Comput Mech
DOI 10.1007/s00466-015-1239-x

ORIGINAL PAPER

The domain interface method: a general-purpose non-intrusive technique for non-conforming domain decomposition problems

M. Cafiero[1] · O. Lloberas-Valls[1] · J. Cante[1,3] · J. Oliver[1,2]

Received: 12 May 2015 / Accepted: 9 December 2015

Abstract A domain decomposition technique is proposed which is capable of properly connecting arbitrary non-conforming interfaces. The strategy essentially consists in considering a fictitious zero-width interface between the non-matching meshes which is discretized using a Delaunay triangulation. Continuity is satisfied across domains through normal and tangential stresses provided by the discretized interface and inserted in the formulation in the form of Lagrange multipliers. The final structure of the global system of equations resembles the dual assembly of substructures where the Lagrange multipliers are employed to nullify the gap between domains. A new approach to handle floating subdomains is outlined which can be implemented without significantly altering the structure of standard industrial finite element codes. The effectiveness of the developed algorithm is demonstrated through a patch test example and a number of tests that highlight the accuracy of the methodology and independence of the results with respect to the framework parameters. Considering its high degree of flexibility and non-intrusive character, the proposed domain decomposition framework is regarded as an attractive alternative to other established techniques such as the mortar approach.

✉ J. Oliver
oliver@cimne.upc.edu

[1] CIMNE – Centre Internacional de Metodes Numerics en Enginyeria, Barcelona, Spain

[2] E.T.S d'Enginyers de Camins, Canals i Ports, Technical University of Catalonia (Barcelona Tech), Campus Nord UPC, Mòdul C-1, c/ Jordi Girona 1-3, 08034 Barcelona, Spain

[3] E.T.S. d'Enginyeries Industrial i Aeronàutica de Terrassa, Technical University of Catalonia (Barcelona Tech), Campus Terrassa UPC, c/ Colom 11, 08222 Terrassa, Spain

Keywords Domain decomposition methods · Non-conforming interface · Weak coupling techniques for non-matching meshes

1 Introduction

Modern engineering applications require sophisticated simulation techniques which deal with increasing complexity and refinement of the computational models. Consequently, detailed finite element discretizations are commonly used in nowadays structural analysis and a number of practical situations are emerging in which special techniques are indispensable to handle non-matching discretizations. In this introduction we focus on engineering applications and computational techniques concerning the assembly and resolution of models involving non-overlapping meshes.

1.1 The need for non-matching mesh assemblies in computational mechanics

Typical scenarios arise when independent mesh discretizations are applied to different parts of a structure and when large models are divided and distributed among different working teams. A common situation is encountered when particular structural components are reused in evolving designs such as the wings among diverse aircraft models with changing fuselages. The meshes of the structural components are most likely non-matching and need to be assembled along common edges using special techniques that account for the non-conforming interfaces [34,35].

The field of contact mechanics [51] has significantly boosted the formulation of new assembly techniques since the most general situations between contact surfaces are

Published online: 28 December 2015

encountered therein, e.g. sliding bodies over a surface or rolling and rebounding of different discretized entities. Other somehow related applications are connected with fluid structure interaction [7] and multiphysics [14,41] analysis where different discretizations are taken into account due to the distinct physical nature of interacting components.

An emerging set of techniques intimately related with computational material design are the so-called multiscale and multiresolution methods [13,17,27]. The idea is to account for the lower scale components and their interactions with an upper scale level. Multiresolution techniques based on mesh adaptivity [33] and concurrent multiscale analysis such as global/local approaches [8], variational multiscale methods [25,26,36] and multiscale domain decomposition methods [15,20,28] are examples in which lower (fine) scale discretizations are glued to upper (coarse) scale models. This can be performed selectively during the computations at areas of interest, e.g. stress concentrations, crack growth and appearance of non-linear effects, by "zooming" into these regions and substituting a part of the domain discretization by its corresponding refined model [30,31]. As a result, a number of non-conforming interfaces between different scale discretizations arise which need to be handled by appropriate techniques.

Most of the above mentioned applications involve complex models which upon discretization, e.g. using finite elements, lead to large systems of equations. It is then not surprising that most of the existing methodologies to connect different meshes are frequently encountered in domain decomposition techniques [19]. These techniques roughly divide the complex model into subdomains and distribute the corresponding calculations among different processor units. They can be viewed as powerful parallel solvers typically formed by a blend of direct solvers that account for the local domain factorizations and iterative solvers for an interface problem that accounts for the connectivity of all domains. It is precisely in the generation of such connectivity where the techniques discussed in this paper play a crucial role since they ensure continuity of the solution field across all conforming and non-conforming interfaces.

1.2 Non-overlapping domain decomposition analysis with non-conforming interfaces

In most of the domain decomposition applications the geometrical compatibility of the interface Γ_I is assumed to be satisfied between non-matching meshes [44] meaning that the boundaries of the domains at the common interface are identical in the undeformed configuration. However, there are practical situations in which this is not the case, e.g. curved interfaces with different piece-wise linear discretizations (cf. Fig. 1). Extensions to geometrical incompatibility have been specially addressed by contact domain techniques [43,51]. Domain decomposition techniques designed to tackle non-conforming interfaces typically employ strong or weak coupling constraints to satisfy compatibility of the solution field at the interface.

Strong coupling constraints refer essentially to collocation techniques where one constraint is assigned to each degree of freedom (DOF) while weak coupling techniques refer to constraints enforced in an integral or average sense along portions of the interface surface assigning one constraint per group of DOFs belonging to the interface portion. In the contact domain literature these techniques are often referred to as node-to-segment techniques and segment-to-segment techniques, respectively.

Generally, a reference displacement field $\mathbf{u}_{\Gamma_\mathrm{I}}$ is outlined from which the interface constraints are generated. Such field can be chosen based on one of the domain discretizations at the common interface or by a third auxiliary interface discretization with a particular optimal distribution of the DOFs such that the gap between domains is minimized in some norm (cf. Fig. 2). Note that if the coarsest discretization is utilized for the reference solution field the solution can be artificially stiffened if the fine mesh is not obtained by h or p refinement of the coarse mesh at the interface for the strong coupling case. For this reason, weak coupling techniques are preferred in the context of a general non-conforming interface since they relax the constraint such that stiffening and locking effects do not occur. Mortar methods and their formulation in terms of finite elements constitute the most general technique for non-conforming interfaces in which geometrical incompatibilities can be handled [5]. The most challenging applications of such techniques are identified in time-dependent domains, e.g. contact problems, and when highly refined domains are considered at local areas without the need for an expensive mesh adaption, e.g. multiresolution analysis.

The enforcement of constraints at a non-conforming interface is typically done with the introduction of Lagrange multipliers [1]. Basically, an extra term is added into the variational statement which corresponds to the work performed at the non-conforming interface in terms of the existing gap and interface stresses. In the context of the finite element method different discretizations are generated at each domain and distribution functions are associated with the Lagrange multipliers at the interface leading to a discretized weak form of the equilibrium equations and the interface compatibility constraints. The distribution functions for the Lagrange multipliers and shape functions for the finite elements should be properly selected to fulfill the Ladyzhenskaya–Babuška–Brezzi (LBB) condition (also known as the *inf-sup* condition) [2] in order to guarantee that both discretizations converge to the right solution upon

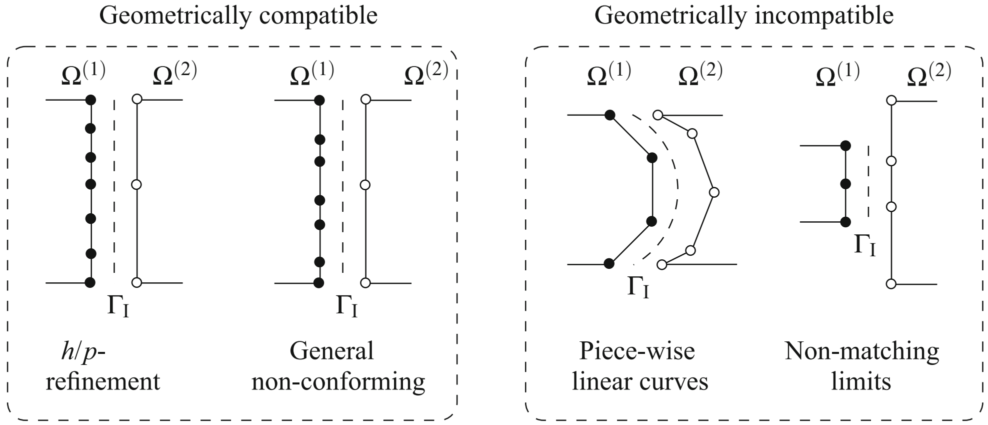


Fig. 1 Different situations for a non-conforming interface

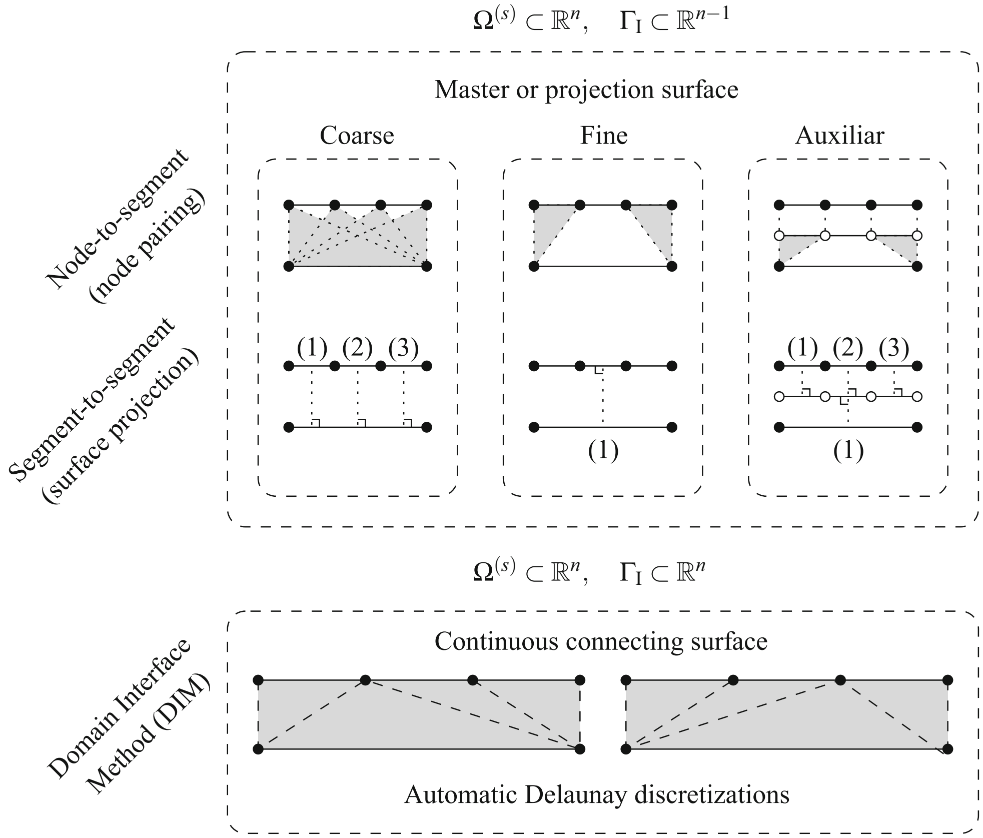

$\Omega^{(s)} \subset \mathbb{R}^n, \quad \Gamma_{\rm I} \subset \mathbb{R}^{n-1}$
Master or projection surface
Coarse
Fine
Auxiliar
Node-to-segment (node pairing)
Segment-to-segment (surface projection)
(1)
(2)
(3)
$\Omega^{(s)} \subset \mathbb{R}^n, \quad \Gamma_{\rm I} \subset \mathbb{R}^{n}$
Continuous connecting surface
Domain Interface Method (DIM)
Automatic Delaunay discretizations

Fig. 2 Connecting non-conforming interfaces with different strategies

mesh refinement. Other techniques to enforce the constraints are related with the introduction of a penalty term which associates a high cost to the violation of the compatibility constraint. This is the case of penalty methods which have the advantage of not incorporating extra DOFs to the system of equations although the penalty term can influence the solution. Methods based on Augmented Lagrange multipliers seek for an optimal compromise between penalty and Lagrange multipliers allowing an exact enforcement in combination with a penalty-like regularization which improves the numerical treatment. In such methods the constraint violation is also penalized but, quite in contrast, the solution is not influenced by the penalty term. In fact, the convexity of the functional is increased to facilitate the search of its minimum. Explicit elimination of the constraints can be performed as well leading to a system of equations with no extra unknowns. However, these methods are not straightforward to implement for the case of non-matching meshes since they require to compute a null space of the compatibility matrices used in the equivalent dual formulation or constructing a projection operator which can be demanding in terms of storage. The reader is referred to the work of Rixen [45] for an overview of such techniques in the context of domain decomposition methods. Yet another method to enforce the constraints without the introduction of Lagrange multipliers was introduced by Nitsche [37] which can be regarded as an intermediate technique between the Lagrange multiplier and the penalty method. It essentially modifies the weak form by adding a term including a positive constant parameter that enforces the Dirichlet boundary condition. Such modification depends on the particular problem but it does not lead to an ill-conditioned system as the penalty method would do for large values of the penalty parameter. In fact the stabilization term exhibits less influence on the solution than standard penalty methods and, in practice, large values are not needed in order to ensure convergence and a proper enforcement of the constraints. A Nitsche method to handle the interface constraints derived from domain decomposition methods was introduced in [4]. In order to avoid integrating products of functions on unrelated meshes, the Lagrange multiplier method can be adopted to enforce the interface constraints and a Nitsche method can be employed to stabilize the system. This is accomplished by introducing an extra term in the variational principle that couples the multipliers with the stress fields at the interface [23]. Although the system of equations is augmented due to the introduction of the Lagrange unknowns, no constraints are needed for the discretization of the hybrid solution field since stabilization is accounted for by the extra "penalty" term. This technique has proven to be specially useful for the constraint enforcement in contact domain methods [21,38] and will be utilized in the present contribution.

Domain decomposition frameworks typically found in literature that account for non-conforming interfaces are based on the introduction of Lagrange multipliers to weakly enforce the compatibility constraints. This is the case for the mortar approach [5] which is currently the most general and well established methodology. It essentially consists on a segment-to-segment discretization strategy where one of the domain surfaces at the interface is considered the 'mortar' (master) surface whilst the other is the 'non-mortar' (or slave). There are also variants of the approach where a third intermediate surface is considered with a reference displacement, however it obviously leads to an increase of the number of DOFs. Constraints are, therefore, weakly enforced by minimizing the gap with respect to the mortar surface. This methodology presents an obvious disadvantage when the selected mortar discretization is significantly coarser than the non-mortar one since a higher error can be obtained at the interface and might not satisfy the patch test. The dual domain decomposition method proposed by Herry et al. [24] presents a highly accurate technique to glue non-conforming interfaces by means of Lagrange Multipliers. It basically uses a third interface discretization with optimal location of the DOFs such that the kinematic continuity at the interface is exact. However, the technique is only valid for geometrically compatible non-conforming interfaces and, for this reason, arbitrary curved interfaces and interfaces which do not share the limit nodes can not be considered therein. The localized Lagrange multipliers (LLM) method proposed by Park et al. [39] consists in the introduction of a third interface surface with a specific discretization in order to collocate Lagrange multipliers to enforce the constraints at the non-matching meshes. Such discretization is performed in order to *a priori* satisfy the constant stress patch condition. The technique can be viewed as a general and optimal node-to-segment approach applied to the connection frame but still arbitrary highly irregular grids and geometrically incompatible interfaces are not addressed.

In contrast with the above mentioned techniques we propose a general and flexible methodology to account for the most complex interface situations, i.e. geometrically incompatible and arbitrary non-conforming. The main idea in the domain interface method (DIM) is to explicitly discretize the interface through a Delaunay triangulation. In all previously introduced techniques one slave node or segment belonging to a domain interface was somehow projected on the other domain interface or on an auxiliary one. Therefore, the interface constraints were formulated in a domain which is one dimension lower than the subdivided domains. In the DIM, the interface constraints are formulated on an intermediate interface of the same dimension as the adjacent decomposed domains (cf. Fig. 2). Consequently, the interface surface is continuous and uniquely defined upon the Delaunay discretization without any assumptions on the master/slave side. This results in full and non-overlapping connections leading to satisfactory results concerning the constant stress patch test. The geometrical details, weak form and FE implementation are given in Sect. 2 and a number of representative simulations are commented in Sect. 3 in order to highlight the advantages and applicability of the proposed approach against other established methodologies.

2 Formulation of the DIM method

In this section the necessary geometrical aspects of the methodology are introduced and strong and weak forms of the problem are outlined. The solvability of the resulting system is discussed in terms of its stabilization and possible resolution choices including a parallel scheme. The main idea behind the DIM method concerns an explicit meshing of the interface between domains and is inspired in the methodology introduced in [21,38] for contact mechanics. Rather than a particularization of the contact domain method for the case of tied contact the DIM equations stem from exporting the concept of the interface domain and the generality of the contact interface connections to the family of domain decomposition methods. In this manner, a new set of techniques within the domain decomposition methods is devised such as a non-intrusive methodology to handle rigid body modes (RBMs) without the need for extending the solution field to the RBM intensities as it is frequently done in established methodologies [11]. For the sake of completeness the methodology is introduced considering a finite strain case. Infinitesimal deformations are recovered by considering the necessary simplifications in the presented theory (i.e. small displacements compared to the domain dimensions and negligible gradients of such displacements). Compact notation will be utilized for tensor quantities throughout the document unless a different notation is specifically mentioned.

2.1 Geometrical description of the DIM

Consider the structure assembly depicted in Fig. 3 (top) where the domain Ω is composed by the union of N_{s} non-overlapping domains $\Omega^{(s)}$. At each domain $\Omega^{(s)}$ one can identify the regions were Dirichlet $\Gamma_{\mathrm{u}}^{(s)}$ and Neumann $\Gamma_{\sigma}^{(s)}$ boundary conditions are imposed. The interface $\Gamma_{\mathrm{I}}^{(s)} = \partial\Omega^{(s)} \cap \partial\Omega^{(q)}$ with outward unit normal $\boldsymbol{\nu}^{(s)}$ where $\partial\Omega$ stands for the domain boundaries of the adjacent domains s and q. Discretizations of the two bodies, e.g. using finite elements (FE), leads to a number of $N_{\lambda}^{(s)}$ vertices at the domain boundary $\partial\Omega^{(s)}$ located at the vicinity of $\Gamma_{\mathrm{I}}^{(s)}$ which need to be involved in the interface discretization.

The interface generation (cf. Fig. 3) starts with a fictitious contraction of the vertices V_i in the direction $-\boldsymbol{\nu}^{(s)}$ by a magnitude k. The result will be independent on the chosen magnitude k but in our analyses $k \approx h_{\mathrm{e}}$, being h_{e} an average of the equivalent FE size. The fictitious coordinates $\mathbf{x}_i'$ are utilized to generate a Delaunay triangulation which defines the interface domain

$$D = \bigcup_{p=1}^{N_{\mathrm{p}}} D^{(p)}, \tag{1}$$

where $D^{(p)}$ represents each of the interface patches up to N_{p}. The interface surface $\Gamma_{\mathrm{D}}^{(s)} = \partial\Omega^{(s)} \cap \partial D$ and, for the case of a geometrically compatible interface, $\Gamma_{\mathrm{I}}^{(s)} = \Gamma_{\mathrm{D}}^{(s)}$.

Remark 2.1 Note that, upon discretization of the domain interface D using triangular linear elements $D^{(p)}$, the integrals over a geometrically incompatible interface when $h \to 0$ converge to a bounded value:

$$\sum_{p=1}^{N_{\mathrm{p}}} \lim_{h_p \to 0} \int_{D^{(p)}} \frac{1}{h_p} (\bullet)\, \mathrm{d}D = \frac{1}{2} \sum_{p=1}^{N_{\mathrm{p}}} \int_{L^{(p)}} (\bullet)\, \mathrm{d}L \; + \mathcal{O}(h), \tag{2}$$

where $\mathcal{O}(h)$ corresponds to the error caused by approximating the interface geometry with piece-wise linear segments. Considering the case of geometrically compatible interfaces $\mathcal{O}(h) \to 0$ and

$$\sum_{p=1}^{N_{\mathrm{p}}} \lim_{h_p \to 0} \int_{D^{(p)}} \frac{1}{h_p} (\bullet)\, \mathrm{d}D \approx \frac{1}{2} \sum_{p=1}^{N_{\mathrm{p}}} \int_{L^{(p)}} (\bullet)\, \mathrm{d}L. \tag{3}$$

Since the integration is performed over $D^{(p)}$ when $h \to 0$ it is pointed out that both adjacent domain interfaces $\Gamma_{\mathrm{D}}^{(s)}$ and $\Gamma_{\mathrm{D}}^{(q)}$ are automatically accounted for. This is in contrast with other methodologies, e.g. the mortar method [5], where only one of the adjacent interface discretizations is considered.

