A non-invasive implementation of a mixed domain decomposition method for frictional contact problems

Paul Oumaziz1 · Pierre Gosselet1 · Pierre-Alain Boucard1 · Stéphane Guinard2

Abstract A non-invasive implementation of the Latin domain decomposition method for frictional contact problems is described. The formulation implies to deal with mixed (Robin) conditions on the faces of the subdomains, which is not a classical feature of commercial software. Therefore we propose a new implementation of the linear stage of the Latin method with a non-local search direction built as the stiffness of a layer of elements on the interfaces. This choice enables us to implement the method within the open source software Code_Aster, and to derive 2D and 3D examples with similar performance as the standard Latin method.

Keywords Latin method · Domain decomposition · Contact problems · Assemblies

1 Introduction

Industrialists more and more consider conducting the simulation of their products at the scale of the (sub)system instead of the scale of the parts. Moreover they wish to be able to compute critical localized phenomena correctly (like the initiation of cracks) which imposes to use very fine meshes and results in large number of degrees of freedom. It is thus crucial to design robust methods capable of handling large discrete models with large contact zones with friction.

Handling large linear systems can be efficiently managed in parallel using iterative solvers based on domain decomposition methods (DDM): Schwarz methods [19], balancing domain decomposition [32], finite element tearing and interconnecting [15–17] as well as their constrained counterparts [10,18]. Robustness of domain decomposition methods was recently significantly improved by the introduction of adapted augmentation spaces obtained through prior quasilocal analysis [41] or multi-preconditioned approaches [23]. However these methods are not yet implemented in commercial software due to their invasive nature.

Regarding contact and friction, a first possibility is to deal with these phenomena in an outer loop; in which case the domain decomposition can be considered as an algebraic tool meant to efficiently solve linear(ized) systems and the DDM interfaces do not have to match the contact surfaces. A second possibility is to embed them in the DDM by forcing the DD interfaces to match the physical interfaces and adapt the solver to that type of connection between subdomains.

Contact laws are highly non-regular and complex to handle numerically [43]. A possibility is to use augmented Lagrangian formulations [1,39,40,42]. Because of the lack of regularity of the resulting systems (they are only B-differentiable [7]), classical Newton method does not apply and adapted solvers must be used [8]. The linearized systems arising from these formulations are often poorly conditioned and solvers based on domain decomposition methods are prone to converge slowly in particular when the contact zone is close to the subdomain’s boundaries. It is not uncommon to use methods with simpler iterations (but slower convergence and not as good parallel efficiency) like Uzawa solvers.
or adapted Gauss-Seidel solvers [25]; conjugate gradients can also be adapted [34] for the global constrained system.

It seems more efficient to specifically design domain decomposition methods capable of dealing with contact at the interface between subdomains. The dual framework is favorable for such adaptation [11–14] since it does not enforce the continuity of the displacement field; it can be viewed as providing a good subdomain-based preconditioner to a constrained conjugate gradient. Another possibility is to use mixed approaches such as the Latin method [28] which we focus on in this paper.

In the Latin method, subdomains and interfaces are considered as different entities with their own mechanical behavior, so that contact laws correspond to interface behavior. The principle of the method is to separate the equations on the subdomains (which in our case are linear) from the nonlinear local formulation of contact problems, and to formulate the problem as the search of a fixed point. The method is non-incremental in the sense that one iteration of the method provides a full approximation of the solution on the whole time-space domain.

Two successive problems are defined and solved at each iteration: independent linear systems set on subdomains (linear stage) followed by independent nonlinear pointwise equations on the interfaces with explicit solutions (local stage). The steps of the methods are linked by alternating directions [20] which can be interpreted as interface stiffness (Robin conditions), this connects the Latin method with the dual framework is favor- able for the formulation of the non-invasive Latin approach is given in Sect. 3. The contact and friction are treated in Sect. 4. The last section focuses on the numerical studies. A comparison between a “standard” Latin method and the new non-invasive formulation is shown. A 2D academic case is presented to validate the contact computation, then a 3D assembly with frictional contact is treated.

2 Reference problem in substructured form

2.1 Substructured problem

2.1.1 Notations

We consider a family of non-overlapping subdomains \((\Omega_E)_{E \in \{1, N\}}\) of \(\mathbb{R}^d\) \((d = 1, 2, 3)\). Each subdomain corresponds to a part of an assembly. We assume that each part is constituted by an isotropic linear elastic material whose Hooke’s tensor is written \(K\). Small perturbations and quasi-static isotherm evolution are assumed. The displacement field is designated by \(u\), the stress field by \(\sigma\) and the strain field by \(\varepsilon\). When needed the subscript \(E\) is used for the restriction to subdomain \(\Omega_E\).

Each part \(\Omega_E\) is subjected to Neumann boundary conditions \(\tilde{F}_d\) on \(\partial F \Omega_E\), Dirichlet boundary conditions \(u_d\) on \(\partial_u \Omega_E\) and a body force \(f\). It also interacts with any neighboring subdomain \(\Omega_E'\) through the interface \(\Gamma_{EE'} \subset \partial \Omega_E\). The problem is now reconsidered with a non-overlapping domain decomposition \((\Omega_E)_{E \in \{1, N\}}\) with \(N\) subdomains. The definition of subdomains implies in the same time the definition of interfaces between them that represent their interaction. Let \(\Gamma_{EE'}\) denote the interface between the subdomain \(\Omega_E\) and the subdomain \(\Omega_{E'}\). Let \(E\) be the set of all the sub-domains and \(G\) the set of all the interfaces.

Remark 1 It is clear that we have \(\partial \Omega_E \cap \partial \Omega_{E'} \subset \Gamma_{EE'}\), but \(\Gamma_{EE'}\) also contains all portions of the boundary which may enter in contact. Thanks to the small displacement hypothesis, this information does not need to be updated during the solving (one piece of the boundary can only interact with one given neighbor).

For each interface \(\Gamma_{EE'}\), displacement and force fields are defined on the two sides (Fig. 1). They are respectively written \((w_{EE'}, w_{EE'})\) and \((\tilde{f}_{EE'}, \tilde{f}_{EE'})\). These fields shall satisfy trace relations \(w_{EE'} = \text{tr}(u_{E})\) and \(\tilde{f}_{EE'} = \tilde{f}_{EE'} \).
dependence of the solution. We use dot notations for the interface and subdomain velocity fields. In practice a simple backward Euler scheme is used for the time integration.

2.1.2 Weak formulation on the subdomains

For any subdomain \( E \) and time \( t \), let us define the local admissibility space \( \mathcal{U}^0_E \) and the associated vector space \( \mathcal{U}^1_E : \mathcal{U}^1_E = \{ u \in H^1(\Omega_E), u(0) = u_0 \} \). In practice a simple backward Euler scheme is used for the time integration.

