A stable and optimally convergent LaTIn-CutFEM algorithm for multiple unilateral contact problems

S. Claus[1*], P. Kerfriden[1]

[1] Cardiff University, School of Engineering,
The Parade, CF243AA Cardiff, United Kingdom

Abstract

In this paper, we propose a novel unfitted finite element method for the simulation of multiple body contact. The computational mesh is generated independently of the geometry of the interacting solids, which can be arbitrarily complex. The key novelty of the approach is the combination of elements of the CutFEM technology, namely the enrichment of the solution field via the definition of overlapping fictitious domains with a dedicated penalty-type regularisation of discrete operators, and the LaTIn hybrid-mixed formulation of complex interface conditions. Furthermore, the novel P1-P1 discretisation scheme that we propose for the unfitted LaTIn solver is shown to be stable, robust and optimally convergent with mesh refinement. Finally, the paper introduces a high-performance 3D level-set/CutFEM framework for the versatile and robust solution of contact problems involving multiple bodies of complex geometries, with more than two bodies interacting at a single point.

keywords: unilateral contact, LaTIn, nonconforming finite element, CutFEM, ghost penalty, multiple level sets, composite materials

1 Introduction

Unfitted or non-conforming finite element methods uncouple the description of the geometry from the representation of the solution field itself. Typically, the geometry of the computational domain is projected over a regular background grid. In this setting, boundaries or interfaces between objects cut through elements of the corresponding mesh. Non-conforming methods are attractive for applications where coupling analysis codes and third party meshing libraries is either impractical, numerically expensive and/or prone to errors. In particular, contact problems in engineering applications often involve a large assembly of interacting solids of complex geometry. These assemblies can be extremely challenging to mesh due to sharp angles or small gaps between assembly parts. Here, the decoupling of the geometry from the finite element mesh, as featured within non-conforming finite element frameworks, can help alleviate this meshing burden. Nonetheless, a number of specific challenges needs to be addressed for unfitted numerical solutions to be computable. Firstly, integrals need to be calculated over cut elements, which requires specialised numerical quadratures. Secondly, unfitted approaches require stabilisation. This is because combinations of degrees of freedom may be poorly controlled in regions where contributions from cut elements to integral forms are small. Finally, in the context of multiple interacting materials, enrichment of the finite element solution space is required in order to allow for numerical jumps or kinks to develop over embedded interfaces. Failure to do so may severely impair the convergence rate of the unfitted finite element solver.

Over the last two decades, several encompassing frameworks have been developed to provide guidance for the development of unfitted finite element solvers. The eXtended Finite Element Method (XFEM) [36, 18, 7, 21] relies on the Partition of Unity Method [34] to enrich the approximation space. Within this popular framework, non-conforming discontinuities can be introduced in an elegant and robust manner. Field discontinuities can

---

*clauss2@cardiff.ac.uk

also be enabled by making use of Embedded Discontinuity Methods [39, 37, 31], whereby discontinuous shape functions are directly defined at the element level. The CutFEM framework [22, 10], which is fundamental to the developments reported in this paper, uses an alternative concept to introduce non-conforming field discontinuities, and to perform computations involving non-conforming geometrical domains. In the context of multiple interacting materials, CutFEM solvers enrich the solution space by overlaying (*i.e.* doubling or tripling) cut elements. Subsequently, penalty techniques such as the Nitsche method are employed to disallow unwanted discontinuities, for instance to represent jumps in fluxes whilst preserving the continuity of the primal field [22]. CutFEM solvers are usually equipped with a ghost-penalty stabilisation technique [9], which results in optimal convergence rates whilst ensuring that the condition number of the system matrix degrades at the same rate as that of the conforming finite element method when the mesh is refined.

Extending unfitted finite element methods to nonlinear problems, and especially to problems involving complex interface behaviour, remains a challenge that needs to be addressed on a case-by-case basis. The present paper focuses on problems involving unilateral contact between elastic solids. This class of problems is important to engineers, and they are typically encountered when modelling joints between components in large-scale engineering assemblies. Numerically, unilateral contact is notoriously difficulty to treat, even in the context of fitted finite element approaches. References on the matter include [45, 30, 42, 1, 33]. Of particular interest to us are LaTIn-based solvers [13, 28, 8, 32, 29, 2], which treat unilateral contact through a two-field formulation of the interface equations, and resolve these fields iteratively using a dedicated two-stage solver. A number of researchers have demonstrated the versatility of LaTIn solvers, owing to the fact that the predictor stage (so-called "linear stage"), is weakly dependent on the type of interface conditions, and completely independent of the current interface state. As a result, switching between nonlinear interface conditions is relatively painless [24]. Moreover, dedicated space-time Model Order Reduction techniques may take advantage of the predictor invariance to reduce the cost of time-dependent or multi-query computations [29, 24, 19]. However, a noteworthy issue related to the hybrid-mixed LaTIn formulation is the difficulty of discretising displacement and traction interface fields in a consistent manner. Whilst a piecewise constant discretisation of interface fields seems to be physically sound when associated with P1 finite element spaces for the bulk equations, such a discretisation scheme allows for the onset of numerically uncontrolled deformation modes whose associated amplitude creep up with increasing LaTIn iteration count, and eventually alter the convergence rate of the finite element solver. A claimed remedy to this numerical plague is to introduce additional degrees of freedom for the bulk fields, by means of either h-refinement or p-refinement. An alternative to this numerically expensive procedure has recently been proposed in [20], as discussed later on. Although both approaches seem to regularise the contact problem efficiently, the effect on the convergence of the mixed finite element solver with mesh refinement has not been thoroughly studied.

Over the years, several attempts to develop stable unfitted finite element solvers for unilateral contact have been met with success. XFEM-based solutions have been proposed in [16, 25, 40, 17, 32, 20, 38]. Of particular importance to us are LaTIn-based unfitted solvers, as the LaTIn is also a fundamental technological element for the present contribution. The first of these solvers was proposed in [16] to solve crack propagation problems (*i.e.* the historically favoured playground for XFEM). In [40], the XFEM-LaTIn idea was extended in order to simulate fatigue crack propagation. In [32], the authors compared a LaTIn formulation to a penalty/Newton approach, and claimed the superiority of the latter in terms of convergence rate and stability. Finally, the authors of [20] proposed a regularised, LaTIn-type formulation of contact for XFEM. The main idea of the approach is to relax the kinematic continuity condition between the bulk primal field and its interface trace and to enforce it weakly via a regularised mortar method.

In contrast to XFEM-based developments, the CutFEM framework is relatively underdeveloped in the context of unfitted unilateral contact. We mention the recent work described in [14, 15], which extends the Nitsche-type treatment of interface conditions that is usually employed in CutFEM solvers to the case of unilateral contact. The resulting fitted finite element solver is proven to be optimally convergent, stable and numerically efficient. A very first nonconforming CutFEM strategy for unilateral contact is discussed in [11], whereby the contact conditions are treated by an augmented Lagrange multiplier formulation, and the discrete equations are stabilised by a ghost-penalty regularisation.

The present paper proposes a novel LaTIn-CutFEM solver for unilateral contact between multiple interacting solid bodies. The solid bodies will be described by linearised elasticity, which implies that the location of

contact interfaces is known *a priori*. Our algorithm relies on the traditional CutFEM enrichment strategy [22]. However, contact conditions over nonconforming interfaces will be enforced via the LaTIn method. The resulting algorithm is a two-stage solver, whereby the linear stage (*i.e.* linear predictor) consists in solving a series of independent linear problems for each of the solid bodies, whilst the local stage (*i.e.* interface corrector) is nonlinear. The method naturally inherits the coarse-grain parallel characteristics of LaTIn-based domain decomposition methods [27, 28, 24], together with the previously mentioned versatility of LaTIn solvers with regards to the type of interface conditions being dealt with (*i.e.* unilateral contact with Coulomb friction [8, 12], frictional sliding [5, 6], cohesive fracture models with transition to unilateral frictionless contact [24]).

Two algorithmic elements of the proposed solver require particular care. Firstly, the LaTIn-CutFEM scheme needs to be stabilised. We regularise each of the subproblems of the linear stage using a ghost penalty technique [9]. We will show that, when using the appropriate scaling for the ghost penalty terms, the condition number of the system matrices "degrades optimally" with mesh refinement (*i.e.* it degrades at the same rate as that of the conforming finite element method). Secondly, we need to be careful when discretising the contact interface fields. As mentioned before, a naive discretisation of the LaTIn hybrid-mixed formulation leads to the onset of spurious interface solutions. We will show that the P1-P1 discretisation scheme that we propose (*i.e.* P1 for the bulk primal field, P1 for both its trace and the trace of its normal flux) yields optimal convergence rates with mesh refinement, and eliminates the onset of instabilities. Our scheme "nonlocalises" the local stage of the LaTIn, as it relies on a two-scale treatment of the interface fields and nonlinear equations. We will also show that the nonlinear solver of the LaTIn converges fast, in the sense that it quickly leads to iterates whereby the discretisation error is larger than the algorithmic error, typically within twenty iterations. To our best knowledge, these convergence results have never been reported in the context of LaTIn solvers, even in the context of conforming geometries.

Our formal developments are accompanied by a high-performance computer implementation. The core of our implementation is the finite element C++/Python library FEniCS [4], which, in particular, proposes a range of high-level tools to rapidly developed finite element solvers. The CutFEM C++ library, LibCutFEM, developed by S. Claus and A. Massing, and partially described in [10], defines additional tools that are specific to unfitted finite elements. This library forms the basis for the LaTIn-CutFEM code. We will present results in two and three space dimensions, which is made seamless by the FEniCS framework. Some of our examples exhibit a high level of geometric complexity, which is handled through a robust implementation of multiple level sets. In particular, we will study the convergence of contact problems exhibiting multiple interacting bodies (*i.e.* more than two) at a single point, (see other contributions on the topic, in a different context, in [5, 6]).

This paper is organised as follows. The LaTIn hybrid-mixed formulation of contact is presented in a continuous setting in Section 2, together with the associated two-stage nonlinear algorithm. The non-conforming regularised LaTIn-CutFEM formulation is presented in Section 3. Section 4 is dedicated to numerical investigations, where we demonstrate the optimal convergence and stability properties of the proposed framework. We also showcase the versatility of our high-performance computer implementation.

2 LaTIn method for unilateral contact: the continuous setting

2.1 Domain decomposition

We consider a problem of linear elasticity over a 2 or 3-dimensional spatial domain Ω composed of n_{d} non-overlapping subdomains Ω^i, where $i \in \mathcal{I}_\Omega := \{1, \ldots, n_{\mathrm{d}}\}$. The domain decomposition satisfies $\bigcup_{i \in \mathcal{I}_\Omega} \Omega^i = \Omega$. Let us denote the set of all subdomains by Ξ_Ω. Subdomains may be multiply-connected. The boundary of subdomain Ω^i is denoted by $\partial \Omega^i$, for any $i \in \mathcal{I}_\Omega$, whilst $\partial \Omega$ denotes the boundary of computational domain Ω. Furthermore, we define interface domains $\Gamma^{i,j} := \partial \Omega^i \cap \partial \Omega^j$ for any $(i,j) \in \mathcal{I}_\Omega \times \mathcal{I}_\Omega$ such that the two indexes satisfy $i < j$ and the result of intersection $\partial \Omega^i \cap \partial \Omega^j$ is of non-zero measure. The union of these interfaces is denoted by Γ. The set of pairs (i,j) of indexes such that $\Gamma^{i,j}$ is defined as a valid interface is denoted by $\mathcal{I}_\Gamma$, and the corresponding set of interfaces is denotes by Ξ_Γ. We also denote the union of all the interfaces of subdomain Ω^i by $\Gamma^i = \bigcup_{(i,j) \in \mathcal{I}_\Gamma, (k=i \text{ or } l=i)} \Gamma^{k,l}$, for any subdomain index $i \in \mathcal{I}_\Omega$. We denote the outward normal pointing from Ω^i to Ω^j on $\Gamma^{i,j}$ by $\underline{n}^{i,j}$. We further note that we have $\underline{n}^{i,j} = -\underline{n}^{j,i}$, and we denote the outward normal to Ω^i over Γ^i by $\underline{n}^i$. As an illustration for this set of definitions, consider the case of the multiphase
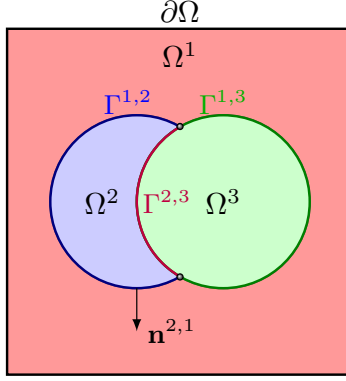
Figure 1: Schematic representation of a composite material made of three distinct phases.

domain depicted in Figure 1. The cubic domain Ω comprises three subdomains Ω^1, Ω^2 and Ω^3. There are three interfaces between the domains: $\Gamma^{1,2}$, $\Gamma^{1,3}$ and $\Gamma^{2,3}$. Therefore, the interface of individual subdomains reads as $\Gamma^1 = \Gamma^{1,2} \cup \Gamma^{1,3}$, $\Gamma^2 = \Gamma^{1,2} \cup \Gamma^{2,3}$ and $\Gamma^3 = \Gamma^{1,3} \cup \Gamma^{2,3}$.

As further detailed in the next subsection, subdomain field variables will be governed by the equations of linear elasticity, while contact equations to be solved over interfaces will couple subdomain problems to one another. The LaTIn framework [26] is particularly well suited to the treatment of such problems where complex nonlinear equations need to be solved over embedded interfaces [8, 29, 3]. In this particular context, the LaTIn method relies on two main ingredients:

- a hybrid description of bulk and interface fields, and a mixed representation of the interface unknowns,

- a two-step iterative solution algorithm for the solution of the global nonlinear problem formulated in its hybrid-mixed form.

Both these steps are described below.

2.2 Hybrid-Mixed LaTIn formulation of unilateral contact problems

Hybrid fields For every subdomain Ω^i, where $i \in \mathcal{I}_\Omega$, we introduce a displacement field $\underline{u}^i : \Omega^i \to \mathbb{R}^d$, where $d = 2$ (2D problem) or $d = 3$ (3D problem).
For every domain interface Γ^i with $i \in \mathcal{I}_\Omega$, we define two field variables:

- a force field $\underline{F}^i : \Gamma^i \to \mathbb{R}^d$ that represents a density of tractions (*i.e.* flux) applied to subdomain i by adjacent subdomains through interface Γ^i.

- a displacement field $\underline{W}^i : \Gamma^i \to \mathbb{R}^d$ that represents the trace of $\underline{u}^i$ over Γ^i.

As detailed below, $\underline{u}^i$ will be governed by linear elasticity, while over an interface $\Gamma^{i,j}$ between two domains Ω^i and Ω^j, where $(i,j) \in \mathcal{I}_\Gamma$, interface models will provide a link between the restriction of fields $\underline{F}^i$, $\underline{W}^i$ to $\Gamma^{i,j}$ and the restriction of fields $\underline{F}^j$, $\underline{W}^j$ to $\Gamma^{i,j}$.

Bulk equations and boundary data For every subdomain Ω^i, $i \in \mathcal{I}_\Omega$, the displacement field $\underline{u}^i \in [H^1(\Omega^i)]^d$ is required to be sufficiently regular (continuous in particular), and to satisfy the following equations:

- Static equilibrium equation and constitutive relation:
  For all $\delta \underline{u}^i \in [H^1(\Omega^i)]^d$, the displacement field $\underline{u}^i \in [H^1(\Omega^i)]^d$ satisfies

$$\int_{\Omega^i} \underline{\underline{\sigma}}(\underline{u}^i; \underset{\sim}{D}^i) : \underline{\underline{\epsilon}}(\delta \underline{u}^i) \, d\Omega = \int_{\Omega^i} \underline{f} \cdot \delta \underline{u}^i \, d\Omega + \int_{\partial \Omega^i} \left(\underline{\underline{\sigma}}(\underline{u}^i; \underset{\sim}{D}^i) \cdot \underline{n}^i \right) \cdot \delta \underline{u}^i \, d\Gamma. \tag{1}$$

In equation (1), $\underline{f}$ is a body force and $\underline{n}^i$ is the outer normal of domain Ω^i. The stress function is defined as $\underline{\underline{\sigma}}(\underline{u}^i; \underset{\sim}{D}^i) := \underset{\sim}{D}^i : \underline{\underline{\epsilon}}(\underline{u}^i)$, where the strain is the symmetric part of the displacement gradient, *i.e.*

4

$\underline{\underline{\epsilon}}(\underline{u}^i) := \frac{1}{2}\left(\underline{\underline{\mathrm{grad}}}\,\underline{u}^i + (\underline{\underline{\mathrm{grad}}}\,\underline{u}^i)^T\right)$ and $\underline{D}^i$ is a fourth-order Hooke tensor. In the case of isotropic elasticity, we have that $\underline{\underline{D}}^i : \underline{\underline{\epsilon}}(\underline{u}^i) = \lambda^i \,\mathrm{Tr}(\underline{\underline{\epsilon}}(\underline{u}^i))\,\underline{\underline{I}}_d + 2\,\mu^i\,\underline{\underline{\epsilon}}(\underline{u}^i)$, where λ^i and μ^i are the two Lamé coefficients. These two coefficients are related to the Poisson and Young's modulii, respectively denoted by ν and E^i, through $\lambda^i = \frac{E^i \nu}{(1+\nu)(1-2\nu)}$ and $\mu^i = \frac{E^i}{2(1+\nu)}$. All these coefficients are assumed to be constant in Ω^i. The dependency of the stress function to the Hooke tensor field will be omitted when possible. However, remember that the Hooke tensor may depend on the spatial coordinates. In particular, it may be discontinuous across an interface between connected subdomains. Such cases will be treated in the numerical example section.

- Boundary Conditions:
  We shall assume that the boundary $\partial\Omega$ of domain Ω is decomposed into a Neumann part and a Dirichlet part, *i.e.* $\partial\Omega = \partial\Omega^{\mathrm{t}} \cup \partial\Omega^{\mathrm{u}}$. For Neumann boundary conditions, we specify that

$$\underline{\underline{\sigma}}(\underline{u}^i) \cdot \underline{n}^i = \underline{T} \qquad \text{over } \partial\Omega^{\mathrm{t}} \cap \partial\Omega^i\,, \tag{2}$$

  whilst for Dirichlet boundary conditions, we set

$$\underline{u}^i = \underline{U} \qquad \text{over } \partial\Omega^{\mathrm{u}} \cap \partial\Omega^i\,. \tag{3}$$

  Here, $\underline{T}$ is a density of prescribed tractions per unit of surface, $\underline{U}$ is a field of prescribed displacement.