Consider the assembly of two non-overlapping domains $\Omega^{(s)}$, $s = 1, 2$, (cf. Fig. 3) undergoing finite strains. The corresponding deformation maps are denoted by $\phi_t^{(s)}(\mathbf{X}) \equiv \phi^{(s)}(\mathbf{X}, t) \;:\; \Omega_0^{(s)} \times [0, T] \to \Omega_t^{(s)}$, where material points $\mathbf{X}^{(s)} \in \Omega_0^{(s)}$ at the reference configuration are mapped onto the current configuration $\mathbf{x}^{(s)} = \phi_t^{(s)}(\mathbf{X}) \in \Omega_t^{(s)}$. Such a mapping can be expressed in terms of the total displacement field $\mathbf{U}_t^{(s)}(\mathbf{X}^{(s)})$ which denotes the difference between reference and current coordinates satisfying

$$\mathbf{x}^{(s)} = \phi_t^{(s)}(\mathbf{X}) = \mathbf{X}^{(s)} + \mathbf{U}_t^{(s)}(\mathbf{X}^{(s)}). \tag{4}$$

In our quasi-static analyses a pseudo-time domain $t \in [0, t]$ is considered with subdivisions in discrete intervals $[t_n, t_{n+1}]$ of incremental time length $\Delta t = t_{n+1} - t_n$. Configurations at the previous t_n and current t_{n+1} times are denoted as $\Omega_n^{(s)} = \phi_{t_n}^{(s)}(\Omega_0^{(s)}) \equiv \phi_n^{(s)}(\Omega_0^{(s)})$ and $\Omega_{n+1}^{(s)} = \phi_{t_{n+1}}^{(s)}(\Omega_0^{(s)}) \equiv \phi_{n+1}^{(s)}(\Omega_0^{(s)})$, respectively. Therefore, an expression of the incremental motion $\phi^{(s)}$ can be found by substitution of the current and previous configurations in (4) as

$$\left.\begin{array}{l} \mathbf{x}_{n+1}^{(s)} = \phi_{t_{n+1}}^{(s)}(\mathbf{X}^{(s)}) \\ \mathbf{x}_{n}^{(s)} = \phi_{t_{n}}^{(s)}(\mathbf{X}^{(s)}) \end{array}\right\} \Rightarrow \begin{array}{l} \mathbf{x}_{n+1}^{(s)} = \phi_{n+1}^{(s)}\left((\phi_n^{(s)})^{-1}(\mathbf{x}_n^{(s)})\right) = \phi^{(s)}(\mathbf{x}_n^{(s)}) \\ \forall \mathbf{x}_n^{(s)} \in \Omega_n^{(s)}. \end{array} \tag{5}$$

With the incremental motion in hand, the incremental field

$$\mathbf{u}^{(s)}(\mathbf{x}_n^{(s)}) = \phi^{(s)}(\mathbf{x}_n^{(s)}) - \mathbf{x}_n^{(s)} = \mathbf{x}_{n+1}^{(s)} - \mathbf{x}_n^{(s)}, \quad \forall \mathbf{x}_n^{(s)} \in \Omega_n^{(s)}. \tag{6}$$

The incremental motion of the interface domain

$$\phi^{\mathrm{D}}(\mathbf{x}_n^{(s)}) \equiv \mathbf{x}_{n+1}(\mathbf{x}_n) = \mathbf{x}_n + \mathbf{u}^{\mathrm{D}}(\mathbf{x}_n), \quad \forall \mathbf{x}_n \in D_n^{(p)}, \tag{7}$$

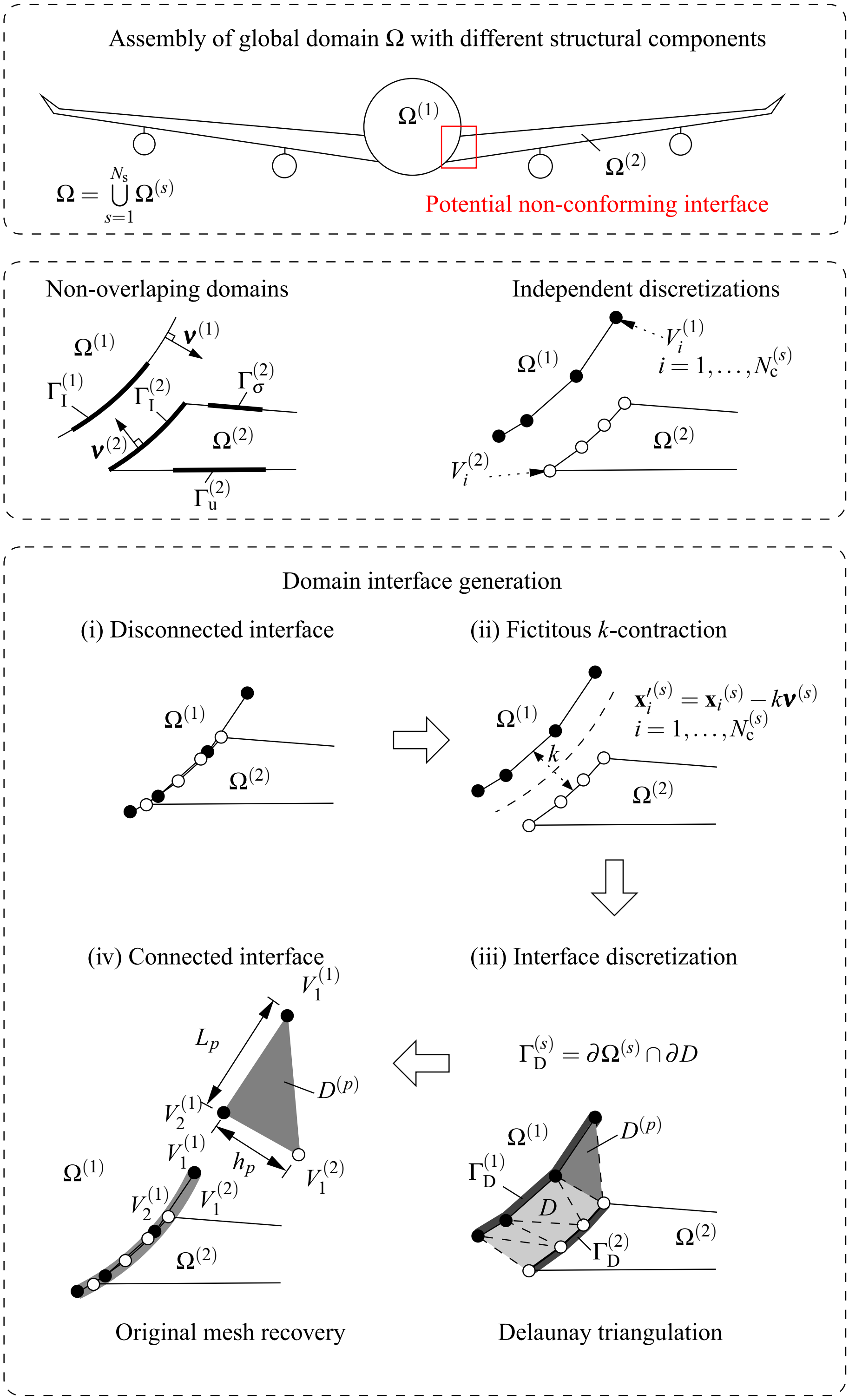

Assembly of global domain Ω with different structural components
$\Omega^{(1)}$
$\Omega^{(2)}$
$\Omega = \bigcup_{s=1}^{N_s} \Omega^{(s)}$
Potential non-conforming interface
Non-overlaping domains
$\Omega^{(1)}$
$\boldsymbol{\nu}^{(1)}$
$\Gamma_{\mathrm{I}}^{(1)}$
$\Gamma_{\mathrm{I}}^{(2)}$
$\Gamma_{\sigma}^{(2)}$
$\boldsymbol{\nu}^{(2)}$
$\Omega^{(2)}$
$\Gamma_{\mathrm{u}}^{(2)}$
Independent discretizations
$V_i^{(1)}$
$i = 1,\ldots,N_{\mathrm{c}}^{(s)}$
$\Omega^{(1)}$
$\Omega^{(2)}$
$V_i^{(2)}$
Domain interface generation
(i) Disconnected interface
$\Omega^{(1)}$
$\Omega^{(2)}$
(ii) Fictitous k-contraction
$\Omega^{(1)}$
$\mathbf{x}_i'^{(s)} = \mathbf{x}_i^{(s)} - k\boldsymbol{\nu}^{(s)}$
$i = 1,\ldots,N_{\mathrm{c}}^{(s)}$
k
$\Omega^{(2)}$
(iv) Connected interface
$V_1^{(1)}$
L_p
$D^{(p)}$
$V_2^{(1)}$
$V_1^{(1)}$
h_p
$V_1^{(2)}$
$\Omega^{(1)}$
$V_2^{(1)}$
$V_1^{(2)}$
$\Omega^{(2)}$
Original mesh recovery
(iii) Interface discretization
$\Gamma_{\mathrm{D}}^{(s)} = \partial\Omega^{(s)} \cap \partial D$
$\Omega^{(1)}$
$D^{(p)}$
$\Gamma_{\mathrm{D}}^{(1)}$
D
$\Gamma_{\mathrm{D}}^{(2)}$
$\Omega^{(2)}$
Delaunay triangulation

Fig. 3 Generation of the domain interface: (*i*) domain discretizations, (*ii*) fictitious domain contractions, (*iii*) Delaunay triangulations and (*iv*) original mesh recovery

$D_n^{(p)}$ denoting the interface domain D at time t_n. Therefore, $D_{n+1} = \phi^{\mathrm{D}}(D_n)$ and $\gamma_{\mathrm{D}} = \phi^{\mathrm{D}}(\Gamma_{\mathrm{D}})$ which represent the current and previous domain interface and interface surfaces, respectively (cf. Fig. 4). The incremental displacement field at the interface domain $\mathbf{u}^{\mathrm{D}}$ is calculated by linearly interpolating the displacement increments d_i^{D} of the interface element vertices as

$$\mathbf{u}^{\mathrm{D}}(\mathbf{x}_n) \equiv \mathbf{u}^{(p)}(\mathbf{x}_n) = \sum_{i=1}^{3} \mathbb{N}_i(\mathbf{x}_n) d_i^{\mathrm{D}}, \quad \forall \mathbf{x}_n \in D_n^{(p)}, \tag{8}$$

$\mathbb{N}_i$ denoting linear shape functions for three-node triangular finite elements. The incremental gradient deformation tensor

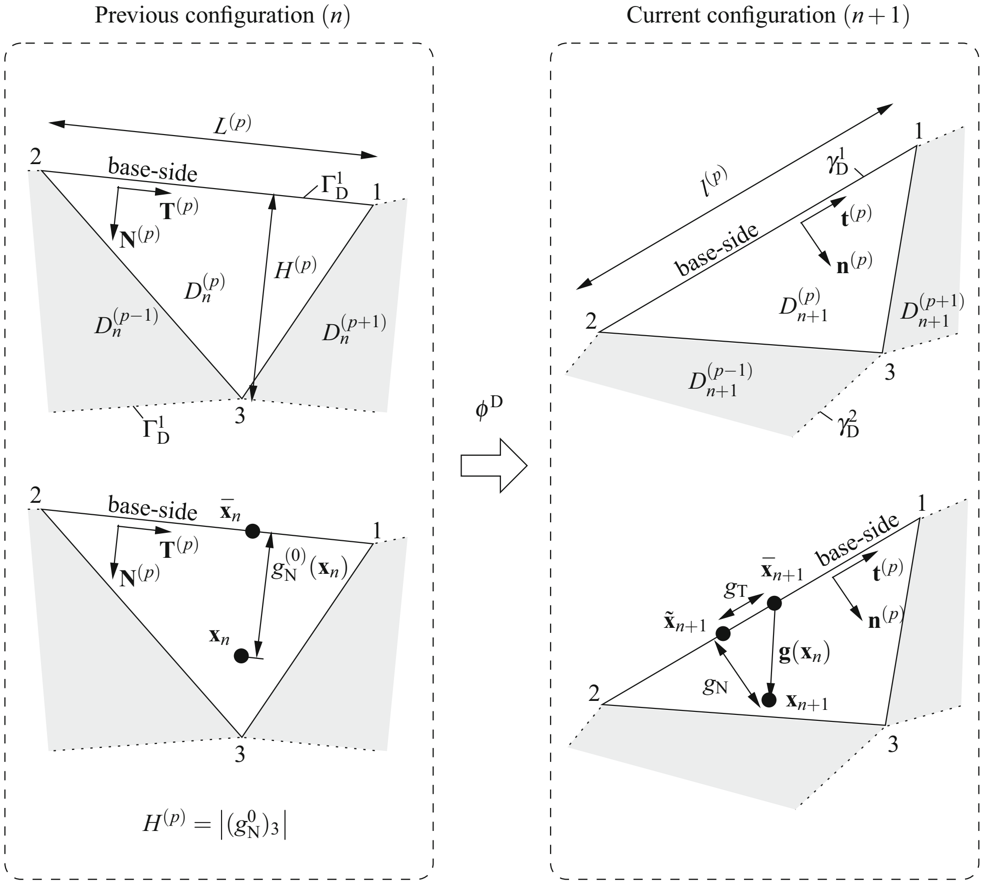


Fig. 4 Geometrical definition of the gap $\mathbf{g}(\mathbf{x}_n)$ according to the previous (*left*) and current (*right*) configurations

$$\mathbf{f}^{\mathrm{D}} = \bar{\boldsymbol{\nabla}}\left(\phi^{\mathrm{D}}(\mathbf{x}_n)\right) = \frac{\partial \mathbf{x}_{n+1}}{\partial \mathbf{x}_n} = \mathbf{1} + \bar{\boldsymbol{\nabla}}(\mathbf{u}^{\mathrm{D}}), \tag{9}$$

where $\mathbf{1}$ represents de second order unity tensor and $\bar{\boldsymbol{\nabla}}$ denotes the material gradient with respect to the reference previous configuration n.

Remark 2.2 It is important to note that $\mathbf{f}^{\mathrm{D}}(\mathbf{x}_n) \equiv \mathbf{f}^{(p)} = constant, \quad \forall \mathbf{x}_n \in D^{(p)}$ due to the linear character of the incremental displacements defined in (8).

As depicted in Fig. 4, each interface patch $D^{(p)}$ contains a base-line defined on $\Gamma_{\mathrm{D}}^{(p)}$ with unit normal vector $\mathbf{N}^{(p)}$ in the sense of the normal to the adjacent domain $\boldsymbol{\nu}^{(s)}$. The tangential vector $\mathbf{T}^{(p)} = \hat{\mathbf{e}} \times \mathbf{N}^{(p)}$, where $\hat{\mathbf{e}}$ denotes the out-of-plane unit vector forming the ordered triplet of unit vectors $\left\{\mathbf{N}^{(p)}, \mathbf{T}^{(p)}, \hat{\mathbf{e}}\right\}$. Considering the incremental motion ϕ^{D}, the current tangential and normal unit vectors read

$$\mathbf{t}^{(p)} = \frac{\phi^{\mathrm{D}}(\mathbf{T}^{(p)})}{\left|\left|\phi^{\mathrm{D}}(\mathbf{T}^{(p)})\right|\right|} = \frac{\mathbf{f}^{(p)} \cdot \mathbf{T}^{(p)}}{\left|\left|\mathbf{f}^{(p)} \cdot \mathbf{T}^{(p)}\right|\right|} \tag{10}$$
$$\mathbf{n}^{(p)} = \mathbf{t}^{(p)} \times \hat{\mathbf{e}}.$$

The definition of the normal and tangential vectors depends on the local base-line for every interface patch, therefore these vectors are constant within every patch (cf. 10) but discontinuous across the interface patches.

The initial normal gap g_{N}^{0} (cf. Fig. 4) is defined at the previous configuration n for a given point $\mathbf{x}_n$ and its normal projection to the base-line $\bar{\mathbf{x}}_n$ as

$$g_{\mathrm{N}}^{0}(\mathbf{x}_n) = (\mathbf{x}_n - \bar{\mathbf{x}}_n) \cdot \mathbf{N}^{(p)}. \tag{11}$$

The final gap vector

$$\mathbf{g}(\mathbf{x}_n) = \mathbf{x}_{n+1} - \bar{\mathbf{x}}_{n+1} = \phi^{\mathrm{D}}(\mathbf{x}_n) - \phi^{\mathrm{D}}(\bar{\mathbf{x}}_n), \tag{12}$$

where $\mathbf{x}_{n+1}$ and $\bar{\mathbf{x}}_{n+1}$ stand for the convected points $\mathbf{x}_n$ and $\bar{\mathbf{x}}_n$, respectively. The final gap can be expressed as a sum of its normal and tangential projections onto the current base-side as

$$\mathbf{g}(\mathbf{x}_n) = g_{\mathrm{N}}(\mathbf{x}_n)\mathbf{n}^{(p)} + g_{\mathrm{T}}(\mathbf{x}_n)\mathbf{t}^{(p)} \Rightarrow \left\{\begin{array}{l} g_{\mathrm{N}}(\mathbf{x}_n) = \mathbf{g}(\mathbf{x}_n) \cdot \mathbf{n}^{(p)} \\ g_{\mathrm{T}}(\mathbf{x}_n) = \mathbf{g}(\mathbf{x}_n) \cdot \mathbf{t}^{(p)}. \end{array}\right. \tag{13}$$

Consequently, the normal gap $g_{\mathrm{N}}(\mathbf{x}_n)$ can be seen as a projection in the direction of $\mathbf{n}^{(p)}$ which denotes penetration for negative values. Similarly, the tangential gap $g_{\mathrm{T}}(\mathbf{x}_n)$ represents the slid distance in the sense of $\mathbf{t}^{(p)}$. It should be stressed that the gap definition utilized in this manuscript and already introduced in [21,38] is not standard in the sense that it is defined as a patch-wise continuous function throughout the patch (p) and not only defined at the interface nodes as other methodologies would consider. For future use into the variational statement it is convenient to express the gap in terms of the displacement field and, to this end, a Taylor series expansion of $\phi^{\mathrm{D}}(\mathbf{x}_n)$ is considered around $\bar{\mathbf{x}}_n$ up to second order terms, with $\mathbf{x}_n - \bar{\mathbf{x}}_n = g^0_{\mathrm{N}}(\mathbf{x}_n)\mathbf{N}^{(p)}$. In this spirit

$$\begin{aligned} \mathbf{x}_{n+1} &= \phi^{\mathrm{D}}(\mathbf{x}_n) = \phi^{\mathrm{D}}\left(\bar{\mathbf{x}}_n + g^0_{\mathrm{N}}(\mathbf{x}_n))\mathbf{N}^{(p)}\right) \\ &= \phi^{\mathrm{D}}(\bar{\mathbf{x}}_n) + \bar{\nabla}\left(\phi^{\mathrm{D}}(\bar{\mathbf{x}}_n)\right)\cdot\left(g^0_{\mathrm{N}}(\mathbf{x}_n))\mathbf{N}^{(p)}\right) + \mathcal{O}({\mathbf{x}_n}^2) \\ &= \bar{\mathbf{x}}_{n+1} + g^0_{\mathrm{N}}(\mathbf{x}_n))\mathbf{f}^{(p)}\cdot\mathbf{N}^{(p)} + \mathcal{O}({\mathbf{x}_n}^2). \end{aligned} \tag{14}$$

Second order $\mathcal{O}({\mathbf{x}_n}^2)$ and higher order terms can be neglected due to the linear character of $\mathbf{f}^{(p)}$ and, therefore, the final expression for the gap vector as a function of the displacements reads

$$\begin{aligned} \mathbf{g}(\mathbf{x}_n) &= \mathbf{x}_{n+1} - \bar{\mathbf{x}}_{n+1} = g^0_{\mathrm{N}}(\mathbf{x}_n))\mathbf{f}^{(p)}\cdot\mathbf{N}^{(p)} \\ &= g^0_{\mathrm{N}}(\mathbf{x}_n))\left(\mathbf{1} + \bar{\nabla}(\mathbf{u}^{(p)})\right)\cdot\mathbf{N}^{(p)}. \end{aligned} \tag{15}$$

In the same spirit as in (13), the final gap $\mathbf{g}(\mathbf{x}_n) = g_{\mathrm{N}}(\mathbf{x}_n)\mathbf{n}^{(p)} + g_{\mathrm{T}}(\mathbf{x}_n)\mathbf{t}^{(p)}$ with

$$\begin{aligned} g_{\mathrm{N}}(\mathbf{x}_n) &= \mathbf{n}^{(p)}\cdot\mathbf{g}(\mathbf{x}_n) = g^0_{\mathrm{N}}(\mathbf{x}_n))\mathbf{n}^{(p)}\cdot\mathbf{f}^{(p)}\cdot\mathbf{N}^{(p)} \\ &= g^0_{\mathrm{N}}(\mathbf{x}_n))\mathbf{n}^{(p)}\cdot\left(\mathbf{N}^{(p)} + \bar{\nabla}(\mathbf{u}^{(p)})\cdot\mathbf{N}^{(p)}\right), \\ g_{\mathrm{T}}(\mathbf{x}_n) &= \mathbf{t}^{(p)}\cdot\mathbf{g}(\mathbf{x}_n) = g^0_{\mathrm{N}}(\mathbf{x}_n))\mathbf{t}^{(p)}\cdot\mathbf{f}^{(p)}\cdot\mathbf{N}^{(p)} \\ &= g^0_{\mathrm{N}}(\mathbf{x}_n))\mathbf{t}^{(p)}\cdot\left(\mathbf{N}^{(p)} + \bar{\nabla}(\mathbf{u}^{(p)})\cdot\mathbf{N}^{(p)}\right). \end{aligned} \tag{16}$$

Remark 2.3 The expressions of the gap match the ones obtained in node-to-segment techniques [51] when $\mathbf{x}_n$ is considered the slave node and $\bar{\mathbf{x}}_n$ is chosen as the master one. However, as already pointed out by Oliver et al. [38], the expressions in (16) can be regarded more general since they are defined continuously throughout the interface patch $D^{(p)}$ and not only for the vertices. Therefore the strategy is comparable to segment-to-segment techniques without the need for a definition of the master and slave surfaces.

In order to formulate the variational principle in Sect. 2.2 it is convenient to work with a dimensionless measure of the gap. In this view, the normal and tangential gap intensities are written relative to the absolute value of the original normal gap as

$$\bar{g}_{\mathrm{N}}(\mathbf{x}_n) = \frac{g_{\mathrm{N}}(\mathbf{x}_n)}{\left|g^0_{\mathrm{N}}(\mathbf{x}_n)\right|} \quad \text{and} \quad \bar{g}_{\mathrm{T}}(\mathbf{x}_n) = \frac{g_{\mathrm{T}}(\mathbf{x}_n)}{\left|g^0_{\mathrm{T}}(\mathbf{x}_n)\right|}. \tag{17}$$

Remark 2.4 It is important to note that the gap intensity results singular when a perfect connection is fulfilled at the previous configuration, i.e. $g^0_{\mathrm{T}}(\mathbf{x}_n) = 0$. However, the integral terms added to the variational statement that account for the work at the interface converge to a bounded value despite the kernel being unbounded as shown in (2).