The weak formulation can be written as:

\[
\forall \Omega_E \in \mathcal{E}, \forall t, \quad \begin{cases}
\text{Find } u_E(t) \in \mathcal{U}^1_E, \text{ such that } \forall u_E \in \mathcal{U}^0_E, \\
\int_{\Omega_E} \sigma_E(t) \cdot \varepsilon(u_E) \, d\Omega = \int_{\Omega_E} f_E(t) \cdot \varepsilon(u_E) \, d\Omega \\
+ \int_{\partial \Omega_E} E(t) \cdot \varepsilon(u_E) \, d\Gamma + \sum_{E' \in \Gamma_{EE'}} \int_{E'} f_{EE'}(t) \cdot \varepsilon(u_E) \, d\Gamma \\
\sigma_E(t) = \mathcal{K}_E : \varepsilon(u_E(t)) \\
w_{EE'}(t) = \text{tr } (\varepsilon(u_E(t)))_{\Gamma_{EE'}}
\end{cases}
\]

where one can recognize the conservation of the momentum of the subdomain submitted to given loads and neighbors’ interactions, the constitutive equation and the trace relation on the interfaces.

2.1.3 Interface behavior

The interface \( \Gamma_{EE'} \) is granted a mechanical behavior given by a relation of the form:

\[
f_{EE'}(t) = b_{EE'}(w_{EE'}(t')) - b_{EE'}(w_{EE'}(t'), t') \leq t; w_{EE'}(0) = w_{EE'}(0)
\]

where \( b_{EE'} \) is a nonlinear evolution law characterizing the behavior of the interface. Initial conditions (corresponding to \( t = 0 \) are also given). Note that \( b_{EE'} \) is not independent from \( b_{EE'} \) because the balance of the interface must be preserved:

\[
f_{EE'} + f_{EE'} = 0
\]

Such a relation can be used to model spring interfaces or cohesive interfaces [26, 37]. The interface \( f_{EE'} \) can also be granted kinematic constraints, like perfect cohesion or contact. Typically a perfect interface is characterized by:

\[
\forall t \begin{cases}
f_{EE'}(t) + f_{EE'}(t) = 0 \\
\dot{w}_{EE'}(t) = \dot{w}_{EE'}(t)
\end{cases},
\]

and

\[
\begin{cases}
f_{EE'}(0) = -f_{EE'}(0) \\
\dot{w}_{EE'}(0) = \dot{w}_{EE'}(0)
\end{cases}
\]

Intuitively, perfect and contact interfaces can be considered as limit cases of mechanical behaviors (2) (think of a perfect interface as an infinitely stiff interface). In the following we use the term “interface behavior” and the notation \( b_{EE'} \) for any kind of interface, except when being more specific is required.

2.2 Discrete formulation

The continuous formulation above is rather standard in the Latin literature. We now propose to completely describe the resulting discrete system, using operators and notations inspired from [22].

2.2.1 Discretization

A classical FE is used in the subdomains for the displacement field. Let \( \phi_E^{w} \) be the matrix of shape function and \( u_E \) the vector of nodal displacement, such that \( u_E = \phi_E^{w} u_E \).

The interface displacements are also discretized with shape functions \( \phi_{EE'}^{w} : w_{EE'} = \phi_{EE'}^{w} w_{EE'} \). We choose \( \phi_{EE'}^{w} = \phi_{EE'}^{E} \).

Subdomains are connected to the interface by a mortar-like approach [2, 3] which amounts to choosing a discrete space of Lagrange multipliers (homogeneous to interface tractions) \( \mathcal{F}_{EE'} \) spanned by the shape functions \( \phi_{EE'}^{F} \), such that \( \int_{\Gamma_{EE'}} \phi_{EE'}^{F} \phi_{EE'}^{w} \) is invertible, and ensuring the following work equivalence:

\[
\forall f \in \mathcal{F}_{EE'}, \int_{\Gamma_{EE'}} f \cdot (w_{EE'} - \text{tr } u_E) dS = 0
\]
In the end, we can define the following discrete trace operator $N_{EE'}$:

$$\mathbf{w}_{EE'} = N_{EE'} \mathbf{u}_E$$  \hspace{1cm} (6)

Note that in our experiments, subdomains have matching grids, so that we use $\phi^w_{EE'} = \text{tr} \left( \phi^w \right)_{|\Gamma_{EE'}}$ and $N_{EE'}$ is a simple boolean matrix.

In the classic description of the Latin method, the space which the interface traction ($f_{EE'}$) belongs to is also discretized. We prefer to directly use the nodal reaction $f_{EE'}$ which represents the work of the traction in the discretized interface displacements:

$$\forall \mathbf{w}^*_{EE'} = \phi^w_{EE'} \mathbf{w}^*_{EE}, \quad \int_{\Gamma_{EE'}} f_{EE'}^{s^*}_{EE} \cdot \mathbf{w}^*_{EE'} \, d\mathbf{S}
= \int_{\Gamma_{EE'}} f_{EE'}^{s^*}_{EE} \cdot \left( \phi^w_{EE'} \mathbf{w}^*_{EE'} \right) \, d\mathbf{S}
= \mathbf{w}^s_{EE'} f_{EE'}$$  \hspace{1cm} (7)

To easily handle the many interfaces of one subdomain, we define concatenated operators:

$$\mathbf{w}_E = \begin{pmatrix} \vdots \\ \mathbf{w}_{EE'} \end{pmatrix}, \quad \mathbf{f}_E = \begin{pmatrix} \vdots \\ f_{EE'} \end{pmatrix},$$

$$\mathbf{N}_E = \begin{pmatrix} \vdots \\ \mathbf{N}_{EE'} \end{pmatrix}, \quad E' \text{ spans all neighbors}$$  \hspace{1cm} (8)

Note that in the presence of multiple points (points belonging to more than one interface), depending on the chosen discretization and mortar space, the matrix $\mathbf{N}_E$ may not be full ranked. In the case of interface of kinematic constraints, this results in a partial indetermination in $\mathbf{f}_E$, only the efforts applied to the subdomain $N^H_{EE'}$ being mechanically meaningful; see [33] for an investigation of the issue.

Regarding the time discretization, we use a backward Euler scheme on a regular grid of time step $\Delta t$. The time instant is mentioned as a superscript.