Interface governing equations Any interface $\Gamma^{i,j}$, where $(i,j) \in \mathcal{I}_\Gamma$, will exhibit behaviour relating the surface tractions to the displacement fields. Although extensions are relatively straightforward, we consider only the case of frictionless unilateral contact, which is characterised by the following equations:

- Newton's third law,
$$\underline{F}^i + \underline{F}^j = \underline{0}\,, \tag{4}$$

- Signorini's law of unilateral contact, which reads as
$$\begin{aligned}
\left(\underline{W}^j - \underline{W}^i\right) \cdot \underline{n}^i &\geq 0\,, \\
\underline{F}^i \cdot \underline{n}^i &\leq 0\,, \\
\left(\left(\underline{W}^j - \underline{W}^i\right) \cdot \underline{n}^i\right) \cdot \left(\underline{F}^i \cdot \underline{n}^i\right) &= 0\,,
\end{aligned} \tag{5}$$

- the condition that the tangential component of tractions must vanish
$$\left(\underline{\underline{I}}_d - \underline{n}^i \otimes \underline{n}^i\right) \cdot \underline{F}^i = \underline{0}\,. \tag{6}$$

Interface compatibility conditions for the hybrid-mixed formulation In order to close the system of equations, the interface fields must be linked to the bulk variables. This is done by enforcing the continuity condition

$$\underline{F}^i = \sigma(\underline{u}^i) \cdot \underline{n}^i\,, \tag{7}$$

together with the kinematic condition

$$\underline{W}^i = \underline{u}^i \tag{8}$$

over all interfaces Γ^i, with $i \in \mathcal{I}_\Omega$.

2.3 Nonlinear iterative solver

We employ the standard LaTIn algorithm to solve the contact problem written in its hybrid-mixed form. The LaTIn algorithm consists in searching for mixed interface fields $(\underline{F}^i, \underline{W}^i)$ in two stages (see Figure 2). In the so-called linear stage, we search for interface fields that satisfy the bulk equations and the interface compatibility equations. The resulting problem is global, at least over each individual subdomain, but remains linear (similar to a predictor in a Newton algorithm). In the so-called local stage, we seek interface fields that satisfy the interface contact laws. The resulting problems are potentially non-linear (if contact is unilateral) but they are local *a priori* (this is similar to an nonlinear update in a Newton solver). As both primal and dual interface fields are unknown, search directions must be defined to close each of these two problems. Details about the algorithm are provided below.
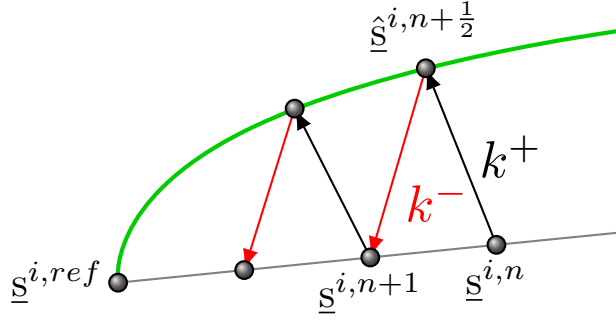
Figure 2: Schematic of LaTIn algorithm.

Local Stage. For all Γ^i with $i \in \mathcal{I}_\Omega$, we search for interface fields $\hat{\underline{s}}^{i,n+\frac{1}{2}} := (\widehat{\underline{F}}^{i,n+\frac{1}{2}}, \widehat{\underline{W}}^{i,n+\frac{1}{2}})$ satisfying unilateral contact equations (4)-(6), closed by the ascent search direction

$$\left(\widehat{\underline{F}}^{i,n+\frac{1}{2}} - \underline{F}^{i,n} \right) - k^+ \left(\widehat{\underline{W}}^{i,n+\frac{1}{2}} - \underline{W}^{i,n} \right) = \underline{0} \,. \tag{9}$$

Here, $n \in \mathbb{R}^+$ denotes the iteration index, $\cdot^{n+\frac{1}{2}}$ denotes a quantity predicted by the $(n+1)-$th local stage of the algorithm (*i.e.* half iterate). The search direction "stiffness" k^+ is a strictly positive scalar parameter.

Linear Stage. For all Γ^i with $i \in \mathcal{I}_\Omega$, we search for interface fields $\underline{s}^{i,n+1} := (\underline{F}^{i,n+1}, \underline{W}^{i,n+1})$ satisfying the interface compatibility conditions (7) - (8) and the subdomain equations (1) - (3), closed by the descent search direction

$$\left(\underline{F}^{i,n+1} - \widehat{\underline{F}}^{i,n+\frac{1}{2}} \right) + k^- \left(\underline{W}^{i,n+1} - \widehat{\underline{W}}^{i,n+\frac{1}{2}} \right) = \underline{0} \,. \tag{10}$$

We choose $k^- = k^+$, which is the classical setting of the LaTIn algorithm (conjugate search directions [26]). We further set $k^- = \xi \frac{E}{L}$, where ξ is a non-dimensional strictly positive parameter (set to 1 unless stated otherwise), E is a reference Young's modulus (set to be the Young modulus of the matrix phase, *i.e.* of the composite materials that we will analyse in all the examples of the numerical section), and L is a reference length of the problem.

In a variational setting, and dropping iteration index n, the linear stage can be written as a set of linear problems that read: *For any $i \in \mathcal{I}_\Omega$, find $\underline{u}^i \in [H^1(\Omega^i)]^d$ and compatible with the Dirichlet conditions such that for all $\delta \underline{u}^i \in [H_0^1(\Omega^i)]^d$*

$$a_D^i(\underline{u}^i, \delta\underline{u}^i) + a_k^i(\underline{u}^i, \delta\underline{u}^i) = l_f^i(\delta\underline{u}^i) + l_k^i(\delta\underline{u}^i) \,, \tag{11}$$

where the bilinear forms a_D^i and a_k^i are given by

$$a_D^i(\underline{u}^i, \delta\underline{u}^i) = \int_{\Omega^i} \underline{\underline{\epsilon}}(\underline{u}^i) : \underset{\sim}{D}^i : \underline{\underline{\epsilon}}(\delta\underline{u}^i) \, d\Omega \tag{12}$$

$$a_k^i(\underline{u}^i, \delta\underline{u}^i) = \int_{\Gamma^i} k^- \, \underline{u}^i \cdot \delta\underline{u}^i \, d\Gamma \,, \tag{13}$$

and the linear forms l_f^i and l_k^i are given by

$$l_f^i(\delta\underline{u}^i) = \int_{\Omega^i} \underline{f} \cdot \delta\underline{u}^i \, d\Omega + \int_{\partial\Omega^i \cap \partial\Omega^t} \underline{T}^i \cdot \delta\underline{u}^i \, d\Gamma \,, \tag{14}$$

$$l_k^i(\delta\underline{u}^i) = \int_{\Gamma^i} \left(\widehat{\underline{F}}^i + k^- \, \widehat{\underline{W}}^i \right) \cdot \delta\underline{u}^i \, d\Gamma \,. \tag{15}$$

Remark. *Problem (11) is always well-posed (at least at the continuous level), owing to the effect of the LaTIn augmentation terms a_k^i, which ensure that the bilinear forms a^i remain strictly coercive.*

6

(a) Level set description.

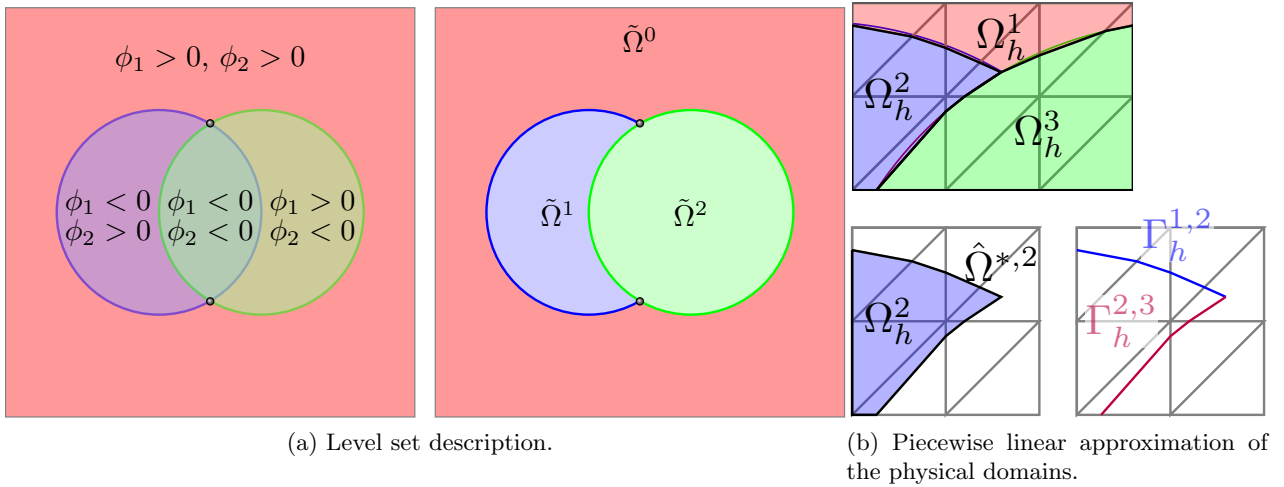(b) Piecewise linear approximation of the physical domains.

Figure 3: Schematic of multiple level set description of domain and its piecewise linear approximation.

Convergence. A relaxation step is performed at the end of the linear stage to control the convergence properties of the algorithm (the interested reader is referred to [26] for proofs of convergence of the algorithm). The relaxation step reads as

$$\begin{cases} \underline{F}^{i,n+1} \leftarrow \eta\, \underline{F}^{i,n+1} + (1-\eta)\, \underline{F}^{i,n}, \\ \underline{W}^{i,n+1} \leftarrow \eta\, \underline{W}^{i,n+1} + (1-\eta)\, \underline{W}^{i,n} \end{cases} \tag{16}$$

for all $i \in \mathcal{I}_\Omega$. The robustness of the algorithm is not very sensitive to the choice of relaxation parameter $\eta \in [0\,1]$, but the convergence rate is. Based on our experience of the method, we set $\eta = 0.85$.

3 LaTin-CutFEM solver for embedded unilateral contact problems

We present the proposed unfitted finite element strategy by firstly detailing the method that we use to describe the geometry of the subdomains. And secondly, we explain how the LaTIn hybrid-mixed formulation of unilateral contact problems are discretised and stabilised in the context of unfitted unilateral contact problems.

3.1 Description and approximation of the geometry

The geometry of the contact problems will be described independently of the background mesh using multiple level set functions whose zero level sets describe the interface locations (see similar applications of multiple level set approaches in [35, 43, 41, 44], amongst other contributions). Our general strategy is to describe each subdomain of our problem as the union of auxiliary subdomains that can be easily defined using level set functions. These functions can be given in an analytical form or in a numerical form. This approach is well suited to the description of contact problems with nonconforming embedded interfaces and/or nonconforming external boundaries. We will simply interpolate our level set functions in the finite element space, which will automatically yield an analysis suitable, piecewise polynomial approximation of the geometry.

3.1.1 Formal domain decomposition using multiple level sets

Let $\Omega^\square$ be a domain that covers the physical domain Ω, *i.e.* $\Omega \subset \Omega^\square$, and let us decompose $\Omega^\square$ into $n_d^\square$ non-overlapping subdomains $\tilde{\Omega}^i$ with $n_d^\square \geq n_d$. $\Omega^\square$ will be meshed later on, and we therefore require that its geometry is of simple nature. Furthermore, let us assume that any physical subdomain Ω^i is exactly represented by one of the subdomains $\tilde{\Omega}^i$ or by the union of several of these auxiliary subdomains.

The decomposition of $\Omega^\square$ into subdomains $\tilde{\Omega}^i$ is defined through a multiple level set approach. We employ $n_d^\square - 1$ level set functions $\phi^i : \mathbb{R}^d \to \mathbb{R}$, $i \in \{1, \dots n_d^\square - 1\}$ to describe the $n_d^\square$ subdomains $\tilde{\Omega}^i$. This is done as follows. Let $I_{\phi^-}(\boldsymbol{x})$ denote the set of indices of level set functions, which are smaller than zero in $\boldsymbol{x}$, i.e.

$I_{\phi^-}(\boldsymbol{x}) = \{i \in \{1, \ldots, n_d^\square - 1\} : \phi_i(\boldsymbol{x}) < 0\}$. Then for each point $\boldsymbol{x} \in \Omega^\square$, the domain that $\boldsymbol{x}$ belongs to is defined as

$$\boldsymbol{x} \in \begin{cases} \tilde{\Omega}^0 & \text{if } I_{\phi^-} = \emptyset, \\ \tilde{\Omega}^i, \ i = \max(I_{\phi^-}) & \text{else.} \end{cases} \tag{17}$$

This definition implies that there exists a hierarchical relationship between level sets. Indeed, the domain defined by the region of negative level set values of level set ϕ^i overlaps all the domains defined by the negative regions of level sets ϕ^j with $j < i$. Also, domain $\tilde{\Omega}^0$ is defined as the complement to all other domains $\tilde{\Omega}^i$, i.e. $\tilde{\Omega}^0 := \Omega^\square \setminus \bigcup_{i \in I_{\tilde{\Omega}}} \tilde{\Omega}^i$. In our numerical examples, domain $\tilde{\Omega}^0$ represents the matrix phase of a composite material. Figure 3a shows a graphical representation of how the domains are described using multiple level sets.

We now define the interface between $\tilde{\Omega}^i$ and $\tilde{\Omega}^j$, $\tilde{\Gamma}^{i,j}$ $(i < j)$, as

$$\begin{aligned} \tilde{\Gamma}^{i,j} &:= \{\boldsymbol{x} \in \Omega^\square : \phi^j(\boldsymbol{x}) = 0 \text{ and } \phi^i(\boldsymbol{x}) < 0\} && \text{for } i \in \{1, \ldots, n_d^\square - 1\}, \\ \tilde{\Gamma}^{0,j} &:= \{\boldsymbol{x} \in \Omega^\square : \phi^j(\boldsymbol{x}) = 0 \text{ and } \phi^k(\boldsymbol{x}) > 0 \ \forall k \neq j\}. \end{aligned} \tag{18}$$

Finally, we can formally map the subdomains of $\Omega^\square$ onto that of Ω with the mapping

$$\Omega^i = \mathcal{M}_d^i \left(\{\tilde{\Omega}^j\}_{j \in [\![0, n_d^\square - 1]\!]} \right) = \bigcup_{j \in \tilde{\mathcal{I}}^i \subset [\![0, n_d^\square - 1]\!],} \tilde{\Omega}^j. \tag{19}$$

This mapping means that each of the physical domains Ω^i is defined as the union of particular subsets of subdomains of $\{\tilde{\Omega}^i\}$. Importantly, the union of subdomains $\tilde{\Omega}^i$ that are included in $\bar{\Omega} := \Omega^\square \setminus \Omega$ are void, i.e. there is no partial differential equation written over $\bar{\Omega}$. In fact, $\bar{\Omega} \neq \emptyset$ means that the boundary of the domain will be treated in an implicit way, while $\bar{\Omega} = \emptyset$ means that the boundary $\partial \Omega$ is meshed explicitly.

The interface sets $\tilde{\Gamma}^i$ and $\tilde{\Gamma}^{i,j}$ are completely described by the definition of their counterpart Γ^i and $\Gamma^{i,j}$ and the definition of the mapping of $\tilde{\Omega}^i$. However, it should be pointed out that interfaces between subdomains of $\bar{\Omega}$ and subdomains of $\Omega^\square$ that also belong to Ω are implicitly defined boundaries. For these interfaces, the mapping can be written as

$$\Gamma^{i,j} = \bigcup_{k,l \ | \ \tilde{\Omega}^k \in \left(\mathcal{M}_d^k\right)^{-1}(\Omega^i), \tilde{\Omega}^l \in \left(\mathcal{M}_d^l\right)^{-1}(\Omega^j)} \tilde{\Gamma}^{k,l}, \tag{20}$$

$$\partial \Omega = \bigcup_{\substack{k,l \ | \ \tilde{\Omega}^k \in \bar{\Omega}, \tilde{\Omega}^l \in \left(\mathcal{M}_d^l\right)^{-1}(\Omega^j) \\ \text{or } \tilde{\Omega}^l \in \bar{\Omega}, \tilde{\Omega}^k \in \left(\mathcal{M}_d^k\right)^{-1}(\Omega^i)}} \tilde{\Gamma}^{k,l}, \tag{21}$$

i.e. any interface between two physical domains is a physical interface and anything between a physical domain and a "void" domain is a physical boundary.

This apparently complex formalism allows us to easily and generically define our approximate geometry by simply discretising all our level sets, as explained in the following.

3.1.2 Discretisation of the level set functions and approximate geometry

In this section, we describe the approximation of the physical subdomains Ω^i and their boundaries and interfaces. Let $\mathcal{T}_h$ denote a tessellation of our background domain $\Omega^\square$. Now, lets consider a piecewise linear approximation of our level set functions ϕ_h^i resulting in a piecewise linear (in 2D) or piecewise planar (in 3D) interface approximation $\tilde{\Gamma}_h^{i,j}$. This yields piecewise linear approximations of Ω^i, Ω_h^i, and $\Gamma^{i,j}$, $\Gamma_h^{i,j}$, as illustrated in Figure 3b.

To ensure that the background mesh is fine enough to resolve the interfaces sufficiently, we impose the condition on the background mesh that each $\Gamma_h^{i,j}$ intersects any face of each element $K \in G^i$ at most once and that the mesh is quasi-uniform.

Note that the integrals of the discretised weak formulation introduced in the next section are evaluated over Ω_h^i and $\Gamma_h^{i,j}$, which includes the evaluation of integrals over arbitrary shaped areas in interface regions and integrals over linear/planar interface parts. These integrals over the areas $K_i = K \cap \Omega_h^i$ are performed using quadrature rules which are generated from a standard sub-triangulation as detailed in e.g. [10].

3.2 Mixed Finite element formulation

Using the piecewise polynomial description of our physical subdomains, we can now formulate our LaTIn-CutFEM method. The key features, which make our method stable and optimally convergent are:

- a representation of both primal and dual interface quantities in the trace of the finite element space defined over the subdomains. That means, we develop a stable hybrid P1-P1 scheme, *i.e.* we employ a piecewise linear approximation of the displacement field in the bulk of the subdomains, and piecewise linear approximations of the displacement and force fields over interfaces between subdomains;

- the regularisation of the equations of elasticity over each of the subdomains using the ghost-penalty methodology;

- a weakened enforcement of the contact equation through a stabilised two-scale representation of the interface fields.

3.2.1 Bulk and interface finite element spaces

Fictitious bulk finite element spaces. We integrate the equations of linear elasticity over each subdomain Ω_h^i (see Figure 3b) in our weak finite element formulation. In the case of small overlaps of the physical domain Ω_h^i with an element K, i.e. $K_i = \Omega_h^i \cap K$ small, the resulting finite element system matrix can become ill-conditioned. To circumvent this problem, we will solve the problem of linear elasticity on an extension of the physical domain (the so-called fictitious domains) instead of only on the domain Ω_h^i. More precisely, these fictitious domains are local background meshes for each of the subdomains $i \in \mathcal{I}_\Omega$ defined as the set of elements in the background mesh $\mathcal{T}_h$ that are fully or partially covered by the physical domain Ω_h^i, *i.e.*

$$\widehat{\Omega}^{*,i} := \{K \in \mathcal{T}_h : K \cap \Omega_h^i \neq \emptyset\}. \tag{22}$$

Figure 4 illustrates the three fictitious domains for the three phase particulate composite example mentioned in the previous section.