Inserting (17) in (16) the final expression for the gap reads

$$\begin{aligned} \bar{g}_{\mathrm{N}}(\mathbf{x}_n) &= sign\left(g^0_{\mathrm{N}}(\mathbf{x}_n)\right)\mathbf{n}^{(p)}\cdot\left(\mathbf{N}^{(p)} + \bar{\nabla}(\mathbf{u}^{(p)})\cdot\mathbf{N}^{(p)}\right), \\ \bar{g}_{\mathrm{T}}(\mathbf{x}_n) &= sign\left(g^0_{\mathrm{N}}(\mathbf{x}_n)\right)\mathbf{t}^{(p)}\cdot\left(\mathbf{N}^{(p)} + \bar{\nabla}(\mathbf{u}^{(p)})\cdot\mathbf{N}^{(p)}\right). \end{aligned} \tag{18}$$

The interface traction vector acting at the surface $\Gamma^{(s)}_{\mathrm{D}}$ can be expressed in terms of the first Piola–Kirchoff stress tensor $\mathbf{P}^{(s)}$ at time $n+1$ with respect to the configuration at time n and the normal vector $\mathbf{N}^{(p)} = \boldsymbol{\nu}^{(s)}_n$ as

$$\mathbf{t}_{\mathrm{I}}(\mathbf{x}_n, \mathbf{N})^{(s)} = \mathbf{P}^{(s)}\cdot\mathbf{N}^{(p)}, \quad \forall\mathbf{x}_n \in \Gamma^{(s)}_{\mathrm{D}} \tag{19}$$

with normal and tangential components w.r.t. the current normal and tangential vectors

$$\left.\begin{aligned} t_{\mathrm{I,N}}(\mathbf{x}_n) &= \mathbf{n}^{(p)}\cdot\mathbf{P}^{(s)}\cdot\mathbf{N}^{(p)} \\ t_{\mathrm{I,T}}(\mathbf{x}_n) &= \mathbf{t}^{(p)}\cdot\mathbf{P}^{(s)}\cdot\mathbf{N}^{(p)} \end{aligned}\right\} \quad \forall\mathbf{x}_n \in \Gamma^{(s)}_{\mathrm{D}} \tag{20}$$

and are defined at the surface $\gamma^{(s)}_{\mathrm{D}}$ (cf. Fig. 4). Displacement compatibility between domains is enforced through the Lagrange multipliers defined constant at each patch $D^{(p)}$ as

$$\left.\begin{aligned} \lambda_{\mathrm{N}}(\mathbf{x}_n) &= t_{\mathrm{I,N}}(\mathbf{x}_n) \\ \lambda_{\mathrm{T}}(\mathbf{x}_n) &= t_{\mathrm{I,T}}(\mathbf{x}_n) \end{aligned}\right\} \quad \forall\mathbf{x}_n \in \Gamma^{(s)}_{\mathrm{D}} \tag{21}$$

which can be seen as the normal and tangential stresses that connect the adjacent domains (cf. Fig. 5).

Remark 2.5 For the case of tied contact within the present domain decompositon framework there is no need for a splitting between normal and tangential contributions. However, this splitting is introduced for the sake of completeness presenting a methodology that serves as a basis to tackle, if needed, more complex phenomena in which a different treatment can be considered for the tangential and normal interface components. Such scenarios may involve, for instance, sliding between domains. Additionally, the imposition of particular boundary conditions utilizing the Lagrange

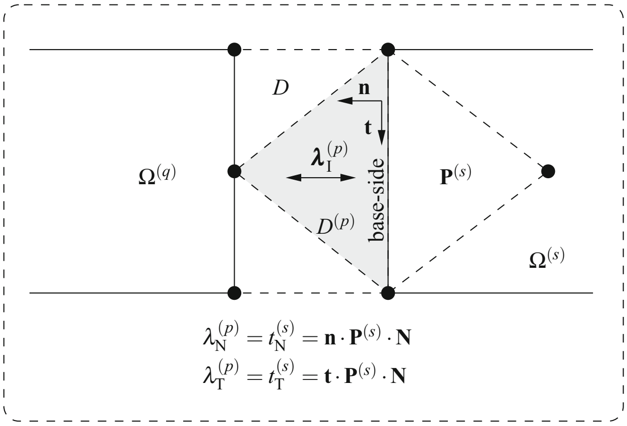

D
$\mathbf{n}$
$\mathbf{t}$
$\boldsymbol{\lambda}_{\mathrm{I}}^{(p)}$
$\Omega^{(q)}$
$\mathbf{P}^{(s)}$
base-side
$D^{(p)}$
$\Omega^{(s)}$
$\lambda_{\mathrm{N}}^{(p)} = t_{\mathrm{N}}^{(s)} = \mathbf{n} \cdot \mathbf{P}^{(s)} \cdot \mathbf{N}$
$\lambda_{\mathrm{T}}^{(p)} = t_{\mathrm{T}}^{(s)} = \mathbf{t} \cdot \mathbf{P}^{(s)} \cdot \mathbf{N}$

Fig. 5 Lagrange multiplier identification at the interface

multiplier framework can be considered as well in which the tangential component is treated differently due to friction or sliding. Specific cases involving fluid structure interaction could in this manner be treated as well. The case of normal tying along an interface segment between fully tied limit corners has been studied already within a multiscale Domain Decomposition framework [30] to avoid undesirable stress concentrations at heterogeneous non-conforming interfaces. Such a splitting induces a non-linearity which would indeed have an impact within a small deformation setting but for the large strain formulation presented in this manuscript the price of the splitting is considered low compared to the benefits of increasing the applicability of the methodology.

2.2 Strong and weak forms of the problem

The strong form of the equilibrium problem at domain $\Omega^{(s)}$ can be written as:

$$\text{FIND:} \begin{cases} \mathbf{u}^{(s)}(\mathbf{x}_n^{(s)}) : \quad \Omega_n^{(s)} \rightarrow \mathbb{R}^2, \\ \boldsymbol{\lambda}_{\mathrm{I}}(\mathbf{x}_n) = \lambda_{\mathrm{N}}(\mathbf{x}_n)\mathbf{n}^{(p)} + \lambda_{\mathrm{T}}(\mathbf{x}_n)\mathbf{t}^{(p)} : \quad D_n \rightarrow \mathbb{R}^2, \end{cases} \tag{22}$$

FULFILLING:

$$\text{Equilibrium equation:} \quad \bar{\boldsymbol{\nabla}} \cdot \mathbf{P}^{(s)} = \mathbf{0}, \quad \text{in } \Omega_n^{(s)} \tag{23}$$

$$\text{Constitutive model:} \quad \mathbf{P}^{(s)} = \boldsymbol{\Sigma}^{(s)}(\mathbf{u}^{(s)}), \quad \text{in } \Omega_n^{(s)} \tag{24}$$

$$\text{Dirichlet's boundary conditions:} \quad \mathbf{u}^{(s)} = \hat{\mathbf{u}}^{(s)}, \quad \text{in } \Gamma_u^{(s)} \tag{25}$$

$$\text{Neumann's boundary conditions:} \quad \mathbf{t}^{(s)} = \hat{\mathbf{t}}^{(s)}, \quad \text{in } \Gamma_\sigma^{(s)} \tag{26}$$

$$\text{Lagrange multiplier identification:} \quad \boldsymbol{\lambda}_{\mathrm{I}} = \mathbf{t}^{(s)}, \quad \text{in } \Gamma_{\mathrm{D}}^{(s)} \tag{27}$$

$$\text{Compatibility constraints:} \quad \begin{cases} \bar{g}_{\mathrm{N}}(\mathbf{u}^{\mathrm{D}}) = 0 \\ \bar{g}_{\mathrm{T}}(\mathbf{u}^{\mathrm{D}}) = 0 \end{cases}, \quad \text{in } D_n, \tag{28}$$

$\bar{\boldsymbol{\nabla}}$ being the material gradient with respect to the previous configuration n, $\mathbf{P}^{(s)}$ the first Piola–Kirchoff corresponding to the previous configuration $\Omega_n^{(s)}$, $\hat{\mathbf{t}}^{(s)}$ and $\hat{\mathbf{u}}^{(s)}$ the prescribed tractions and displacements at $\Gamma_\sigma^{(s)}$ and $\Gamma_u^{(s)}$, respectively.

Remark 2.6 Note that the compatibility constraints are enforced to nullify the normal $\bar{g}_{\mathrm{N}}(\mathbf{u}^{\mathrm{D}})$ and tangential $\bar{g}_{\mathrm{T}}(\mathbf{u}^{\mathrm{D}})$ components of the effective gap satisfying, in this manner, displacement compatibility across domains as indicated in (28). The gaps are defined by means of the incremental displacements at the interface $\mathbf{u}^{\mathrm{D}}$ which are calculated by interpolating the displacements at the vertices according to (8). In addition, the displacements at the vertices of the interface patch are calculated with the domain displacements $\mathbf{u}^{(s)}(\mathbf{x}_n)$.

The weak form of the problem stated in (23–28) is expressed through the virtual work principle and variational statement of the constraint equations. To this end, the solution $\boldsymbol{\mathcal{V}}$ and weighting spaces $\boldsymbol{\mathcal{V}}_0$ for the displacement field as well as the Lagrange multiplier space $\boldsymbol{\mathcal{L}}$ for the corresponding solution and weighting functions are defined as:

$$\boldsymbol{\mathcal{V}} := \left\{ \mathbf{u} \big/ \mathbf{u}\big|_{\Omega^{(s)}} \in \mathcal{H}^1(\Omega^{(s)}), \quad \mathbf{u}^{(s)} = \hat{\mathbf{u}}^{(s)} \quad \text{in} \quad \Gamma_u^{(s)}, \quad s: 1 \ldots N_{\mathrm{s}} \right\}, \tag{29}$$

$$\boldsymbol{\mathcal{V}}_0 := \left\{ \delta\mathbf{u} \big/ \delta\mathbf{u}\big|_{\Omega^{(s)}} \in \mathcal{H}^1(\Omega^{(s)}), \quad \delta\mathbf{u}^{(s)} = \mathbf{0} \quad \text{in} \quad \Gamma_u^{(s)}, \quad s: 1 \ldots N_{\mathrm{s}} \right\}, \tag{30}$$

$$\mathcal{L} := L^2(\mathrm{D}), \tag{31}$$

$\mathcal{H}^1(\Omega^{(s)})$ and $L^2(\mathrm{D})$ being the Sovolev space of functions with square integrable derivatives and the Lebesgue space of square integrable functions, respectively. The variational statement reads:

$$\text{FIND:} \begin{cases} \mathbf{u}^{(s)} \in \boldsymbol{\mathcal{V}} : \quad \Omega_n^{(s)} \rightarrow \mathbb{R}^2 \\ \boldsymbol{\lambda}_{\mathrm{I}} \in \boldsymbol{\mathcal{L}} : \quad D_n \rightarrow \mathbb{R}^2 \end{cases} \tag{32}$$

FULFILLING:

$$\delta\Pi_{\mathrm{mec}}(\mathbf{u}, \boldsymbol{\lambda}_{\mathrm{I}}, \delta\mathbf{u}) := \delta\Pi_{\mathrm{int,ext}}(\mathbf{u}, \delta\mathbf{u}) + \delta\Pi_{\mathrm{I}}(\mathbf{u}, \boldsymbol{\lambda}_{\mathrm{I}}, \delta\mathbf{u}) = 0, \quad \forall \delta\mathbf{u}^{(s)} \in \boldsymbol{\mathcal{V}}_0. \tag{33}$$

AND

$$\delta\Pi_{\lambda_{\mathrm{N}}}(\mathbf{u}^{\mathrm{D}}, \delta\lambda_{\mathrm{N}}) = 0, \quad \forall \delta\lambda_{\mathrm{N}} \in \mathcal{L}, \tag{34}$$

$$\delta\Pi_{\lambda_{\mathrm{T}}}(\mathbf{u}^{\mathrm{D}}, \delta\lambda_{\mathrm{T}}) = 0, \quad \forall \delta\lambda_{\mathrm{T}} \in \mathcal{L}. \tag{35}$$

The part of the mechanical work corresponding to internal and external forces

$$\delta\Pi_{\mathrm{int,ext}}(\mathbf{u}, \delta\mathbf{u}) = \delta\Pi_{\mathrm{int}}(\mathbf{u}, \delta\mathbf{u}) - \delta\Pi_{\mathrm{ext}}(\delta\mathbf{u}) \tag{36}$$

with

$$\delta\Pi_{\mathrm{int}}(\mathbf{u}, \delta\mathbf{u}) = \sum_{s=1}^{N_s} \left\{ \int_{\Omega_n^{(s)}} \mathbf{P}^{(s)} : \bar{\boldsymbol{\nabla}}(\delta\mathbf{u}^{(s)}) \, d\Omega \right\} \tag{37}$$

and

$$\delta\Pi_{\text{ext}}(\delta\mathbf{u}) = \sum_{s=1}^{N_s}\left\{\int_{\Gamma_\sigma^{(s)}} \hat{\mathbf{t}}^{(s)} \cdot \delta\mathbf{u}^{(s)}\, d\Gamma\right\}. \tag{38}$$

The work performed at the interface

$$\begin{aligned}\delta\Pi_{\text{I}}(\mathbf{u}, \boldsymbol{\lambda}_{\text{I}}) &= \int_{D_n} \boldsymbol{\lambda}_{\text{I}} \cdot \delta\bar{\mathbf{g}}(\mathbf{u}^{\text{D}})\, dD \\ &= \int_{D_n} \lambda_{\text{N}}\delta\bar{g}_{\text{N}}(\mathbf{u}^{\text{D}})\, dD + \int_{D_n} \lambda_{\text{T}}\delta\bar{g}_{\text{T}}(\mathbf{u}^{\text{D}})\, dD,\end{aligned} \tag{39}$$

where $\delta\bar{\mathbf{g}}$ denotes the gap intensity variations. It is worth noting that the resulting variational principles are considered on the interface volume (surface in 2D) D_n and not at the interface surface (segments in 2D) as it is done in other established methodologies. Following the expressions of the gap intensity variations developed in [21,38] the interface normal and tangential work contributions can be written in terms of the displacement variations as:

$$\int_{D_n} \lambda_{\text{N}}\delta\bar{g}_{\text{N}}(\mathbf{u}^{\text{D}})\, dD = \int_{D_n} \lambda_{\text{N}}\bar{g}_{\text{N}}\mathbf{n} \cdot \boldsymbol{\nabla}(\delta\mathbf{u}^{\text{D}}) \cdot \mathbf{n}\, dD, \tag{40}$$

$$\begin{aligned}\int_{D_n} \lambda_{\text{T}}\delta\bar{g}_{\text{T}}(\mathbf{u}^{\text{D}})\, dD &= \int_{D_n} \lambda_{\text{T}}\bar{g}_{\text{N}}\mathbf{n} \cdot \boldsymbol{\nabla}(\delta\mathbf{u}^{\text{D}}) \cdot \mathbf{t}\, dD \\ &\quad + \int_{D_n} \lambda_{\text{T}}\bar{g}_{\text{N}}\mathbf{t} \cdot \boldsymbol{\nabla}(\delta\mathbf{u}^{\text{D}}) \cdot \mathbf{n}\, dD \\ &\quad + \int_{D_n} \lambda_{\text{T}}\bar{g}_{\text{T}}\mathbf{t} \cdot \boldsymbol{\nabla}(\delta\mathbf{u}^{\text{D}}) \cdot \mathbf{t}\, dD,\end{aligned} \tag{41}$$

where $\boldsymbol{\nabla} = \dfrac{\partial(\bullet)}{\partial\mathbf{x}_{n+1}}$ denotes the spatial gradient, i.e. taking derivatives with respect to the current configuration $n+1$.

Finally, the variational statements of the constraint equations

$$\begin{aligned}\delta\Pi_{\lambda_{\text{N}}}(\mathbf{u}^{\text{D}}, \delta\lambda_{\text{N}}) &= \int_{D_n} \delta\lambda_{\text{N}}\bar{g}_{\text{N}}(\mathbf{u}^{\text{D}})\, dD \\ &= \int_{D_n} \delta\lambda_{\text{N}}\, sign\left(g_{\text{N}}^0\right) \mathbf{n} \cdot \left(\mathbf{N} + \bar{\boldsymbol{\nabla}}(\mathbf{u}^{\text{D}}) \cdot \mathbf{N}\right)\, dD,\end{aligned} \tag{42}$$

$$\begin{aligned}\delta\Pi_{\lambda_{\text{T}}}(\mathbf{u}^{\text{D}}, \delta\lambda_{\text{T}}) &= \int_{D_n} \delta\lambda_{\text{T}}\bar{g}_{\text{T}}(\mathbf{u}^{\text{D}})\, dD \\ &= \int_{D_n} \delta\lambda_{\text{T}}\, sign\left(g_{\text{N}}^0\right) \mathbf{t} \cdot \left(\mathbf{N} + \bar{\boldsymbol{\nabla}}(\mathbf{u}^{\text{D}}) \cdot \mathbf{N}\right)\, dD,\end{aligned} \tag{43}$$

which force the interface work to nullify in an average sense along the domain interface D_n.

Remark 2.7 The equilibrium equation in (23) and imposed tractions at the boundary (26) correspond to the Euler–Lagrange equations and natural boundary conditions associated to the virtual work principle in (33). In the same spirit, the constraint equations in (28) correspond to the Euler–Lagrange equations associated to the constraint variational Eqs. (34, 35).

2.3 Discretization using FE and lambda-solvability of the resulting system

Consider a Galerkin-based discretization in which the displacement solution field and its variations are interpolated using the shape functions $\mathbb{N}$ as:

$$\mathbf{u}^{(s)}(\mathbf{x}_n) = \sum_a \mathbb{N}_a(\mathbf{x}_n)\mathbf{d}_a^{(s)} \quad \forall\mathbf{x}_n \in \Omega_n^{(s)}, \tag{44}$$

$$\delta\mathbf{u}^{(s)}(\mathbf{x}_n) = \sum_a \mathbb{N}_a(\mathbf{x}_n)\delta\mathbf{d}_a^{(s)} \quad \forall\mathbf{x}_n \in \Omega_n^{(s)}, \tag{45}$$

where the subscript a denotes the discrete nodes corresponding to the displacement interpolation. In a similar fashion, the displacements $\mathbf{u}^{\text{D}}$, Lagrange multipliers $\boldsymbol{\lambda}_{\text{I}}$ and its corresponding variations at the interface patch D are discretized as:

$$\mathbf{u}^{\text{D}}(\mathbf{x}_n) = \sum_a \mathbb{N}_a(\mathbf{x}_n)\mathbf{d}_a^{\text{D}} \quad \forall\mathbf{x}_n \in D_n, \tag{46}$$

$$\delta\mathbf{u}^{\text{D}}(\mathbf{x}_n) = \sum_a \mathbb{N}_a(\mathbf{x}_n)\delta\mathbf{d}_a^{\text{D}} \quad \forall\mathbf{x}_n \in D_n, \tag{47}$$

$$\boldsymbol{\lambda}_{\text{I}}(\mathbf{x}_n) = \sum_b \Psi_b(\mathbf{x}_n)\boldsymbol{\Lambda}_b \quad \forall\mathbf{x}_n \in D_n, \tag{48}$$

$$\delta\boldsymbol{\lambda}_{\text{I}}(\mathbf{x}_n) = \sum_b \Psi_b(\mathbf{x}_n)\delta\boldsymbol{\Lambda}_b \quad \forall\mathbf{x}_n \in D_n, \tag{49}$$

where the subscript b stands for the the discrete nodes corresponding to the Lagrange multipliers interpolation using the shape functions Ψ which read:

$$\Psi(\mathbf{x}_n) = \begin{cases} 1 & \forall\mathbf{x}_n \in D_n^{(p)} \\ 0 & \forall\mathbf{x}_n \notin D_n^{(p)} \end{cases}. \tag{50}$$

Remark 2.8 It should be noted that a piece-wise constant interpolation of the Lagrange multipliers might not lead to optimal spatial convergence rates. We have not observed any critical convergence behaviour in any of our simulations. However, it is observed in our analyses that the theoretical convergence rates might not be fully recovered due to use of piece-wise constant Lagrange multipliers. A more theoretical and practical study regarding convergence rates with respect to the choice of the Lagrange multiplier space is out of the scope of this work and could be considered as a future research topic for a more mathematically oriented contribution.