In the end, the discrete counterpart to system (1) can be written as:

Subdomains: $\forall \Omega_E \in \mathcal{E}, \forall t$,
\[ \begin{cases} \mathbf{K}_E \mathbf{u}_t = f^t_{EE} + N^T_{EE} f_{EE} \\ \mathbf{w}^t_E = \mathbf{N}_E \mathbf{u}_t \\ f_{EE}^t + f^t_{EE'} = 0, \forall t \end{cases} \]

Interfaces: $\forall \Gamma_{EE'} \in \mathcal{G}$,
\[ \begin{cases} f_{EE}^t - \mathbf{b}_{EE'} \cdot \mathbf{w}^t_E \\ -\mathbf{w}^t_{EE'}, t' \leq t; \mathbf{w}^0_{EE'} - \mathbf{w}^0_{EE'}, \forall t \end{cases} \]

Initial conditions

where $f^t_{EE}$ stands for the generalized forces associated with given loads and displacements (it is assumed that non zeros Dirichlet conditions were substituted) and $\mathbf{w}^t_{EE'} - \mathbf{w}^t_{EE'} - \Delta t$.

### 2.2.2 Global notations

Global variables are used to simplify the writing of all relations. Let us consider a given discrete interface variable $\mathbf{x}_E$. As said earlier $\mathbf{x}_E$ represents the gathering of all the $(\mathbf{x}_{EE'})_E$. $\mathbf{x}$ will represent the gathering of all the $\mathbf{x}_E$ defined on each subdomain $\Omega_E$, same procedure applies to operators:

$$\mathbf{x} = \begin{pmatrix} \vdots \\ \mathbf{x}_E \end{pmatrix},$$

$$\mathbf{K} = \begin{pmatrix} \mathbf{K}_E \\ 0 \\ \vdots \end{pmatrix}, \text{ where } E \text{ spans all subdomains}$$  \hspace{1cm} (10)

### 2.2.3 Sum and difference assembly operators

We introduce operators which enable the communication between neighboring sub-domains: the operator $\mathbf{A}$ add vectors on the interface whereas the operator $\mathbf{B}$ does the subtraction:

$$(\mathbf{A} \mathbf{f})|_{\Gamma_{EE'}} = f_{EE'} + f'_{EE'}$$

$$(\mathbf{B} \mathbf{w})|_{\Gamma_{EE'}} = \mathbf{w}_{EE'} - \mathbf{w}_{EE'}$$  \hspace{1cm} (11)

There is an arbitrary sign convention in the difference operator which plays no role in the following. Note that because multiple points are handled in the matrix $\mathbf{N}$, operators $\mathbf{A}$ and $\mathbf{B}$ are full-ranked.

**Remark 2** If we consider only one interface $\Gamma_{EE'}$, with identical ordering on both sides, these operators have the following matrix representation:

$$\mathbf{A}|_{\Gamma_{EE'}} = \begin{bmatrix} 1 & 1 \\ \vdots & \vdots \\ 1 & 1 \end{bmatrix}$$

$$\mathbf{B}|_{\Gamma_{EE'}} = \begin{bmatrix} -1 & 1 \\ \vdots & \vdots \\ -1 & 1 \end{bmatrix}$$  \hspace{1cm} (12)

The size of these operators is $n_{dof}^{int} \times 2n_{dof}^{int}$ with $n_{dof}^{int}$ the number of degrees of freedom of the interfaces.
The following property is fundamental to handle interface quantities:

**Proposition 1** Range($B^T$) $\oplus$ Range($A^T$) = Range($N$).

In words, any quantity defined on the two sides of an interface can be uniquely decomposed into the sum of a “continuous field” (with identical values on both sides) and a “balanced field” (with opposite values on the two sides). This is basically the odd-even decomposition of a function.

When needed, for regular enough interfaces, we will split the normal $n$ and tangential $\tau$ components.

### 2.2.4 Discrete problem

With these notations, the discrete problem to be solved can be written as:

$$\begin{cases}
Ku' = f_d' + N^T \tau' & \text{Equilibrium of the subdomains} \\
w' = Nu' & \text{Trace of the subdomain displacement} \\
f' = B^T b (Bw', \tau' \leq t; B\hat{w}) & \text{Interfaces' behavior}
\end{cases}$$

(13)

This notation for the interface’s behavior makes it clear that the equilibrium is ensured: $Af = AB^T b = 0$. In the case of perfect interfaces, the equations become:

$$\forall t, \ A f' = 0 \quad \text{and} \quad Bw' = 0; \quad \text{and initial conditions}$$

(14)

Contact is detailed in Sect. 4.

### 3 The non-invasive quasi-static Latin method

#### 3.1 Principle of the Latin method

##### 3.1.1 Separation of the equations

In order to solve the problem (13), the Latin method [28] is applied. The first idea of the Latin method is to separate the equations into two groups. The first group concerns the subdomain equations, which in our case are all linear. The second group gathers the equations which manage the behavior of the interfaces, these equations are all local (i.e. pointwise) in space. In order to handle friction easily, we chose to use the velocity as the main kinematic unknown.

Historically the separation of equations was made between linear and local equations. We will keep this denomination even if subdomain/interface would be more appropriate in our case.

These two groups of equations define respectively two sets of partial solutions to the problem: $A$ and $L$. Note that the method is non-incremental in the sense that each iteration gives an approximation defined in the whole time/space domain. The solution of the whole problem is the intersection of these two sets:

$$\forall t, \ (f', \ w') \in A \cup L \text{ solutions to } \begin{cases}
Ku' = f_d' + N^T \tau' \\
w' = Nu'
\end{cases}$$

(15)

The set $A$ is an affine space often called space of admissible fields. The set $L$ is in general a manifold; in the case of perfect interfaces, it is a vector space. Unknowns are limited to the interface velocities and efforts because they are sufficient to recover any other field thanks to the trace relations, the linear behavior of the subdomains and the time integrator.

##### 3.1.2 Iterations

In order to reach the solution of the whole problem, an iterative method is applied where partial solutions are found alternatively in each set $A$ and $L$.

The so-called local stage consists in, starting from a partial solution $s_n = (f_n, w_n) \in A$, searching for a partial solution $\hat{s}_n = (\hat{f}_n, \hat{w}_n) \in L$. This is made possible by enforcing a search direction of the form:

$$f_n - \hat{f}_n - k_v^+ (\hat{w}_n - \hat{w}) = 0$$

(17)

where $k_v^+$ is an operator which can be chosen by the user. In order to benefit from the local character of the equations which define $L$, $k_v^+$ is often chosen to be diagonal.

The so-called linear stage consists in, starting from a partial solution $\hat{s}_n = (\hat{f}_n, \hat{w}_n) \in L$, searching for a partial solution $s_{n+1} = (f_{n+1}, w_{n+1}) \in A$. This is made possible by enforcing a search direction of the form:

$$f_{n+1} - \hat{f}_n + k_v^- (\hat{w}_{n+1} - \hat{w}_n) = 0$$

(18)

where $k_v^-$ is an operator which can be chosen by the user.