To make this extension from the physical domain Ω_h^i onto the fictitious domain $\widehat{\Omega}^{*,i}$ without loosing weak consistency, we will employ a regularisation technique on the set of elements intersected by interface Γ_h^i. We denote this set of elements by

$$G^i := \{K \in \mathcal{T}_h : K \cap \Gamma_h^i \neq \emptyset\} \tag{23}$$

and define a set of element faces associated with G^i

$$\mathcal{F}_G^i := \{F \in \mathcal{F}(K, K') : K \in G^i \text{ or } K' \in G^i\}, \tag{24}$$

which we call *ghost penalty faces*. Here, $\mathcal{F}(K, K') = K \cap K'$ denotes the face between element $K \in \mathcal{T}_h$ and its neighbouring element $K' \in \mathcal{T}_h$. The ghost penalty faces, $\mathcal{F}_G^i$, and the set of intersected elements, G^i, are illustrated in Figure 5.

Now, we define our finite element space over the fictitious domains as: Let $\mathcal{U}_h$ denote the vector valued space of continuous piecewise linear polynomials defined on the background mesh $\mathcal{T}_h$. Then, for each $i \in I_\Omega$, we define the space

$$\mathcal{U}_h^i = \mathcal{U}_h|_{\widehat{\Omega}^{*,i}}, \tag{25}$$

i.e. $\mathcal{U}_h^i$ is defined as the restriction of $\mathcal{U}_h$ to the fictitious domain mesh $\widehat{\Omega}^{*,i} \subset \mathcal{T}_h$.

Recall that two different fictitious domains overlap in cells that are intersected by interfaces. This overlapping, subdomain-based definition of approximation spaces, which is consistent with the classical CutFEM paradigm [23, 10], naturally leads to a displacement field that may be discontinuous across interfaces.
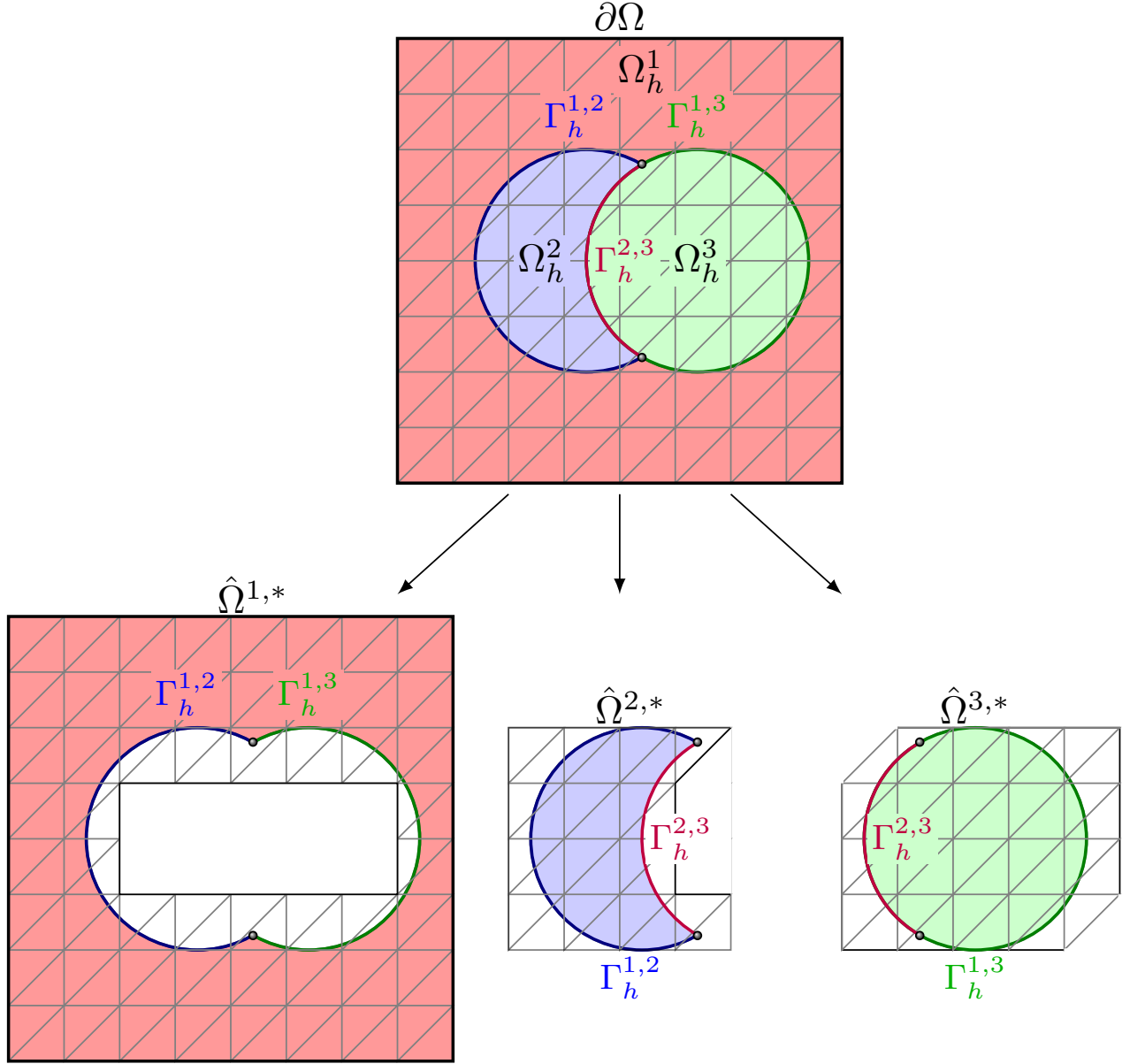
Figure 4: Schematic representation of fictitious domain meshes $\hat{\Omega}^{i,*}$.

Piecewise linear spaces for the LaTIn interface quantities. We will seek continuous piecewise linear approximations of the interface displacements $\underline{W}_h^{i,j}$ and interface forces $\underline{F}_h^{i,j}$, for all linearly interpolated subdomain interfaces $\Gamma_h^{i,j}$, with $(i,j) \in \mathcal{I}_\Gamma$. Similarly to the bulk equations, we will extend the interface quantities from the interface $\Gamma_h^{i,j}$ onto the entire band of elements which are intersected by the interface $\Gamma_h^{i,j}$ defined as

$$G^{i,j} := \{K \in \mathcal{T}_h : K \cap \Gamma_h^{i,j} \neq \emptyset\} \tag{26}$$

to prevent ill-conditioning. These extensions will be denoted by $\underline{W}_h^{*,i,j}$ and $\underline{F}_h^{*,i,j}$, respectively. In particular, these extensions fulfil

$$\underline{W}_h^{*,i,j}\big|_{\Gamma_h^{i,j}} = \underline{W}_h^{i,j}\big|_{\Gamma_h^{i,j}}, \quad \underline{F}_h^{*,i,j}\big|_{\Gamma_h^{i,j}} = \underline{F}_h^{i,j}\big|_{\Gamma_h^{i,j}}. \tag{27}$$

Algebraic operations on interface quantities $\underline{W}_h^{i,j}$ and $\underline{F}_h^{i,j}$ are then performed implicitly through manipulating the nodal values of their volume extensions. Now let us formally define the continuous piecewise linear finite element space in which we will seek the interface quantity extensions $\underline{W}_h^{*,i,j}$ and $\underline{F}_h^{*,i,j}$ as

$$\mathcal{V}_h^{i,j} = \mathcal{U}_h\big|_{G^{i,j}}, \tag{28}$$
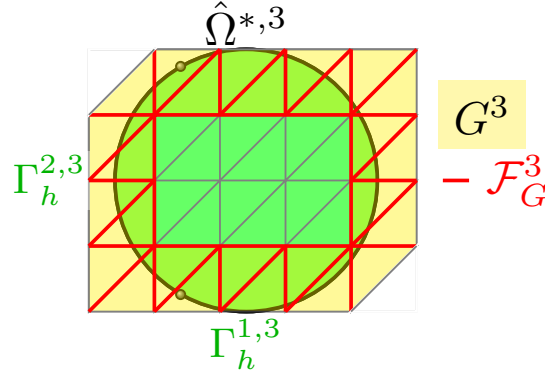
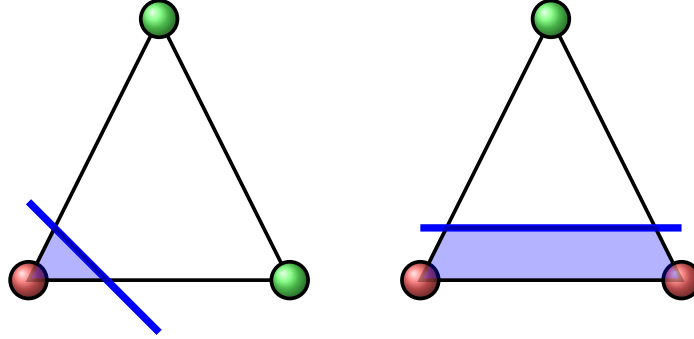Figure 5: Schematic representation of ghost penalty faces.

Figure 6: Cases of interface-element intersections that lead to ill conditioning of the system matrices in the abscence of appropriate regularisation.

i.e. the vector valued space of continuous piecewise linear polynomials on $G^{i,j}$. Let $N_{d,G^{i,j}}$ denote the number of degrees of freedom in $\mathcal{V}_h^{i,j}$.

3.2.2 Regularised formulation of the bulk equations

As briefly mentioned above, integration over element parts $K_i = K \cap \Omega_h^i$ in the interface region can lead to conditioning issues. For instance, cutting an element in an arbitrary location may result in elements that have an almost-zero intersection with the physical domain (see Figure 6). As a consequence, degrees of freedom of such elements can move in an uncontrolled manner, which results in severe ill-conditioning and convergence issues for linear and nonlinear solution algorithms.

In the context of our LaTIn-CutFEM method, we will address this issue through a ghost penalty extension of the coercivity of the linear elasticity problems to the entire fictitious domains. This is done by penalising the jump of fictitious tractions (*i.e.* defined over the entire fictitious domain) across elements that are cut by an unfitted LaTIn interface. When the penalty term is scaled correctly, this consistent approach ensures that the convergence of the condition number with mesh refinement is that of a conforming FEM. The interested reader is referred to [9, 10] for extensive derivations and applications of the ghost penalty approach in the context of Nitsche's method.

The ghost-penalty regularised variational statements associated with the static equilibrium of the subdomains read as

For all $i \in \mathcal{I}_\Omega$, field $\underline{u}_h^{*,i} \in \mathcal{U}_h^i$ defined over the fictitious subdomain $\hat{\Omega}^{*,i}$ must satisfy, for any variation $\delta\underline{u} \in \mathcal{U}_h^i$
,

$$
\begin{aligned}
a_{\mathrm{D}}^i(\underline{u}_h^{*,i}, \delta\underline{u}) \quad &+ \quad a_{\mathrm{t}}^i(\underline{u}_h^{*,i}, \delta\underline{u}) + a_{\mathrm{N}}^i(\underline{u}_h^{*,i}, \delta\underline{u}) + j_{\mathrm{u}}^i(\underline{u}_h^{*,i}, \delta\underline{u}) \\
&= \quad l_{\mathrm{f}}^i(\delta\underline{u}) + l_{\mathrm{N}}^i(\delta\underline{u}) .
\end{aligned}
\tag{29}
$$

The terms introduced in the previous formulation are listed and commented below.

11

- **"Mechanical" virtual work.** The virtual work of the internal forces in domain Ω_h^i, $i \in \mathcal{I}_\Omega$, reads as

$$a_D^i(\underline{u}_h^{*,i}, \delta\underline{u}) = \int_{\Omega_h^i} \underline{\underline{\epsilon}}(\underline{u}_h^{*,i}) : \underline{\underline{D}} : \underline{\underline{\epsilon}}(\delta\underline{u})\, d\Omega\,, \tag{30}$$

the work of the tractions from neighbouring subdomains reads as

$$a_t^i(\underline{u}_h^{*,i}, \delta\underline{u}) = -\int_{\Gamma_h^i} \left(\left(\underline{\underline{D}} : \underline{\underline{\epsilon}}(\underline{u}_h^{*,i}) \right) \cdot \underline{n}^i \right) \cdot \delta\underline{u}\, d\Gamma\,, \tag{31}$$

and the work of the external forces, excluding that of the interface, reads as

$$l_f^i(\delta\underline{u}) = \int_{\Omega_h^i} \underline{f} \cdot \delta\underline{u}\, d\Omega + \int_{\partial\Omega_h^t \cap \partial\Omega_h^i} \underline{T} \cdot \delta\underline{u}\, d\Gamma\,. \tag{32}$$

- **Nitsche terms for non-conforming Dirichlet boundaries.** On approximate boundary $\partial\Omega_h^u$, Nitsche's method allows us to apply the Dirichlet conditions weakly. The corresponding terms read as

$$
\begin{aligned}
a_N^i(\underline{u}_h^{*,i}, \delta\underline{u}) = &-\int_{\partial\Omega_h^u} \delta\underline{u} \cdot \left(\underline{\underline{D}} : \underline{\underline{\epsilon}}(\underline{u}_h^{*,i}) \right) \cdot \underline{n}\, d\Gamma - \int_{\partial\Omega_h^u} \underline{u}_h^{*,i} \cdot \left(\underline{\underline{D}} : \underline{\underline{\epsilon}}(\delta\underline{u}) \right) \cdot \underline{n}\, d\Gamma \\
&+ \int_{\partial\Omega_h^u} \frac{\alpha E^i}{h}\, \underline{u}_h^{*,i} \cdot \delta\underline{u}\, d\Gamma\,,
\end{aligned}
\tag{33}
$$

and

$$l_N^i(\delta\underline{u}) = \int_{\partial\Omega^{u,h}} \frac{\alpha E^i}{h}\, \underline{U}^h \cdot \delta\underline{u}\, d\Gamma\,. \tag{34}$$

The $1/h$ scaling ensures stability and optimal convergence of the symmetric Nitsche's method (see *e.g.* [10]). Note that, when using penalty-type formulations to enforce Dirichlet conditions, the test field $\delta\underline{u}$ is not required to vanish over Dirichlet boundaries.

- **The ghost penalty term** reads as

$$j_u^i(\underline{u}_h^{*,i}, \delta\underline{u}) = \sum_{F \in \mathcal{F}_G^i} \int_F \frac{\gamma_g\, h}{E^i} \left[\!\!\left[\left(\underline{\underline{D}} : \underline{\underline{\epsilon}}(\underline{u}_h^{*,i}) \right) \cdot \underline{n}_F \right]\!\!\right] \cdot \left[\!\!\left[\left(\underline{\underline{D}} : \underline{\underline{\epsilon}}(\delta\underline{u}) \right) \cdot \underline{n}_F \right]\!\!\right] ds\,. \tag{35}$$

Here, $[\![x \cdot \underline{n}_F]\!]$ denotes the normal jump of the quantity x over the face, F, defined as $[\![x \cdot \underline{n}_F]\!] = x|_K\, n_F - x|_{K'}\, n_F$, where n_F denotes a unit normal to the facet F with fixed but arbitrary orientation. The ghost penalty parameter, $\gamma_g > 0$, is chosen to be sufficiently large to regularise the solution in the interface regions. In the previous definition, we have penalised the jump of stress across all ghost penalty faces as defined in (24). These include intersected faces and faces of intersected elements that couple the element to an element in the interior of Ω_h^i. This way, we extend the traction vector into the fictitious domain to prevent ill-conditioning in a consistent way (i.e. for smooth tractions the jump terms vanish).

3.2.3 Multiscale split of interface quantities and weak compatibility conditions

Heart quantities Forcing piecewise linear quantities $\{(\underline{F}_h^{i,j}, \underline{W}_h^{i,j})\}$ to satisfy the nonlinear equations of unilateral contact is doomed to failure due to the intrinsically nonlinear nature of unilateral contact. Recall that the mainstream approach is to choose these quantities as piecewise constant, which is compatible with complex interface laws. However, such approaches lead to numerical instabilities.

We propose to introduce a new pair of dual interface fields $\{\underline{W}_\heartsuit^{i,j}\}$, $\{\underline{F}_\heartsuit^{i,j}\}$, chosen in $L^2(\Gamma)$. These quantities are the ones that are required to satisfy the contact conditions, as represented in Figure 7. They are then weakly connected to $\{(\underline{F}_h^{i,j}, \underline{W}_h^{i,j})\}$. More precisely, a regularised L^2 projection of $\{\underline{W}_\heartsuit^{i,j}\}$, $\{\underline{F}_\heartsuit^{i,j}\}$ into the finite element piecewise linear interface space is required to equate $\{(\underline{F}_h^{i,j}, \underline{W}_h^{i,j})\}$, whilst the remainder, which we term "fluctuation", will satisfy a Robin condition.

In practice, the contact conditions are satisfied by the heart quantities at a set of quadrature points, more precisely the quadrature points of the subtriangulation of the cut interface (3 quadrature points per element in the sub-triangulation of interface $\{\Gamma_h^{i,j}\}$).
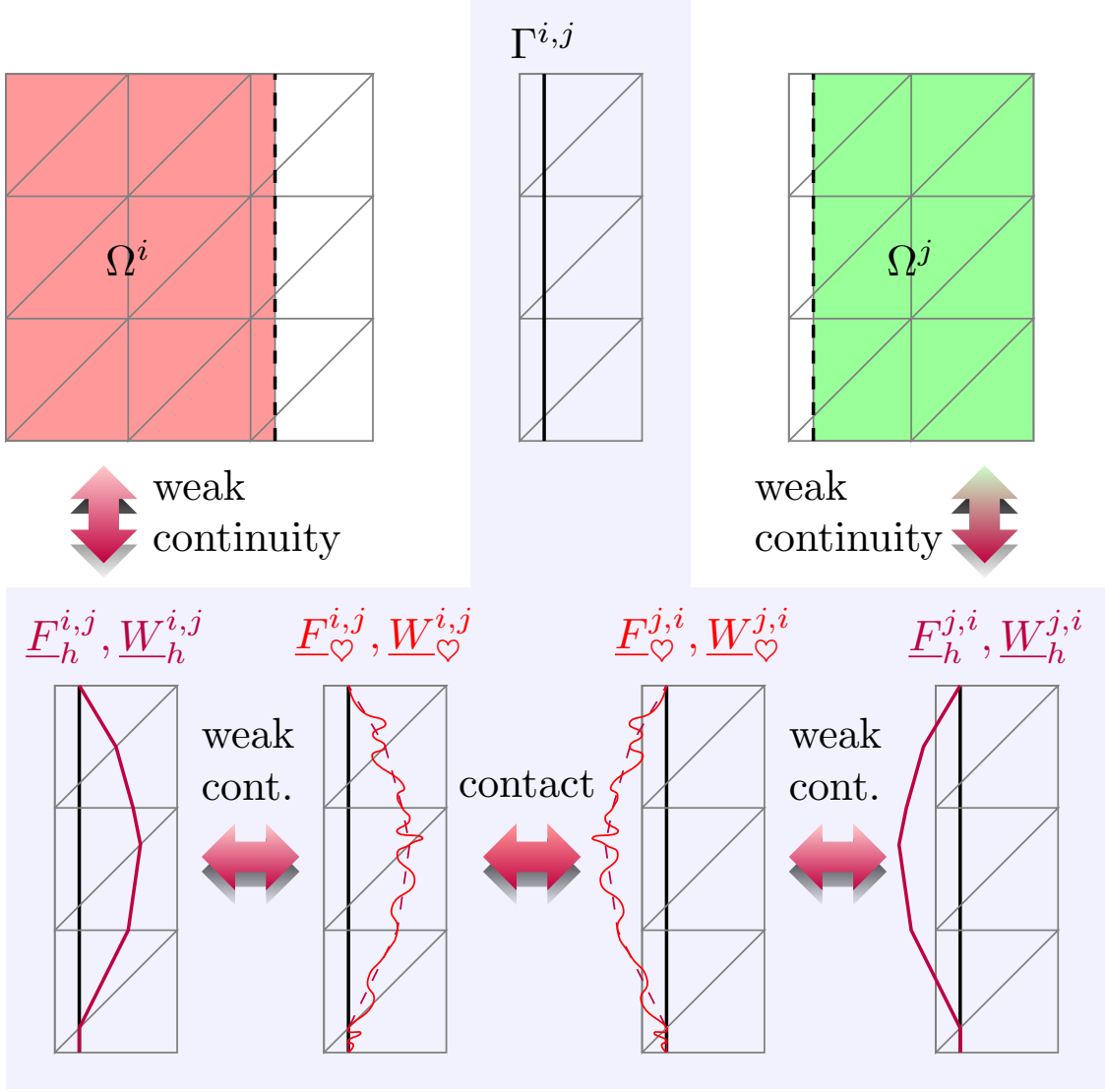
Figure 7: Schematic of LaTIn-CutFEM interface/bulk and interface/interface coupling.