The FE approximation of the virtual work expression in (33) can be written using the above interpolations (44) to (49) as follows:

$$\delta\Pi_{\text{mec}}(\mathbf{u}, \boldsymbol{\lambda}_{\text{I}}, \delta\mathbf{u}) \approx \delta\Pi^{\text{h}}_{\text{mec}}(\mathbf{d}, \boldsymbol{\Lambda}, \delta\mathbf{d}) = \delta\Pi^{\text{h}}_{\text{int,ext}}(\mathbf{d}, \delta\mathbf{d}) + \delta\Pi^{\text{h}}_{\text{I}}(\mathbf{d}, \boldsymbol{\Lambda}, \delta\mathbf{d}). \tag{51}$$

Considering that (51) holds for any virtual displacement $\delta\mathbf{u}$, the residual forces of the variational principle

$$\mathbf{R}_{\text{mech}}(\mathbf{d}, \boldsymbol{\Lambda}) = \mathbf{R}_{\text{int,ext}}(\mathbf{d}) + \mathbf{R}_{\text{I}}(\mathbf{d}, \boldsymbol{\Lambda}), \tag{52}$$

with

$$\begin{aligned}\mathbf{R}_{\text{int,ext}}(\mathbf{d}) &= \sum_{s=1}^{N_{\text{s}}} \left(\mathbf{F}^{(s)}_{\text{int}}(\mathbf{d}^{(s)}) - \mathbf{F}^{(s)}_{\text{ext}}\right) \\ &= \sum_{s=1}^{N_{\text{s}}} \Bigg(\int_{\Omega^{(s)}} \sum_a \left(\mathbf{P}^{(s)}(\mathbf{d}^{(s)}) \cdot \bar{\boldsymbol{\nabla}}(\mathbb{N}_a)\right) \, d\Omega \\ &\quad - \int_{\Omega^{(s)}} \sum_a \left(\mathbb{N}_a \hat{\mathbf{t}}^{(s)}_a\right) \, d\Omega \Bigg),\end{aligned} \tag{53}$$

$$\begin{aligned}\mathbf{R}_{\text{I}}(\mathbf{d}, \boldsymbol{\Lambda}) = &\int_{D_n} \sum_b \left(\Psi_b \Lambda_{\text{N},b}\right) \bar{g}_{\text{N}} \mathbf{n} \left(\sum_a \boldsymbol{\nabla}(\mathbb{N}_a) \right) \cdot \mathbf{n} \, dD \\ &+ \int_{D_n} \sum_b \left(\Psi_b \Lambda_{\text{T},b}\right) \bar{g}_{\text{N}} \mathbf{n} \left(\sum_a \boldsymbol{\nabla}(\mathbb{N}_a) \right) \cdot \mathbf{t} \, dD \\ &+ \int_{D_n} \sum_b \left(\Psi_b \Lambda_{\text{T},b}\right) \bar{g}_{\text{N}} \mathbf{t} \left(\sum_a \boldsymbol{\nabla}(\mathbb{N}_a) \right) \cdot \mathbf{n} \, dD \\ &+ \int_{D_n} \sum_b \left(\Psi_b \Lambda_{\text{T},b}\right) \bar{g}_{\text{T}} \mathbf{t} \left(\sum_a \boldsymbol{\nabla}(\mathbb{N}_a) \right) \cdot \mathbf{t} \, dD,\end{aligned} \tag{54}$$

where it is assumed that $\bar{\mathbf{g}} = \bar{\mathbf{g}}(\mathbf{d}^{\text{D}})$, $\mathbf{n} = \mathbf{n}(\mathbf{d}^{\text{D}})$ and $\mathbf{t} = \mathbf{t}(\mathbf{d}^{\text{D}})$.

Following an analogous procedure for the variational expression of the interface constraints (34, 35)

$$\delta\Pi_{\lambda_{\text{N}}}(\mathbf{u}^{\text{D}}, \delta\lambda_{\text{N}}) \approx \delta\Pi^{\text{h}}_{\lambda\text{N}}(\mathbf{d}^{\text{D}}, \delta\Lambda_{\text{N}}). \tag{55}$$

$$\delta\Pi_{\lambda_{\text{T}}}(\mathbf{u}^{\text{D}}, \delta\lambda_{\text{T}}) \approx \delta\Pi^{\text{h}}_{\lambda\text{T}}(\mathbf{d}^{\text{D}}, \delta\Lambda_{\text{T}}). \tag{56}$$

Considering that (55, 56) hold for any virtual Lagrange multiplier $\delta\lambda_{\text{N}}$ and $\delta\lambda_{\text{T}}$, the residual displacement gap of the constraint variational principle can be written in a matrix form as:

$$\mathbf{R}_{\lambda}(\mathbf{d}^{\text{D}}) = \begin{bmatrix} R_{\lambda_{\text{N}}}(\mathbf{d}^{\text{D}}) \\ R_{\lambda_{\text{T}}}(\mathbf{d}^{\text{D}}) \end{bmatrix}, \tag{57}$$

with

$$R_{\lambda_{\text{N}}}(\mathbf{d}^{\text{D}}) = \int_{D_n} sign(g^0_{\text{N}})\mathbf{n} \cdot \left(\mathbf{N} + \left(\sum_a \bar{\boldsymbol{\nabla}}(\mathbb{N}_a \mathbf{d}^{\text{D}}_a) \cdot \mathbf{N} \right) \right) dD, \tag{58}$$

$$R_{\lambda_{\text{T}}}(\mathbf{d}^{\text{D}}) = \int_{D_n} sign(g^0_{\text{N}})\mathbf{t} \cdot \left(\mathbf{N} + \left(\sum_a \bar{\boldsymbol{\nabla}}(\mathbb{N}_a \mathbf{d}^{\text{D}}_a) \cdot \mathbf{N} \right) \right) dD. \tag{59}$$

The discrete problem to be solved can be specified in terms of the nodal quantities $\mathbf{d}$ and $\boldsymbol{\Lambda}$ as:

FIND **d** AND **Λ** FULFILLING

$$\mathbf{R}_{\text{mech}}(\mathbf{d}, \boldsymbol{\Lambda}) = \mathbf{0}, \tag{60}$$

$$\mathbf{R}_{\lambda}(\mathbf{d}^{\text{D}}) = \mathbf{0}. \tag{61}$$

The set of Eqs. (60, 61) can be solved incrementally via a standard Newton-Raphson procedure on the linearized system

$$\begin{bmatrix} \mathbf{R}_{\text{mech}}(\mathbf{d}, \boldsymbol{\Lambda}) \\ \mathbf{R}_{\lambda}(\mathbf{d}) \end{bmatrix} + \begin{bmatrix} \dfrac{\partial \mathbf{R}_{\text{mech}}(\mathbf{d}, \boldsymbol{\Lambda})}{\partial \mathbf{d}} & \dfrac{\partial \mathbf{R}_{\text{mech}}(\mathbf{d}, \boldsymbol{\Lambda})}{\partial \boldsymbol{\Lambda}} \\ \dfrac{\partial \mathbf{R}_{\lambda}(\mathbf{d})}{\partial \mathbf{d}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{d} \\ \Delta\boldsymbol{\Lambda} \end{bmatrix} = \mathbf{0}, \tag{62}$$

where $\Delta\mathbf{d}$ and $\Delta\boldsymbol{\Lambda}$ denote the solution field increments to be solved at each iteration.

It is important to realize that zero entries appear at the tangent matrix due to the fact that the gap residual $\mathbf{R}_{\lambda}(\mathbf{d})$ depends solely on the nodal displacements $\mathbf{d}$ and not on the Lagrange multipliers $\boldsymbol{\Lambda}$. Consequently, the problem is prone to exhibit instabilities if the adopted solution field discretizations do not satisfy the Ladyzhenskaya–Babuška–Brezzi (LBB) condition [2]. In order to provide a dependency of the Lagrange multipliers $\boldsymbol{\Lambda}$ for the gap residual $\mathbf{R}_{\lambda}(\mathbf{d})$, a Nitsche method [37] is employed based on the work of Heintz and Hansbo [23] who introduced this methodology for the case of linear kinematics. The stabilization procedure utilized in this manuscript has been introduced by Oliver et al. [38] and Hartmann et al. [21] in the context of contact mechanics. It essentially consists in a modification of the constraint variational principles (34, 35) considering a weak format of the identities in (21) leading to

$$\begin{aligned}\delta\tilde{\Pi}_{\lambda_{\text{N}}}(\mathbf{u}^{\text{D}}, \delta\lambda_{\text{N}}) &= \delta\Pi_{\lambda_{\text{N}}}(\mathbf{u}^{\text{D}}, \delta\lambda_{\text{N}}) \\ &\quad + \int_{\partial D_n \cap \Gamma^{(s)}_{\text{D}}} \tau \delta\lambda_{\text{N}} \left(t_{\text{N}}(\mathbf{u}^{(s)}) - \lambda_{\text{N}} \right) \, d\Gamma = 0, \\ &\quad \forall \delta\lambda_{\text{N}} \in \mathcal{L}_{\text{N}},\end{aligned} \tag{63}$$

$$\begin{aligned}\delta\tilde{\Pi}_{\lambda_{\text{T}}}(\mathbf{u}^{\text{D}}, \delta\lambda_{\text{T}}) &= \delta\Pi_{\lambda_{\text{T}}}(\mathbf{u}^{\text{D}}, \delta\lambda_{\text{T}}) \\ &\quad + \int_{\partial D_n \cap \Gamma^{(s)}_{\text{D}}} \tau \delta\lambda_{\text{T}} \left(t_{\text{T}}(\mathbf{u}^{(s)}) - \lambda_{\text{T}} \right) \, d\Gamma = 0, \\ &\quad \forall \delta\lambda_{\text{T}} \in \mathcal{L}_{\text{T}},\end{aligned} \tag{64}$$

with $\tau > 0$ being an additional parameter that penalizes the violation of the identity expressed by the new term. In addition

$$\tau = \frac{\alpha_{\text{stab}}}{E_{\min}} L, \tag{65}$$

where $E_{\min}$ is the minimal Young's modulus of the adjacent domains, L stands for the base-side length of the interface domain element in the previous configuration (cf. Fig. 4) and α_{stab} corresponds to a dimensionless user defined parameter which is regarded independent of the mesh size [21]. Note that the units of the stabilization parameter τ are $[L]^3/[F]$ such that the additional variational term corresponds to an energetic contribution with units $[F][L]$. It should be noted that, since the penalized term is part of the Euler–Lagrange equations of the variational principle (27), it will tend to zero upon mesh refinement. For this reason the stabilization procedure described in (63, 64) is qualified as a consistent penalty method in which, unlike other non-consistent penalty methods, the parameter τ can be made significantly small without affecting the quality of the obtained results.

The new residual displacement gap of the modified variational constraints (63, 64)

$$\tilde{\mathbf{R}}_\lambda(\mathbf{d}^{\text{D}}, \boldsymbol{\Lambda}) = \begin{bmatrix} \tilde{R}_{\lambda_{\text{N}}}(\mathbf{d}^{\text{D}}, \Lambda_{\text{N}}) \\ \tilde{R}_{\lambda_{\text{T}}}(\mathbf{d}^{\text{D}}, \Lambda_{\text{T}}) \end{bmatrix}, \tag{66}$$

with

$$\begin{aligned} \tilde{R}_{\lambda_{\text{N}}}(\mathbf{d}^{\text{D}}, \Lambda_{\text{N}}) = {} & \int_{D_n} sign(g_{\text{N}}^0)\mathbf{n} \cdot \left(\mathbf{N} + \left(\sum_a \bar{\boldsymbol{\nabla}}(\mathbb{N}_a \mathbf{d}_a^{\text{D}}) \cdot \mathbf{N}\right)\right) dD \\ & + \int_{\Gamma_n} \tau \left(\mathbf{n} \cdot \mathbf{P}^{(s)} \cdot \mathbf{N} - \sum_b \left(\Psi_b \Lambda_{\text{N},b}\right)\right) d\Gamma \end{aligned} \tag{67}$$

$$\begin{aligned} \tilde{R}_{\lambda_{\text{T}}}(\mathbf{d}^{\text{D}}, \Lambda_{\text{T}}) = {} & \int_{D_n} sign(g_{\text{N}}^0)\mathbf{t} \cdot \left(\mathbf{N} + \left(\sum_a \bar{\boldsymbol{\nabla}}(\mathbb{N}_a \mathbf{d}_a^{\text{D}}) \cdot \mathbf{N}\right)\right) dD \\ & + \int_{\Gamma_n} \tau \left(\mathbf{t} \cdot \mathbf{P}^{(s)} \cdot \mathbf{N} - \sum_b \left(\Psi_b \Lambda_{\text{T},b}\right)\right) d\Gamma. \end{aligned} \tag{68}$$

The first Piola–Kirchoff stress tensor $\mathbf{P}^{(s)}$ in (67, 68) belongs to domain $\Omega^{(s)}$ and corresponds to the element adjacent to the base-line of the interface element $D^{(p)}$ (cf. Fig. 5). Taking into account the above expressions for the modified constraint residual, the system in (62) results in

$$\begin{bmatrix} \mathbf{R}_{\text{mech}}(\mathbf{d}, \boldsymbol{\Lambda}) \\ \tilde{\mathbf{R}}_\lambda(\mathbf{d}, \boldsymbol{\Lambda}) \end{bmatrix} + \begin{bmatrix} \dfrac{\partial \mathbf{R}_{\text{mech}}(\mathbf{d}, \boldsymbol{\Lambda})}{\partial \mathbf{d}} & \dfrac{\partial \mathbf{R}_{\text{mech}}(\mathbf{d}, \boldsymbol{\Lambda})}{\partial \boldsymbol{\Lambda}} \\ \dfrac{\partial \tilde{\mathbf{R}}_\lambda(\mathbf{d}, \boldsymbol{\Lambda})}{\partial \mathbf{d}} & \dfrac{\partial \tilde{\mathbf{R}}_\lambda(\mathbf{d}, \boldsymbol{\Lambda})}{\partial \boldsymbol{\Lambda}} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{d} \\ \Delta\boldsymbol{\Lambda} \end{bmatrix} = \mathbf{0}. \tag{69}$$

Remark 2.9 The system in (69) is non-symmetric due to the fact that the stabilization term is only introduced in the constraint equations. The consistent symmetric version proposed by Heintz and Hansbo [23] could be utilized too and it would be recommended in those cases where the adopted FE model leads to a symmetric tangent stiffness matrix. In the context of a full parallel scheme, the symmetry of the system in (69) allows the use of efficient iterative solvers such as the preconditioned conjugate gradient [3].

2.4 Parallel system resolution strategies

The dual assembly in (69) can be recast considering the discretized quantities for each domain $\Omega(s)$ in a matrix form as

$$\begin{bmatrix} \mathbf{K}_{\text{dd}}^{(1)} & \mathbf{0} & \mathbf{0} & \mathbf{K}_{\text{d}\Lambda}^{(1)} \\ \mathbf{0} & \ddots & \mathbf{0} & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{K}_{\text{dd}}^{(N_{\text{s}})} & \mathbf{K}_{\text{d}\Lambda}^{(N_{\text{s}})} \\ \mathbf{K}_{\Lambda\text{d}}^{(1)} & \dots & \mathbf{K}_{\Lambda\text{d}}^{(N_{\text{s}})} & \mathbf{K}_{\Lambda\Lambda} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{d}^{(1)} \\ \vdots \\ \Delta\mathbf{d}^{(N_{\text{s}})} \\ \Delta\boldsymbol{\Lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{r}_{\text{d}}^{(1)} \\ \vdots \\ \mathbf{r}_{\text{d}}^{(N_{\text{s}})} \\ \mathbf{r}_\Lambda \end{bmatrix}, \tag{70}$$

where

$$\begin{bmatrix} \mathbf{K}_{\text{dd}}^{(s)} & \mathbf{K}_{\text{d}\Lambda}^{(s)} \\ \mathbf{K}_{\Lambda\text{d}}^{(s)} & \mathbf{K}_{\Lambda\Lambda} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial \left(\mathbf{R}_{\text{mech}}(\mathbf{d}, \boldsymbol{\Lambda})\right)^{(s)}}{\partial \mathbf{d}^{(s)}} & \dfrac{\partial \left(\mathbf{R}_{\text{mech}}(\mathbf{d}, \boldsymbol{\Lambda})\right)^{(s)}}{\partial \boldsymbol{\Lambda}^{(s)}} \\ \dfrac{\partial \left(\tilde{\mathbf{R}}_\lambda(\mathbf{d}, \boldsymbol{\Lambda})\right)^{(s)}}{\partial \mathbf{d}^{(s)}} & \dfrac{\partial \tilde{\mathbf{R}}_\lambda(\mathbf{d}, \boldsymbol{\Lambda})}{\partial \boldsymbol{\Lambda}} \end{bmatrix}, \tag{71}$$

$$\begin{bmatrix} \mathbf{r}_{\text{d}}^{(s)} \\ \mathbf{r}_\Lambda \end{bmatrix} = \begin{bmatrix} \left(\mathbf{R}_{\text{mech}}(\mathbf{d}, \boldsymbol{\Lambda})\right)^{(s)} \\ \tilde{\mathbf{R}}_\lambda(\mathbf{d}, \boldsymbol{\Lambda}) \end{bmatrix} \tag{72}$$

and

$$\mathbf{d}^{(s)} = \mathbf{d}(\mathbf{x}_n)|\mathbf{x}_n \in \Omega^{(s)}, \tag{73}$$

$$\boldsymbol{\Lambda}^{(s)} = \boldsymbol{\Lambda}(\mathbf{x}_n)|\mathbf{x}_n \in \Gamma_{\text{I}}^{(s)} = \Gamma_{\text{I}} \cap \Omega^{(s)}, \tag{74}$$

$$\left(\mathbf{R}_{\text{mech}}(\mathbf{d}, \boldsymbol{\Lambda})\right)^{(s)} = \mathbf{R}_{\text{mech}}(\mathbf{d}, \boldsymbol{\Lambda}), \quad \forall \mathbf{d} = \mathbf{d}^{(s)} \quad \text{and} \quad \forall \boldsymbol{\Lambda} = \boldsymbol{\Lambda}^{(s)}, \tag{75}$$

$$\left(\tilde{\mathbf{R}}_\lambda(\mathbf{d}, \boldsymbol{\Lambda})\right)^{(s)} = \tilde{\mathbf{R}}_\lambda(\mathbf{d}, \boldsymbol{\Lambda}), \quad \forall \mathbf{d} = \mathbf{d}^{(s)} \quad \text{and} \quad \forall \boldsymbol{\Lambda} = \boldsymbol{\Lambda}^{(s)}. \tag{76}$$

The system in (70) is expected to be large and, therefore, suitable to be solved using a parallel scheme. Given a reasonable amount of memory, the sparse system in (70) can be tackled with a direct parallel solver using a moderate number of processors. These algorithms are referred to as multi-frontal or block-LU methods [46] and are based on independent simultaneous factorizations of the domain matrices. Although these techniques do not scale well in massively parallel computers, they provide the same robustness as traditional direct methods which make them attractive when dealing with general non-symmetric and ill-conditioned systems. Additionally, block-LU solvers account for automatic load-balancing and multi-threading which is specially effective when dealing with domains with a significant difference in terms of the number of DOF [9,32].

Another option is to express the global system (70) in terms of the interface flexibility problem

$$\mathbf{F}_{\mathrm{I}}\Delta\boldsymbol{\Lambda} = \Delta\mathbf{g}_{\mathrm{I}}, \tag{77}$$

$$\mathbf{F}_{\mathrm{I}} = \mathbf{K}_{\Lambda\Lambda} - \sum_{s=1}^{N_{\mathrm{s}}} \left(\mathbf{K}_{\Lambda\mathrm{d}}^{(s)} \left(\mathbf{K}_{\mathrm{dd}}\right)^{-1} \mathbf{K}_{\mathrm{d}\Lambda}^{(s)}\right), \tag{78}$$

$$\Delta\mathbf{g}_{\mathrm{I}} = \mathbf{r}_{\Lambda} - \sum_{s=1}^{N_{\mathrm{s}}} \left(\mathbf{K}_{\Lambda\mathrm{d}}^{(s)} \left(\mathbf{K}_{\mathrm{dd}}\right)^{-1} \mathbf{r}_{\mathrm{d}}^{(s)}\right), \tag{79}$$

assuming that the matrices $\mathbf{K}_{\mathrm{dd}}^{(s)}$ are not singular, i.e. they do not exhibit rigid body modes (RBMs) as observed in floating domains (cf. following Sect. 2.5 for a detailed explanation). The flexibility of the interface and the interface displacement gap increment are denoted by $\mathbf{F}_{\mathrm{I}}$ and $\Delta\mathbf{g}_{\mathrm{I}}$, respectively and can be interpreted as the condensation of the domain stiffness matrices and residual forces at the interface Γ_{I}. The domain displacement increments $\Delta\mathbf{d}^{(s)}$ can be independently calculated for each domain after the solution of the Lagrange multipliers $\Delta\boldsymbol{\Lambda}$ as

$$\Delta\mathbf{d}^{(s)} = \left(\mathbf{K}_{\mathrm{dd}}^{(s)}\right)^{-1} \left(\mathbf{r}_{\mathrm{d}}^{(s)} - \mathbf{K}_{\mathrm{d}\Lambda}^{(s)}\Delta\boldsymbol{\Lambda}\right). \tag{80}$$

In this second option, a blend of direct solvers are employed to independently compute the factorizations of the domain stiffness and an iterative solver is utilized for the solution of the interface problem in (77) as commonly done in most substructuring and domain decomposition techniques [10, 11]. Consequently, the domain stiffness factorizations, the resolution of the interface problem and the computation of the domain solution fields are inherently parallel tasks and, for this reason, the methodology scales well in massively parallel computers. Moreover, since the interface problem does not have to be explicitly assembled, the required memory profile for such a parallel resolution is significantly lower than for the parallel direct solution of the global system (70).

If the flexibility matrix $\mathbf{F}_{\mathrm{I}}$ is symmetric, the interface problem in (77) can be solved by preconditioned Conjugate Gradient iterations, otherwise a Bi-Conjugate Gradient Stabilized (Bi-CGSTAB) or a Generalized Minimal Residual method (GMRES) can be employed [3]. In our case, the non-symmetry of the flexibility problem (77) can be caused by the stabilization procedure outlined in (63, 64), which only affects the constraint variational principle, or in those cases where the constitutive equations render a non-symmetric stiffness matrix. For the case of ill-conditioned systems, e.g. domains with high stiffness contrasts due to heterogeneous components or undergoing damage growth and coalescence, robust and efficient preconditioners are generally hard to find (cf. [29,47,49]).

The main goal of this contribution is not the parallel assessment of the proposed domain decomposition technique but rather the introduction of the novel concepts for handling non-conforming meshes and its performance in general assembly situations. However, it is highlighted that the algorithm is perfectly compatible with a full parallel scheme as the ones explained above. For clarity, in all the examples presented in Sect. 3 the flexibility problem in (77) was explicitly assembled and solved through standard direct solvers using an LU factorization.