The iterations are illustrated by Fig. 2. Note that a relaxation is often applied at the end of linear stages:

$$s_{n+1} \leftarrow s_n + \alpha (s_{n+1} - s_n), \quad \text{with } 0 < \alpha \leq 1$$

(19)

**Proposition 2** The convergence of the iterations is proved [28] when $k_v^- = k_v^+$ are symmetric definite positive operators and $\alpha = 0.5$ for maximal monotone behaviors $b$. 

© Springer
3.1.3 Interpretation of the search directions

Considering an interface \( \Gamma_{EE'} \) search directions are often written at the continuous level as a relation of the form:

\[
\begin{align*}
\hat{f}_{EE'} - \hat{\kappa}_{EE'} \hat{w}_{EE'} &= 0 \quad (20)
\end{align*}
\]

This leads to the following considerations:

- In its most general meaning, \( k_{EE'} \) is an operator from \( H^{1/2}(\Gamma_{EE'}) \) to its dual. Choosing it to be linear is a classical hypothesis to preserve the linearity of the computation of the sub-domains. But it has no reason for being local.
- In the literature, a relation like (20), is often referred to as a generalized Robin condition. A possible way to implement it is to use Ventcel conditions [24].
- [9] proposes a non-local form of the search direction by choosing the following writing:

\[
\left( \hat{f}_{EE'} - \hat{\kappa}_{EE'} \hat{w}_{EE'} \right) = 0
\]

where \( J_{EE'} \) is the Riesz isomorphism from \( H^{1/2}(\Gamma_{EE'}) \) to its dual, \( \hat{k}_{EE'} \) can be chosen to be scalar. This choice leads to dense matrices of search direction which are not computationally tractable.
- In practice, \( k_{EE'} \) is often taken as a scalar or a local orthotropic operator [38] (without taking care to use Riesz’ isomorphism). In that case the search direction is a classical Robin condition.
- Even when \( k_{EE'} \) is chosen to be scalar, there remains two possible interpretations of the search direction.

First, the “weak” interpretation with discretization leads to:

\[
\int_{\Gamma_{EE'}} \left( f_{EE'} - \hat{\kappa}_{EE'} \hat{w}^* \right) dS = 0
\]

and the discrete search direction \( \hat{k}_{EE'} \) is some sort of weighted \( L^2 \)-mass matrix of the interface.

Second the “strong” (collocation) interpretation where one directly writes:

\[
f_{EE'} - \hat{k}_{EE'} \hat{w}_{EE'} = 0
\]

which has the evident advantage of making the search direction local.
- Note that a local search direction can also be derived from a “weak” interpretation by lumping \( k_{EE'} \) or simply keeping only the diagonal.

3.2 The non-invasive Latin method

The mono-scale Latin method is the direct application of the two principles exposed previously. The linear stage is presented first since it is mainly impacted by the non-invasive hypothesis. Several possibilities to tackle the local stage are presented afterwards; in this section we detail the case of perfect interfaces and leave the contact and friction to the next section.

3.2.1 Linear stage

The linear stage consists in solving problems defined on the sub-domains. Equations (15) and (18) lead to the following equations which are independent per sub-domains, and where \( \left( \hat{f}^{t+\Delta t}, \hat{w}^{t+\Delta t} \right) \) are known from the previous local stage:

\[
\begin{align*}
\left( K + N^T \frac{k_n}{\Delta t} N \right) u^{t+\Delta t} &= f^{t+\Delta t} + N^T \left( \hat{f}^{t+\Delta t} + k_v \hat{w}^{t+\Delta t} \right) \\
\end{align*}
\]

Then compute

\[
\begin{align*}
\hat{w}^{t+\Delta t} &= N u^{t+\Delta t} - N (u^{t+\Delta t} - u^t) / \Delta t \\
\hat{f}^{t+\Delta t} &= \hat{f}^{t+\Delta t} + k_v \left( \hat{w}^{t+\Delta t} - \hat{w}^{t+\Delta t} \right)
\end{align*}
\]

Remark 3 Equation (24) can be interpreted as the discretization of the subdomain’s equilibrium under a generalized Robin boundary condition. The term \( N^T \hat{k}_v N/\Delta t \) in (24) corresponds to the interface impedance, it is a non-standard term in commercial software for mechanical problems (there
often exist implementations for thermal problems, since it corresponds to convection conditions).

3.2.2 Non-invasive implementation for the linear stage

Here the aim is to propose a new implementation of the generalized Robin condition prone to be implemented in a commercial software. First note that other options are possible:

- first, the interface could be connected to the subdomain by node-to-node springs elements leading to a local operator.  
- second, cohesive elements could be used in their elastic regime. Depending on the chosen operator, a local operator can be obtained.

The idea which we investigate in this study is to consider \( k^- = k/\Delta t \) as a generalized Robin condition which can be realized by adding a layer of elements on the boundary of the subdomains. This strategy leads to a non-local search direction in the sense that it couples neighboring nodes together (the associated matrix is non-diagonal).

The layer of elements added to the boundary of \( \Omega_E \) regarding the interface \( \Gamma_{EE'} \) is written \( \theta_{EE'} \). The Hooke tensor associated with its behavior is \( \mathcal{K}_{EE'} \). The addition is called a sole. A zero Dirichlet condition is imposed on the other part of the boundary of the sole.

At the continuous level, the search direction operator written in displacement \( k_{EE'} \) is defined by the following relation:

\[
\hat{f}_{EE'} = k_{EE'} (w_{EE'}) \\
\Leftrightarrow \begin{cases}
| \begin{align*}
& \forall u \in H^1(\theta_{EE'}), \ u = w_{EE'} \text{ on } \Gamma_{EE'}, \ u = 0 \text{ on } \partial_a \theta_{EE'} \\
& \forall \nu \in H^1(\theta_{EE'}), \ \nu = 0 \text{ on } \Gamma_{EE'} \cup \partial_a \theta_{EE'} \\
& \int_{\theta_{EE'}} \mathcal{K}_{\theta_{EE'}} : \varepsilon(u) \ d\Omega = \int_{\Gamma_{EE'}} f_{EE'} \cdot \nu \ dS
\end{align*} \end{cases}
\]

At the discrete level, if \( \mathbf{K}_{\theta_{EE'}} \) is the stiffness matrix of the sole (with zero Dirichlet boundary conditions taken into account), then \( k^-_{EE'} \) is the Schur complement of \( \mathbf{K}_{\theta_{EE'}} \) with respect to the degrees of freedom of the interface \( \Gamma_{EE'} \).

Remark 4 This setting resembles the Restricted Additive Schwarz method \([4, 5]\). Indeed the layer of elements added to the boundary of the subdomains may be seen as an overlap between two subdomains. However contrary to the Schwarz method, the boundary of the overlap which is not in contact with the interface is blocked, whereas a displacement that comes from the subdomains is imposed in the Schwarz method. Moreover the material characteristics of the sole is a parameter of the method and need not coincide with the characteristics of the neighbor.