Compatibility between interface and bulk For all interfaces $\Gamma_h^{i,j}$, $(i,j) \in \mathcal{I}_\Gamma$, we couple the interface and bulk quantities through approximate compatibility conditions

$$
\underline{W}_h^{i,j} = \underline{u}_h^{*,i}, \quad \underline{W}_h^{j,i} = \underline{u}_h^{*,j} \qquad \text{on } \Gamma_h^{i,j},
$$
$$
\int_{\Gamma_h^{i,j}} \underline{F}_h^{i,j} \cdot \delta\underline{u} = \int_{\Gamma_h^{i,j}} \left(\left(\underset{\sim}{D} : \underline{\underline{\epsilon}}(\underline{u}_h^{*,i}) \right) \cdot \underline{n}^i \right) \cdot \delta\underline{u} \, d\Gamma \qquad \forall \delta\underline{u} \in \mathcal{U}_h^i,
$$
$$
\int_{\Gamma_h^{i,j}} \underline{F}_h^{j,i} \cdot \delta\underline{u} = \int_{\Gamma_h^{i,j}} \left(\left(\underset{\sim}{D} : \underline{\underline{\epsilon}}(\underline{u}_h^{*,j}) \right) \cdot \underline{n}^i \right) \cdot \delta\underline{u} \, d\Gamma \qquad \forall \delta\underline{u} \in \mathcal{U}_h^j.
$$
(36)

These are standard conditions whereby the kinematic continuity is enforced exactly, whilst the continuity of dual quantities is only enforced on average, *i.e.* in a finite element sense. When solving our problem using the LaTIn algorithm, these two conditions will be introduced in (29) to yield our working expression of the linear stage.

Compatibility between interface and heart We project the heart quantities onto the continuous piecewise linear approximation space $\mathcal{V}_h^{i,j}$ (28) for each $\Gamma_h^{i,j}$ using a stabilised L^2-projection as follows. Find the extensions

$\underline{F}_h^{*,i,j} \in \mathcal{V}_h^{i,j}$ and $\underline{W}_h^{*,i,j} \in \mathcal{V}_h^{i,j}$ such that

$$
\begin{aligned}
\int_{\Gamma_h^{i,j}} (\underline{F}_h^{*,i,j} - \underline{F}_\heartsuit^{i,j})\, \delta \underline{F}_h^{i,j}\, d\Gamma + j_F(\underline{F}_h^{*,i,j}, \delta \underline{F}_h^{i,j}) &= \underline{0} \quad \forall \delta \underline{F}_h^{i,j} \in \mathcal{V}_h^{i,j}, \\
\int_{\Gamma_h^{i,j}} (\underline{W}_h^{\star,i,j} - \underline{W}_\heartsuit^{i,j})\, \delta \underline{W}_h^{i,j}\, d\Gamma + j_F(\underline{W}_h^{*,i,j}, \delta \underline{W}_h^{i,j}) &= \underline{0} \quad \forall \delta \underline{W}_h^{i,j} \in \mathcal{V}_h^{i,j},
\end{aligned}
\tag{37}
$$

where

$$
j_F(\underline{F}_h^{*,i,j}, \delta \underline{F}_h^{i,j}) = \sum_{F \in \mathcal{F}_I^{i,j}} \gamma_\Pi h^2 \int_F \left[\!\!\left[\nabla \underline{F}_h^{*,i,j} \cdot \underline{n}_F \right]\!\!\right] \cdot \left[\!\!\left[\nabla \delta \underline{F}_h^{i,j}, \cdot \underline{n}_F \right]\!\!\right] ds
\tag{38}
$$

regularises the interface fields with a penalty parameter $\gamma_\Pi > 0$ in the band of elements intersected by $\Gamma_h^{i,j}$. Here,

$$
\mathcal{F}_I^{i,j} := \{ F \in \mathcal{F}(K, K') : K \in G^{i,j} \text{ and } K' \in G^{i,j} \}.
\tag{39}
$$

denotes the set of faces intersected by $\Gamma_h^{i,j}$.

Heart closure We close our hybrid-mixed formulation of contact by requiring that the fluctuation of the heart quantities around $\underline{F}_h^{i,j}$ and $\underline{W}_h^{i,j}$ satisfy a Robin condition, which reads as

$$
\left(\underline{F}_h^{i,j} - \underline{F}_\heartsuit^{i,j} \right) - \beta \left(\underline{W}_h^{i,j} - \underline{W}_\heartsuit^{i,j} \right) = 0 \,,
\tag{40}
$$

where β is an algorithmic parameter that is homogeneous to a stiffness. If we set it equal to the LaTIn search direction parameter, the resulting local stage of the LaTIn solver is a classical local stage followed by a "coarse-scale" filtering step, as described in the next section.

This closure introduces a lack of consistency in the formulation. However, we expect the associated consistency error to vanish with mesh refinement with optimal order, as it directly affects the remainder of the heart fields after projection in the finite element space. This will be shown numerically through examining the convergence properties of the method.

It is noticeable that the non-local continuity equations (37) make the local stage non-local. In our opinion, this is not a significant limitation of the method. This is because the non-linearities can still be treated locally and semi-explicitly as per usual, in a first sub-stage, whilst the second sub-stage requires solving a unique set of independent (and small) linear problems, each of these problems corresponding to an interface between two adjacent subdomains.

3.3 Iterative algorithm for the fully discrete mixed cut finite element problem

We are now in the position to use the regularisation and modifications introduced previously to apply the LaTIn algorithm to the mixed interface quantities $\{ (\underline{F}_h^{i,j}, \underline{W}_h^{i,j}) \}$.

3.3.1 Regularised linear stage over the fictitious domains

Linear system of equations

For all $i \in \mathcal{I}_\Omega$, find the displacement field $\underline{u}_h^{*,i} \in \mathcal{U}_h^i$ defined over the fictitious subdomain $\hat{\Omega}^{*,i}$ such that, for any variation $\delta \underline{u} \in \mathcal{U}_h^i$,

$$
\begin{aligned}
a_D^i(\underline{u}_h^{*,i}, \delta \underline{u}) \quad &+ \quad a_k^i(\underline{u}_h^{*,i}, \delta \underline{u}) + a_N^i(\underline{u}_h^{*,i}, \delta \underline{u}) + j_u^i(\underline{u}_h^{*,i}, \delta \underline{u}) \\
&= \quad l_f^i(\delta \underline{u}) + \tilde{l}_k^i(\delta \underline{u}) + l_N^i(\delta \underline{u}) \,.
\end{aligned}
\tag{41}
$$

Two terms corresponding to the LaTIn approximation of the virtual work of the forces applied by neighbouring subdomains have been introduced. These forces are found iteratively as described in Section 2. At a given

iteration of the LaTIn solver, the approximate equilibrium of the subdomain is described via an augmentation term (*i.e.* "penalty" or "regularisation" boundary term induced by the descent search direction), which reads

$$a_{\mathrm{k}}^i(\underline{u}_h^{*,i}, \delta\underline{u}) = \int_{\Gamma_h^i} k^- \, \underline{u}_h^{*,i} \cdot \delta\underline{u} \, d\Gamma \,. \tag{42}$$

Mixed quantities from the previous half-iterations appear on the right-hand side as follows

$$l_{\mathrm{k}}^i(\delta\underline{u}) = \int_{\Gamma_h^i} \left(\widehat{\underline{F}}^i + k^- \, \widehat{\underline{W}}^i \right) \cdot \delta\underline{u} \, d\Gamma \,. \tag{43}$$

Note that the interface quantities $\widehat{\underline{F}}^i$, $\widehat{\underline{W}}^i$ over the interface Γ_h^i are given by the interface quantities $\widehat{\underline{F}}^{i,j}$, $\widehat{\underline{W}}^{i,j}$ by the following sum

$$\int_{\Gamma_h^i} \left(\widehat{\underline{F}}_h^i + k^- \, \widehat{\underline{W}}_h^i \right) \cdot \delta\underline{u}_h \, d\Gamma = \sum_{j \in \mathcal{I}_{\Gamma^i}} \left(\int_{\Gamma_h^{k,l}} \widehat{\underline{F}}_h^{i,j} + k^- \, \widehat{\underline{W}}_h^{i,j} \right) \cdot \delta\underline{u}_h \, d\Gamma, \tag{44}$$

where $(k,l) = (i,j)$ if $i < j$ and $(k,l) = (j,i)$ if $i > j$. Here,

$$\mathcal{I}_{\Gamma^i} = \{j \in \mathcal{I}_\Omega : \Gamma_h^{i,j} \neq \emptyset \text{ if } i < j; \text{ or } \Gamma_h^{j,i} \neq \emptyset \text{ if } i > j\}. \tag{45}$$

For example, in Figure 1, $\Gamma^3 = \Gamma^{1,3} \cup \Gamma^{2,3}$, $\mathcal{I}_{\Gamma^3} = \{1, 2\}$.

Post-processing of interface quantities   For all $(i,j) \in \mathcal{I}_\Gamma$, we seek the extended interface displacement fields, $\underline{W}_h^{*,i,j} \in \mathcal{V}_h^{i,j}$, $\underline{W}_h^{*,j,i} \in \mathcal{V}_h^{j,i}$, which fulfil

$$\begin{aligned}
\underline{W}_h^{*,i,j}|_{\Gamma_h^{i,j}} &= \underline{W}_h^{i,j}|_{\Gamma_h^{i,j}} = \underline{u}_h^{*,i}|_{\Gamma_h^{i,j}}, \\
\underline{W}_h^{*,j,i}|_{\Gamma_h^{i,j}} &= \underline{W}_h^{j,i}|_{\Gamma_h^{i,j}} = \underline{u}_h^{*,j}|_{\Gamma_h^{i,j}}.
\end{aligned} \tag{46}$$

We obtain these extended interface displacement fields by setting $\underline{W}_h^{*,i,j}$ equal to $\underline{u}_h^{*,i}$ and $\underline{W}_h^{*,j,i}$ equal to $\underline{u}_h^{*,j}$ in each degree of freedom in the band of intersected elements $G^{i,j}$, *i.e.*

$$\begin{aligned}
\underline{W}_h^{*,i,j}(\boldsymbol{x}_i) &= \underline{u}_h^{*,i}(\boldsymbol{x}_i) \quad \forall i = 1, \dots, N_{d,G^{i,j}}, \\
\underline{W}_h^{*,j,i}(\boldsymbol{x}_i) &= \underline{u}_h^{*,j}(\boldsymbol{x}_i) \quad \forall i = 1, \dots, N_{d,G^{i,j}}.
\end{aligned} \tag{47}$$

This means, as $\underline{W}_h^{*,i,j}$ and $\underline{W}_h^{*,j,i}$ is equal to $\underline{u}_h^{*,i}$ and $\underline{u}_h^{*,j}$ in all degrees of freedom around the interface, it is also equal to the $\underline{u}_h^{*,i}$ and $\underline{u}_h^{*,j}$ on the restriction to the interface $\Gamma_h^{i,j}$, hence we satisfy (46). The location of the degrees of freedom, $\boldsymbol{x}_i$, in the surface band mesh $G^{i,j}$ are illustrated in Figure 8.

Next, we can obtain the extended interface forces $\underline{F}_h^{*,i,j} \in \mathcal{V}_h^{i,j}$ and $\underline{F}_h^{*,j,i} \in \mathcal{V}_h^{i,j}$ for each $(i,j) \in I_\Gamma$ through the search direction

$$\begin{aligned}
\underline{F}_h^{*,i,j}(\boldsymbol{x}_i) &= \widehat{\underline{F}}_h^{*,i,j}(\boldsymbol{x}_i) + k^-(\widehat{\underline{W}}_h^{*,i,j}(\boldsymbol{x}_i) - \underline{W}_h^{*,i,j}(\boldsymbol{x}_i)) \quad \forall i = 1, \dots, N_{d,G^{i,j}}, \\
\underline{F}_h^{*,j,i}(\boldsymbol{x}_i) &= \widehat{\underline{F}}_h^{*,j,i}(\boldsymbol{x}_i) + k^-(\widehat{\underline{W}}_h^{*,j,i}(\boldsymbol{x}_i) - \underline{W}_h^{*,j,i}(\boldsymbol{x}_i)) \quad \forall i = 1, \dots, N_{d,G^{i,j}}.
\end{aligned} \tag{48}$$

Note that this gives us a piecewise linear continuous representation of the interface quantities.

Multiscale local stage   Having obtained the approximation for the interface force, $\underline{F}_h^{*,i,j}$, and the interface displacement, $\underline{W}_h^{*,i,j}$, we proceed to calculating local quantities following the multi-scale strategy introduced in section 3.2.3. This is done by performing the standard semi-explicit operations of the LaTIn local stage, at each interface quadrature point. Then, in a post-processing step, a linear smoothing is used to implement the multiscale regularisation using the stabilized L^2-projection (37). This is detailed below.
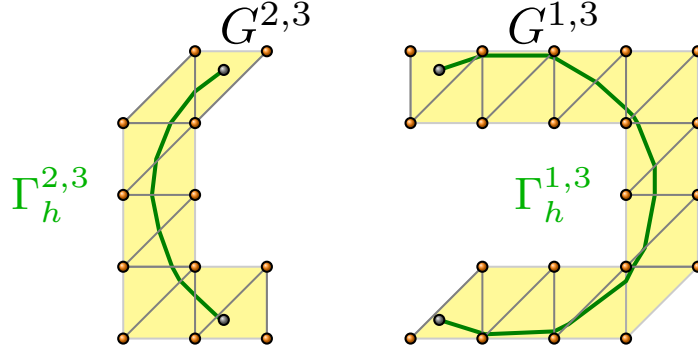
15

Figure 8: Decomposition of intersected cells for extended interface quantities into meshes $G^{i,j}$.

Firstly, to enforce frictionless unilateral contact, we determine the heart quantities for each interface $\Gamma_h^{i,j}$ at each quadrature point $\boldsymbol{x}_q$ along the interface $\Gamma_h^{i,j}$ by

$$\underline{F}_{\heartsuit}^{i,j}(\boldsymbol{x}_q) = \frac{1}{2}(\underline{F}_h^{*,i,j}(\boldsymbol{x}_q) - \underline{F}_h^{*,j,i}(\boldsymbol{x}_q) + k^+(\underline{W}_h^{*,j,i}(\boldsymbol{x}_q) - \underline{W}_h^{*,i,j}(\boldsymbol{x}_q))) \cdot \underline{n}_K^{i,j}(\boldsymbol{x}_q). \tag{49}$$

Here, $\underline{n}_K^{i,j}$ denotes a piecewise constant normal obtained from the level-set function ϕ_h^j

$$\underline{n}_K^{i,j}(\boldsymbol{x}_q) = -\left.\frac{\nabla \phi_h^j(\boldsymbol{x})}{|\nabla \phi_h^j(\boldsymbol{x})|}\right|_{\boldsymbol{x}=\boldsymbol{x_q}}. \tag{50}$$

To fulfil frictionless contact, we test at each quadrature point, if $\underline{F}_{\heartsuit}^{i,j}(\boldsymbol{x}_q) > 0$ (i.e. no contact), in which case we set $\underline{F}_{\heartsuit}^{i,j}(\boldsymbol{x}_q) = 0$. The heart quantity now satisfies frictionless contact at each point and we can determine the local interface force extension $\widehat{\underline{F}}_h^{*,i,j} \in \mathcal{V}_h^{i,j}$ from the stabilised L^2-projection (37). Next, we use $\widehat{\underline{F}}_h^{*,i,j}$ to obtain

$$\widehat{\underline{F}}_h^{*,j,i}(\boldsymbol{x}_i) = -\widehat{\underline{F}}_h^{*,i,j}(\boldsymbol{x}_i) \tag{51}$$

$$\begin{aligned}
\widehat{\underline{W}}_h^{*,i,j}(\boldsymbol{x}_i) &= \underline{W}_h^{*,i,j}(\boldsymbol{x}_i) + \frac{1}{k^+}(\widehat{\underline{F}}_h^{*,i,j}(\boldsymbol{x}_i) - \underline{F}_h^{*,i,j}(\boldsymbol{x}_i)) \\
\widehat{\underline{W}}_h^{*,j,i}(\boldsymbol{x}_i) &= \underline{W}_h^{*,j,i}(\boldsymbol{x}_i) + \frac{1}{k^+}(\widehat{\underline{F}}_h^{*,j,i}(\boldsymbol{x}_i) - \underline{F}_h^{*,j,i}(\boldsymbol{x}_i)).
\end{aligned} \tag{52}$$

for all $i = 1, \ldots, N_{d,G^{i,j}}$. Note that these algebraic relations together with the stabilised L^2-projection (37) give us the continuous and piecewise linear extended interface quantities that we are seeking, *i.e.* $\widehat{\underline{F}}_h^{*,j,i}(\boldsymbol{x}_i) \in \mathcal{V}_h^{j,i}$, $\widehat{\underline{W}}_h^{*,i,j}(\boldsymbol{x}_i) \in \mathcal{V}_h^{i,j}$, $\widehat{\underline{W}}_h^{*,j,i}(\boldsymbol{x}_i) \in \mathcal{V}_h^{j,i}$.

Remark. *The extension of our LaTIn-CutFEM method to other contact laws is relatively straightforward. This is done by simply modifying the local heart quantities at the quadrature points along the interfaces. Therefore, our algorithm inherits one of the central characteristics of the LaTIn method, whereby different type of interface behaviour are implemented by only changing the local quadrature algorithm.*
For instance, for Coulomb friction under proportional loading, we simply extend the computation of the heart quantities by adding a new test and computation of the tangential interface forces and displacements as it is classically done within the LaTIn algorithm [12]. For non-proportional loading a time integration loop is required. We have successfully performed preliminary tests of the LaTIn-CutFEM algorithm for Coulomb friction which will be published in a forthcoming paper. The convergence and stability properties of our approach seem to be unaffected.

3.3.2 Pseudo-code of the LaTIn-CutFEM method

Algorithm 1 summaries the LaTin-CutFEM algorithm presented above.