2.5 A non-intrusive strategy to handle rigid body modes in the DIM

If one or more subdomains $\Omega^{(s)}$ exhibit rigid body modes (RBMs), the corresponding matrices $\mathbf{K}_{\mathrm{dd}}^{(s)}$ are not invertible and the expressions of the interface problem in (77) to (79) need to be generalized in order to handle singular matrices $\mathbf{K}_{\mathrm{dd}}^{(s)}$. In standard non-overlapping dual domain decomposition methods [11,45], the expression for the nodal displacements at domain $\Omega^{(s)}$ reads

$$\Delta\mathbf{d}^{(s)} = \left(\mathbf{K}_{\mathrm{dd}}^{(s)}\right)^{+} \left(\mathbf{r}_{\mathrm{d}}^{(s)} - \mathbf{K}_{\mathrm{d}\Lambda}^{(s)}\Delta\Lambda\right) - \mathbf{R}^{(s)}\boldsymbol{\alpha}^{(s)}, \tag{81}$$

$\mathbf{K}_{\mathrm{dd}}^{(s)+}$, $\mathbf{R}^{(s)}$ and $\boldsymbol{\alpha}^{(s)}$ being a generalized inverse, rigid body modes (RBMs) and their amplitudes, respectively. The RBMs $\mathbf{R}^{(s)}$ span the null space of $\mathbf{K}_{\mathrm{dd}}^{(s)}$ and represent all displacement configurations that do not contribute to the deformation energy, thus satisfying $\mathbf{K}_{\mathrm{dd}}^{(s)}\Delta\mathbf{d}_{\mathrm{rbm}} = \mathbf{0}$ at the corresponding domain $\Omega^{(s)}$. Note that the generalized inverse $\mathbf{K}_{\mathrm{dd}}^{(s)+}$ coincides with the standard inverse $\mathbf{K}_{\mathrm{dd}}^{(s)-1}$ if $\mathbf{K}_{\mathrm{dd}}^{(s)}$ is non-singular and fulfills

$$\mathbf{K}_{\mathrm{dd}}^{(s)}\mathbf{K}_{\mathrm{dd}}^{(s)+}\mathbf{K}_{\mathrm{dd}}^{(s)} = \mathbf{K}_{\mathrm{dd}}^{(s)}, \tag{82}$$

$$\mathbf{K}_{\mathrm{dd}}^{(s)+}\mathbf{K}_{\mathrm{dd}}^{(s)}\mathbf{K}_{\mathrm{dd}}^{(s)+} = \mathbf{K}_{\mathrm{dd}}^{(s)+}. \tag{83}$$

The computation of the generalized inverse can be performed by zeroing the rows and columns corresponding to a zero pivot as in the temporary links method [16]. The rigid body modes $\mathbf{R}^{(s)}$ can be computed as a by-product of the factorization of $\mathbf{K}_{\mathrm{dd}}^{(s)}$ or by geometrical inspection considering the given Dirichlet boundary conditions.

The mechanical work developed by the domain boundary forces $\mathbf{r}_{\mathrm{d}}^{(s)} - \mathbf{K}_{\mathrm{d}\Lambda}^{(s)}\Delta\boldsymbol{\Lambda}$ must vanish in the directions of the rigid body modes and, therefore the following orthogonality condition must be satisfied

$$\mathbf{R}^{(s)\mathrm{T}}\left(\mathbf{r}_{\mathrm{d}}^{(s)} - \mathbf{K}_{\mathrm{d}\Lambda}^{(s)}\Delta\boldsymbol{\Lambda}\right) = \mathbf{0}. \tag{84}$$

Substituting (81) in the stabilized constraint conditions assembled in (70)

$$\sum_{s=1}^{N_\mathrm{s}} \mathbf{K}_{\Lambda \mathrm{d}}^{(s)} \Delta \mathbf{d}^{(s)} + \mathbf{K}_{\Lambda\Lambda} \Delta \boldsymbol{\Lambda} = \mathbf{r}_\Lambda \tag{85}$$

and taking into account the orthogonality condition (84), the following general expression for the interface problem is obtained:

$$\begin{bmatrix} \mathbf{F}_\mathrm{I} & \mathbf{G}_\mathrm{I} \\ \mathbf{G}_\mathrm{I}{}^\mathrm{T} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{\Lambda} \\ \Delta\boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} \Delta\mathbf{g}_\mathrm{I} \\ \Delta\mathbf{e} \end{bmatrix}, \tag{86}$$

where

$$\mathbf{F}_\mathrm{I} = \sum_{s=1}^{N_\mathrm{s}} \mathbf{K}_{\Lambda\mathrm{d}}^{(s)} \mathbf{K}_{\mathrm{dd}}^{(s)^+} \mathbf{K}_{\mathrm{d}\Lambda}^{(s)^\mathrm{T}}, \tag{87}$$

$$\Delta\mathbf{g}_\mathrm{I} = \sum_{s=1}^{N_\mathrm{s}} \mathbf{K}_{\Lambda\mathrm{d}}^{(s)} \mathbf{K}_{\mathrm{dd}}^{(s)^+} \mathbf{r}_\mathrm{d}^{(s)} + \mathbf{K}_{\Lambda\Lambda} \Delta\boldsymbol{\Lambda} - \mathbf{r}_\Lambda, \tag{88}$$

$$\mathbf{G}_\mathrm{I} = \left[\mathbf{K}_{\Lambda\mathrm{d}}^{(1)} \mathbf{R}^{(1)} \ \ldots\ \mathbf{K}_{\Lambda\mathrm{d}}^{(N_\mathrm{s})} \mathbf{R}^{(N_\mathrm{s})} \right], \tag{89}$$

$$\boldsymbol{\alpha} = \left[\boldsymbol{\alpha}^{(1)^\mathrm{T}} \ldots \boldsymbol{\alpha}^{(N_\mathrm{s})^\mathrm{T}} \right]^\mathrm{T}, \quad \text{and} \tag{90}$$

$$\Delta\mathbf{e} = \left[\mathbf{r}_\mathrm{d}^{(1)^\mathrm{T}} \mathbf{R}^{(1)} \ \ldots\ \mathbf{r}_\mathrm{d}\mathbf{f}^{(N_\mathrm{s})^\mathrm{T}} \mathbf{R}^{(N_\mathrm{s})} \right]^\mathrm{T}. \tag{91}$$

In the augmented interface system (86) the operator $\mathbf{G}_\mathrm{I}$ is built considering the rigid body modes of each domain restricted onto the interface, i.e. considering only those configurations of the rigid body modes with components at the interface and neglecting all inner degrees of freedom. The increments $\Delta\mathbf{e}$ correspond to the residual at the interface which is out of balance with respect to the rigid body modes.

The local problems in (81) are solved using direct solvers while an iterative solver is employed for the augmented interface problem (86) which is transformed into a semi-definite system of equations on $\Delta\boldsymbol{\Lambda}$, i.e. eliminating the RBMs from the system, by imposing $\mathbf{G}_\mathrm{I}^\mathrm{T} \Delta\boldsymbol{\Lambda} = \mathbf{e}$ through a projection operator (cf. [12,45] for a more detailed explanation).

Although the methodology presented in (81) to (91) is general for the parallel processing of any dual system (70), its implementation in commercial FE packages is regarded highly intrusive. For this reason, a new methodology is proposed in the context of the DIM method which is capable of handling rigid body modes without the need for augmenting the flexibility system (77). The method essentially adds a new term to the virtual mechanical work expression (33) which penalizes a function of the type $\frac{1}{2}\,||\bar{\mathbf{g}}||^2$. The new term is nullified in the solution since it is related with the Euler–Lagrange equations of the constraint variational principle and, therefore motivates the character of interior penalty method of the proposed procedure in the sense that mesh refinement will force the penalized term to tend to zero. The new virtual work expression reads:

$$\begin{aligned} \delta\tilde{\Pi}_\mathrm{mec}(\mathbf{u}, \boldsymbol{\lambda}_\mathrm{I}, \delta\mathbf{u}) &:= \delta\Pi_\mathrm{mec}(\mathbf{u}, \boldsymbol{\lambda}_\mathrm{I}, \delta\mathbf{u}) \\ &\quad + \delta\Pi_\mathrm{RBM}(\mathbf{u}^\mathrm{D}, \delta\mathbf{u}^\mathrm{D}) = 0, \end{aligned} \tag{92}$$

$$\begin{aligned} \delta\Pi_\mathrm{RBM}(\mathbf{u}^\mathrm{D}, \delta\mathbf{u}^\mathrm{D}) &= \sum_{r=1}^{N_\mathrm{r}} c \int_{\Gamma_\mathrm{D}^{(r)}} \delta\bar{\mathbf{g}} \cdot \bar{\mathbf{g}}\, d\Gamma \\ &= \sum_{r=1}^{N_\mathrm{r}} c \int_{\Gamma_\mathrm{D}^{(r)}} \delta\bar{g}_\mathrm{N} \bar{g}_\mathrm{N}\, d\Gamma \\ &\quad + \sum_{r=1}^{N_\mathrm{r}} c \int_{\Gamma_\mathrm{D}^{(r)}} \delta\bar{g}_\mathrm{T} \bar{g}_\mathrm{T}\, d\Gamma, \end{aligned} \tag{93}$$

$\Gamma_\mathrm{D}^{(r)}$ being the interface segments such that $\Gamma_\mathrm{D}^{(r)} = \Omega^{(s)} \cap D^{(r)}$, N_r representing the number of patches utilized to handle the RBMs (cf. Fig. 6) and c denoting the penalty coefficient utilized to enforce the new condition. The additional term (93) can be expressed in terms of the virtual displacements as

$$\sum_{r=1}^{N_\mathrm{r}} c \int_{\Gamma_\mathrm{D}^{(r)}} \delta\bar{g}_\mathrm{N} \bar{g}_\mathrm{N}\, d\Gamma = \sum_{r=1}^{N_\mathrm{r}} c \int_{\Gamma_\mathrm{D}^{(r)}} \left(\bar{g}_\mathrm{N} \mathbf{n} \cdot \boldsymbol{\nabla}(\delta\mathbf{u}^\mathrm{D}) \cdot \mathbf{n} \right) \bar{g}_\mathrm{N}\, d\Gamma, \tag{94}$$

$$\begin{aligned} \sum_{r=1}^{N_\mathrm{r}} c \int_{\Gamma_\mathrm{D}^{(r)}} \delta\bar{g}_\mathrm{T} \bar{g}_\mathrm{T}\, d\Gamma &= \sum_{r=1}^{N_\mathrm{r}} c \int_{\Gamma_\mathrm{D}^{(r)}} \Big(\bar{g}_\mathrm{N} \mathbf{n} \cdot \boldsymbol{\nabla}(\delta\mathbf{u}^\mathrm{D}) \cdot \mathbf{t} \\ &\quad + \bar{g}_\mathrm{N} \mathbf{t} \cdot \boldsymbol{\nabla}(\delta\mathbf{u}^\mathrm{D}) \cdot \mathbf{n} + \bar{g}_\mathrm{T} \mathbf{t} \cdot \boldsymbol{\nabla}(\delta\mathbf{u}^\mathrm{D}) \cdot \mathbf{t} \Big) \bar{g}_\mathrm{T}\, d\Gamma. \end{aligned} \tag{95}$$

Considering that Eqs. (94, 95) hold for any virtual displacement $\delta\mathbf{u}^\mathrm{D}$, the corresponding residual

$$\mathbf{R}_\mathrm{RBM}(\mathbf{d}^\mathrm{D}) = \mathbf{R}_{\mathrm{RBM},N}(\mathbf{d}^\mathrm{D}) + \mathbf{R}_{\mathrm{RBM},T}(\mathbf{d}^\mathrm{D}), \tag{96}$$

$$\mathbf{R}_{\mathrm{RBM},N}(\mathbf{d}^\mathrm{D}) = \sum_{r=1}^{N_\mathrm{r}} c \int_{\Gamma_\mathrm{D}^{(r)}} \left(\bar{g}_\mathrm{N} \mathbf{n} \left(\sum_a \boldsymbol{\nabla}(\mathbb{N}_a) \right) \cdot \mathbf{n} \right) \bar{g}_\mathrm{N}\, d\Gamma, \tag{97}$$

$$\begin{aligned} \mathbf{R}_{\mathrm{RBM},T}(\mathbf{d}^\mathrm{D}) &= \sum_{r=1}^{N_\mathrm{r}} c \int_{\Gamma_\mathrm{D}^{(r)}} \left(\bar{g}_\mathrm{N} \mathbf{n} \left(\sum_a \boldsymbol{\nabla}(\mathbb{N}_a) \right) \cdot \mathbf{t} \right. \\ &\quad \left. + \bar{g}_\mathrm{N} \mathbf{t} \left(\sum_a \boldsymbol{\nabla}(\mathbb{N}_a) \right) \cdot \mathbf{n} + \bar{g}_\mathrm{T} \mathbf{t} \left(\sum_a \boldsymbol{\nabla}(\mathbb{N}_a) \right) \cdot \mathbf{t} \right) \bar{g}_\mathrm{T}\, d\Gamma. \end{aligned} \tag{98}$$

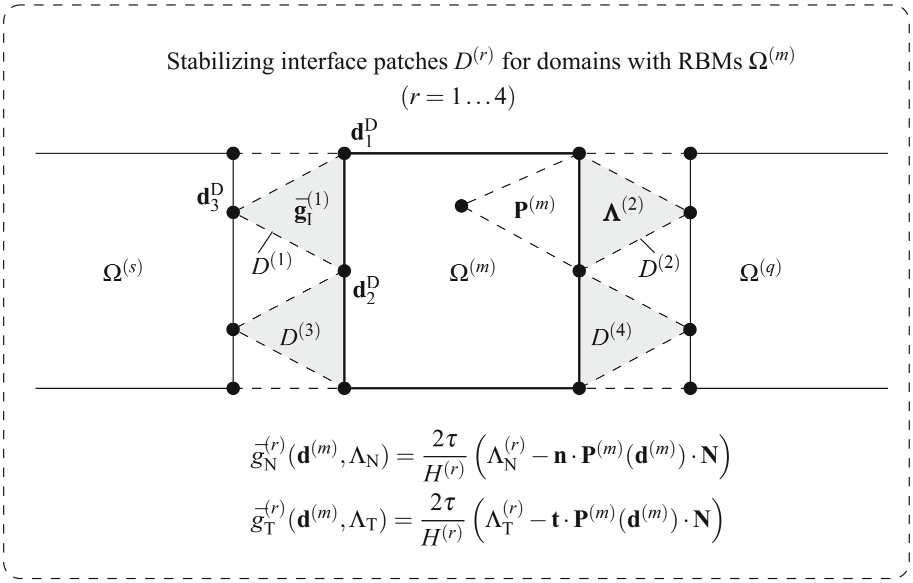
Note that both $\bar{g}_\mathrm{N}$ and $\bar{g}_\mathrm{T}$ are a function of $\mathbf{d}^\mathrm{D}$ and, therefore

$$\frac{\partial \mathbf{R}_\mathrm{RBM}(\mathbf{d}^\mathrm{D})}{\partial \mathbf{d}} = f(\mathbf{d}^\mathrm{D}), \tag{99}$$

$$\frac{\partial \mathbf{R}_\mathrm{RBM}(\mathbf{d}^\mathrm{D})}{\partial \boldsymbol{\Lambda}} = \mathbf{0}. \tag{100}$$

Consequently, there will be contributions to the matrix $\mathbf{K}_\mathrm{dd}^{(s)}$, being $\Omega^{(s)}$ the domain sharing the segment $\Gamma_\mathrm{D}^{(r)}$ with the patch $D^{(r)}$, and also to the adjacent domain $\Omega^{(q)}$ (cf. Fig. 6)

Fig. 6 Handling domains $\Omega^{(m)}$ with RBMs. Stabilizing interface patches $D^{(r)}$, $r = 1 \ldots 4$ correspond to the *shadowed elements*


Stabilizing interface patches $D^{(r)}$ for domains with RBMs $\Omega^{(m)}$
$(r = 1 \ldots 4)$
$\mathbf{d}_1^{\mathrm{D}}$
$\mathbf{d}_3^{\mathrm{D}}$
$\bar{\mathbf{g}}_{\mathrm{I}}^{(1)}$
$\mathbf{P}^{(m)}$
$\mathbf{\Lambda}^{(2)}$
$D^{(1)}$
$\Omega^{(s)}$
$\Omega^{(m)}$
$\mathbf{d}_2^{\mathrm{D}}$
$D^{(2)}$
$\Omega^{(q)}$
$D^{(3)}$
$D^{(4)}$
$\bar{g}_{\mathrm{N}}^{(r)}(\mathbf{d}^{(m)}, \Lambda_{\mathrm{N}}) = \frac{2\tau}{H^{(r)}}\left(\Lambda_{\mathrm{N}}^{(r)} - \mathbf{n} \cdot \mathbf{P}^{(m)}(\mathbf{d}^{(m)}) \cdot \mathbf{N}\right)$
$\bar{g}_{\mathrm{T}}^{(r)}(\mathbf{d}^{(m)}, \Lambda_{\mathrm{T}}) = \frac{2\tau}{H^{(r)}}\left(\Lambda_{\mathrm{T}}^{(r)} - \mathbf{t} \cdot \mathbf{P}^{(m)}(\mathbf{d}^{(m)}) \cdot \mathbf{N}\right)$

which will brake the band structure detailed in (70). In order to avoid the coupling of displacement quantities from adjacent domains it is proposed to express the gap as a function of the interface Lagrange multipliers and surface tractions on the same domain and substitute the expression in the residuals (97, 98).

To this end, we consider the stabilization expressions (63, 64). The use of linear triangular interface patches leads to patch-wise constant normal and tangential vectors $\mathbf{n}^{(r)}$, $\mathbf{t}^{(r)}$ and gap intensities $\bar{g}_{\mathrm{N}}$, $\bar{g}_{\mathrm{T}}$. Therefore the discretized residual Eqs. (67, 68) can be re-written as

$$\int_{D^{(r)}} \bar{g}_{\mathrm{N}}\, dD + \int_{\Gamma_{\mathrm{D}}^{(r)}} \tau \left(t_{\mathrm{N}} - \Lambda_{\mathrm{N}}^{(r)}\right) d\Gamma = 0, \quad p \in \{1, \ldots, N_{\mathrm{p}}\}, \tag{101}$$

$$\int_{D^{(r)}} \bar{g}_{\mathrm{T}}\, dD + \int_{\Gamma_{\mathrm{D}}^{(r)}} \tau \left(t_{\mathrm{T}} - \Lambda_{\mathrm{T}}^{(r)}\right) d\Gamma = 0, \quad p \in \{1, \ldots, N_{\mathrm{p}}\}, \tag{102}$$

where the base-side of the patch is denoted by $\Gamma_{\mathrm{D}}^{(r)}$ (cf. Fig. 4). The volume and surface integrals in (101, 102) can be calculated as

$$\int_{D^{(r)}} dD = \frac{1}{2} L^{(r)} H^{(r)}, \tag{103}$$

$$\int_{\Gamma_{\mathrm{D}}^{(r)}} d\Gamma = L^{(r)}, \tag{104}$$

$L^{(r)}$ being the length of the base-side and $H^{(r)} = \left|g_{\mathrm{N}}^{0}(x_3)\right|$ the absolute value of the initial normal gap corresponding to the interface element vertex 3 (cf. Fig. 4) which is the height of the interface patch $D_n^{(r)}$ in the previous configuration. Considering the integrals (103, 104) and taking into account that t_{N} and t_{T} from the element adjacent to $D^{(r)}$ are constant, the expressions in (101, 102) yield

$$\frac{1}{2} H^{(r)} \bar{g}_{\mathrm{N}} + \tau \left(t_{\mathrm{N}} - \Lambda_{\mathrm{N}}^{(r)}\right) = 0, \quad p \in \{1, \ldots, N_{\mathrm{p}}\}, \tag{105}$$

$$\frac{1}{2} H^{(r)} \bar{g}_{\mathrm{T}} + \tau \left(t_{\mathrm{T}} - \Lambda_{\mathrm{T}}^{(r)}\right) = 0. \quad p \in \{1, \ldots, N_{\mathrm{p}}\}, \tag{106}$$

The effective gap components can be now expressed in terms of the Lagrange multiplier and traction vector components as

$$\bar{g}_{\mathrm{N}}^{(r)} = \frac{g_{\mathrm{N}}(x_3)}{\left|g_{\mathrm{N}}^{0}(x_3)\right|} = \frac{(g_{\mathrm{N}})_3}{H^{(r)}} = \frac{2\tau}{H^{(r)}}\left(\Lambda_{\mathrm{N}}^{(r)} - t_{\mathrm{N}}\right), \tag{107}$$

$$\bar{g}_{\mathrm{T}}^{(r)} = \frac{g_{\mathrm{T}}(x_3)}{\left|g_{\mathrm{T}}^{0}(x_3)\right|} = \frac{(g_{\mathrm{T}})_3}{H^{(r)}} = \frac{2\tau}{H^{(r)}}\left(\Lambda_{\mathrm{T}}^{(r)} - t_{\mathrm{T}}\right). \tag{108}$$