From an implementation point of view, solving the problem of the linear stage \( (\mathbf{K} + \mathbf{N}^T \mathbf{k}^{-} \mathbf{N}) \) corresponds to solving a mechanical problem on the subdomain extended with the sole, loaded by a traction field on the embedded interface (Fig. 3) which is a classic operation in most software. This allows to compute \( \mathbf{w}_{EE'} \) and \( \mathbf{u}_E \).

The nodal reaction \( f_{EE'} \) can be obtained by post-processing the subdomain or by computing a Dirichlet problem on the sole alone.

3.2.3 The local stage

The local stage corresponds to solving a system of equations at the interfaces. Equations (2) and (17) lead to searching for \( (\mathbf{w}, \mathbf{f}) \), solution of:

\[
\begin{align*}
\begin{pmatrix}
\mathbf{f}^t \\
\mathbf{f}^0
\end{pmatrix} &= \mathbf{B}^T \mathbf{b} \\
\hat{\mathbf{f}}^t &= \mathbf{B} \hat{\mathbf{w}}^t, \ t' \leq t, \ \mathbf{B} \hat{\mathbf{w}}^0 \\
\mathbf{w}^t - \mathbf{w}^{t-} - \mathbf{k}_V \left( \mathbf{w}^t - \mathbf{w}^{t-} \right) &= 0
\end{align*}
\]

Constitutive law

Search direction

Backward Euler integration

We briefly detail here the computation for perfect interfaces, whereas next section is dedicated to contact. Note that in that particular case, their is no need for time integration.

\[
\begin{align*}
\begin{pmatrix}
\mathbf{f}^t \\
\mathbf{w}^t
\end{pmatrix} &= 0 \\
\mathbf{B} \hat{\mathbf{w}}^t &= 0 \\
\hat{\mathbf{f}}^t - \mathbf{f}^t - \mathbf{k}_V \left( \mathbf{w}^t - \mathbf{w}^{t-} \right) &= 0
\end{align*}
\]

Equilibrium

Continuity of velocity

Search direction

The equilibrium and continuity relations also impose that \( \mathbf{f} \) and \( \mathbf{w} \) belong to \( \text{Range}(\mathbf{B}^T) \) and \( \text{Range}(\mathbf{A}^T) \) respectively. Thus one can deduce that:
In order to write the contact problem, we assume a sufficient smoothness of the interface so that we can define a local basis at each interface node: \( n \) stands for the normal direction and \( \tau \) for the tangent direction (the tangent basis is \((\tau_1, \tau_2)\) in 3D). Conventionally the same normal vector is used on both neighbors, the orientation of the vector is chosen in agreement with the sign convention of \( B \). Thus we assume all vectors and matrices are related to the local interface frame \((n, \tau)\): the normal and the tangent components of a given variable.

In order to preserve locality, the search direction of the local stage must be a diagonal operator in the normal-tangent frame. In order to obtain such a property while not departing too much from the search direction of the linear stage, the search direction of the local stage is defined by the diagonal of the stiffness matrix of the sole turned in the normal-tangent frame: \( \text{diag}(R_{nt}^T K_{EE} R_{nt}) \).

The local stage for contact interfaces regroups the different relations that govern the contact plus the search directions written in terms of velocities. As for the linear stage, an implicit scheme is applied to link the velocity distribution and the displacement distribution. Thus a sequence indexed by time of problems need to be solved:

\[
\text{Find } \left( \hat{\mathbf{w}}^{t+\Delta t}, \hat{\mathbf{t}}^{t+\Delta t} \right) \text{ solution to:}
\]

\[
\begin{align*}
B_n \hat{\mathbf{w}}^{t+\Delta t} + j_n & = 0 \\
A \hat{\mathbf{t}}^{t+\Delta t} & = 0 \\
B_n \hat{\mathbf{f}}^{t+\Delta t} & = 0 \\
B_n \hat{\mathbf{t}}^{t+\Delta t} & = 0 \\
(B_n \hat{\mathbf{f}}^{t+\Delta t})^T (B_n \hat{\mathbf{w}}^{t+\Delta t} + j_n) & = 0 \\
\hat{\mathbf{t}}^{t+\Delta t} - \mathbf{t}^{t+\Delta t} - k_V^{+} \left( \hat{\mathbf{w}}^{t+\Delta t} - \hat{\mathbf{w}}^{t+\Delta t} \right) & = 0 \\
\hat{\mathbf{w}}^{t+\Delta t} & = \hat{\mathbf{w}}^{t} + \Delta t \hat{\mathbf{w}}^{t+\Delta t}, \quad \hat{\mathbf{w}}^{0} \text{ given}
\end{align*}
\]

where hat-less quantities and quantities at time step \( t \) are known from the previous time step and the linear stage, \( j_n \) is the initial normal gap.

To solve the problem, the time integration scheme is used in the non-interpenetration relation, and then the search direction is applied to reveal \( \hat{\mathbf{t}}^{t+\Delta t} \), so that we can control the sign of the normal component.

\[
\begin{align*}
B_n \hat{\mathbf{w}}^{t+\Delta t} + j_n & = B_n \hat{\mathbf{w}}^{t} + j_n \\
+ \Delta t B_n \left( \hat{\mathbf{w}}^{t+\Delta t} + k_V^{+} \left( \hat{\mathbf{f}}^{t+\Delta t} - \mathbf{f}^{t+\Delta t} \right) \right) & \geq 0
\end{align*}
\]

which can be reorganized as:

\[
\begin{align*}
B_n \hat{\mathbf{w}}^{t} + j_n + \Delta t B_n \hat{\mathbf{w}}^{t+\Delta t} - \Delta t B_n k_V^{+} \mathbf{f}^{t+\Delta t} + C_n & \\
+ \Delta t B_n k_V^{+} \mathbf{f}^{t+\Delta t} & \geq 0
\end{align*}
\]
$C_n$ is a computable quantity which enables us to evaluate node-wise the feasibility of a contactless solution. The analysis that follows must be conducted on each node of the interface separately. The non-bold characters indicate that the computations are conducted on single nodes (with potentially fine grain parallelism).

No contact case \[ C_n > 0 \]

In that case, the solution $\hat{f}_t^{i+\Delta t} = 0$ and $B_n \hat{w}_t^{i+\Delta t} + j_n = C_n > 0$ satisfies all equations.

Contact case \[ C_n \leq 0 \]

In that case, the solution is $\Delta t B_n k_{V_n}^{-1} \hat{f}_t^{i+\Delta t} = -C_n \geq 0$ and $B_n \hat{w}_t^{i+\Delta t} + j_n = 0$. To compute $\hat{f}_t^{i+\Delta t}$ one simply has to remember that it can be expressed as $\hat{f}_t^{i+\Delta t} = B_n^{T} f_n^*$ for some $f_n^*$ (indeed $\hat{f}_t^{i+\Delta t}$ verifies $A_n \hat{f}_t^{i+\Delta t} = 0$), and thus:

$$\hat{f}_t^{i+\Delta t} = -B_n^{T} \left( B_n k_{V_n}^{-1} B_n^{T} \right)^{-1} \frac{C_n}{\Delta t}$$

(35)

$\hat{w}_t^{i+\Delta t}$ then can be deduced using the search direction and the time integration.

