side is initialized with $(\rho,\ u_X,\ u_Y, p)^T = (0.125, 0, 0, 0.1)^T$.

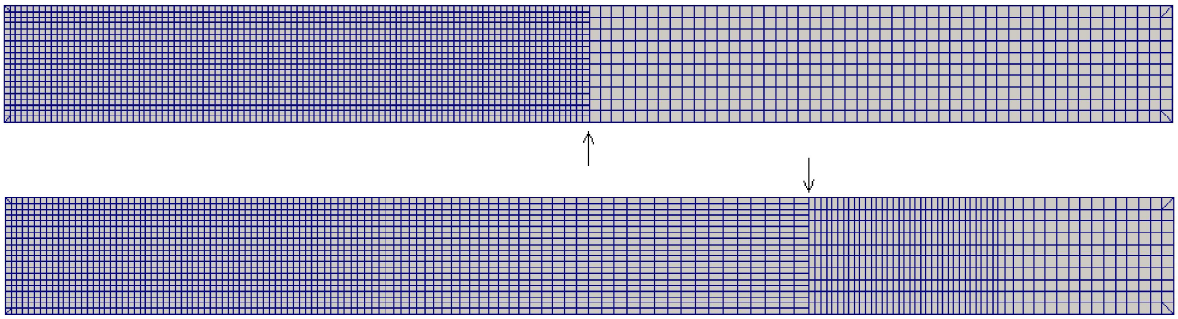

Figure 10: Presentation of two meshes resulted from the computation of the numerical solution in the Sod test case with the presence of a slide line at the initial discontinuity. (top) initial mesh, (bottom) mesh at $t = 0.2$. The slide line is pointed with arrows on the two figures.

The initial mesh, as well as the mesh compute at the time t = 0.2, are proposed on the figure (10). First, we check that the scheme is stable under CFL condition. Then, we do not observe any kind of instability that would break the symmetry of the test case, particularly near the slide line, and hence cause a crash of the simulation Moreover, the numerical solution converges to the analytical solution, as seen on the figure 11.

.

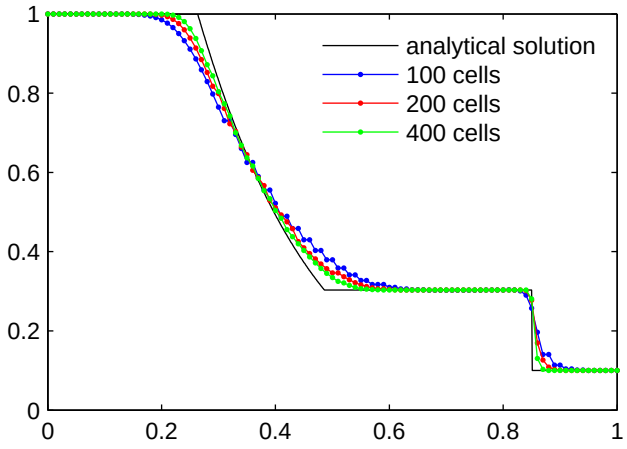

Figure 11: Study of the convergence of the numerical solution for the Sod test case with an artificial slide line transverse to the flow and introduced at the initial discontinuity

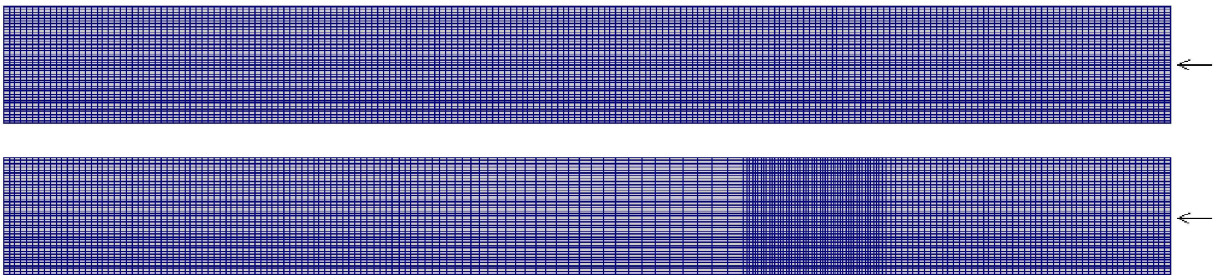

Figure 12: Presentation of two meshes resulted from the computation of the numerical solution in the Sod test case with the presence of a slide line parallel to the flow set at y = 0.25. (top) initial mesh, (bottom) mesh at t = 0.2. The slide line is pointed with arrows on the two figures.