Algorithm 1 LaTin-CutFEM algorithm for unilateral contact

1: Approximate level set functions through linear interpolation $\rightarrow \phi_h^i$
2: Compute intersection of level set functions with background mesh and obtain linear approximations of $\Gamma_h^{i,j}$ and Ω_h^i
3: Set $\widehat{\underline{W}}_h^{*,i,j} = \mathbf{0}$, $\widehat{\underline{W}}_h^{*,j,i} = \mathbf{0}$ and $\widehat{\underline{F}}_h^{*,i,j} = \mathbf{0}$, $\widehat{\underline{F}}_h^{*,j,i} = \mathbf{0}$, $\forall (i,j) \in \mathcal{I}_\Gamma$.
4: **while** $it < it_{max}$ **do**   ▷ it_{max} is maximum iteration count
5:     **procedure** LINEAR STAGE($\widehat{\underline{W}}_h^{*,i,j}, \widehat{\underline{F}}_h^{*,i,j}$)
6:         Find $\underline{u}_h^{*,i}$ by solving bulk equation (41) $\forall i \in \mathcal{I}_\Omega$
7:         Determine $\underline{W}_h^{*,i,j}$ from (47) and $\underline{F}_h^{*,i,j}$ from (48) using $\underline{u}_h^{*,i}$.
8:         Perform relaxation (16) for $\underline{W}_h^{*,i,j}$ and $\underline{F}_h^{*,i,j}$.
9:         **return** $\underline{u}_h^{*,i}$, $\underline{W}_h^{*,i,j}$, $\underline{F}_h^{*,i,j}$
10:     **end procedure**
11:     **procedure** LOCAL STAGE($\underline{u}_h^{*,i}$, $\underline{W}_h^{*i,j}$, $\underline{F}_h^{*,i,j}$)
12:         for **contact interfaces** $\Gamma_h^{i,j}$:
13:             Compute normals $\underline{n}_K^{i,j}$ (50)
14:             Obtain $\widehat{\underline{F}}_h^{*,i,j}$ from (37), (51) and $\widehat{\underline{W}}_h^{*,i,j}$ from (52)
15:         **return** $\widehat{\underline{F}}_h^{*,i,j}$, $\widehat{\underline{W}}_h^{*,i,j}$
16:     **end procedure**
17:     it +=1
18: **end while**

---

4 Numerical investigations

4.1 Controlled Condition Number: Square Domain with a Crack

In our first numerical example, we demonstrate that the ghost penalty terms introduced in (35) prevent ill conditioning of the system matrices associated with the linear stage of our LaTIn solver (recall equation (41)). To fully control "bad cuts" (see Figure 6 for two typical bad cut cases), we design a crack test inspired by previous studies published in [32] and [6]. The computational domain of our test case is decomposed into 4 subdomains with two vertical and one diagonal "crack" interface (see Figure 9). "Good" (large K_i) and "bad" (small K_i) cut cases can be constructed through shifting these interfaces.

The geometry is defined by the following three level set functions

$$\phi_1(x,y) = y - x - \epsilon_y h \,, \tag{53}$$

$$\phi_2(x,y) = x - \frac{1}{3} - \epsilon_x h \,, \tag{54}$$

$$\phi_3(x,y) = \frac{2}{3} + \epsilon_x h - x \,, \tag{55}$$

where the parameters ϵ_x and ϵ_y are chosen to range from 0.5 (good cut) to 10^{-11} (bad cut). We investigate two test cases: (i) the vertical interfaces are kept fixed at a good cut position with $\epsilon_x = 0.5$ and the diagonal crack is moved in the interval $10^{-11} \leq \epsilon_y \leq 0.25$ (i.e. "simply bad" cut, see Figure 9a, 9b) ; (ii) both ϵ_x and ϵ_y are varied in the interval $10^{-11} \leq \epsilon_x = \epsilon_y \leq 0.25$ (i.e. "doubly bad" cut, see Figure 9c, 9d).

Our LaTIn-based domain decomposition strategy leads to uncoupled sets of linear systems of equations. Therefore, we will only report the conditioning of the worst subproblem, which we denote by κ. Figure 10 shows this condition number for test case (i) (see Figure 10a) and test case (ii) (see Figure 10b). In both cases, if the ghost penalty regularisation is switched off, e.g. $\gamma_g = 0$, then the condition number increases drastically with decreasing ϵ (i.e. worsening the cut scenario). This problem appears despite the regularisation effect of the LaTIn augmentation terms. This is because the search direction coefficient is chosen so as to scale with macroscopic properties of the problem at hand. In other words, this term is chosen to ensure the fast convergence of the LaTIn solver, and not to repair instabilities created by local bad cuts. However, with ghost

(a) $\epsilon_x = 0.5$, $\epsilon_y = 0.25$.

(b) $\epsilon_x = 0.5$, $\epsilon_y = 10^{-10}$.

(c) $\epsilon_x = \epsilon_y$, $\epsilon_y = 0.25$.

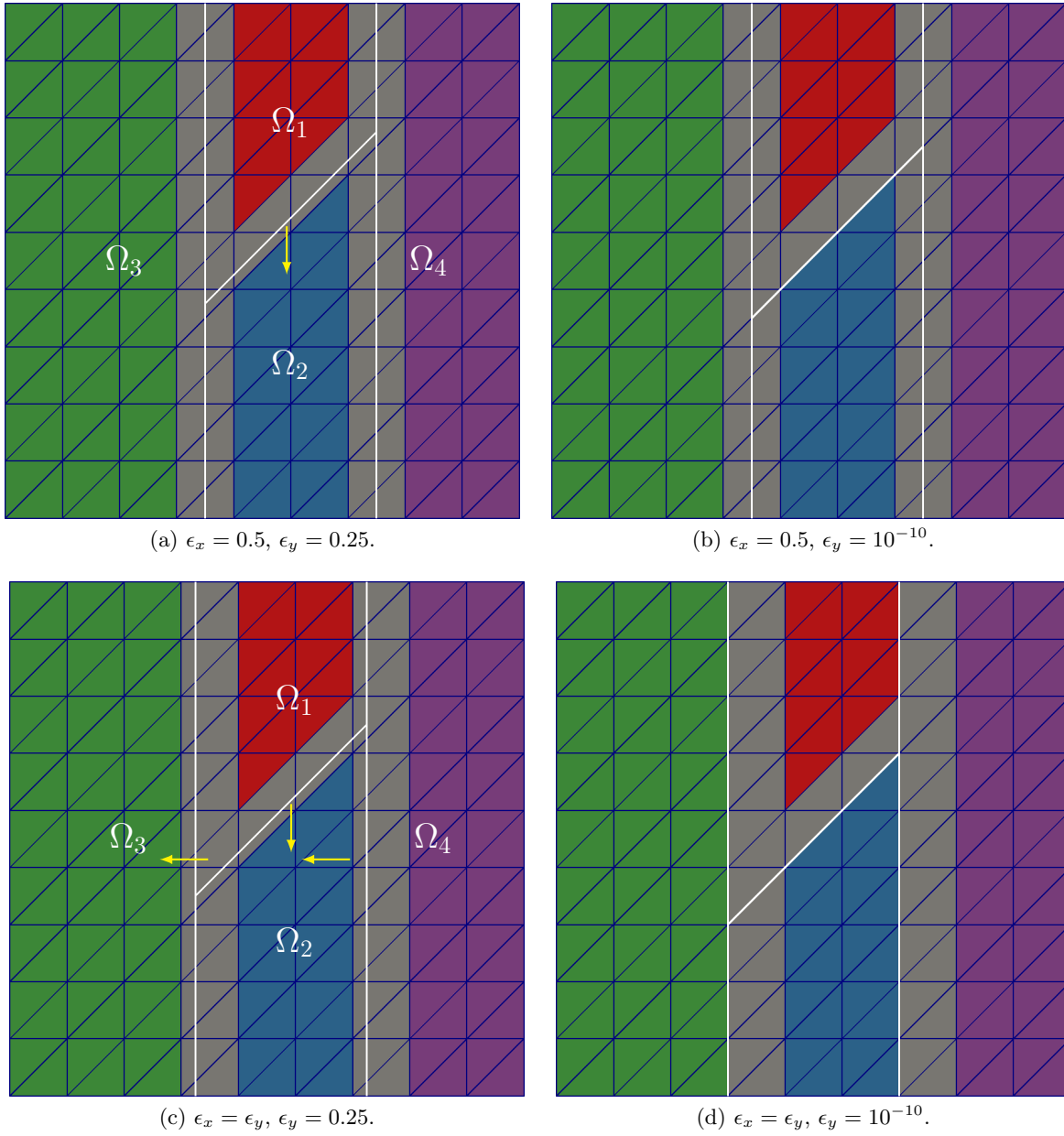(d) $\epsilon_x = \epsilon_y$, $\epsilon_y = 10^{-10}$.

Figure 9: Domains and interface cut locations for the two condition number test cases (i): (a), (b) and (ii): (c), (d).

penalty regularisation the increasingly unstable behaviour with decreasing ϵ is cured. Even a very small value of the ghost penalty parameter, *e.g.* $\gamma_g = 10^{-3}$, is sufficient to guarantee a bounded condition number for very small ϵ.

Most importantly, the ghost penalty regularisation does not impact the scaling of the condition number of the system matrices with mesh refinement. This is demonstrated in Figure 11. In fact, this scaling is typical for standard P1 finite element matrices. In other words, through the scaling that we have chosen for the ghost penalty regularisation, we ensure that the regularisation properties stated previously are consistent throughout the mesh refinement process.

(a) $\epsilon_x = 0.5$, $\epsilon = \epsilon_y$.

(b) $\epsilon = \epsilon_x = \epsilon_y$.
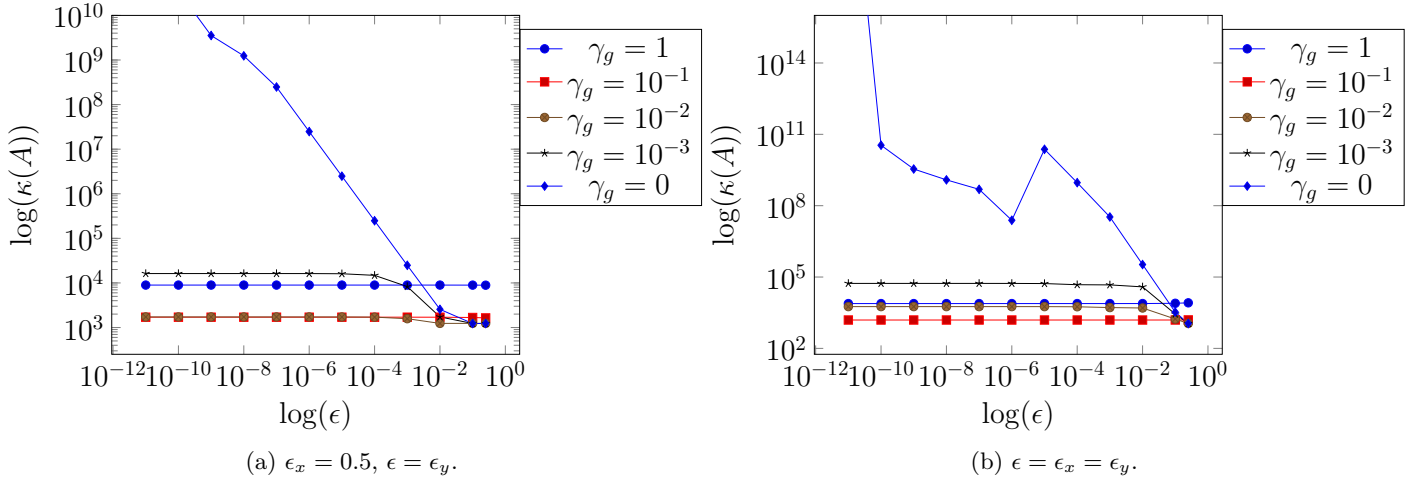
Figure 10: Condition number dependence on cut location for test case (i): (a) and test case (ii): (b).
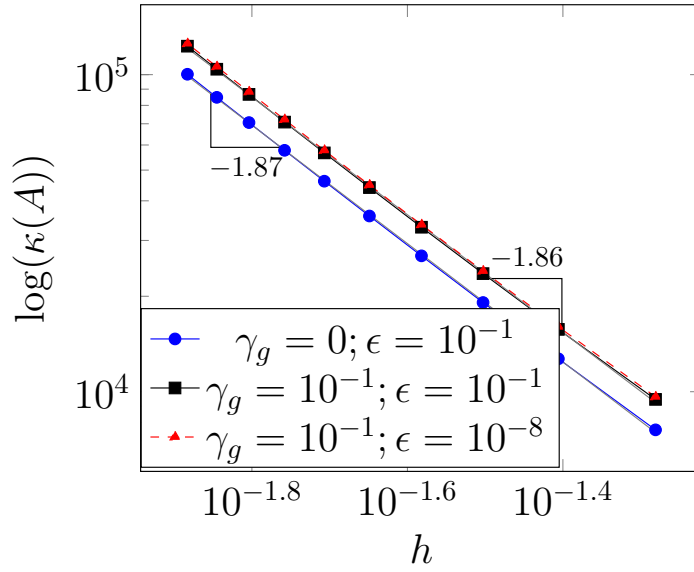
Figure 11: Condition number dependence on mesh refinement.

4.2 Convergence with mesh refinement

We will now proceed to studying the convergence properties of the LaTIn-CutFEM algorithm, through three test cases: (i) a problem with a simple elliptical inclusion in contact with a matrix, (ii) a problem with two inclusions, which exhibits multiply connected points, and (iii) a problem that features a large elastic contrast between interacting elastic phases.

4.2.1 Elliptical inclusion

We start with our first convergence study. Let us consider an elliptical inclusion Ω_1 in a rectangular domain $\Omega_2 = \Omega \setminus \Omega_1$ with $\Omega = [-1.2, 1.2] \times [-1.2, 1.2]$. The elliptical inclusion is described by the level set function

$$\phi_1(x,y) = \sqrt{\left(\frac{x}{a}\right)^2 + \left(\frac{y}{b}\right)^2} - r, \tag{56}$$

where $r = 0.654545$, $a = 1$, $b = 0.5$. We apply a displacement of $\underline{u} = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$ on the top of the outer domain, a zero displacement at the bottom and zero Neumann conditions at the side of the domain (see Figure 12a). We choose $E^1 = E^2 = 1.0$, $k^+ = k^- = 1.0$, $\gamma_g = 0.1$, $\gamma_{\Pi} = 0.1$ and $\alpha = 10$.

19

The elliptical inclusion and the rectangular domain interact through unilateral contact interface Γ_C. Figure 13 shows the displacement and stress components for a fine mesh with $h = 0.00375$ after 200 LaTIn iterations (the LaTIn algorithm has reached a converged solution state, as shown later on). We observe that the inclusion and the background block material are not in contact on the left and right of the ellipse and are in contact on the top. There are stress concentrations where the contact boundary changes from "in contact" to "not in contact". We utilise this fine mesh solution as a numerical reference to investigate the convergence of our LaTin-CutFEM algorithm with mesh refinement and with the number of LaTIn iterations.

(a) Schematic and boundary conditions.

(b) Coarsest Mesh.

(c) First hierarchical refinement.

(d) Finest mesh.
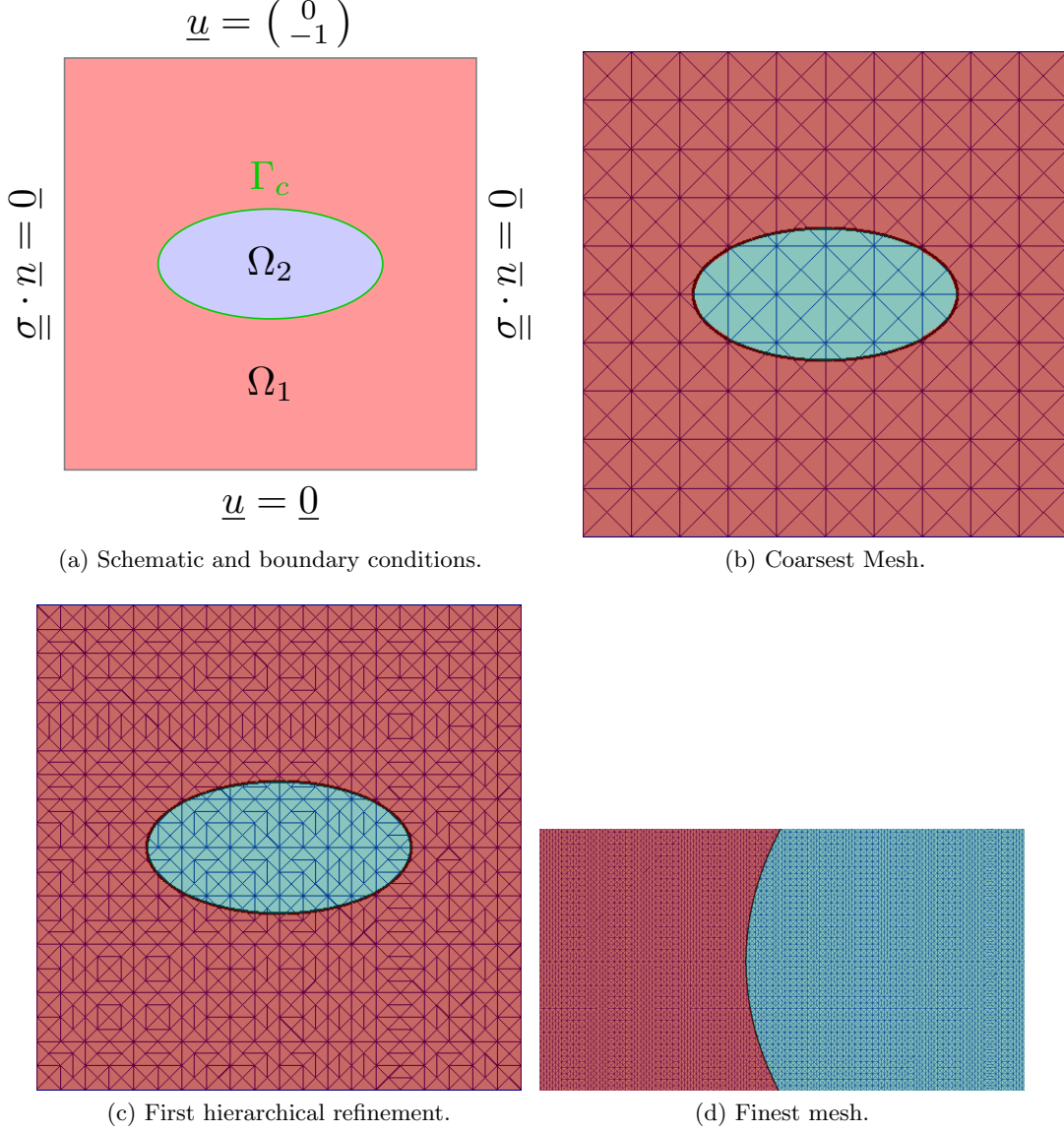
Figure 12: Schematic of elliptical inclusion with boundary conditions and regular background mesh refinement strategy.