Substituting (107, 108) in (97, 98) we obtain

$$\mathbf{R}_{\mathrm{RBM}}(\mathbf{d}, \mathbf{\Lambda}) = \mathbf{R}_{\mathrm{RBM},N}(\mathbf{d}, \Lambda_{\mathrm{N}}) + \mathbf{R}_{\mathrm{RBM},T}(\mathbf{d}, \mathbf{\Lambda}), \tag{109}$$

$$\mathbf{R}_{\mathrm{RBM},N}(\mathbf{d}, \Lambda_{\mathrm{N}}) = \sum_{r=1}^{N_{\mathrm{r}}} c \int_{\Gamma_{\mathrm{D}}^{(r)}} \left(\bar{g}_{\mathrm{N}}(\mathbf{d}, \Lambda_{\mathrm{N}})\mathbf{n}\left(\sum_a \nabla(\mathbb{N}_a)\right) \cdot \mathbf{n}\right) \bar{g}_{\mathrm{N}}(\mathbf{d}, \Lambda_{\mathrm{N}})\, d\Gamma, \tag{110}$$

$$\begin{aligned} \mathbf{R}_{\mathrm{RBM},T}(\mathbf{d}, \mathbf{\Lambda}) = \sum_{r=1}^{N_{\mathrm{r}}} c \int_{\Gamma_{\mathrm{D}}^{(r)}} \Bigg\{ & \bar{g}_{\mathrm{N}}(\mathbf{d}, \Lambda_{\mathrm{N}})\mathbf{n}\left(\sum_a \nabla(\mathbb{N}_a)\right) \cdot \mathbf{t} \\ & + \bar{g}_{\mathrm{N}}(\mathbf{d}, \Lambda_{\mathrm{N}})\mathbf{t}\left(\sum_a \nabla(\mathbb{N}_a)\right) \cdot \mathbf{n} \qquad (111) \\ & + \bar{g}_{\mathrm{T}}(\mathbf{d}, \Lambda_{\mathrm{T}})\mathbf{t}\left(\sum_a \nabla(\mathbb{N}_a)\right) \cdot \mathbf{t} \Bigg\} \bar{g}_{\mathrm{T}}(\mathbf{d}, \Lambda_{\mathrm{T}})\, d\Gamma, \\ & \qquad (112) \end{aligned}$$

with

$$\bar{g}_{\mathrm{N}}^{(r)}(\mathbf{d}, \Lambda_{\mathrm{N}}) = \frac{2\tau^{(r)}}{H} \left(\Lambda_{\mathrm{N}}^{(r)} - \mathbf{n} \cdot \mathbf{P}_{\mathrm{I}}(\mathbf{d}) \cdot \mathbf{N} \right), \tag{113}$$

$$\bar{g}_{\mathrm{T}}^{(r)}(\mathbf{d}, \Lambda_{\mathrm{T}}) = \frac{2\tau^{(r)}}{H} \left(\Lambda_{\mathrm{T}}^{(r)} - \mathbf{t} \cdot \mathbf{P}_{\mathrm{I}}(\mathbf{d}) \cdot \mathbf{N} \right). \tag{114}$$

Note that the modifications to the mechanical residual of those domains that exhibit RBMs $\Omega^{(m)}$ are performed involving nodal displacements $\mathbf{d}^{(m)}$ of the same domain and Lagrange multipliers defined at the interface patch $D^{(r)}$ assuming that only those interface patches $D^{(r)}$ adjacent to $\Omega^{(m)}$ are selected (cf. Fig. 6). The new submatrices of the global dual system (70) corresponding to domains with rigid body modes $\Omega^{(m)}$ can be defined as

$$\begin{bmatrix} \mathbf{K}_{\mathrm{dd}}^{(m)} & \mathbf{K}_{\mathrm{d}\Lambda}^{(m)} \\ \mathbf{K}_{\Lambda\mathrm{d}}^{(m)} & \mathbf{K}_{\Lambda\Lambda} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial \left(\tilde{\mathbf{R}}_{\mathrm{mech}}(\mathbf{d}, \boldsymbol{\Lambda}) \right)^{(m)}}{\partial \mathbf{d}^{(m)}} & \dfrac{\partial \left(\tilde{\mathbf{R}}_{\mathrm{mech}}(\mathbf{d}, \boldsymbol{\Lambda}) \right)^{(m)}}{\partial \boldsymbol{\Lambda}^{(m)}} \\ \dfrac{\partial \left(\tilde{\mathbf{R}}_{\lambda}(\mathbf{d}, \boldsymbol{\Lambda}) \right)^{(m)}}{\partial \mathbf{d}^{(m)}} & \dfrac{\partial \tilde{\mathbf{R}}_{\lambda}(\mathbf{d}, \boldsymbol{\Lambda})}{\partial \boldsymbol{\Lambda}} \end{bmatrix}, \tag{115}$$

$$\begin{bmatrix} \mathbf{r}_{\mathrm{d}}^{(m)} \\ \mathbf{r}_{\Lambda} \end{bmatrix} = \begin{bmatrix} \left(\tilde{\mathbf{R}}_{\mathrm{mech}}(\mathbf{d}, \boldsymbol{\Lambda}) \right)^{(m)} \\ \tilde{\mathbf{R}}_{\lambda}(\mathbf{d}, \boldsymbol{\Lambda}) \end{bmatrix}, \tag{116}$$

where

$$\left(\tilde{\mathbf{R}}_{\mathrm{mech}}(\mathbf{d}, \boldsymbol{\Lambda}) \right)^{(m)} = \left(\mathbf{R}_{\mathrm{mech}}(\mathbf{d}, \boldsymbol{\Lambda}) \right)^{(m)} + \left(\mathbf{R}_{\mathrm{RBM}}(\mathbf{d}, \boldsymbol{\Lambda}) \right)^{(m)}. \tag{117}$$

Consequently, the bandwidth in (70) is kept also for the contributions to domains with RBMs and, for this reason, the system can be processed in a parallel fashion as detailed in (78–80).

Remark 2.10 Equations (113, 114) are obtained assuming linear (constant strain) interface patches as adopted in the examples presented in this work. If a higher order interpolation is employed at the interface patches, a mean value of the normal and tangential traction t_{N} and t_{T} can be used along $\Gamma_{\mathrm{D}}^{(p)}$ such that the relations (113, 114) can still be utilized.

Remark 2.11 In all our computations we considered all interface patches $D^{(r)}$ adjacent to domain $\Omega^{(m)}$. If the base-line of the patch $D^{(r)}$ chosen to avoid the RBM is not located at $\partial\Omega^{(m)}$, contributions to adjacent domains would be expected in (70) and, therefore, the system could not be properly processed in a parallel fashion.

2.6 Iterative scheme for the non-linear DIM

The linearized set of equations in (70) obtained with a Newton-like scheme is solved iteratively for each load/time step Δt as done in the so-called Newton-Krylov-Schur methods [6,12]. In this view, a first type of iterations refer to the solution of the non-linear problem with successive linear approximations. A second type of iterations arise from the solution of the flexibility problem in (77) where usually Conjugate Gradient or GMRES iterates are considered. Finally, the Schur complements are utilized for the local solutions at each domain $\Omega^{(s)}$ (80).

Assuming a fix domain decomposition and a given Delaunay interface discretization, the iterative scheme for the non-linear DIM framework is summarized in Box 1.

Box 1 Iterative scheme for the non-linear DIM

Initialize: $\mathbf{d}_0^{(s)}$, $\Lambda_0^{(s)}$, $\mathbf{K}_{\mathrm{dd},0}^{(s)}$, $\mathbf{K}_{\mathrm{d}\Lambda,0}^{(s)}$, $\mathbf{K}_{\Lambda\mathrm{d},0}^{(s)}$, $\mathbf{K}_{\Lambda\Lambda,0}$, $\mathbf{r}_{\mathrm{d},0}^{(s)}and\mathbf{r}_{\Lambda,0}$
New load step $t_{n+1} = t_n + \Delta t$:

1. Add new external force increment to the residual $\mathbf{r}_{\mathrm{d}}^{(s)} \leftarrow \mathbf{f}_{\mathrm{ext},n+1}^{(s)}$
2. Update tangent stiffness $\mathbf{K}_{\mathrm{dd},i}^{(s)}$, $\mathbf{K}_{\mathrm{d}\Lambda,i}^{(s)}$, $\mathbf{K}_{\Lambda\mathrm{d},i}^{(s)}$, $\mathbf{K}_{\Lambda\Lambda,i}$ using nodal data from last iteration i
    - If floating domain $\mathbf{K}_{\mathrm{dd},i}^{(s)}$, $\mathbf{K}_{\mathrm{d}\Lambda,i}^{(s)}$, $\mathbf{r}_{\mathrm{d},i}^{(s)} \leftarrow \tilde{\mathbf{R}}_{\mathrm{mech},i}(\mathbf{d}, \boldsymbol{\Lambda})^{(s)}$
    - Otherwise $\mathbf{K}_{\mathrm{dd},i}^{(s)}$, $\mathbf{K}_{\mathrm{d}\Lambda,i}^{(s)}$, $\mathbf{r}_{\mathrm{d},i}^{(s)} \leftarrow \mathbf{R}_{\mathrm{mech},i}(\mathbf{d}, \boldsymbol{\Lambda})^{(s)}$
3. Parallel solver:
    - Factorize local stiffness and compute the flexibility components $\mathbf{F}_{\mathrm{I},i}$ and $\Delta\mathbf{g}_{\mathrm{I},i}$
    - Solve for $\Delta\boldsymbol{\Lambda}_{i+1} = \left(\mathbf{F}_{\mathrm{I},i}\right)^{-1} \Delta\mathbf{g}_{\mathrm{I},i}$ using an iterative solver.
    - Compute displacement increments $\Delta\mathbf{d}_{i+1}^{(s)} = \left(\mathbf{K}_{\mathrm{dd},i}^{(s)}\right)^{-1} \left(\mathbf{r}_{\mathrm{d},i}^{(s)} - \mathbf{K}_{\mathrm{d}\Lambda,i}^{(s)} \Delta\Lambda_{i+1}\right)$
4. Update domain internal variables, assemble internal forces and compute new residuals $\mathbf{r}_{\mathrm{d},i+1}^{(s)}$ and $\mathbf{r}_{\Lambda,i+1}^{(s)}$
5. Check for convergence:
    If converged, go to 1
    Else, add new iteration i and go to 2

All operations involving domain FE quantities (i.e. with superscript s) require a loop over all domains $\Omega^{(s)}$

3 Framework validation through representative simulations

In the following we present a number of academic examples which highlight the accuracy and convergence properties of the framework. Attention is focused on the continuity of the solution at the interface, the convergence rate upon mesh refinement and a qualitative comparison of the advantages and eventual pitfalls against existing formulations. Infinitesimal strain theory is utilized in all examples except from the one reported in Sect. 3.5 where finite strain theory is considered. Additionally, the stabilization parameter τ (cf. Eqs. 63–68) is set to 10^{-7} in all our computations except from the results reported by the end of Sect. 3.4 where a sensitivity analysis is performed varying the values of τ.

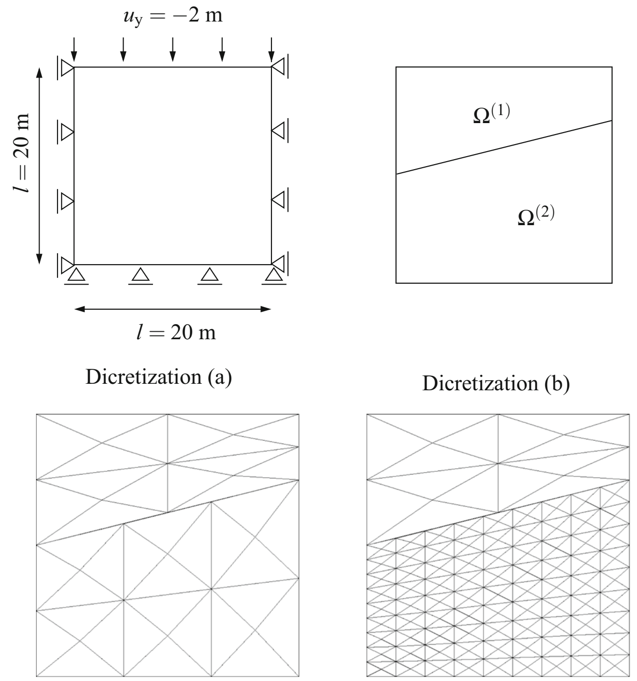


Fig. 7 Geometry, boundary conditions and domain decomposition (*top*). Domain discretizations using linear T3 elements (*bottom*)

3.1 Patch test

The so-called ‘patch test’ is specially selected to verify the correct transference of information throughout the interface. A compression analysis is performed on a two-dimensional homogeneous quadrilateral specimen. Plane strain conditions are assumed and a linear elastic constitutive law is considered with Young’s modulus $E = 2.1 \times 10^2$ MPa and Poisson’s ratio $\nu = 0.3$. The geometry, boundary conditions and domain discretizations are depicted in Fig. 7.

The quadrilateral specimen is submitted to an homogeneous strain state where

$$\boldsymbol{\varepsilon} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \dfrac{u_y}{l} & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{118}$$

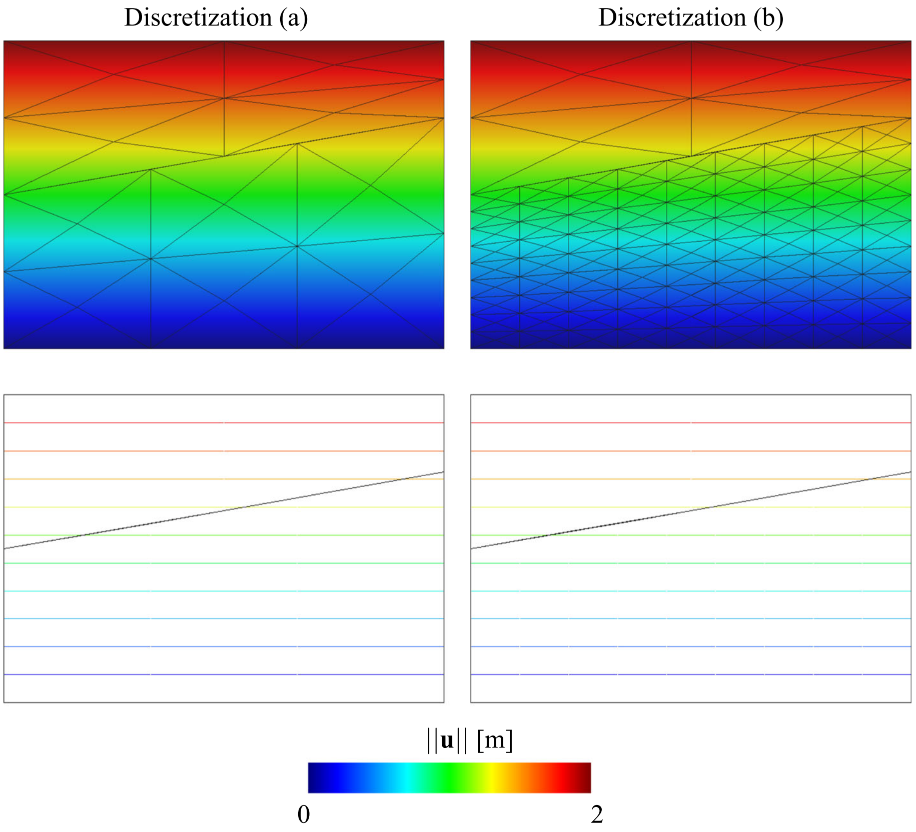
Considering the linear elastic constitutive relation

$$\boldsymbol{\sigma} = \lambda Tr(\boldsymbol{\varepsilon})\mathbf{1} + 2\mu\boldsymbol{\varepsilon} \tag{119}$$

with Lamé constants

$$\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)}, \quad \mu = \frac{E}{2(1+\nu)} \tag{120}$$

Fig. 8 Total displacement $||\mathbf{u}||$ contours (*top*) and contour lines (*bottom*) within the deformed configurations for discretizations (**a**) and (**b**). 3 × displacement magnification



and **1** being the second order unity tensor, the analytical solution reads $\sigma_x = -1.2115 \times 10^4$ Pa and $\sigma_y = -2.8269 \times 10^4$ Pa. These values are obviously constant throughout the specimen, and are taken as the reference solution with nine significant decimal digits when compared to the stress distributions obtained through our simulations using double precision floating points.

Displacement contours for both discretizations are shown in Fig. 8 within the corresponding deformed configurations. Contour lines show that continuity is satisfied throughout the whole specimen at this observation scale. The horizontal and vertical stress fields are constant and identical for both cases according to the machine precision and therefore not reported. However the relative error e_r between the numerical and reference stresses is 1.2×10^{-8} for the horizontal stress and 1.1×10^{-8} for the vertical stress.

Since the maximum relative error is order 10^{-8}, it is concluded that the proposed methodology passes the patch test and provides an adequate transference of information across a non-conforming interface.

3.2 Patch test with floating subdomains

In the same spirit as the previous example a patch test is set up using a domain discretization containing one floating domain. The objective is, therefore, to verify that the algorithm to handle floating domains developed in Sect. 2.5 does not affect the resulting solution. To this end, a biaxial compression test is imposed to a quadrilateral specimen divided into nine domains as shown in Fig. 9 and a linear elastic constitutive law is considered with plane strain conditions. The parameter c used to overcome the appearance of RBMs (cf. 93–98) is set to 10^{-4} in all examples that require the treatment of floating domains.

Note that interface corner points are correctly connected using the interface discretization. The boundary conditions result in an homogeneous strain state where

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \frac{\delta_x}{l} & 0 & 0 \\ 0 & \frac{\delta_y}{l} & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad (121)$$

with $\delta_x = u_x^R - u_x^L$ and $\delta_y = u_y^T - u_y^B$. Considering the linear elastic constitutive relation and Lamé constants in (119, 120), respectively, the analytical stresses read $\sigma_x = -9.9615 \times 10^4$ Pa and $\sigma_y = -8.8846 \times 10^4$ Pa.

Displacement contour lines in the deformed configuration plotted in Fig. 10 show that displacement continuity is again satisfied throughout the whole specimen at this observation scale. The obtained horizontal and vertical stress fields are obviously constant and the relative error e_r between the

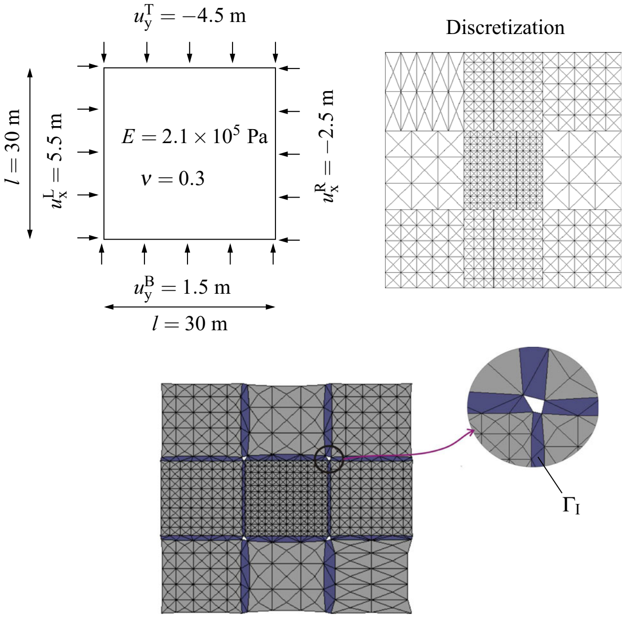


Fig. 9 Geometry, boundary conditions and material parameters (*top left*). Domain discretizations using linear T3 elements (*top right*). Detail of the interface discretization Γ_I at corner points (*bottom*)

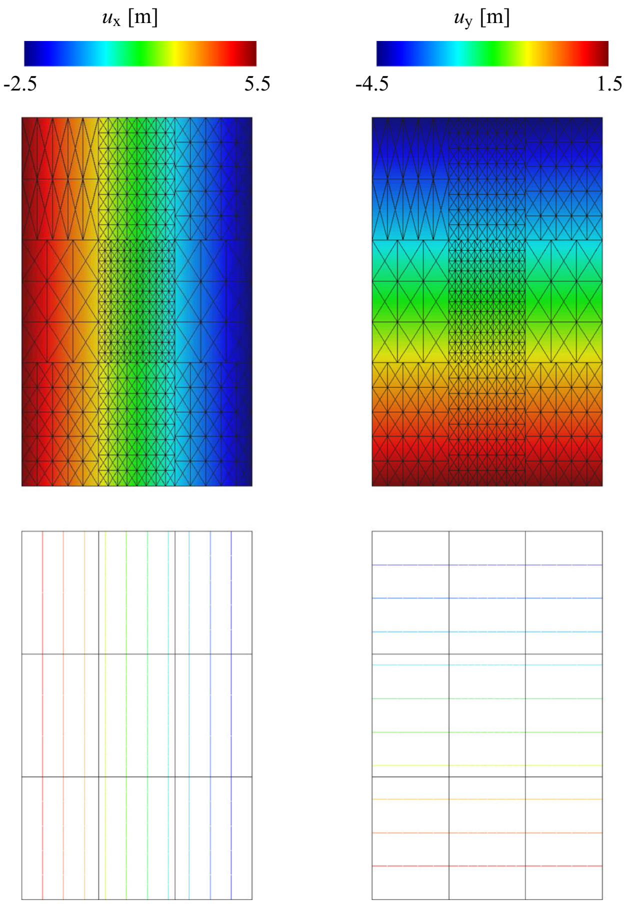


Fig. 10 Horizontal (*left*) and vertical (*right*) displacement contours (*top*) and contour lines (*bottom*) within the deformed configurations. 1.75 × displacement magnification

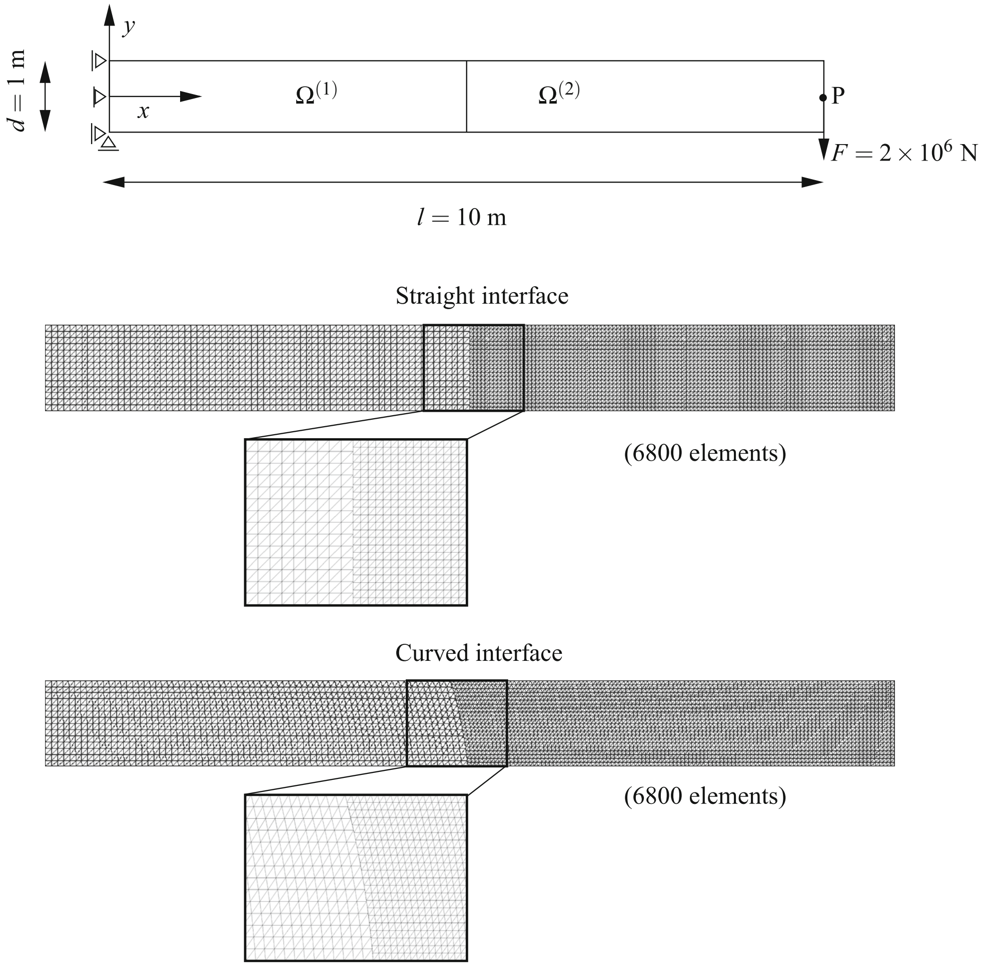


Fig. 11 Boundary conditions and domain discretizations for the cantilever beam test

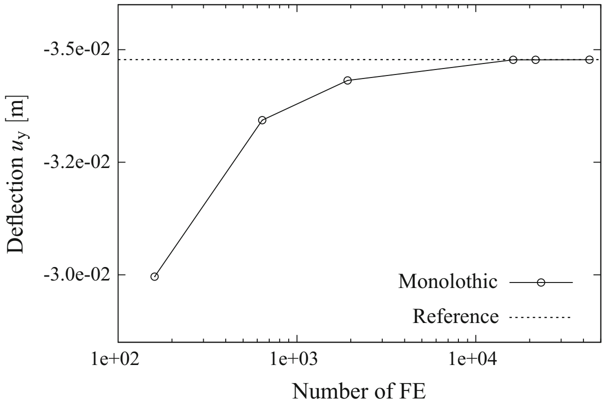


Fig. 12 Mesh sensitivity analysis with a monolithic approach

numerical and reference stresses is 1.5×10^{-9} for the horizontal stress and 1.8×10^{-9} for the vertical stress.