For the tangential component of the interface distributions, we use the fact that $\hat{f}_t^{i+\Delta t} = 0$. Then with the search direction and the time integration scheme, we obtain:

$$\hat{w}_t^{i+\Delta t} = \hat{w}_t^{i+\Delta t} - k_{V_t}^{-1} \hat{f}_t^{i+\Delta t}$$

(36)

In conclusion, it is possible to define a node-wise indicator which gives the contact status. This indicator can be expressed only with the distribution $\left( \hat{w}_t, \hat{w}_t^{i+\Delta t}, f_t^{i+\Delta t} \right)$ known from the previous linear stage and previous time step of the local stage. Then the computation of the solutions $\left( \hat{w}_t^{i+\Delta t}, \hat{w}_t^{i+\Delta t}, \hat{f}_t^{i+\Delta t} \right)$ is explicit.

4.2 Coulomb friction

The frictional aspect is taken into account by adding new conditions in the case of contact. First the normal components can be determined as explained in previous subsection. Then in the case of contact, another indicator is computed to evaluate the possibility of sliding.

Let $\mu$ be Coulomb’s friction coefficient. The friction conditions can be written nodewise as:

$$\begin{align*}
\text{If } \| \hat{f}_t^{i+\Delta t} \| < \mu | f_n^{i+\Delta t} | & \text{ then } B_t \hat{u}_t^{i+\Delta t} = 0 \\
\text{If } \| \hat{f}_t^{i+\Delta t} \| = \mu | f_n^{i+\Delta t} | & \text{ then } \exists \lambda > 0, B_t \hat{u}_t^{i+\Delta t} \\
& = -\lambda \left( B_t B_t^{T} \right)^{-1} B_t \hat{f}_t^{i+\Delta t}
\end{align*}$$

(37)

where $\hat{f}_t^{i+\Delta t}$ is still submitted to the balance condition $A_t \hat{f}_t^{i+\Delta t} = 0$.

With the same idea as for frictionless contact, it is possible to define an indicator that specifies the status of the node. The tangential component of the force distribution is computed and compared to the norm of the normal one. Beforehand, as the search direction is positive definite, a decomposition of the tangential velocity distribution is done:

$$\exists (\alpha, \beta), \hat{u}_t^{i+\Delta t} = A_t^{T} \alpha + k_{V_t}^{-1} B_t^{T} \beta$$

(38)

$A_t^{T} \alpha$ corresponds to the sticking part whereas $k_{V_t}^{-1} B_t^{T} \beta$ corresponds to the sliding part.

With such an expression for the velocity distribution, the search direction is used to compute the force distribution:

$$\hat{f}_t^{i+\Delta t} = f_t^{i+\Delta t} + k_{V_t}^{-1} \left( \hat{w}_t^{i+\Delta t} - \hat{u}_t^{i+\Delta t} \right)$$

(39)

Applying the equilibrium of the force distribution, $\alpha$ can be determined as:

$$\alpha = \left( A_t k_{V_t}^{T} A_t^{T} \right)^{-1} A_t \left[ k_{V_t}^{-1} \hat{w}_t^{i+\Delta t} - f_t^{i+\Delta t} \right]$$

(40)

Thus the expression of the force distribution becomes:

$$\hat{f}_t^{i+\Delta t} = \left[ I - k_{V_t}^{+} A_t^{T} \left( A_t k_{V_t}^{+} A_t^{T} \right)^{-1} A_t \right] f_t^{i+\Delta t} - k_{V_t}^{+} \left[ I - A_t^{T} \left( A_t k_{V_t}^{+} A_t^{T} \right)^{-1} A_t k_{V_t}^{+} \right] \hat{u}_t^{i+\Delta t} + B_t^{T} \beta$$

(41)

$G_t$ represents the sticking contribution to the force distribution. It only depends on contributions from the linear stage and so can be computed directly at the local stage. It is used as a nodewise indicator for the possibility of sliding.

Sticking case \[ \| G_t \| \leq \mu | \hat{f}_t^{i+\Delta t} | \]

We can directly set $\beta = 0, \hat{f}_t^{i+\Delta t} = G_t$. The velocity distribution is computed with the expression of $\alpha$.

$$\begin{align*}
\hat{w}_t^{i+\Delta t} & = A_t^{T} \left( A_t k_{V_t}^{+} A_t^{T} \right)^{-1} A_t \left( k_{V_t}^{+} \hat{w}_t^{i+\Delta t} - f_t^{i+\Delta t} \right) \\
\hat{w}_t^{i+\Delta t} & = \hat{w}_t^{i} + \Delta t \hat{u}_t^{i+\Delta t}
\end{align*}$$

(42)
Sliding case: If \( \| \mathbf{G}_t \| > \mu | \mathbf{f}_n^{t+\Delta t}| \)

\[ \beta \text{ and } \lambda > 0 \text{ must be found such that } \| \mathbf{f}_t^{t+\Delta t} \| = \mu | \mathbf{f}_n^{t+\Delta t}| \text{ and } B_t \mathbf{w}_t^{t+\Delta t} = -\lambda (B_t B_t^T)^{-1} B_t \mathbf{f}_t^{t+\Delta t}. \]

With the expression of the tangential component of the velocity, the traction can be expressed as a function of \( \beta \) and \( \lambda \):

\[
(B_t B_t^T)^{-1} B_t \mathbf{f}_t^{t+\Delta t} = -\frac{1}{\lambda} B_t k^+_t \mathbf{v}_t \beta \tag{43}
\]

Moreover \( \mathbf{f}_t^{t+\Delta t} = \mathbf{G}_t + B_t^T \beta \). Thus the relation between \( \beta \) and \( \lambda \) is:

\[
(B_t B_t^T)^{-1} B_t G_t + \beta = -B_t k^+_t \mathbf{v}_t \frac{\beta}{\lambda} \tag{44}
\]

and

\[
\beta = \lambda \left( \mathbb{I} + B_t k^+_t \mathbf{v}_t \frac{1}{\lambda} B_t B_t^T \right)^{-1} B_t G_t \tag{45}
\]

The tangential force distribution can be expressed as a function of \( \lambda \):

\[
\mathbf{f}_t^{t+\Delta t} = \left( \mathbb{I} - B_t \left( \mathbb{I} + \frac{B_t k^+_t \mathbf{v}_t}{\lambda} B_t B_t^T \right)^{-1} \right) B_t G_t \tag{46}
\]