The same test case is investigated with a slide line initially parallel to the flow. It is set at the position y = 0.25. The mesh is uniform and composed of square cells of side 5×10^{-3}. The results are proposed on the figure 12. Our methods is efficient and robust, as in the previous case.

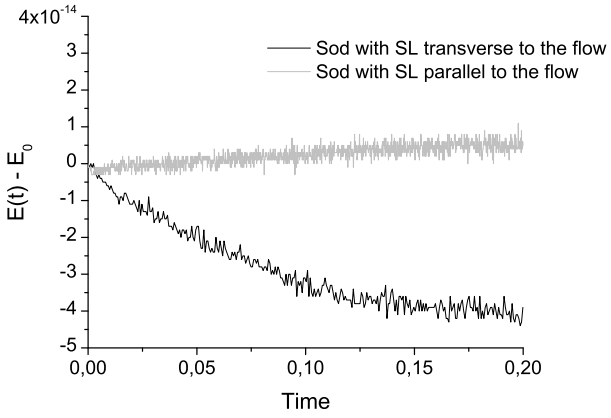

Figure 13: Variation $E(t)E_0$ in the Sod test case. $E_0 = E(0)$ is the initial total energy.

We end this section by proposing on the figure 13 a numerical result about the conservation of total energy as a function of time for both previous cases. We represent on this figure the variation $E(t)E_0$, where $E_0 = E(0)$ is the initial total energy. This result shows that the total energy is preserved up to the machine precision, as expected.

4.3. The test case of Caramana

We apply our method to a test case similar to the one described by Caramana in the reference [7]. The main difference is that we use a different initial mesh. A schematics of the test case is proposed on the figure 14. We respectively denote $\mathcal{M}_1$ and $\mathcal{M}_2$ the lower and

the upper part of the mesh around the slide line. At the initial time, all velocities are zero. The density in $\mathcal{M}_2$ is unity, while it is 10 in $\mathcal{M}_2$. The pressure is 10^{-8} everywhere except in the right lower part of $\mathcal{M}_2$ where it is 20. The pressure will be computed from the ideal gas law with $\gamma = 5/3$. The mesh $\mathcal{M}$ is globally composed of square cells of side 5.10^{-3} except in the high pressure region where the mesh is highly refined, and cells are quadrangles with dimensions $5.10^{-4} \times 5.10^{-3}$.

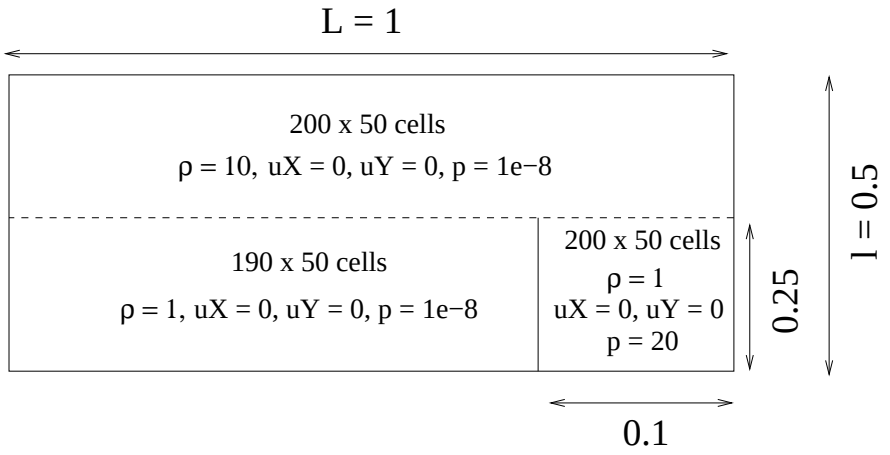

Figure 14: schematics of the test case similar to the Caramana test case. The slide line is denoted by a dash line.

We propose a result of our computation on the figure 15, at time t = 0.3. It shows a good agreement with the one presented by Caramana in the previous reference. One may notice that the slide line is subjected to local oscillations in some areas, that may be prevented by refining the mesh.

Again, we end this section by presenting on the figure 16 the result about the conservation of total energy. The latter is preserved up to the machine precision, as expected.