Both relative errors are significantly small and, therefore, it is concluded that the patch test is successfully passed in those cases where floating subdomains are present. This indicates that the proposed non-intrusive methodology to handle floating domains does not affect the accuracy of the domain decomposition framework.

3.3 Cantilever beam test

The following example is based on a test proposed by Herry et al. [24] where the deflection at one end of a cantilever beam is measured in order to assess the performance of the domain decomposition method when splitting the beam into two domains connected at a non-conforming interface. The material is linear elastic with Young's modulus $E = 2.1 \times 10^{11}$ N/m^2 and Poisson's ratio $\nu = 0.3$. The two-dimensional test is conducted under plane strain conditions and the geometry, boundary conditions and domain discretizations are shown in Fig. 11.

The reference solution for the right end deflection is obtained through simulations with a monolithic approach, i.e. considering a single discretization for the whole specimen and employing a standard FE approach. A mesh sensitivity analysis is performed (cf. Fig. 12) and the reference deflec-

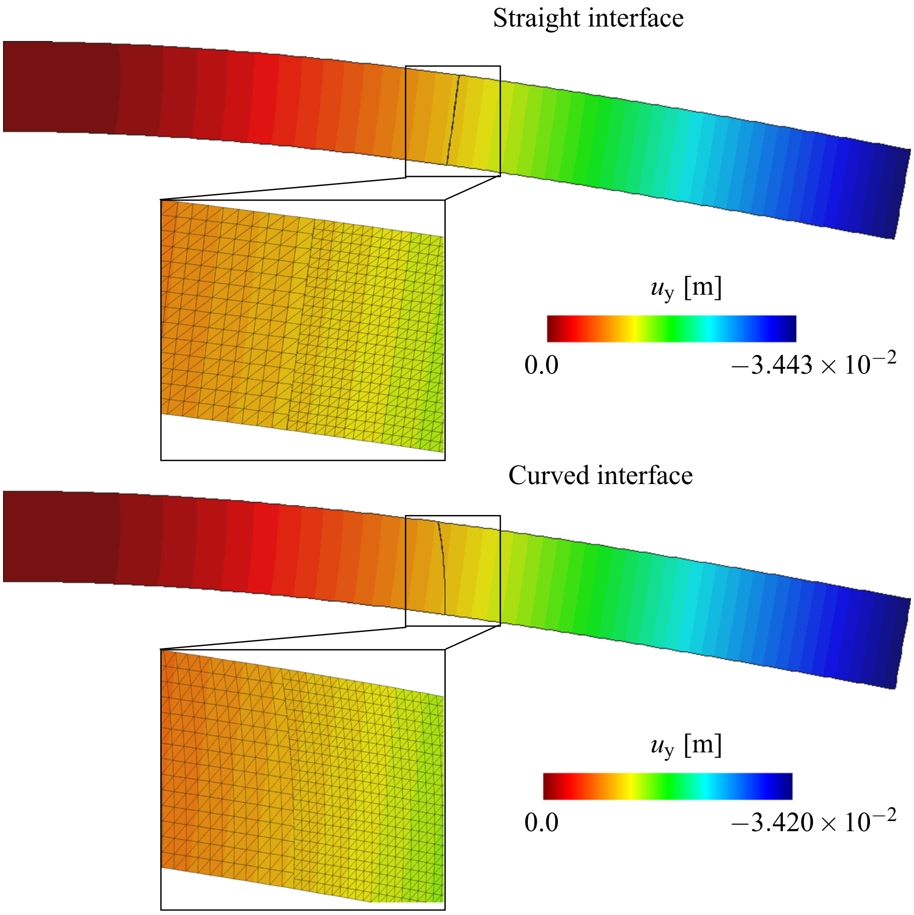


Fig. 13 Vertical displacement contours employing straight and curved interfaces. 35 × displacement magnification

Table 1 Cantilever beam test. Accuracy of the proposed approach compared to the reference numerical solution

	Straight interface	Curved interface
Number of Λ	36	36
$\frac{u_y}{u_y^{\text{ref}}}$	99.00 %	98.33 %

Table 2 Cantilever beam test. Accuracy of similar approaches compared to a reference theoretical solution

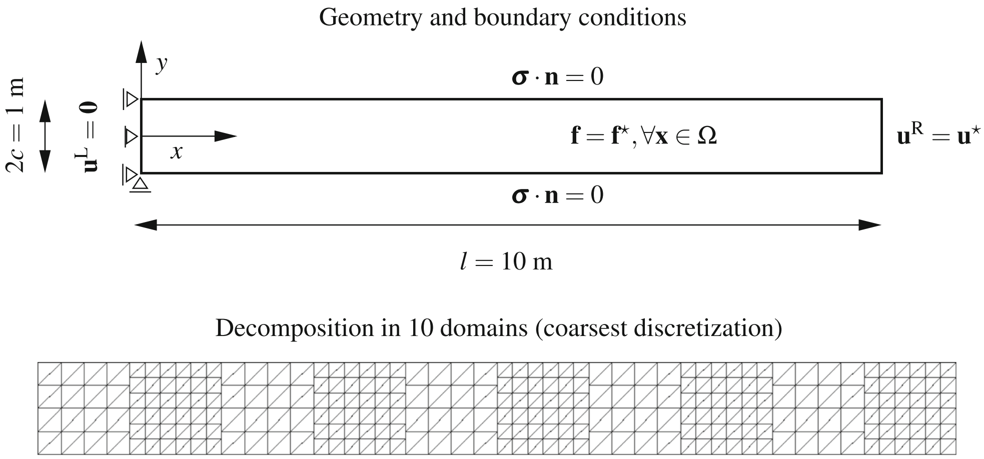
	Coarse mortar mesh	Fine mortar mesh	Dual DD method
Number of Λ	15	23	35
$\frac{u_y}{u_y^{\text{ref}}}$	79.06 %	99.86 %	99.84 %

tion $u_y^{\text{ref}} = -3.478 \times 10^{-2}$ m is selected which corresponds to a mesh discretization similar to the one chosen in the domain decomposition (DD) approach.

The vertical displacement contours for both straight and curved interfaces are shown in Fig. 13. Note that displacement continuity is fulfilled across the straight and curved interfaces.

The high accuracy of the DD approach compared to the reference monolithic solution is proven in Table 1. Both decompositions provide accuracy percentages which are above 98 %.

A similar example was carried out by Herry et al. [24] where the cantilever beam was analyzed using bilinear quadrilateral FE. Table 2 is reproduced from this study and compares the performance between mortar methods and a dual domain decomposition approach developed for non-matching meshes. It is worth noting that the accuracy of the deflection provided in the present contribution is very much comparable to the accuracies of the above mentioned approaches. It is noticeable in Table 2 that the original mortar method performs poorly when the coarse mesh is used to define the mortar surface. This is obviously regarded as a considerable drawback of the method since in a general case the distinction between coarse and fine discretizations at the interface might not be straightforward. The approach based on dual domain decomposition methods does not suffer from this shortcoming since a third surface is constructed with a


Geometry and boundary conditions

Decomposition in 10 domains (coarsest discretization)

Fig. 14 Boundary conditions (*top*) and domain decomposition using the coarsest FE discretization (*bottom*)

particular arrangement of Lagrange multipliers that leads to an optimum matching condition regarding the kinematic continuity. However, this approach is based on the assumption that the three surfaces have the same geometry. This implies that, upon discretization, a number of nodes, e.g. extreme nodes, are common. This assumption is reasonable for cases in which the domains are originated from the decomposition of a continuous body. Conversely, if the domains are glued at a common surface and discretized independently, this condition might not be realistic.

More advanced mortar methods employed nowadays [40–42,50] consider a carefully chosen Lagrange multiplier space based on stability and optimality considerations or even a third auxiliary surface with an optimal node collocation to minimize the error at the interface integrals. This may not lead to the suboptimal performance shown in Table 2 but, for the case of a third auxiliary surface, extra degrees of freedom (possibly condensible) are required and, therefore, the cost and complexity of the formulation is increased. For the case of an heterogeneous interface, the performance in terms of accuracy of such advanced mortar methods would not be affected. However, the DIM method outlined in this contribution for two-dimensional applications distinguishes from these techniques in the sense that no extra projection surfaces and extra DOFs are required for a general geometrically incompatible interface since this is automatically taken into account by the interface mesh. It is not the author's intention to provide a review study of all recent mortar variants but rather compare the accuracy of the DIM method with situations in which original mortar methods would perform optimally. We believe that more advanced mortar technologies would provide a comparable accuracy to the one observed by original mortar techniques when the finest mesh is selected as the mortar surface. In such scenarios, the DIM method would be considered a computationally cheaper alternative.

3.4 Convergence analysis and dependence on the stabilization parameter τ

The effect of mesh refinement at the interface is studied in this section. Based on the work of Girault et al. [18] a test is set up in a two dimensional cantilever beam under plane strain conditions. Infinitesimal strain theory is utilized and the material follows a linear elastic constitutive law with Young's modulus $E = 2.0 \times 10^5\ \mathrm{N/m^2}$ and Poisson's ratio $\nu = 0.3$.

As sketched in Fig. 14, homogeneous displacement boundary conditions are imposed at the left end of the beam $\mathbf{u}^{\mathrm{L}} = \mathbf{0}$ and the right end follows the displacement $\mathbf{u}^{\mathrm{R}} = \mathbf{u}^{\star}$ where

$$\mathbf{u}^{\star} = \begin{bmatrix} -3\alpha x^2 y \\ \alpha x^3 + \dfrac{3\alpha\lambda}{\lambda + 2\mu} x(y^2 - c^2) \end{bmatrix}, \qquad (122)$$

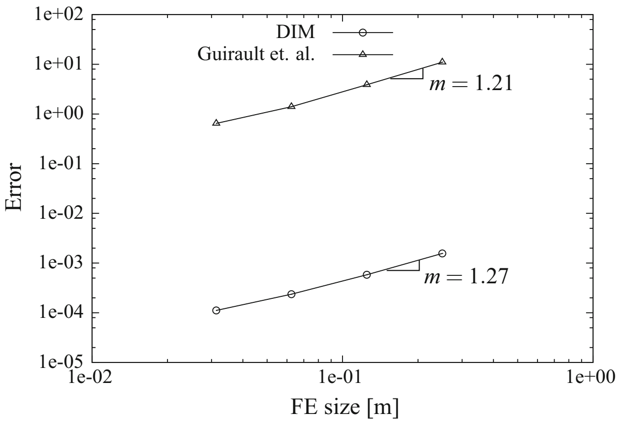


Fig. 15 Local convergence error based on the displacement jump norm $\left|\left|[\![\mathbf{u}]\!]\right|\right|_{\mathrm{L}_2}$ in (124) for different mesh discretizations

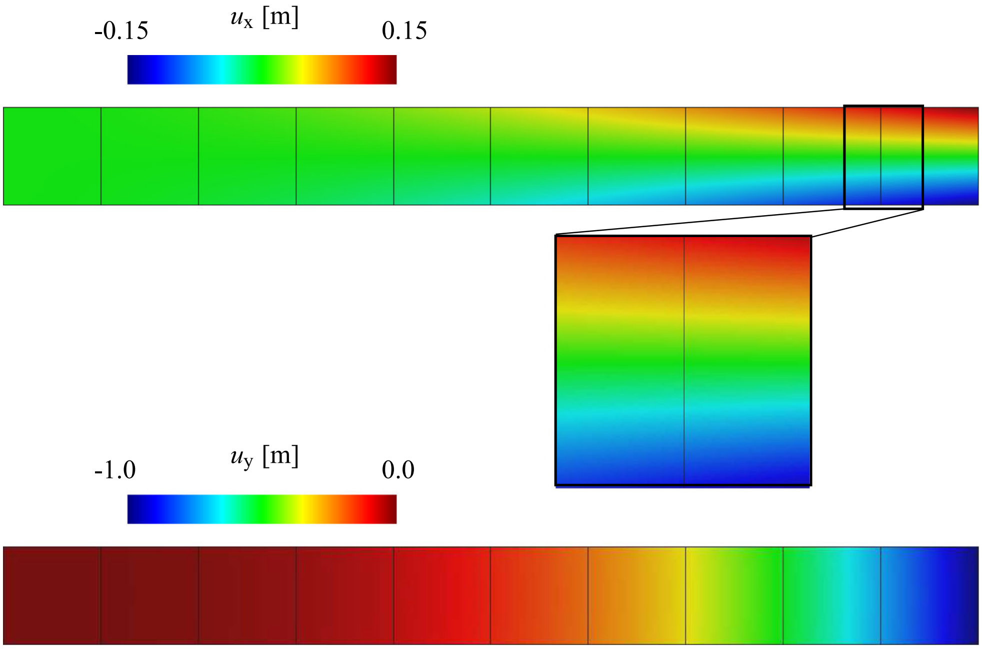


Fig. 16 Horizontal (*top*) and vertical (*bottom*) displacement contours for the finest FE discretization

λ and μ being the Lamé constants described in (120) and $\alpha = -1.0 \times 10^{-3}\mathrm{m}^{-2}$. The body force field on the beam reads

$$\mathbf{f}^{\star} = \begin{bmatrix} 6\alpha\mu \dfrac{3\lambda + 4\mu}{\lambda + 2\mu} y \\ 0 \end{bmatrix}. \tag{123}$$

This condition is translated into homogeneous tractions $\boldsymbol{\sigma} \cdot \mathbf{n} = 0$ at the top and bottom surfaces of the beam as indicated in Fig. 14.

The beam is split into ten domains as indicated in Fig. 14 and an alternate regular discretization is considered between domains such that all interfaces are non-conforming. Four levels of refinement are employed which correspond to element sizes ranging from $1/4$ to $1/32$ m for one set of domains and $1/6$ to $1/48$ m for the alternate set of domains.

The error is measured taking into account the displacement jump $[\![\mathbf{u}]\!]$ at the interface $\Gamma_{i,j}$ between domains Ω_i and Ω_j as

$$\left|\left|[\![\mathbf{u}]\!]\right|\right|_{\mathrm{L}_2} = \left(\int_{\Gamma_{i,j}} \left| \mathbf{u}_i - P_{\Gamma_i}(\mathbf{u}_j) \right|^2 \, d\Gamma_{i,j} \right)^{1/2}, \quad \forall \Gamma_{i,j} \in \Gamma_{\mathrm{I}}, \tag{124}$$

where $P_{\Gamma_i}(\mathbf{u}_j)$ denotes the projection of the coarse domain interface displacements u_j onto the fine interface discretization Γ_i. It can be observed that the error $\left|\left|[\![\mathbf{u}]\!]\right|\right|_{\mathrm{L}_2}$ reduces with decreasing element size and the rate of change m is about 1.27 between the first two discretizations (cf. Fig. 15). This is in accordance with the results presented by Girault et al. [18] where a different error is computed which takes into account the prescribed boundary and Lagrange multipliers at the interface.

Additionally, the displacement norm $||\mathbf{u} - \mathbf{u}_\mathrm{h}||_{\mathrm{L}_2}$ is calculated using the approximated $\mathbf{u}_\mathrm{h}$ and theoretical solution $\mathbf{u}$ throughout the specimen as:

$$||\mathbf{u} - \mathbf{u}_\mathrm{h}||_{\mathrm{L}_2} = \left(\int_{\Omega} |\mathbf{u} - \mathbf{u}_\mathrm{h}|^2 \, d\Omega \right)^{1/2}. \tag{125}$$

The convergence study summarized in Fig. 15 shows satisfactory results when the framework is compared to similar techniques. Displacement jumps at the interfaces are hardly visible at the displacement contour plots given in Fig. 16 which reveals that displacement continuity is sufficiently satisfied in the present approach.

Results in Fig. 17 show the expected convergence order $\mathcal{O}(h^2)$ using the displacement norm in (125) for a monolithic analysis. When the analysis is performed using the proposed domain decomposition approach using different mesh refinements and non-conforming interfaces, the obtained convergence order is found lower than $\mathcal{O}(h^2)$. The reason for this suboptimal behaviour is related with the use of piecewise constant Lagrange multipliers at each interface patch. Although this choice has a positive impact on the simplicity of the framework implementation, the exact theoretical convergence order may not be recovered. A more thorough study considering higher interpolations of the Lagrange multiplier

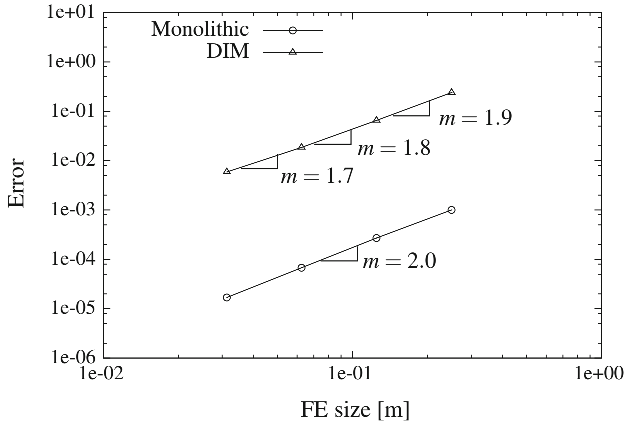


Fig. 17 Global convergence error based on the displacement norm $||\mathbf{u} - \mathbf{u}_h||_{L_2}$ in (125) for different mesh discretizations

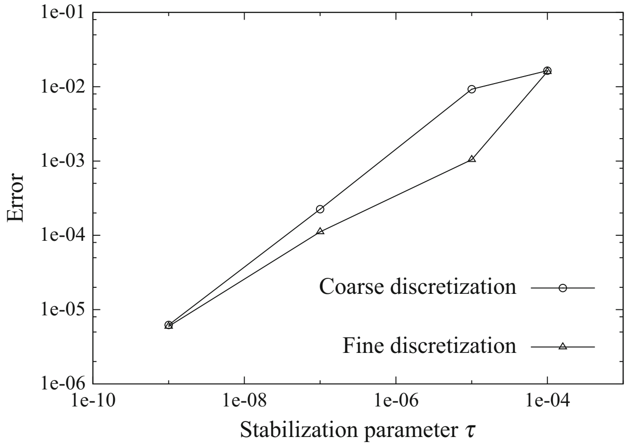


Fig. 18 Error $\left|\left|[\![\mathbf{u}]\!]\right|\right|_{L_2}$ for different values of the stabilization parameter τ

field and its impact on the presented framework is considered a topic for future research.

The sensitivity of the analysis with respect to the stabilization parameter τ is reported in Fig. 18 where the jump error $\left|\left|[\![\mathbf{u}]\!]\right|\right|_{L_2}$ is plotted against different values of τ for the coarsest and finest discretization sets. It is observed that the error diminishes with decreasing values of τ. Additionally, one can utilize remarkably low values of the stabilization parameter upon mesh refinement which confirms the consistency character indicated in Sect. 2.3. In general, coarser meshes tend to give higher error values for a given value of the stabilization parameter τ.