And \( \lambda > 0 \) is determined by the relation \( \| \mathbf{f}_t^{t+\Delta t} \| = \mu | \mathbf{f}_n^{t+\Delta t}| \) given from the sliding assumption. It is more convenient to consider the intermediate quantity \( \left( B_t B_t^T \right)^{-1} B_t f_t^{\mathbf{f}_t^{t+\Delta t}} \) which is non-redundant between the two sides of the interface. As \( \| (B_t B_t^T)^{-1} B_t f_t^{\mathbf{f}_t^{t+\Delta t}} \| = \mu | (B_n B_n^T)^{-1} B_n f_n^{\mathbf{f}_n^{t+\Delta t}}| \), we must ensure:

\[
\left| \mathbb{I} - \left( \mathbb{I} + \frac{B_t k^+_t \mathbf{v}_t}{\lambda} B_t B_t^T \right)^{-1} \right| \left( B_t B_t^T \right)^{-1} B_t G_t \quad G_t' = \frac{\mu | \mathbf{f}_n^{t+\Delta t}|}{| \mathbf{G}_t |} \tag{47}
\]

Note that the square matrices are all diagonal.

Remark 8  
- \( f_t^{t+\Delta t} \) can be expressed as \( f_t^{t+\Delta t} = \gamma G_t' \) as \( \gamma = \left( \mathbb{I} - \left( \mathbb{I} + \frac{B_t k^+_t \mathbf{v}_t}{\lambda} B_t B_t^T \right)^{-1} \right) \)

- In the 2D case, \( \gamma \in \mathbb{R}^+ \) and \( \gamma G_t' \) so the \( \gamma \) is easily computed as \( \gamma = \frac{\mu | \mathbf{f}_n^{t+\Delta t}|}{| \mathbf{G}_t |} \) and therefore:

\[
\mathbf{f}_t^{t+\Delta t} = \mu | \mathbf{f}_n^{t+\Delta t}| \frac{G_t'}{| \mathbf{G}_t |} \tag{48}
\]

- In the 3D case, the Eq. (47) is a polynomial equation of the fourth degree. However, in the case of the two tangential search directions being equal, \( \gamma = \gamma \mathbb{I} \) with \( \gamma > 0 \), a similar result as previously is obtained: \( \gamma = \frac{\mu | \mathbf{f}_n^{t+\Delta t}|}{| \mathbf{G}_t |} \) and therefore:

\[
\mathbf{f}_t^{t+\Delta t} = \mu | \mathbf{f}_n^{t+\Delta t}| \frac{G_t'}{| \mathbf{G}_t |} \tag{49}
\]

- The velocity distribution \( \mathbf{w}_t^{t+\Delta t} \) is computed from the relation of the search direction and the expression of the force distribution \( \mathbf{f}_t^{t+\Delta t} \). Then the displacement is computed from the time integration scheme.

The Latin method for frictional contact is summarized in Algorithm 1.

5 Numerical examples

First a validation of the non-invasive implementation is presented in 5.1 with a comparison to a “standard” Latin with an invasive modification of the stiffness operator. This validation is performed on a simple traction beam study. Moreover the influence of the search direction on the convergence is shown. An academic test case for contact with friction is presented in 5.2. The objective is to validate the frictional contact solution by comparing results with [6]. A 3D study of a bolted assembly involving frictional contact with initial gap and preloaded interfaces is developed in 5.3. The computations are made with Code_Aster driven by a python instance.

5.1 Comparison “standard Latin”/non-invasive Latin

The comparison between a “standard” invasive Latin method and the non-invasive one is performed on a simple traction beam problem fixed on one side and under a load \( F \) on the other side (Fig. 4). The Young modulus is \( E \), the length \( L \) and a section \( S \). A sub-structuring in 5 subdomains is chosen. In this 1D case, the search direction is reduced to only one parameter per interface.

For the standard computation we use \( k^- = k^+ = \frac{ES}{L} \) [28]. For the non-invasive version, the parameter is reduced to the Young modulus of the soles and therefore it is easy to obtain equivalent search directions for the two cases by
Algorithm 1: Summary of the Latin method

**Input:** Initialisation

**while** Error criterion < objective **do**

**Local stage:**

foreach time step **do**

foreach interface **do**

if interface is perfect **then**

- Non-local computation on a double sole to obtain displacement: \(w^{t+\Delta t}\)
- Time scheme to compute velocity: \(\dot{w}^{t+\Delta t}\)
- Search direction to compute force: \(\hat{f}^{t+\Delta t}\)

else

- Contact interface:
  foreach each node on the interface **do**
    Compute the contact indicator \(C_n\)
    if \(C_n \geq 0\) **then**
      No contact computation
    else
      Contact computation
      Compute the sticking indicator \(G\)
      if \(\|G\| \leq \mu |\hat{f}^{t+\Delta t}|\) **then**
        Sticking computation
      else
        Sliding computation
    end
  end
end

**end**

**Linear stage:**

foreach time step **do**

foreach subdomain **do**

- Computation of displacement \(w^{t+\Delta t}\) by solving a problem on subdomain plus soles
- Time scheme to compute velocity: \(\dot{w}^{t+\Delta t}\)
- Search direction to compute force: \(f^{t+\Delta t}\)
- Relaxation step: \(s_{n+1} \leftarrow s_n + \alpha(s_{n+1} - s_n)\)

end

end

end

Choosing \(E_s\) the Young modulus of the soles as \(E_s = \frac{E_Ls}{L}\) with \(L_s\) the length of the soles. As the soles are composed of a layer of one element, the size \(L_s\) is equal to the discretization \(h\). A parametric computation is conducted to study the impact of the search direction on the convergence.

The numerical values of the parameters are given on the Table 1.

As seen on the Fig. 5 the standard and the non-invasive Latin method gives the same results. This validates the non-invasive implementation.

![Fig. 4](image1.png)  (a) Traction beam, a reference problem, b sub-structured problem

![Fig. 5](image2.png)  Comparison between the standard and the non-invasive Latin

![Fig. 6](image3.png)  Parametric study of the search direction

On Fig. 6 the parametric study is presented to illustrate the influence of the search direction on the convergence. We use the multiplicative parameter \(\alpha\) as \(k^- = k^+ = \alpha \frac{E_s}{L}\) for several sub-structuring. The dimensions of the global structure

| Parameters for test case 1 |
|-----------------------------|
| **Parameters**               | **Value** |
| \(E\)                       | 200 GPa   |
| \(L\)                       | 100 mm    |
| \(S\)                       | 10 mm     |
| \(h = L_s\)                 | 1 mm      |

![Table 1](image4.png)
and the size of the discretization are the same for all the sub-structuring cases. The results show that an optimal search direction exists and that it does not depend on the choice of the sub-structuring. The optimal search direction depends on the global structure. These two results are classical for the Latin computation.