4.4. Sliding rings

The last test case we proposed is inspired from the recent work of Kucharik [8]. It consists in an inner ring sliding over an outer ring at the same density ($\rho = 1$) and the same pressure ($p = 1$). The outer ring is fixed while the inner one rotates at a constant velocity $\omega = 0.25$. The inner radius is $R_1 = 0.25$ and the outer radius is $R_2 = 1$. The slide line is set at $r = 0.5$. Well denote $\mathcal{M}_1$ and $\mathcal{M}_2$ the part of the mesh used to discretize the conservation laws respectively in the inner ring and the outer ring. $\mathcal{M}_1$ is composed of 96 angular sectors of 25 quadrangle cells, so that the dimensions of a cell is 0.01×0.016. $\mathcal{M}_2$ is composed of 95 angular sectors of 10 quadrangle cells, so that the dimension of a cell is 0.05 0.016.

We propose on figures 17 and 18 the meshes computed respectively at times $t = 0$ and $t = 0.3$. A pressure boundary condition is applied for the computation of the velocity of

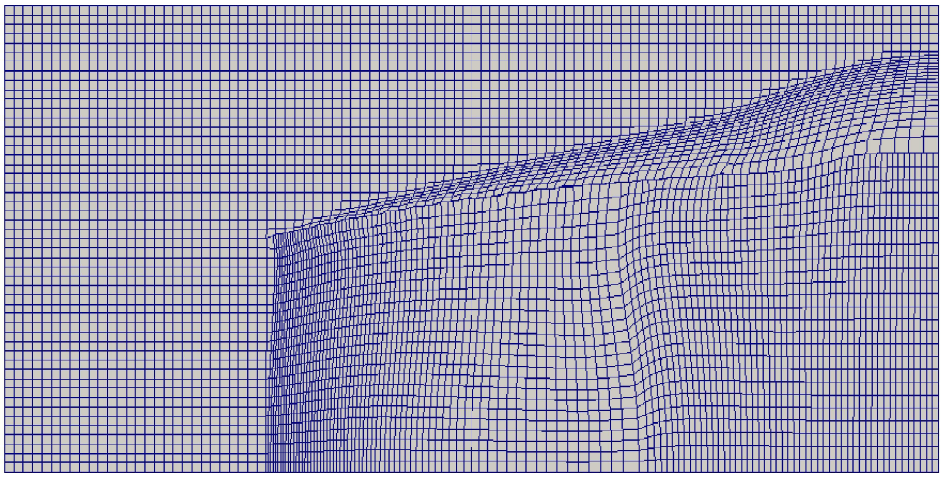

Figure 15: Presentation of the mesh computed in the Caramana test case at time t = 0.3

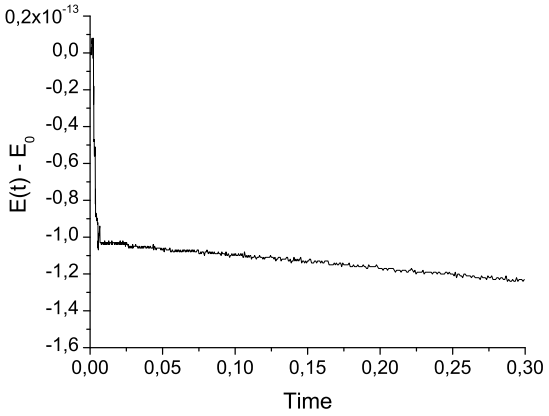

Figure 16: Variation $E(t)E_0$ in the Caramana test case. $E_0 = E(0)$ is the initial total energy.

the outer nodes while a wall condition is applied on the inner ones. Both conditions consist in modifying the matrices $A_r \equiv \tilde{A}_r$ and $\mathbf{b} \equiv \tilde{b}_r$ for all $r \in [1 : N]$ respectively to take into account the exterior pressure value or the normal vector to the wall in the nodal solver (See [9] for details). We observe that our method preserves the initial position and the circular shape of the slide line, without any stabilization procedure.

Momentum and total energy are preserved up to the machine precision (10^{-14} in our case), as seen on figure 19. Until t = 0.05, the change in energy undergoes some oscillations of amplitude 10^{-10}, which remains relatively small and does not contradict the conservation of total energy. After that time, the latter is preserved up to 10^{-14}.