3.5 Geometrically incompatible non-matching meshes

The following example consists in the assembly of a simplified plane-like structure where two geometrically incompatible interfaces are identified which connect the wings to the plane fuselage (cf. Fig. 19). A distributed force of 0.5 N/m is applied at the bottom part of both wings to emulate the lift experienced during flight conditions. Several points at the fuselage section are imposed null Dirichlet boundary conditions in order to restrict the global rigid body modes. The

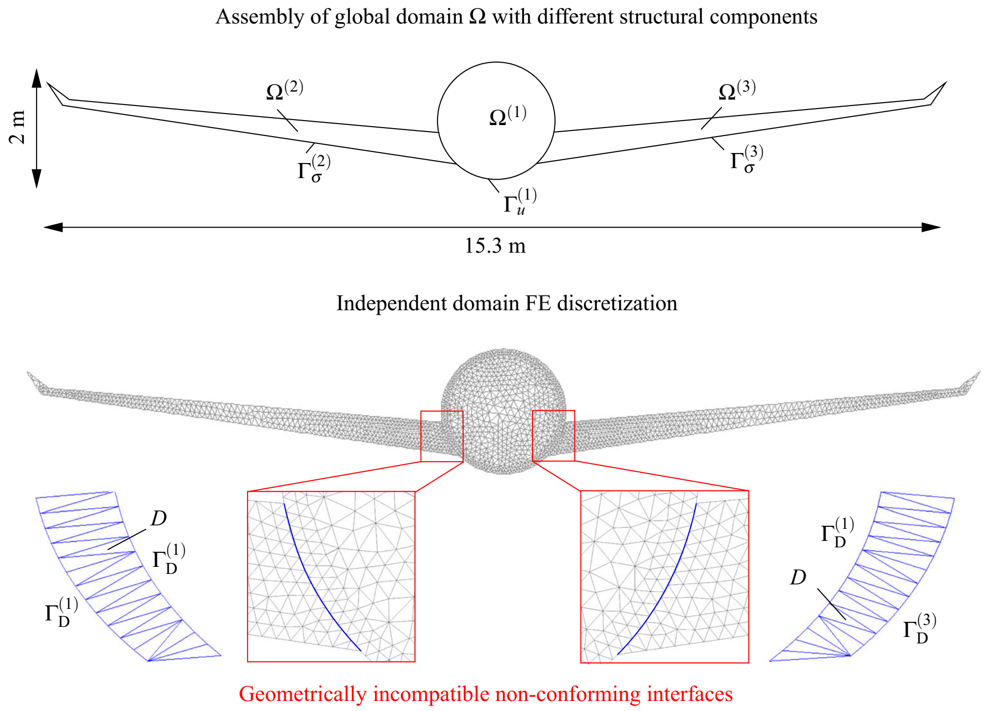


Fig. 19 Plane structure assembly. Geometry (*top*) and domain decomposition (*bottom*)

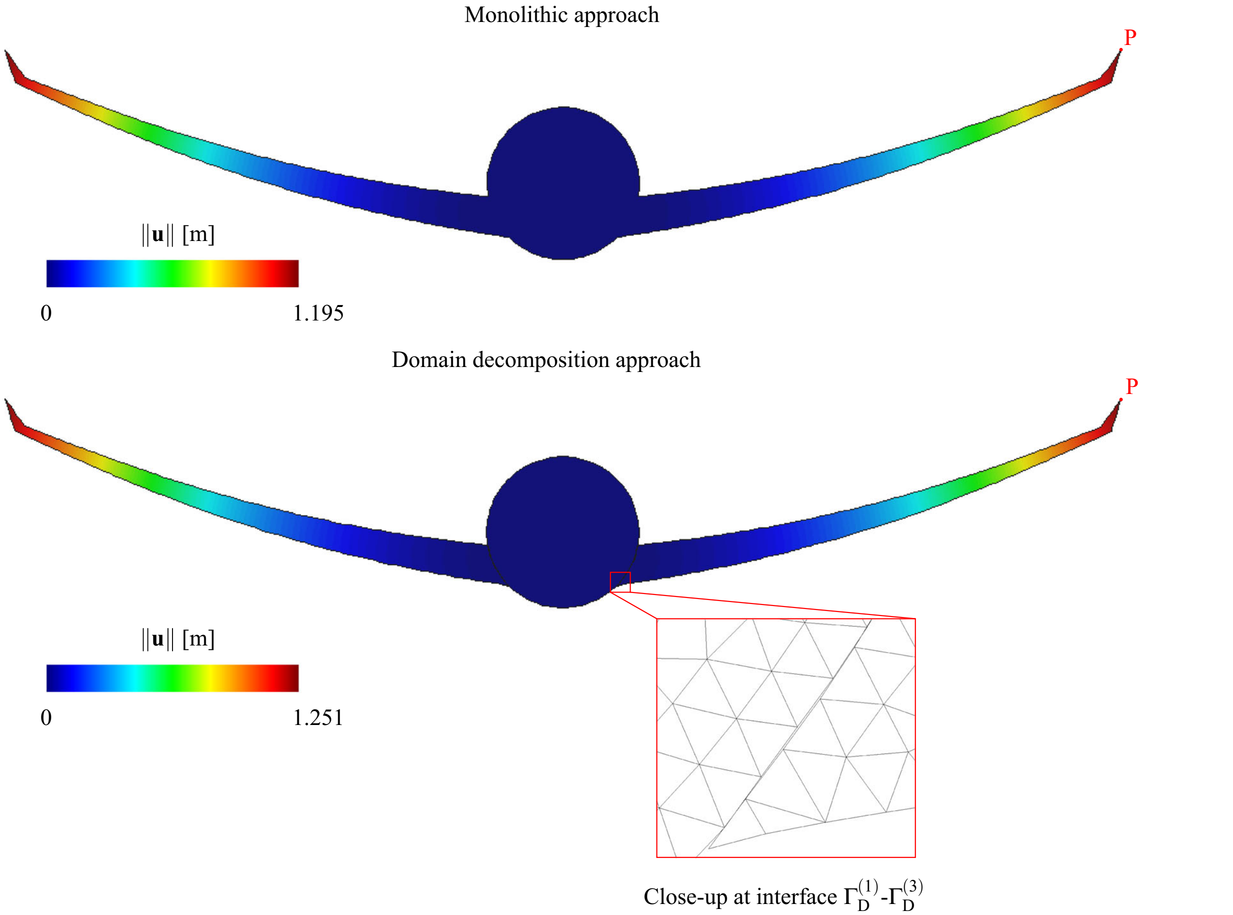

Monolithic approach
P
||**u**|| [m]
0
1.195
Domain decomposition approach
P
||**u**|| [m]
0
1.251
Close-up at interface $\Gamma_D^{(1)}$-$\Gamma_D^{(3)}$

Fig. 20 Total displacement contours of the monolithic (*top*) and domain decomposition approaches (*bottom*) in the deformed configurations

Table 3 Deflections at point P using the monolithic and domain decomposition approaches

	Monolithic $\|\|\mathbf{u}_{\mathrm{M}}\|\|$ [m]	Domain decomposition $\|\|\mathbf{u}_{\mathrm{DDM}}\|\|$ [m]	Relative error $e_{\mathrm{r}} = \dfrac{\|\|\mathbf{u}_{\mathrm{M}}\|\| - \|\|\mathbf{u}_{\mathrm{DDM}}\|\|}{\|\|\mathbf{u}_{\mathrm{DDM}}\|\|}$
Point P	1.195	1.251	4.676 %

material is considered hyperelastic [48] satisfying

$$\mathbf{S} = \lambda tr(\mathbf{E})\mathbf{1} + 2\mu\mathbf{E}, \tag{126}$$

S being the second Piola–Kirchhoff stress tensor, **E** the Green–Lagrange strain, and λ and μ the Lamé constants. Due to the simplicity of the model, i.e. discretized by solid elements and not thin-shells, the material parameters are chosen arbitrarily and are not sought to mimic a real situation but rather illustrate a target case for the use of geometrically incompatible non-conforming interfaces. In the current simulations the Young's modulus and Poisson's ratio are set to 2.0×10^4 MPa and 0.3, respectively.

In order to assess the performance of the DIM in this example, the vertical displacement of the wing end (point P) is monitored and compared to the one obtained with a reference monolithic approach considering a similar spatial discretization. As it can be observed in the contour plots in Fig. 20 and the deflections reported in Table 3, the results for the geometrically incompatible non-conforming interface are remarkably close to the ones obtained with a monolithic approach used as the reference solution for this problem. Note that the interface gap depicted in the close-up in Fig. 20 is hardly visible and it can be concluded that the methodology shows satisfactory results and proves to be remarkably competent against the most demanding cases.

4 Conclusions and future perspectives

The kernel of the proposed DIM to be used in domain decomposition methods and presented in this contribution resides in an explicit discretization of the interface by means of a zero-thickness Delaunay triangulation. This is accomplished through a fictitious contraction of the subdomains at the interface which allows for a proper discretization between the shrunk domain boundaries. The fictitious contraction has no impact on the solution of the problem since all calculations are performed using the original coordinates. Moreover, it is shown that the integrals over the zero-thickness interface are bounded despite the fact that the integrand is not bounded. The method is grounded in the so called Nitsche methods in which a stabilization term is added at the constraint equations. In this manner, zero diagonal terms are not present at the global system and instabilities are avoided if the LBB condition is not fulfilled by the chosen discretization. This process is viewed as a consistent penalty method since the stabilization term vanishes with progressive mesh refinement and the penalty factor can be made significantly small without affecting the results.

The methodology is inherited from the field of contact mechanics and, for this reason, is regarded as more general than existing domain decomposition strategies since there is no need for a fixed interface geometry that needs to be shared by the decomposed domains at both sides of the interface. This is for instance the case of some dual domain decomposition techniques in which the limit DOFs at the interface need to be common in both adjacent domain discretizations. Moreover, the generation of interface patches is independent of the choice of slave and master sides in contrast with early mortar methods and, for this reason, the methodology is regarded less prone to errors related with such choice. More evolved mortar methodologies are able to automatically handle these situations by considering, for instance, an extra interface surface from which a particular distribution of DOFs serves to construct an optimal set of interface constraints between adjacent meshes. However, such an intermediate surface involves calculations over extra DOFs which could increase the computational cost and complexity of the approach.

A new non-intrusive strategy to handle floating domains is outlined which adds an extra stabilization term to the energy functional with contributions of all adjacent interface patches. This avoids the calculation of a pseudo-inverse at floating domains and does not destroy the band structure of the global system. For this reason, it does not spoil a possible parallel solution as done with existing dual domain decomposition methods.

The DIM method passes the patch test also for the case of floating subdomains providing continuity of the stress field across the interface which indicates that all new ingredients do not affect the accuracy of the solution when compared to other established techniques. Remarkable continuity of the displacement field across the interface is shown in all reported experiments. A comparable accuracy degree has been observed with independence of the shape of the interfaces. In addition, good convergence rates are reached upon mesh refinement similar to other accurate techniques for non-conforming interfaces although theoretical convergence rates could not be obtained exactly due to the piece-wise constant interpolation of the Lagrange multipliers.

The algorithmic treatment of the subdomains allows for a parallel solution scheme analogous to well established techniques such as dual domain decomposition techniques. In this manner the local factorizations can be tackled by direct solvers and the resulting interface problem can be assembled in a matrix-free fashion and solved with the use of iterative solvers. A full parallel version of the framework involves the construction of adequate preconditioners for the interface problem which was out of the scope of this contribution and it is left as a future research line. In the same spirit, a 3D extension of the method is left as a topic for further research. Preliminary 3D works have been successfully performed for the contact domain case [22] and it is presumed that the technology shows sufficient potential to successfully perform in large 3D cases although challenges are expected concerning a robust 3D Delaunay tetrahedralization for the most complex interface geometries. In any case, the domain decomposition formulation applied to monoscale analysis or ’static’ spatial discretizations has the advantage of performing the interface meshing once at the beginning of the analysis and, therefore, the cost of a new Delaunay tetrahedralization (which could be certainly more involved for a 3D case) is negligible compared to the cost of the whole analysis.

The framework presented in this contribution provides the basis to study complex deformation phenomena involving large strains, e.g. bulk metal forming, and shows a clear potential to tackle multifield applications, e.g. mixed formulations for incompressible and thermo-mechanical problems. In this view, the field of Lagrange Multipliers needs to be extended in order to account for the temperature and pressure fields and it is planned for a future contribution.

The explicit Delaunay triangulation of the interface is expected to positively impact the field of multiresolution problems and adaptive multiresolution analysis. Since arbitrary discretizations can be handled at non-conforming interfaces, adaptive multiresolution analyses can be performed employing an independent ‘on-the-fly’ refinement at particular domains of interest without meshing restrictions. In the same spirit, existing independent discretizations of particular domains can be easily reused and incorporated to the calculations.

Due to its versatility and generality the DIM method is viewed as an attractive alternative to mortar methods and other established dual domain decomposition methods in which the interface geometry and its limits are restricted to the boundary discretization of the connected domains.

Acknowledgments The research leading to these results has received funding from the European Research Council under the European Union’s Seventh Framework Programme (FP/2007-2013)/ERC Grant

Agreement No. 320815 (ERC Advanced Grant Project "Advanced tools for computational design of engineering materials" COMP-DES-MAT). Oriol Lloberas-Valls gratefully acknowledges the funding received from the Spanish Ministry of Economy and Competitiveness through the "Juan de la Cierva" Postdoctoral Junior Grant: JCI-2012-13782 and the National Research Plan 2014: MAT2014-60919-R.

References

1. Babuška I (1973) The finite element method with lagrangian multipliers. Numer Math 20(3):179–192
2. Babuška I, Narasimhan R (1997) The Babuška–Brezzi condition and the patch test: an example. Comput Methods Appl Mech Eng 140(12):183–199
3. Barrett R, Berry M, Chan TF, Demmel J, Donato JM, Dongarra J, Eijkhout V, Pozo R, Romine C, Van Der Vorst H (1993) Templates for the solution of linear systems: building blocks for iterative methods. SIAM Press, Philadelphia, PA
4. Becker R, Hansbo P, Stenberg R (2003) A finite element method for domain decomposition with non-matching grids. ESAIM Math Model Numer Anal 37:209–225
5. Bernardi C, Maday Y, Patera A (1994) A new nonconforming approach to domain decomposition: the mortar element method. In: Brezis H, Lions JL (eds) Nonlinear partial differential equations and their application, volume XI of College de France Seminar. Pitman, London, pp 13–51
6. Cresta P, Allix O, Rey C, Guinard S (2007) Nonlinear localization strategies for domain decomposition methods: application to post-buckling analyses. Comput Methods Appl Mech Eng 196(8):1436–1446
7. de Boer A, van Zuijlen AH, Bijl H (2007) Review of coupling methods for non-matching meshes. Comput Methods Appl Mech Eng 196(8):1515–1525
8. Duarte CA, Kim DJ (2008) Analysis and applications of a generalized finite element method with global-local enrichment functions. Comput Methods Appl Mech Eng 197(6–8):487–504
9. Everdij FPX, Lloberas-Valls O, Rixen DJ, Simone A, Sluys LJ (2013) Domain decomposition and parallel direct solvers as an adaptive multiscale strategy for damage simulation in materials. In: Proceedings of the 22nd international conference on domain decomposition methods (DD22), Lugano, Switzerland
10. Farhat C (1990) Which parallel finite element algorithm for which architecture and which problem? Eng Comput 7(3):186–195
11. Farhat C, Roux FX (1991) A method of finite element tearing and interconnecting and its parallel solution algorithm. Int J Numer Methods Eng 32(6):1205–1227
12. Farhat C, Pierson K, Lesoinne M (2000) The second generation FETI methods and their application to the parallel solution of large-scale linear and geometrically non-linear structural analysis problems. Comput Methods Appl Mech Eng 184(2–4):333–374
13. Feyel F, Chaboche JL (2000) FE^2 multiscale approach for modelling the elastoviscoplastic behaviour of long fibre SiC/Ti composite materials. Comput Methods Appl Mech Eng 183(3–4):309–330
14. Flemisch B, Kaltenbacher M, Triebenbacher S, Wohlmuth B (2011) Non-matching grids for a flexible discretization in computational acoustics. Commun Comput Phys 11(2):472–488
15. Gendre L, Allix O, Gosselet P (2011) A two-scale approximation of the schur complement and its use for nonintrusive coupling. Int J Numer Methods Eng. doi:10.1002/nme.3142
16. Géradin M, Rixen DJ (2015) Mechanical vibrations: theory and applications, 3rd edn. Willey. ISBN 978-1-118-90020-8
17. Ghosh S, Bai J, Raghavan P (2007) Concurrent multi-level model for damage evolution in microstructurally debonding composites. Mech Mater 39(3):241–266
18. Girault V, Pencheva GV, Wheeler MF, Wildey TM (2009) Domain decomposition for linear elasticity with dg jumps and mortars. Computer Methods in Applied Mechanics and Engineering 198(2126):1751–1765
19. Gosselet P, Rey C (2006) Non-overlapping domain decomposition methods in structural mechanics. Arch Comput Methods Eng 13(4):515–572
20. Guidault PA, Allix O, Champaney L, Navarro JP (2007) A two-scale approach with homogenization for the computation of cracked structures. Comput Struct 85(17–18):1360–1371
21. Hartmann S, Oliver J, Weyler R, Cante JC, Hernández JA (2009) A contact domain method for large deformation frictional contact problems. Part 2: numerical aspects. Comput Methods Appl Mech Eng 198(3336):2607–2631
22. Hartmann S, Weyler R, Oliver J, Cante JC, Hernández JA (2010) A 3d frictionless contact domain method for large deformation problems. Comput Model Eng Sci 55(3):211–269
23. Heintz P, Hansbo P (2006) Stabilized lagrange multiplier methods for bilateral elastic contact with friction. Comput Methods Appl Mech Eng 195(3336):4323–4333
24. Herry B, Di Valentin L, Combescure A (2002) An approach to the connection between subdomains with non-matching meshes for transient mechanical analysis. Int J Numer Methods Eng 55(8):973–1003
25. Hughes TJR, Feijóo GR, Mazzei L, Quincy JB (1998) The variational multiscale method—a paradigm for computational mechanics. Comput Methods Appl Mech Eng 166(1–2):3–24
26. Hund A, Ramm E (2007) Locality constraints within multiscale model for non-linear material behaviour. Int J Numer Methods Eng 70(13):1613–1632
27. Kouznetsova V, Brekelmans WAM, Baaijens FPT (2001) An approach to micro-macro modeling of heterogeneous materials. Comput Mech 27(1):37–48
28. Ladevèze P, Loiseau O, Dureisseix D (2001) A micro-macro and parallel computational strategy for highly heterogeneous structures. Int J Numer Methods Eng 52(12):121–138
29. Lloberas-Valls O (2013) Multiscale domain decomposition analysis of quasi-brittle materials. Ph.D. Thesis, Delft University of Technology
30. Lloberas-Valls O, Rixen DJ, Simone A, Sluys LJ (2012) On micro-to-macro connections in domain decomposition multiscale methods. Comput Methods Appl Mech Eng 225–228:177–196
31. Lloberas-Valls O, Rixen DJ, Simone A, Sluys LJ (2012) Multiscale domain decomposition analysis of quasi-brittle heterogeneous materials. Int J Numer Methods Eng 89(11):1337–1366
32. Lloberas-Valls O, Everdij FPX, Rixen DJ, Simone A, Sluys LJ (2014) Multiscale analysis of damage using dual and primal domain decomposition techniques. In: Oñate E, Oliver X, Huerta A (eds) Proceedings of the 11th. World congress on computational mechanics—WCCM XI, Barcelona, Spain
33. Loehnert S, Prange C, Wriggers P (2012) Error controlled adaptive multiscale XFEM simulation of cracks. Int J Fract 178(1):147–156
34. MacPherson I, Rodgers J, Allen C, Fenwick C (2006) Sliding and non-matching grid methods for helicopter simulations. In: 44th AIAA aerospace sciences meeting and exhibit. American Institute of Aeronautics and Astronautics
35. McGee W, Seshaiyer P (2005) Non-conforming finite element methods for nonmatching grids in three dimensions. In: Domain

decomposition methods in science and engineering, number 40 in lecture notes in computational science and engineering. Springer, Berlin, pp 327–334

36. Mergheim J (2009) A variational multiscale method to model crack propagation at finite strains. Int J Numer Methods Eng 80(3):269–289
37. Nitsche J (1971) Über ein variationsprinzip zur lösung von dirichlet-problemen bei verwendung von teilräumen, die keinen randbedingungen unterworfen sind. Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg 36(1):9–15
38. Oliver J, Hartmann S, Cante JC, Weyler R, Hernández JA (2009) A contact domain method for large deformation frictional contact problems. Part 1: theoretical basis. Comput Methods Appl Mech Eng 198(3336):2591–2606
39. Park KC, Felippa CA, Rebel G (2002) A simple algorithm for localized construction of non-matching structural interfaces. Int J Numer Methods Eng 53(9):2117–2142
40. Popp A, Wohlmuth BI, Gee MW, Wall WA (2012) Dual quadratic mortar finite element methods for 3d finite deformation contact. SIAM J Sci Comput 34(4):B421–B446
41. Popp A, Gee MW, Wall WA (2013) Mortar methods for single- and multi-field applications in computational mechanics. In: Resch M, Wang X, Bez W, Focht E, Kobayashi H (eds) Sustained simulation performance 2012. Springer, Berlin, pp 133–154
42. Puso MA (2004) A 3d mortar method for solid mechanics. Int J Numer Methods Eng 59(3):315–336
43. Puso MA, Laursen TA (2004) A mortar segment-to-segment contact method for large deformation solid mechanics. Comput Methods Appl Mech Eng 193(68):601–629
44. Quiroz L, Beckers P (1995) Non-conforming mesh gluing in the finite elements method. Int J Numer Methods Eng 38(13):2165–2184
45. Rixen DJ (1997) Substructuring and dual methods in structural analysis. PhD thesis, Publications de la Faculté des Sciences Appliquées, no. 175, Université de Liège, Belgium
46. Rixen DJ (2001) Parallel processing. In: Encyclopedia of vibration. Elsevier, Oxford, pp 990–1001
47. Rixen DJ, Farhat C (1999) A simple and efficient extension of a class of substructure based preconditioners to heterogeneous structural mechanics problems. Int J Numer Methods Eng 44(4):489–516
48. Simo JC, Hughes TJ (1998) Computational inelasticity, volume 7 of interdisciplinary applied mathematics. Springer, New York
49. Spillane N, Rixen DJ (2013) Automatic spectral coarse spaces for robust FETI and BDD algorithms. Int J Numer Methods Eng 95(11):953–990
50. Wohlmuth BI (2000) A mortar finite element method using dual spaces for the lagrange multiplier. SIAM J Numer Anal 38(3):989–1012
51. Wriggers P (1995) Finite element algorithms for contact problems. Arch Comput Methods Eng 2(4):1–49