Actually as the optimal search direction is not evident to compute for any structure, we keep a “non-optimal” standard one but sufficient to reach good level of convergence (indicator $< 10^{-5}$). The rule to choose $k^+ = k^- = \frac{E_S}{L}$ is chosen. For a 2D or a 3D structure search direction is chosen by adjusting the Young modulus of the soles to ensure that soles and sub-domains have approximately the same rigidity.

### 5.2 An academic 2D friction case

The developed contact management is exemplified on an academic use case [6]. The quasi-static problem is presented on the Fig. 7. Two interfaces with frictional contact link respectively the subdomains (1, 2) and the subdomains (2, 3); the parameters $\mu_1$ and $\mu_2$ are Coulomb’s friction coefficients. An additional interface enables to manage a boundary condition of unilateral contact with an initial gap $j$. The structure is under loads $F_1$ and $F_2$ defined by two time steps $t_0$ and $t_1$. Firstly, the structure is preloaded with the load $F_1$ before being loaded with $F_2$. The materials are linear elastic, defined by the Young modulus $E$ and the Poisson ratio $\nu$. Each sub-

![Fig. 7 Test case for frictional contact](image)

| Table 2 Parameters for test case 2 |
|-----------------------------------|
| Parameters | Value/range |
| $E$        | 210 GPa     |
| $h$        | 50 mm       |
| $j$        | 0.04 mm     |
| $F_{max}^1$ | 50 MPa      |
| $F_{max}^2$ | 30 MPa      |
| $\mu_1$    | $\in [0, 0.6]$ |
| $\mu_2$    | $\in [0, 0.6]$ |

![Fig. 8 Response surface of the reaction force $(\mu_1, \mu_2)$](image)

Structure is discretized by $25 \times 25$ elements. The values of the different parameters are given in Table 2.

Depending on the chosen Coulomb coefficients, the sub-domain 2 may or may not touch the contact boundary with initial gap. In order to illustrate this behavior, the force reaction on the boundary condition is computed at convergence of the algorithm and plotted depending on the two parameters $\mu_1$ and $\mu_2$ (Fig. 8).

Our results correspond to those of Champaney and Boucard in [6]. We obtain the different configurations in which the contact on the interface with gap is reached or not. On Fig. 9a–c contact is reached, Coulomb coefficients are too weak to retain the subdomain 2. On the other hand, on Fig. 9d, no contact is obtained on the interface with gap.

This 2D computation with frictional contact validated our non-invasive implementation. Relaxing the condition $k^+ = k^- = \frac{E_S}{L}$ by using of the diagonal of the stiffness of the soles as a local search direction seems not to impair the convergence.

### 5.3 A 3D contact problem

A 3D contact problem of a bolted assembly is presented here (Fig. 10). The purpose of this example is to show that the non-invasive approach permits to tackle a complex 3D case with frictional contact. The assembly is composed of two
plates and three bolts, symmetry is exploited to limit the size of the problem. The problem is composed of more than 1.2 million degrees of freedom. Each plate is decomposed to three sub-domains. The bolts are also sub-structured into three subdomains. One is the nut whereas the other two compose the screw. The interface between the two subdomains of the screw permits to impose the preload.

The Young modulus is $E = 200$ GPa, the Poisson’s ratio is $\nu = 0.3$. The frictional interface has a Coulomb coefficient of $\mu = 0.1$ and the gap between the screw and the plate is 0.001 mm. These parameters are recalled in Table 3. The
The problem is computed in four time steps: the first corresponds to the pre-load of the structure, the other ones represent a linear traction force $F$ of the upper plate (applied on the right side of the plate) (Fig. 11). The lower plate is clamped on the other side of the assembly (on the left).

**Remark 9** The preload is managed with a contact interface. An initial negative gap is imposed at the time step $t_1$, $B_{n|t=t_1} = -j_n$ with $j_n > 0$, which results in a tension stress in the screw.

The deformed shape at the last time step ($\times 500$) is shown on the Fig. 12. Slip is observed and provokes contact between the screws and the plates. On this figure, Von Mises stress is also illustrated and it shows that several parts that are solicited in the assembly: stress concentrates around the screws and near the clamp.

On Fig. 13a, we represent the elements in compression ($\sigma_{zz}$ is negative) when the preload is imposed. On this example we find the classical Rötscher’s pressure cone under the head of the screws [35] and the tension in the body of the screws on Fig. 13b. The shearing stress $\sigma_{xz}$ at the last time step is shown on Fig. 14a. Stress concentrates under heads of screws, due to relative movements of plates. The different contact zones are represented in black on Fig. 15 and the detachment of the two plates is visible between the screws. The error indicator is presented on Fig. 16.

![Von Mises stress (MPa) at the last time increment- deformed shape ×500](image1)

![Stress in MPa at the time step $t_1$, a Rotscher’s cone after preload: $\sigma_{zz} < 0$, b Tension in the screw after preload: $\sigma_{zz} > 0$](image2)

![Stress in MPa at the last time step $t_f$, a negative shearing stress $\sigma_{xz}$, b negative shearing stress $\sigma_{xz}$](image3)
6 Conclusion

In this paper we have presented a non-invasive derivation of a mixed domain decomposition capable of dealing with frictional contact interfaces. In particular the linear stage of the Latin method has been modified to handle Robin conditions over the boundaries of the subdomains. A nonlocal stiffness operator has been proposed to compute the search direction on the interfaces. It transforms the standard formulation into a simple assembly of stiffness operators over the subdomains that any commercial software is able to compute.

The implementation of this derivation has been validated on a simple test case. Classical results of the Latin method about the influence of the search directions can be applied to our new non-invasive method.

In order to remain efficient, the contact formulation imposes a node-wise search direction at the local stage, which implies using different search directions at the local and linear stages ($k_l^+ \neq k_l^-$). We choose to use the diagonal of the stiffness operator of the sole in the local frame. According to that operation we transform the nonlocal behavior of the interfaces into a decoupled behavior suited to contact formulation. Numerical examples on a 2D academic case confirm the convergence of the algorithm and a first 3D contact problem of a bolted assembly has been presented.

Future works will consist in insuring the scalability of the method. This multi-scale approach, in the sense of [29], modifies the linear stage by introducing a macro problem that ensures a global equilibrium of force distributions. This approach will be seen as a particular choice of the search direction that highlights a specific macro projector. The multi-scale scalability will be coupled with a parallel implementation of the algorithm.

Acknowledgements We thank Airbus Group Innovations and EDF for their financial and technical support.

References

1. Alart P, Curnier A (1991) A mixed formulation for frictional contact problems prone to Newton like solution methods. Comput Methods Appl Mech Eng 92(3):353–375