However, a curious behaviour of the nodes belonging to the slide line can be noticed : the

latter seem to slow down and consequently, the mesh curves at the slide line, as shown on the figure 20. The latter represents a part of the mesh around the slide line at the time $t = 0.3$. The mesh curves until the simulation crash at larger times, due to a too strong deformation. Such behaviour doesnt not result from the new nodal solver, but, according to us, from the treatment of the boundary conditions in the GLACE scheme. To precise this argument, we refer to the result proposed on the figure 21 which represents a zoomed region near the inner boundary in the mesh at time $t = 0.3$. As said above, the velocity of the nodes belonging to the inner boundary is computed using the traditional formulation of the nodal solver (7) with modified $\tilde{A}_r$ and $\tilde{\mathbf{b}_r}$. We however observe the same behaviour: the nodes seem to slow down. We do not have any certainties about this behaviour, but we assume that it results from the non-preservation of the angular moment $\mathbf{L} = \mathbf{r} \times \mathbf{p}$ (where $\mathbf{r}$ and $\mathbf{p}$ are respectively the position vector and the momentum vector) in the GLACE scheme. Indeed, for the j-th cell, the angular moment is discretized as:

$$\begin{aligned}
\frac{d}{dt}\mathbf{L}_j &= \frac{d}{dt}\left(\mathbf{r}_j \times \mathbf{p}_j\right) \\
&= \frac{d}{dt}\mathbf{r}_j \times \mathbf{p}_j + \mathbf{r}_j \times \frac{d}{dt}\mathbf{p}_j \\
&= \mathbf{r}_j \times \frac{d}{dt}\mathbf{p}_j \\
&= \mathbf{r}_j \times \left(- \sum_{r \in \mathcal{N}(j)} \mathbf{C}_{j,r} p_{j,r} \right) \\
&= - \sum_{r \in \mathcal{N}(j)} \left(\mathbf{r}_j \times \mathbf{C}_{j,r}\right) p_{jr}
\end{aligned}$$

which is generally non zero.

Further investigations must be performed to precisely understand the reasons that forces the mesh to curve and to propose a remedy.

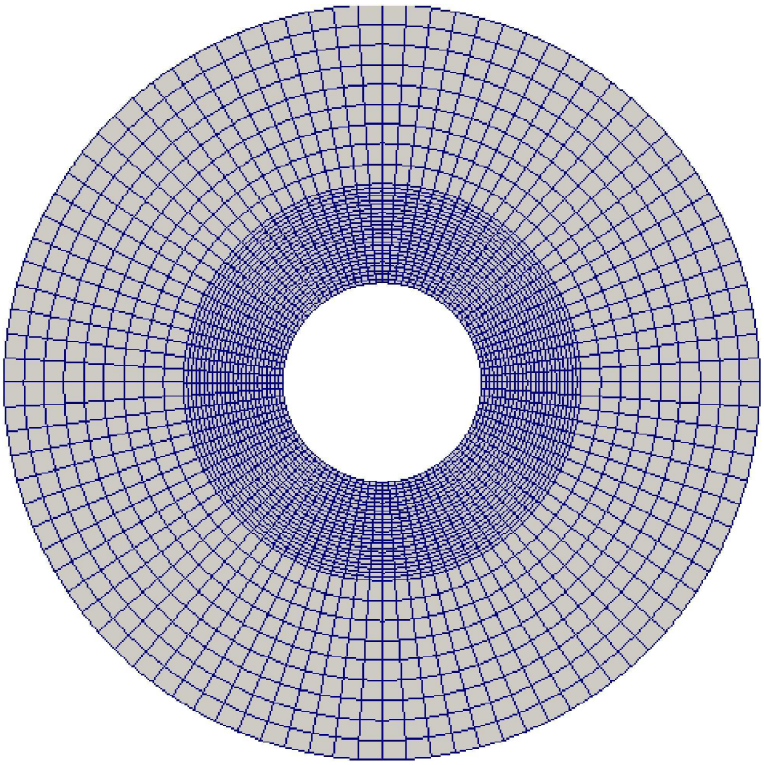

Figure 17: 2D initial mesh for the Sliding ring test case.