Convergence of algorithmically converged solutions with mesh refinement. The coarsest background mesh is the regular mesh shown in Figure 12b. This mesh is refined in a hierarchical manner (each triangle of the coarser mesh is subdivided into 4 triangles) to produce the finer levels of discretisation. We perform 200 iterations of the LaTIn solver for each of these meshes. We now investigate the rate of convergence of the

(a) Horizontal displacement u_x.

(b) Vertical displacement u_y.

(c) Stress component σ_{xx}.

(d) Stress component σ_{xy}.
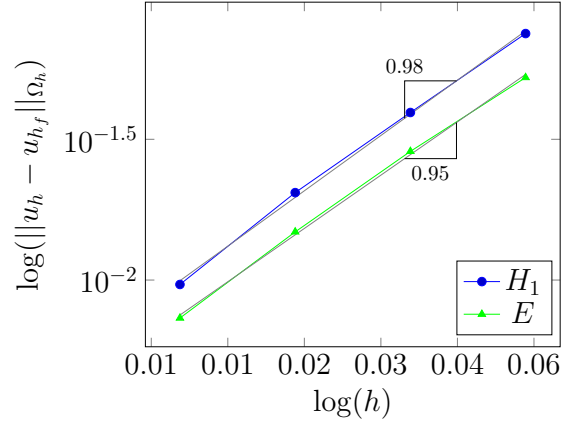
(e) Stress component σ_{yy}.
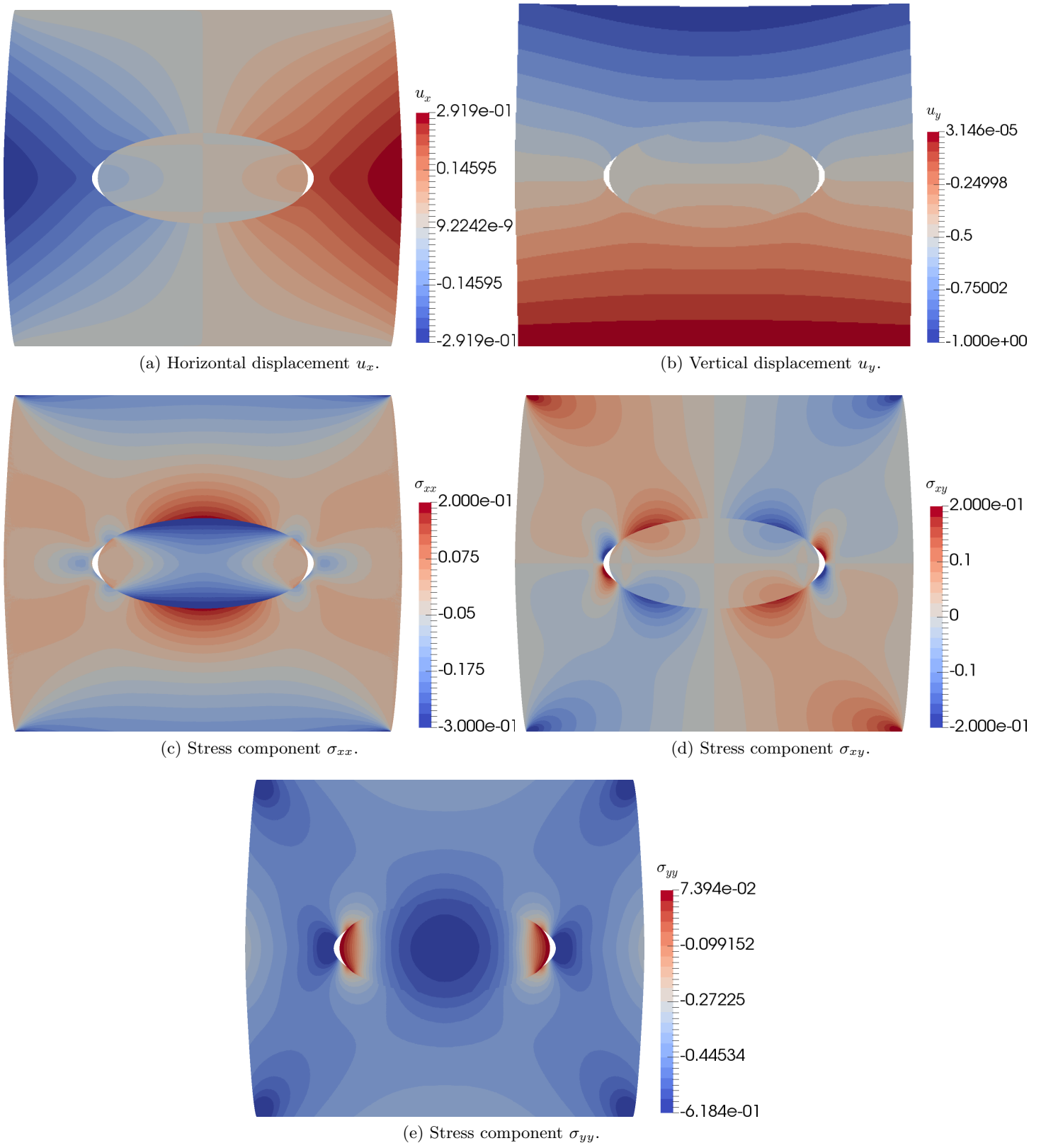
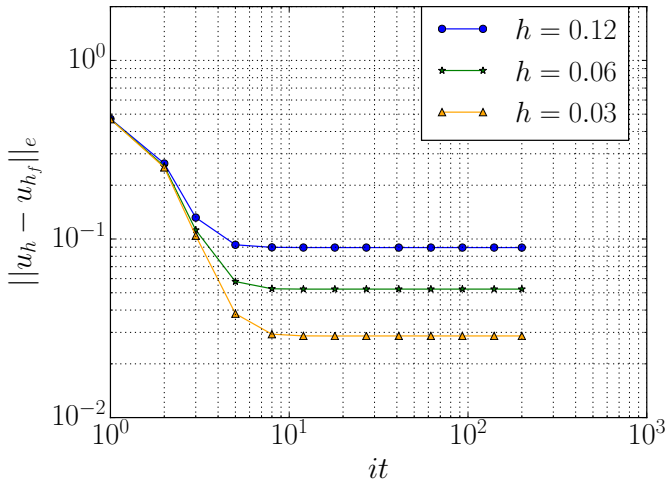Figure 13: Displacement and stress components of elliptical inclusion for the finest mesh.
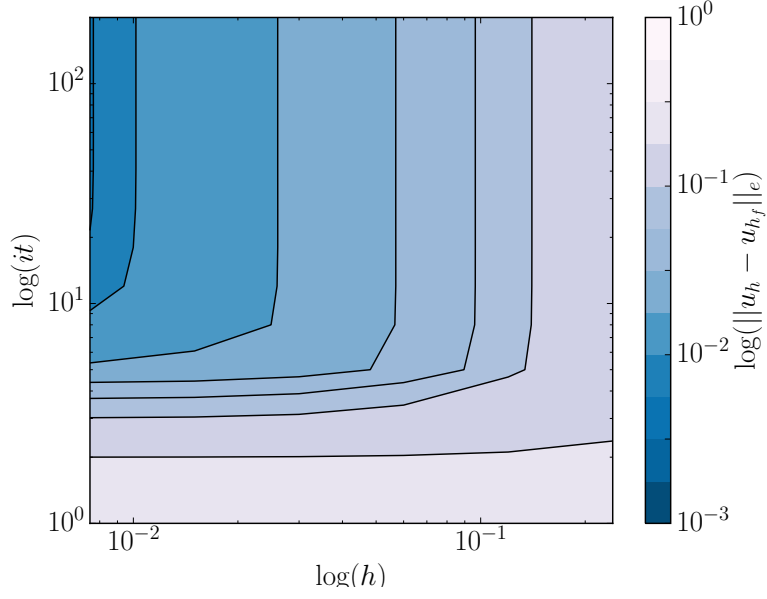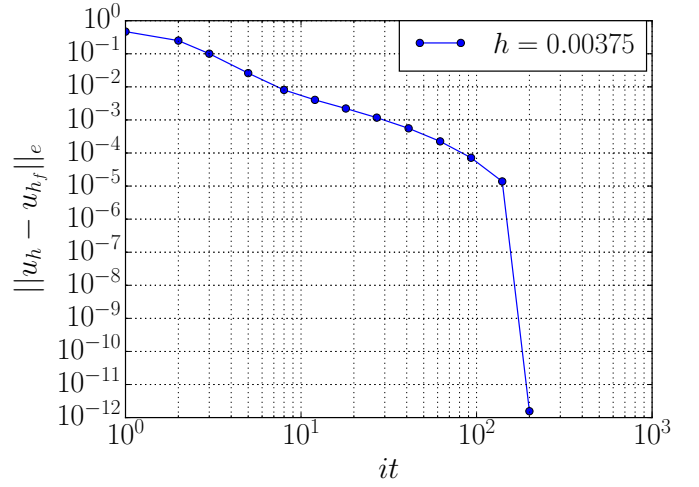
(a) Energy norm error vs iterations and mesh size.
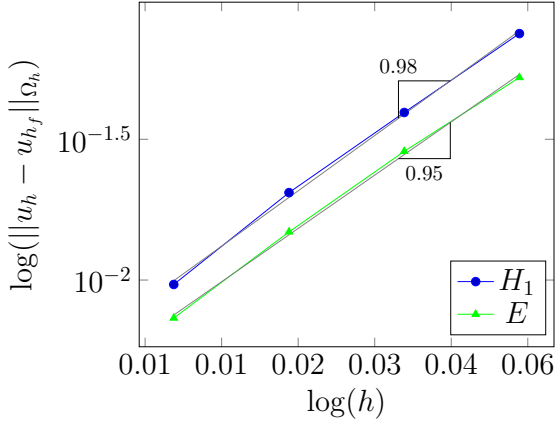
(b) Convergence rates for 2 quadrature points per interface segment.

(c) Convergence rates for 4 quadrature points per interface segment.

(d) Energy norm error with iteration count for coarser mesh sizes.

(e) Energy norm error with iteration count for finest solution.

Figure 14: Convergence rates and energy norm error with iteration count for ellipse-shaped inclusion.

numerical solution in the H_1-norm, which is defined as

$$\left\| \underline{u}_h - \underline{u}_{h_f} \right\|_1 = \sqrt{\sum_{i \in \mathcal{I}_\Omega} \int_{\Omega_{h_f}^i} ((\underline{u}_h^i - \underline{u}_{h_f}^i)^2 + (\nabla \underline{u}_h^i - \nabla \underline{u}_{h_f}^i)^2) d\Omega} \tag{57}$$

and in the energy norm, which reads

$$\left\| \underline{u}_h - \underline{u}_{h_f} \right\|_E = \sqrt{\sum_{i \in \mathcal{I}_\Omega} \int_{\Omega_{h_f}^i} \underline{\underline{\epsilon}}(\underline{u}_h - \underline{u}_{h_f}) : \underline{\underline{D}} : \underline{\underline{\epsilon}}(\underline{u}_h - \underline{u}_{h_f}) \, d\Omega}, \tag{58}$$

where $\underline{u}_{h_f}$ denotes the reference solution (*i.e.* finest mesh). To evaluate these norms, we interpolate the solution on the coarser mesh into the finest mesh and then evaluate the error norm over the linear geometry approximation, $\Omega_{h_f}^i$, of the finest mesh. Figure 14b, 14c shows that the convergence rate of our scheme is 0.98 in the H_1 and 0.95 in the energy norm. This is optimal, in the sense that the proposed mixed numerical solver for unilateral contact uses P1 standard finite elements to solve the bulk equations and the optimal convergence rate in the H^1 and energy norm is therefore expected to be one.

h	H_1-error	E-error
$6.0 \cdot 10^{-2}$	$7.5087 \cdot 10^{-2}$	$5.2406 \cdot 10^{-2}$
$3.0 \cdot 10^{-2}$	$3.9379 \cdot 10^{-2}$	$2.8651 \cdot 10^{-2}$
$1.5 \cdot 10^{-2}$	$2.0446 \cdot 10^{-2}$	$1.4827 \cdot 10^{-2}$
$7.5 \cdot 10^{-3}$	$9.6445 \cdot 10^{-3}$	$7.3295 \cdot 10^{-3}$

(a) Two quadrature points.

h	H_1-error	E-error
$6.0 \cdot 10^{-2}$	$7.5106 \cdot 10^{-2}$	$5.2414 \cdot 10^{-2}$
$3.0 \cdot 10^{-2}$	$3.9379 \cdot 10^{-2}$	$2.8651 \cdot 10^{-2}$
$1.5 \cdot 10^{-2}$	$2.0446 \cdot 10^{-2}$	$1.4827 \cdot 10^{-2}$
$7.5 \cdot 10^{-3}$	$9.6444 \cdot 10^{-3}$	$7.3294 \cdot 10^{-3}$

(b) Four quadrature points.

Table 1: H_1 and energy norm error values for two and four quadrature points per interface segment of elliptical inclusion problem.

Effect of the quadrature order for the multiscale filter. The "heart" interface fields are evaluated at a discrete set of points. This is required to integrate the right-hand side of projection (37). We set those points to be the quadrature points of a standard Gauss quadrature rule, applied to each of the line segments (2D) or triangles (3D) of the subtessalation associated with the embedded contact interfaces. The purpose of the present study is to choose an appropriately large quadrature order for the remainder of our work. Considering Figure 14b, 14c and Table 1, we see that switching from 2 points (*i.e.* Gauss quadrature of third order) to 4 (*i.e.* order 7) leads to only marginal changes in the error norms and to no measurable change in the overall convergence order of the LaTIn-CutFEM algorithm. In other words, order 3 is sufficiently accurate to yield results that cannot be distinguished from that obtained when using an overkill integrator. Therefore, we choose to use order 3 quadrature rules in all our numerical examples. We emphasise that this study indicates that the stable behaviour of the proposed P1-P1 LaTIn scheme is not due to some under-integration of the contact laws, but to the two-scale nature of the ("non-localised") local stage.

Convergence of the LaTIn algorithm. Next, we show that the LaTIn solver converges well with the number of LaTIn iterations. We investigate the evolution of the error in energy norm with LaTIn iterations for a fixed coarser mesh $h > h_f$. More precisely, we measure the error of our numerical solution at LaTIn iteration

$it < it_{max}$ with respect to the finest solution at the maximum iteration $it_{max} := 200$. The results plotted in Figure 14d indicate the LaTin-CutFEM algorithm reaches an algorithmically converged solution at $it \approx 20$, which is indicated by the flat part of the error curves. We can also observe that the error in the solutions remains constant after $it \approx 20$, *i.e.* the remainder is the discretisation error.

We emphasise that these results are non-classical (as far as we are aware, such convergence results for LaTin-based solvers have never been published). Most contributions only report the convergence of the algorithmic error of LaTin solvers. For our finest mesh, the algorithmic error is the one reported in Figure 14e, as the reference solution mesh and that of the approximate one are the same. However, the error curves of Figure 14d represent the *combined effect of algorithmic and discretisation errors*. When seen from this perspective, the convergence of the LaTin solver is faster than usually thought, as the algorithmic error quickly drops to the level of the discretisation error, which can only be overcome by mesh refinement. As a corollary, the algorithmic error alone decreases slowly (this has been acknowledged in several contributions), but there is in fact no reason to make it converge to level of errors that are below the level of pure discretisation error.
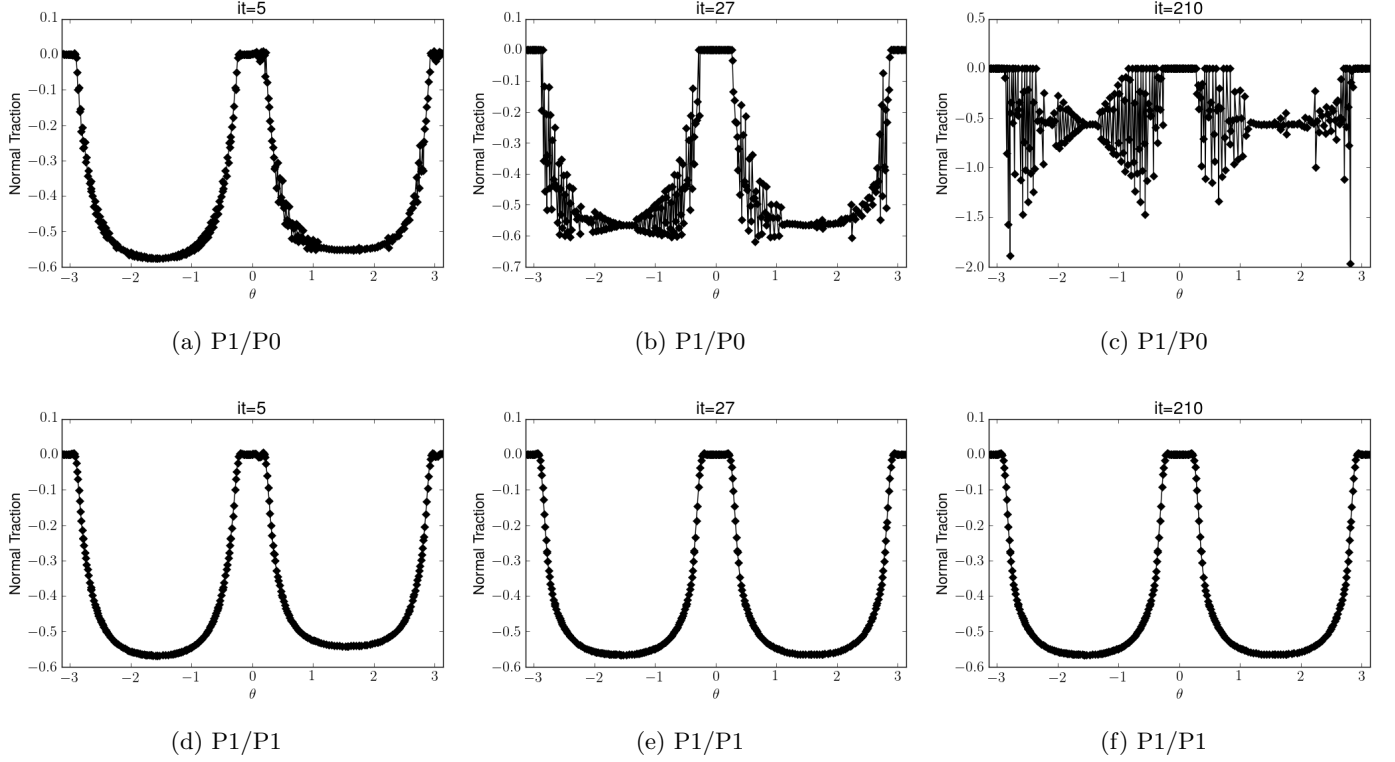
Figure 15: Comparison of current stable P1/P1 discretisation versus unstable P1/P0 discretisation for it= $\{5, 27, 210\}$ LaTIn iterations.