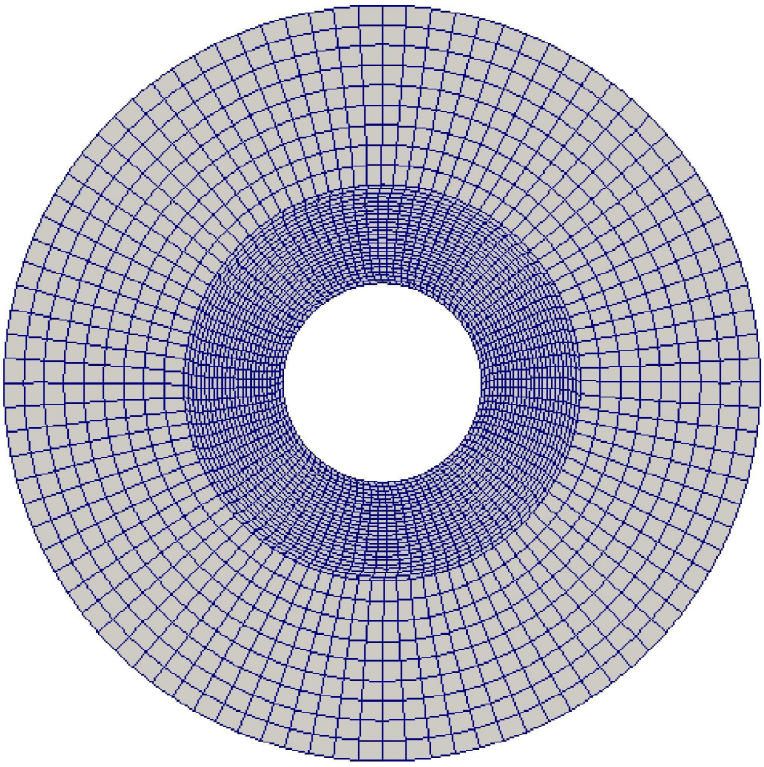

Figure 18: 2D mesh at t = 0.2 for the Sliding ring test case.

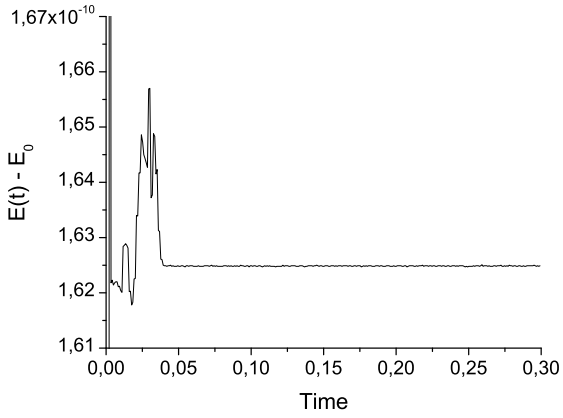

Figure 19: Variation $E(t)E_0$ on the sliding rings test case. $E_0 = E(0)$ is the initial total energy.

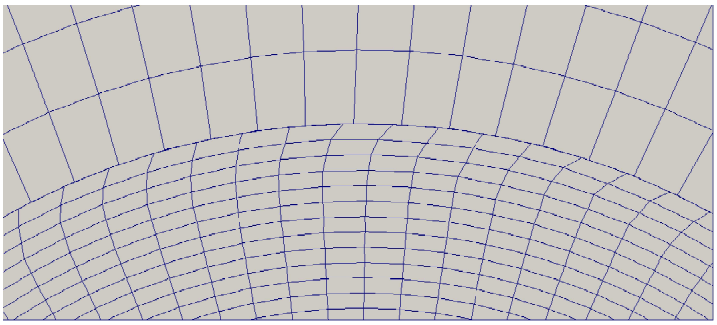

Figure 20: Zoom on the interior boundary at $t = 0.3$ for the Sliding ring test case.

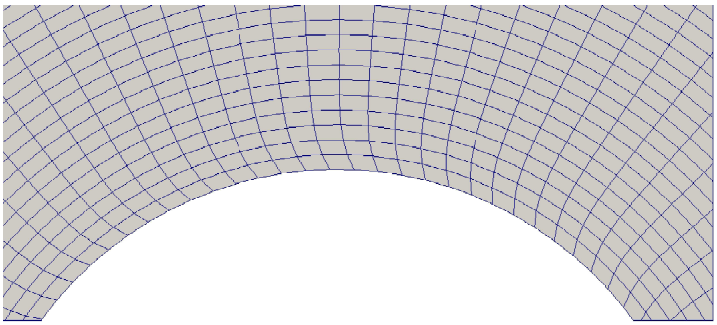

Figure 21: Zoom on the slide line at $t = 0.3$ for the Sliding ring test case.

5. Conclusions and perspectives

A new method to treat slide line in cell-centered Lagrangian schemes has been presented. It is based on a minimization procedure of an objective function under constraints. Such constraints model the sliding condition on each point of the slide line. Our method exploits the double discretization of the velocity in the cell-centered Lagrangian schemes: in particular, the objective function is defined from the nodal velocities and the minimization procedure is actually a reformulation of usual Riemann Solvers encountered in such schemes.
A discrete formulation of the set of admissible velocities $\mathbb{K}$ within which the minimum of the objective function is searched has been proposed. We prove that such formulation satisfies the four properties ①, ②, ③ and ④ that are sufficient conditions to ensure the existence and the uniqueness of the solution of the nodal solver, as well as the preservation of momentum and total energy.
We demonstrate the efficiency and the robustness of our method through several test cases. The slide line is correctly treated for all the proposed numerical test cases. In particular, for symmetrical sliding problems, the symmetry is preserved. We do not observe any hydrodynamical instability around the slide line, preventing the use of any stabilization procedure. In each case, momentum and total energy were preserved up to the machine precision. Finally, in the sliding rings test case, nodes belonging to boundaries that are treated with a sliding condition behave in an unexpected way, which was imputed to the non-preservation of the angular momentum in the employed scheme.
In our previous paper, only impact was considered. In this paper, pure sliding is studied. A future work would be then to treat the case where both impact and sliding occur. In addition, we might imagine that the sliding will not be pure any more but for which an exchange of momentum between the mobiles occurs. Surface tension may also be considered, as well as material strength as in [12].
Finally, our method should be easily extended to the 3D formalism. The key point is to find a convenient expression, similar to the relation (14), for the velocity of the ghost node.

References

[1] G. Clair, B. Després, E. Labourasse, Comput. Methods Appl. Mech. Engrg vol. 261-262 (2013) 56-65.

[2] N. G. Burago, V. N. Kukudzhanov, Mechanics of Solids 40(1) (2005) 35-71.

[3] M. L. Wilkins, Computer Simulation of Dynamic Phenomena, Springer (1999).

[4] D.E. Burton. Advances in the free-lagrange method including contributions on adaptive gridding and the smooth particle hydrodynamics method. Lecture notes in physics, vol. 395. Springer; 1991. p. 267-76.

[5] A.J. Barlow, J. Whittle, Mesh adaptivity and material interface algorithms in a two dimensional Lagrangian hydrocode. Chem.Phys.Reports, 19(2) (2000) 15-26.

[6] A.S. Dawes, Int. J. Numer. Methods Fluids 42(11) (2003) 1189-210.

[7] E.J Caramana, J. Comp. Phys. 228 (2009) 3911-3916.

[8] M. Kucharik, R. Loubére, L. Bednarik, R. Liska, Computers & Fluids, in Press (2012).

[9] G. Carré, S. Del Pino, B. Després, E. Labourasse, J. Comp. Phys. 228 (2009) 5160-5183.

[10] P.-H. Maire, J. Comp. Phys., 228(7) (2009) 2391-2425.

[11] D.J. Benson, Comput. Methods Appl. Mech. Engrg. 99 Issue 2-3 (1992) 235-394.

[12] N. R. Morgan, M. A. Kenamond, D. E. Burton, T. C. Carney, D.J. Ingraham, 250 (2013) 527-554.

[13] S. Bertoluzza, S. Del Pino, E. Labourasse, submitted to J. Sci. Comput. (2013).

[14] B. Després, C. Mazeran, C. R. Mécanique 331 (2003).

[15] A. Claisse, B. Després, E. Labourasse, F. Ledoux, J. Comp. Phys 231(11), 4324-4354(2012).

[16] R. Glowinski, A. Lichnewsky, Computing Methods in Applied Sciences and Engineering, Society for Industrial & Applied Mathematics, U.S. (1991).

[17] G. Allaire, Numerical Analysis and Optimization: An Introduction to Mathematical Modelling and Numerical Simulation, First edition, OUP Oxford (2007).

[18] P.H. Maire, B. Nkonga, Multi-scale Godunov type method for cell-centered discrete Lagrangian hydrodynamics, J. Comput. Phys., 2008, online (october).

[19] G.A. Sod, J. Comp. Phys. 27 (1978) 1-31.