Convergence of the LaTIn algorithm for P1/P0 approximation versus P1/P1 approximation.
Figure 15 shows the comparison of our stable P1/P1 LaTin-CutFEM discretisation with a P1/P0 scheme. In the latter case, the bulk displacement is approximated by continuous piecewise linear polynomials but both interface fields are approximated as piecewise constant (the continuity is enforced in a weak sense, by enforcing that the projection of the corresponding residual in the space of piecewise constant interface fields vanishes). Figure 15 shows the normal traction, $\underline{F}_h^{2,1} \cdot \underline{n}^{2,1}$, along the elliptical contact interface Γ_C in terms of the angle $\theta = \arctan(y/x)$. We can see that with increasing LaTIn iterations the solution for the P1/P1 scheme has reached a converged state at $it = 27$ and the solution remains virtually unchanged at $it = 210$. However, for the P1/P0 scheme, we observe an onset of oscillations in the normal traction at $it = 5$, which grow substantially with LaTIn iterations. At $it = 27$, we see substantial pollution of the normal tractions through these oscillations and at $it = 210$, these oscillation completely destroy the normal tractions.

4.2.2 Multiply-connected interface points: effect of simultaneous enrichments

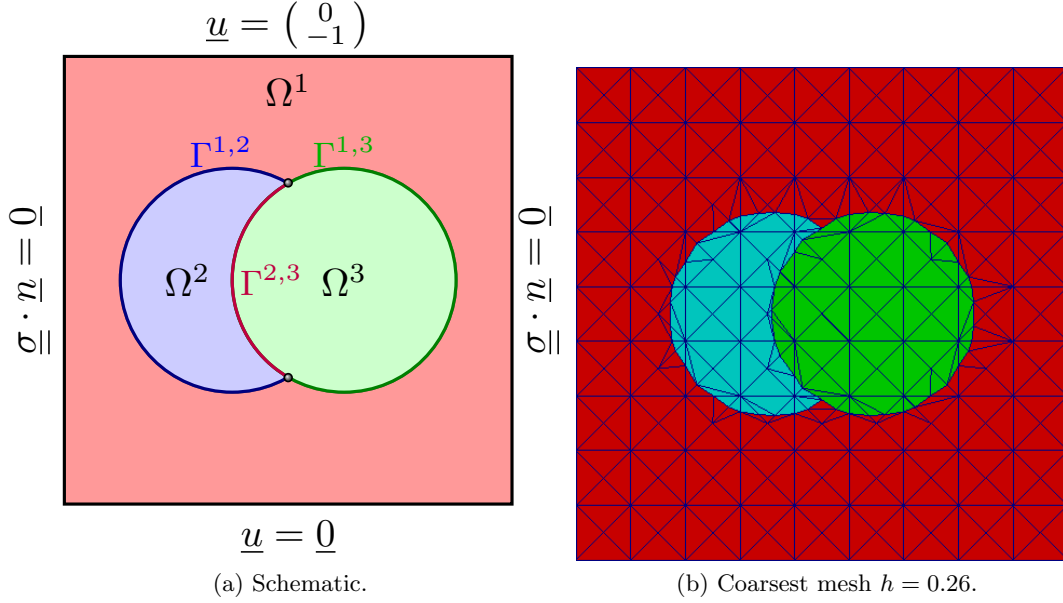(a) Schematic. (b) Coarsest mesh $h = 0.26$.

Figure 16: Schematic of two inclusions with boundary conditions and piecewise linear approximation of the geometry in coarsest mesh used.

In our next example, we consider two intersecting inclusions in a rectangular domain $\Omega = [-1.2, 1.2] \times [-1.2, 1.2]$, which is the example referenced throughout this article. The two circular inclusions are described by the level set functions

$$\phi_1(x, y) = \sqrt{(x - x_M^1)^2 + y^2} - r, \tag{59}$$

$$\phi_2(x, y) = \sqrt{(x - x_M^2)^2 + y^2} - r, \tag{60}$$

where $r = 0.5$, $x_M^1 = -0.25$ and $x_M^2 = 0.25$. We choose $k^+ = k^- = 1.0$, $\gamma_g = 0.1$, $\gamma_\Pi = 0.1$ and $\alpha = 10$. The resulting geometry has two triple junctions which will yield high stress concentrations at these points. We apply a displacement of $\underline{u} = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$ at the top boundary of the rectangular domain, a zero displacement at the bottom and zero Neumann conditions at the side of the domain (see Figure 16a). Figure 19 shows the vertical displacement, the shear stress σ_{xy} and the normal stress σ_{yy} for the two inclusion problem with no contrast in $E^1 = E^2 = E^3 = 1$ on the left and with a contrast of $E^1 = 1$ and $E^2 = E^3 = 10$ on the right. In the case of stiffer inclusions than the matrix material, the inclusions deviate only slightly from their circular shape while in the case of no contrast the inclusions undergo a much stronger deformation.

We choose a mesh configuration as displayed in Figure 16b and hierarchically refine the mesh as described in the previous example. We compare our coarse numerical solution with the finest numerical solution ($h = 0.008$) at LaTin iteration 200. The H_1 and energy norm errors with mesh refinement are shown in Figure 17 for a Young's modulus of $E^1 = E^2 = E^3 = 1$ in each domain and in Figure 18 for a contrast of the Young's modulus of $E^2 = E^3 = 10$ for the inclusions to $E^1 = 1$ in the background domain. In both cases, with and without contrast in the Young's modulus between the inclusions and the background domain, we obtain convergence rates of first order which is optimal for our strategy, which relies on the P1 finite element method.
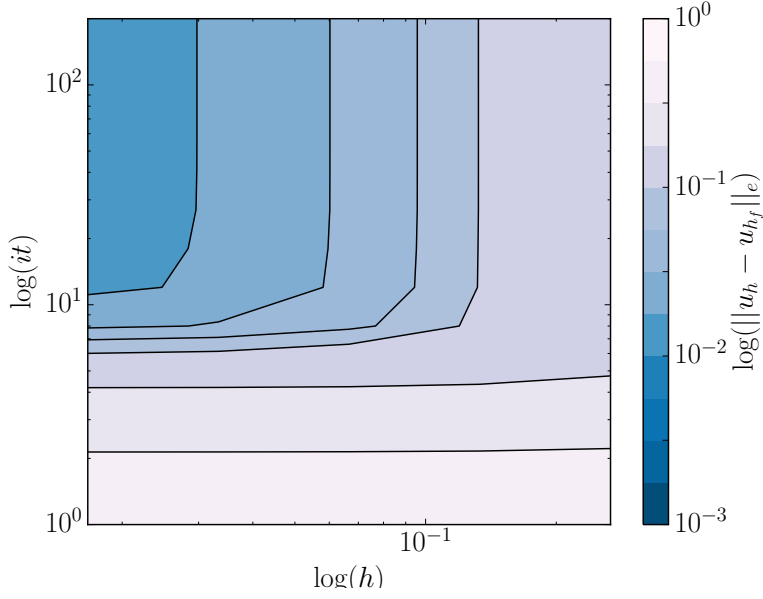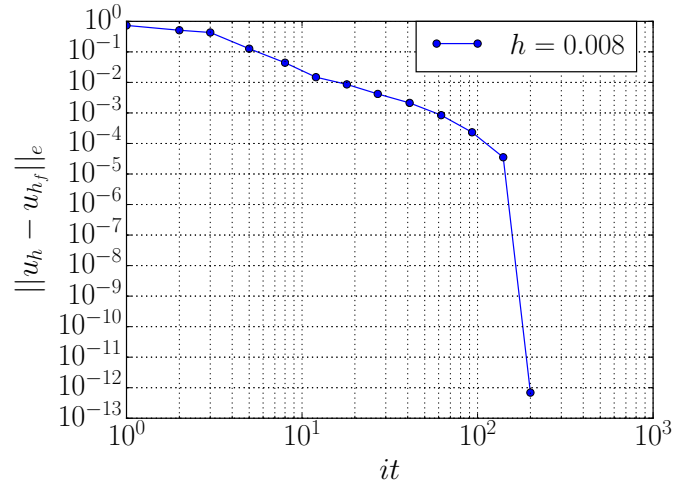
(a) Energy norm error vs iterations and mesh size.
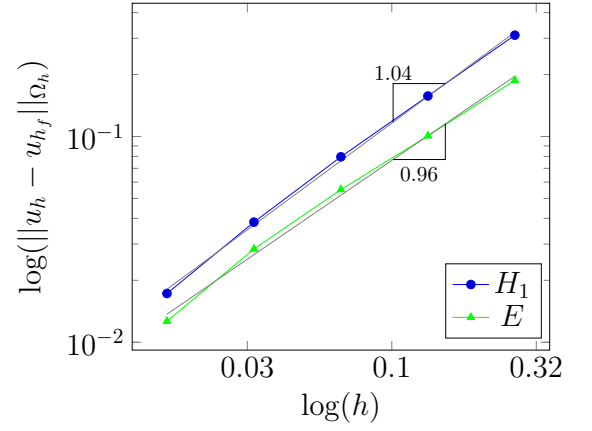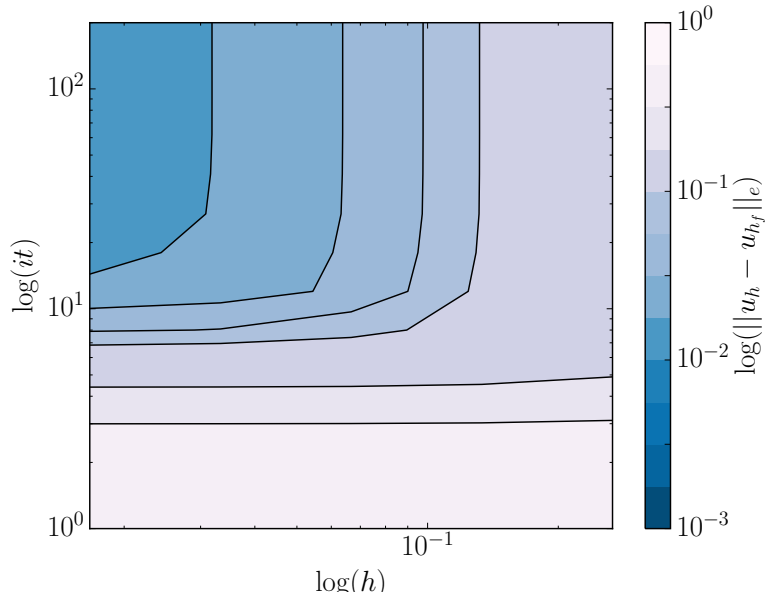
(b) Convergence rates.
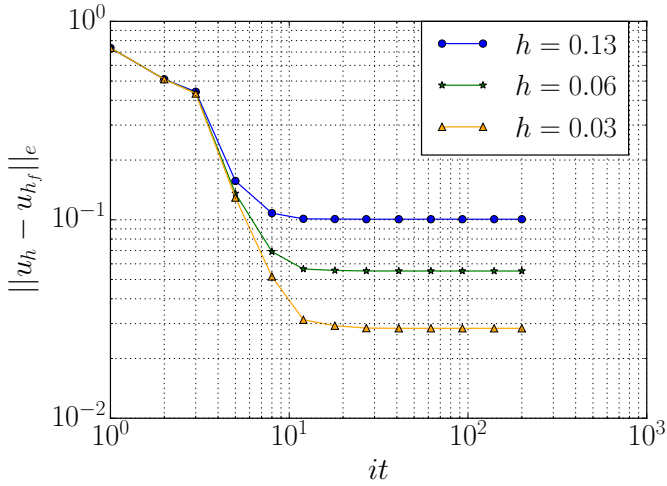
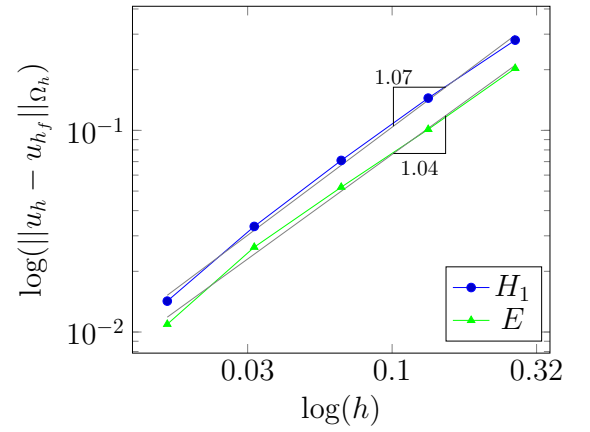(c) Energy norm error with iteration count for coarser mesh sizes. (d) Energy norm error with iteration count for finest solution.

Figure 17: Convergence rates and energy norm error with iteration count for two inclusions with $E^1 = E^2 = E^3 = 1$.

(a) Energy norm error vs iterations and mesh size.

(b) Convergence rates.

Figure 18: Convergence rates and energy norm error with iteration count for two inclusions with $E^2 = E^3 = 10$ in a matrix with $E^1 = 1$.
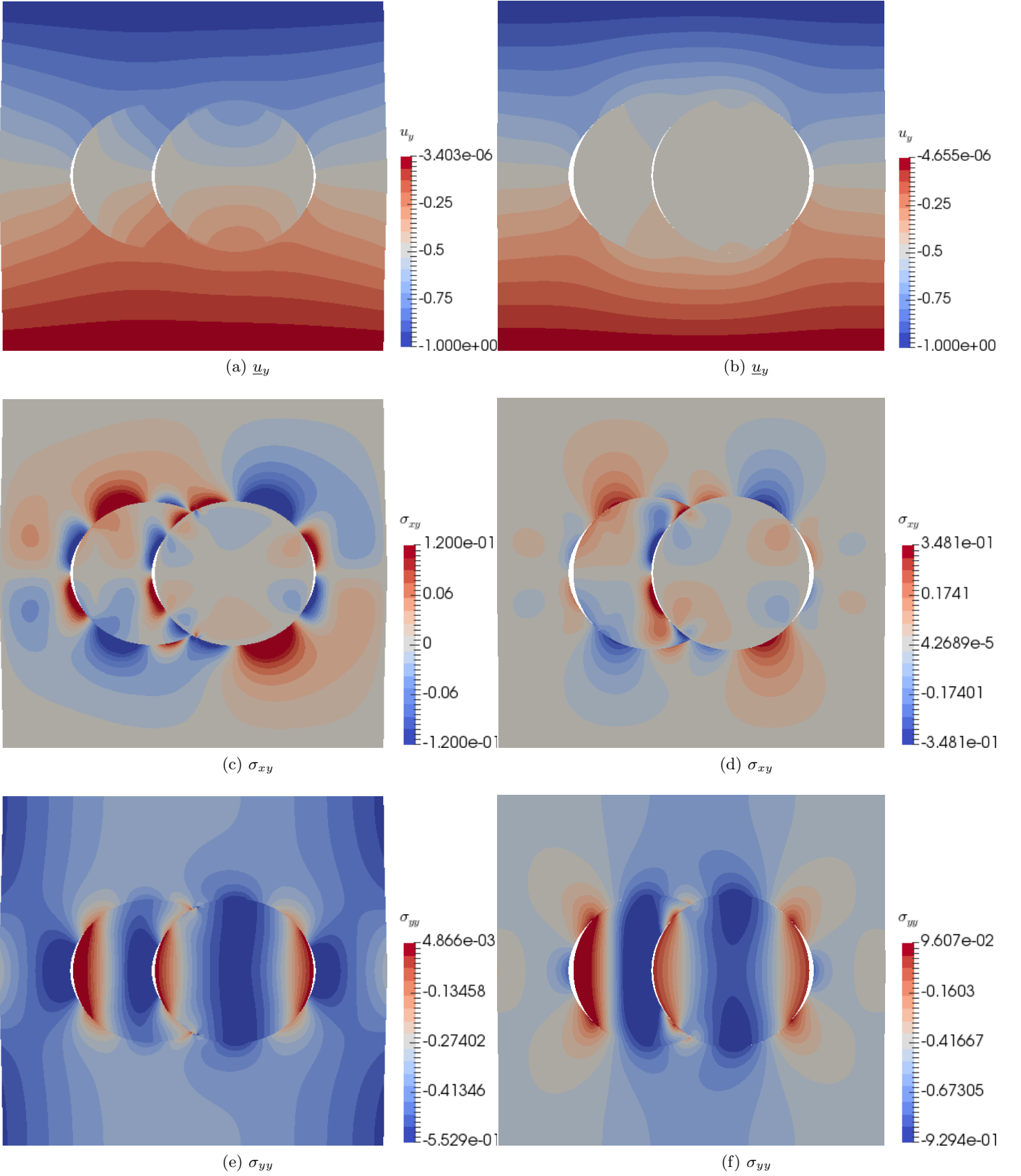
Figure 19: Displacement and stress components of two inclusion for finest mesh with $E^1 = E^2 = E^3 = 1$ (left) and $E^1 = 1, E^2 = E^3 = 10$ (right).

28

4.3 An advanced 3D geometry: braided composite material

In this section, we consider a 3D braided composite geometry formed of interwoven threads. Each thread is created through the motion of an ellipsoid, given by

$$\phi_j^{thread}(x,y,z) = \sqrt{\left(\frac{x - x_M^j}{a}\right)^2 + \left(\frac{y - y_M^j}{b}\right)^2 + \left(\frac{z - z_M^j}{c}\right)^2} - r, \tag{61}$$

along a sinusoidal path defined by (x_M^j, y_M^j, z_M^j). For the ellipsoid, we choose $r = 1.0$, $a = 0.2$, $b = 0.2$ (width of the thread) and $c = 0.1$ (height of the thread). The three threads in x-direction in our example geometry are described by $x_M^j = x$, $y_M^j = \left[-\frac{\lambda}{2}, 0, \frac{\lambda}{2}\right]$, $z_M^j = A\sin\left(\left(\frac{2\pi(x+z_C^j)}{\lambda}\right)\right)$, $z_C^j = \left[\frac{\lambda}{2}, 0, \frac{\lambda}{2}\right]$. Here, $\lambda = 1$ is the length of the path oscillation (sinus period), $A = 0.1$ is the amplitude of the path oscillation and z_C^j is the phase shift. Each new thread is shifted in y-direction by half the period length and their phase is shifted by half the period length. We form a union of all threads in x-direction to obtain one level set function, *i.e.* $\phi_x(x,y,z) = \min_j(\phi_j^{thread}(x,y,z))$. The threads in the y-direction are combined into a second level set function analogously. They are shifted by $\frac{\lambda}{4}$ with respect to the threads in the x-direction to build the interwoven geometry.

We apply the boundary conditions displayed in Figure 20 to a domain $\Omega = \left[-\frac{1}{2}, -\frac{3}{4}, -\frac{1}{4}\right] \times \left[1, \frac{3}{4}, \frac{1}{4}\right]$. We set a Young's modulus of one in all subdomains and pull the threads and the matrix in y-direction. We choose $k^+ = k^- = 1.0$, $\gamma_g = 0.1$, $\gamma_\Pi = 0.1$ and $\alpha = 10$.

The resulting normal stress component σ_{yy} and the shear stress component σ_{xy} are displayed in Figure 21. Figure 22 shows the displacement in y-direction and the stress component σ_{zz} in a slice of the domain indicated in red in Figure 22a. We can qualitatively verify the numerical solution by inspecting the slice displayed in Figure 22b. A vertical axial stress component (compression) is transferred through interfaces where the tangent plane is horizontal only when the vertical jump of displacement vanishes, which means that composite phases are in contact. Conversely, we see that when phases are not in contact at interface points where the tangent plane is horizontal, then the vertical compressive stress is zero.

Let us also point out that neither in 3D nor in 2D did we observe the development of numerical instabilities around contact zones, which we believe is due to the proposed P1-P1 mixed LaTIn formulation, together with the globalisation of the local stage and of course the ghost-penalty regularisation of the linear stages.
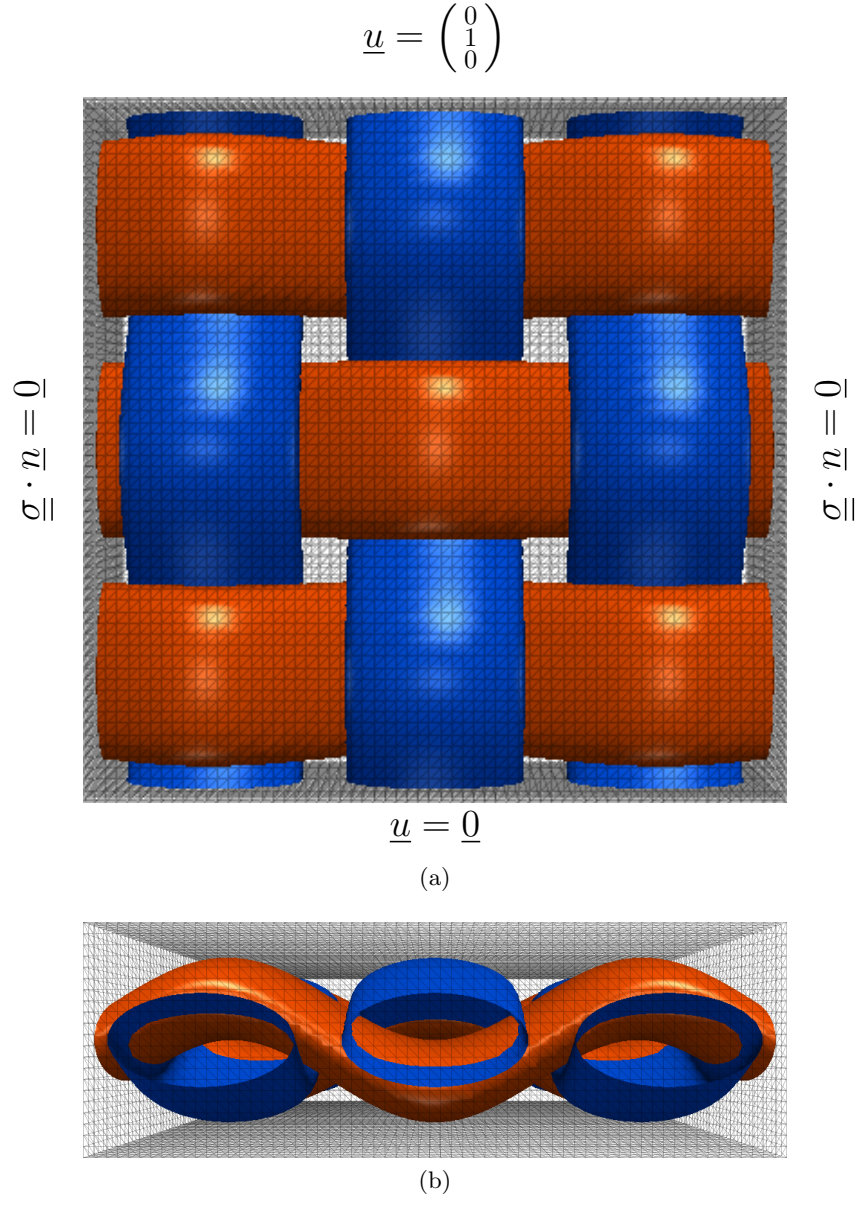
$$\underline{u} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

(a)

(b)

Figure 20: Zero level set contours of braided composite with mesh and boundary conditions.

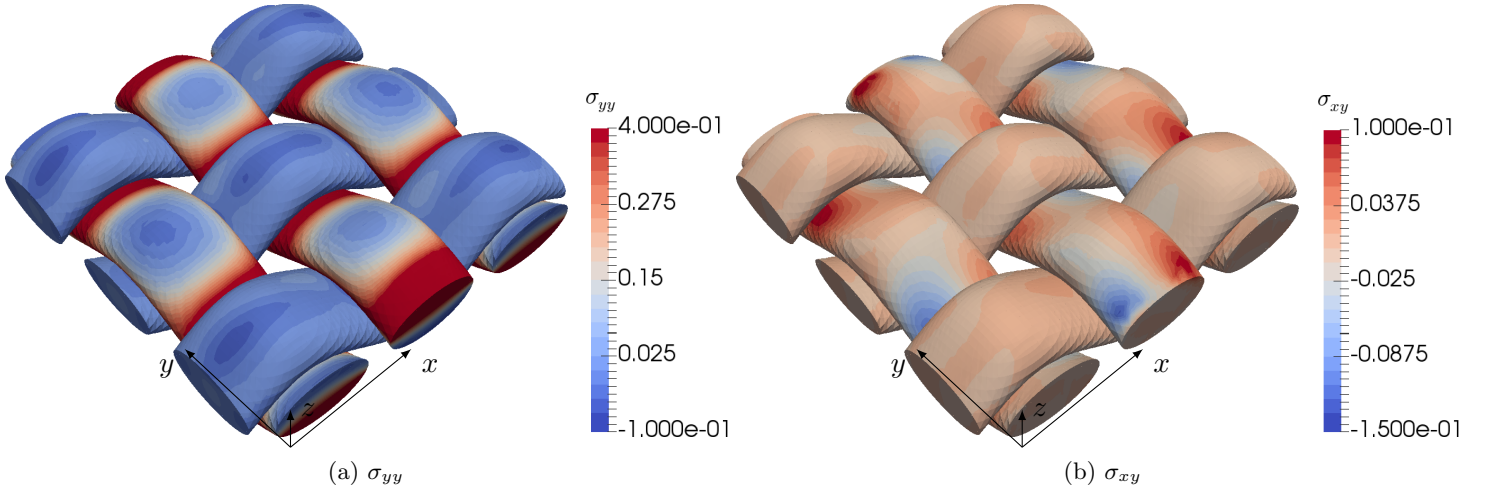(a) σ_{yy}

(b) σ_{xy}

Figure 21: The normal stress component σ_{yy} and the shear stress component σ_{xy} of the braided composite.

(a) Diplacement in y-direction.

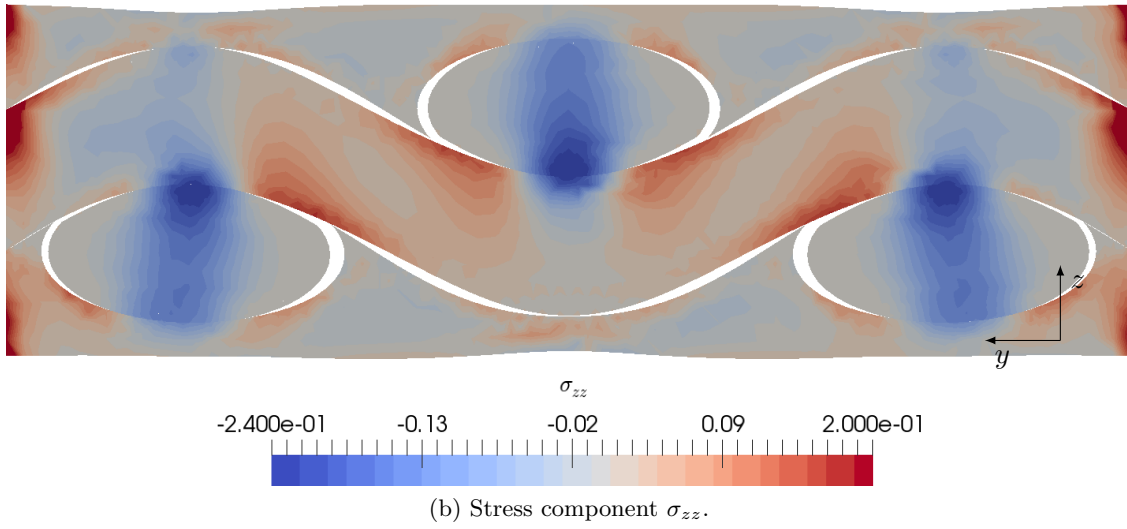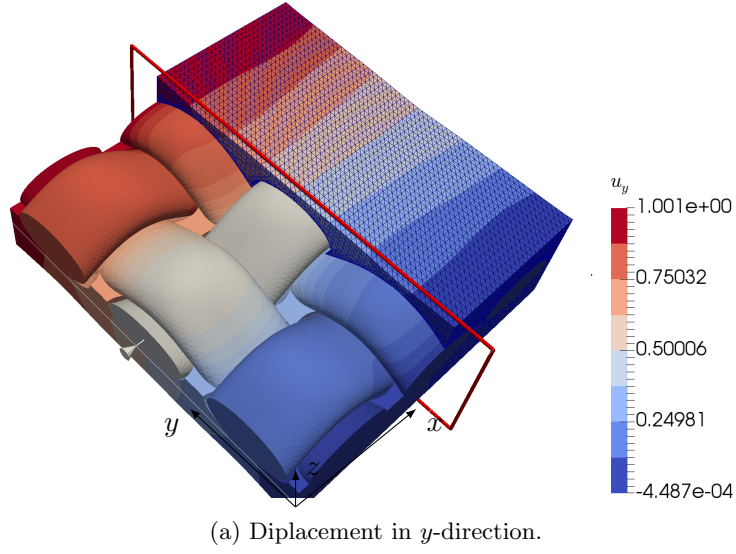(b) Stress component σ_{zz}.

Figure 22: Displacement in y-direction and normal stress component σ_{zz} in a slice of the domain indicated in red in (a).

5 Conclusions

We have presented a novel LaTIn-CutFEM approach for the simulation of multiple-body contact mechanics. The solver uses the enrichment technique that is the main characteristic of the CutFEM technology: the introduction of overlapping meshes (*i.e.* duplication of interface elements) to allow the representation of discontinuities in the solution fields across embedded interfaces. In order for the proposed scheme to be numerically stable, a P1/P1 reformulation of the LaTIn hybrid-mixed formulation has also been developed. Specifically, a weak enforcement of the contact conditions by a piecewise linear pair of interface fields prevents the growth and onset of spurious solution modes. Moreover, we have employed the ghost-penalty method, which prevents ill-conditioning of system matrices in the presence of "bad cuts". We have demonstrated that our scheme is optimally convergent with mesh refinement, and that it is as stable as a conforming, primal finite element formulation.

We have also demonstrated the capabilities and notably the high geometrical flexibility of our implementation of the proposed algorithm. The latter property has been obtained through the use of a versatile multiple-level set approach to describe the geometry of complex interacting solids. In particular, we have shown that our algorithm is robust in the case where multiple simultaneous enrichments are performed, which happens when several solids interact at the same point.

This work is expected to be a strong basis for further contributions to the field. For now, our main focus is to develop a robust stopping criterion for LaTIn. Indeed, we have shown that, in contrast to widespread belief, the LaTIn algorithm is relatively fast. In our examples, where, arguably, the default search direction parameter is close to its optimum value, 20 iterations lead to solution states whereby the algorithmic error is blurred into the discretisation error. We hope to be able to make use of this observation to develop rational estimates stopping criteria for LaTIn formulations of complex interface laws.

Acknowledgments

References

[1] P. Alart and A. Curnier. A mixed formulation for frictional contact problems prone to Newton like solution methods. *Computer Methods in Applied Mechanics and Engineering*, 92(3):353 – 375, 1991.

[2] O. Allix, P. Kerfriden, and P. Gosselet. On the control of the load increments for a proper description of multiple delamination in a domain decomposition framework. *International Journal for Numerical Methods in Engineering*, 83:1518–1540, 2010.

[3] O. Allix, P. Kerfriden, and P. Gosselet. A relocalization technique for the multiscale computation of delamination in composite structures. *Computer Modeling in Engineering and Sciences*, 55(3):271–291, 2010.

[4] Martin Alnæs, Jan Blechta, Johan Hake, August Johansson, Benjamin Kehlet, Anders Logg, Chris Richardson, Johannes Ring, Marie E Rognes, and Garth N Wells. The FEniCS project version 1.5. *Archive of Numerical Software*, 3(100):9–23, 2015.

[5] Chandrasekhar Annavarapu, Martin Hautefeuille, and John E Dolbow. A Nitsche stabilized finite element method for frictional sliding on embedded interfaces. part ii: intersecting interfaces. *Computer Methods in Applied Mechanics and Engineering*, 267:318–341, 2013.

[6] Chandrasekhar Annavarapu, Martin Hautefeuille, and John E Dolbow. A Nitsche stabilized finite element method for frictional sliding on embedded interfaces. part i: single interface. *Computer Methods in Applied Mechanics and Engineering*, 268:417–436, 2014.

[7] S. Bordas and B. Moran. Enriched finite elements and level sets for damage tolerance assessment of complex structures. *Engineering Fracture Mechanics*, 73(9):1176–1201, 2006.

[8] P.-A. Boucard and L. Champaney. A suitable computational strategy for the parametric analysis of problems with multiple contact. *International Journal for Numerical Methods in Engineering*, 57(9):1259–1281, 2003.

[9] E. Burman. Ghost penalty. *C. R. Math. Acad. Sci. Paris*, 348(21-22):1217–1220, 2010.

[10] Erik Burman, Susanne Claus, Peter Hansbo, Mats G. Larson, and André Massing. CutFEM: Discretizing geometry and partial differential equations. *International Journal for Numerical Methods in Engineering*, 104(7):472–501, 2015.

[11] Erik Burman and Peter Hansbo. Deriving robust unfitted finite element methods from augmented Lagrangian formulations. *arXiv preprint arXiv:1702.08340*, 2017.

[12] L. Champaney, P. A. Boucard, and S. Guinard. Adaptive multi-analysis strategy for contact problems with friction. *Computational Mechanics*, 42(2):305–315, 2008.

[13] L. Champaney, J.Y. Cognard, and P. Ladevèze. Modular analysis of assemblages of three-dimensional structures with unilateral contact conditions. *Computers and Structures*, 73:249–266, 1999.

[14] Franz Chouly and Patrick Hild. A Nitsche-based method for unilateral contact problems: numerical analysis. *SIAM Journal on Numerical Analysis*, 51(2):1295–1307, 2013.

[15] Franz Chouly, Patrick Hild, and Yves Renard. Symmetric and non-symmetric variants of Nitsche's method for contact problems in elasticity: theory and numerical experiments. *Mathematics of Computation*, 84(293):1089–1112, 2015.

[16] John Dolbow, Nicolas Moës, and Ted Belytschko. An extended finite element method for modeling crack growth with frictional contact. *Computer methods in applied Mechanics and engineering*, 190(51):6825–6846, 2001.

[17] T. Elguedj, A. Gravouil, and A. Combescure. A mixed augmented Lagrangian-extended finite element method for modelling elastic–plastic fatigue crack growth with unilateral contact. *International Journal for Numerical Methods in Engineering*, 71(13):1569–1597, 2007.

[18] Thomas-Peter Fries and Ted Belytschko. The intrinsic XFEM: a method for arbitrary discontinuities without additional unknowns. *International journal for numerical methods in engineering*, 68(13):1358–1385, 2006.

[19] A. Giacoma, D. Dureisseix, A. Gravouil, and M. Rochette. A multiscale large time increment/fas algorithm with time-space model reduction for frictional contact problems. *International Journal for Numerical Methods in Engineering*, 97(3):207–230, 2014.

[20] A. Gravouil, E. Pierres, and M. C. Baietto. Stabilized global–local X-FEM for 3d non-planar frictional crack using relevant meshes. *International Journal for Numerical Methods in Engineering*, 88(13):1449–1475, 2011.

[21] P.-A. Guidault, O. Allix, L. Champaney, and S. Cornuault. A multiscale extended finite element method for crack propagation. *Computer Methods in Applied Mechanics and Engineering*, 197(5):381–399, 2008.

[22] Anita Hansbo and Peter Hansbo. An unfitted finite element method, based on Nitsches method, for elliptic interface problems. *Computer methods in applied mechanics and engineering*, 191(47):5537–5552, 2002.

[23] Anita Hansbo and Peter Hansbo. A finite element method for the simulation of strong and weak discontinuities in solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, 193(33–35):3523 – 3540, 2004.

[24] P. Kerfriden, O. Allix, and P. Gosselet. A three-scale domain decomposition method for the 3D analysis of debonding in laminates. *Computational Mechanics*, 44(3):343–362, 2009.

[25] AR Khoei and M Nikbakht. Contact friction modeling with the extended finite element method (X-FEM). *Journal of Materials Processing Technology*, 177(1):58–62, 2006.

[26] P. Ladevèze. *Nonlinear Computational Structural Mechanics - New Approaches and Non-Incremental Methods of Calculation.* Springer Verlag, 1999.

[27] P. Ladevèze and D. Dureisseix. A 2-level and mixed domain decomposition approach for structural analysis. *Contemporary Mathematics*, 218:246–253, 1998.

[28] P. Ladevèze, A. Nouy, and O. Loiseau. A multiscale computational approach for contact problems. *Computer Methods in Applied Mechanics and Engineering*, 191:4869–4891, 2002.

[29] P. Ladevèze, J.C. Passieux, and D. Néron. The LaTIn multiscale computational method and the proper generalized decomposition. *Computer Methods in Applied Mechanics and Engineering*, 199(21):1287–1296, 2009.

[30] Tod A. Laursen. *Computational Contact and Impact Mechanics.* Springer-Verlag Berlin Heidelberg, 2003.

[31] C. Linder and F. Armero. Finite elements with embedded strong discontinuities for the modeling of failure in solids. *International Journal for Numerical Methods in Engineering*, 72(12):1391–1433, 2007.

[32] Fushen Liu and Ronaldo I Borja. A contact algorithm for frictional crack propagation with the extended finite element method. *International Journal for Numerical methods in engineering*, 76(10):1489–1512, 2008.

[33] T. W. McDevitt and T. A. Laursen. A mortar-finite element formulation for frictional contact problems. *International Journal for Numerical Methods in Engineering*, 48(10):1525–1547, 2000.

[34] J. Melenk and I. Babuška. The partition of unity finite element method: basic theory and applications. *Computer Methods in Applied Mechanics and Engineering*, 39:289–314, 1996.

[35] Barry Merriman, James K Bence, and Stanley J Osher. Motion of multiple junctions: A level set approach. *Journal of Computational Physics*, 112(2):334–363, 1994.

[36] N. Moës, J. Dolbow, and T. Belytschko. A finite element method for crack growth without remeshing. *International Journal of Engineering Science*, 46:131–150, 1999.

[37] J. Mosler and G. Meschke. 3d modelling of strong discontinuities in elastoplastic solids: fixed and rotating localization formulations. *International Journal for Numerical Methods in Engineering*, 57(11):1553–1576, 2003.

[38] DS Mueller-Hoeppe, P Wriggers, and S Loehnert. Crack face contact for a hexahedral-based XFEM formulation. *Computational Mechanics*, 49(6):725–734, 2012.

[39] J. Oliver, M. Cervera, and O. Manzoli. Strong discontinuities and continuum plasticity models: the strong discontinuity approach. *International Journal of Plasticity*, 15(3):319 – 351, 1999.

[40] Rachelle Ribeaucourt, M-C Baietto-Dubourg, and Anthony Gravouil. A new fatigue frictional contact crack propagation model with the coupled X-FEM/LATIN method. *Computer Methods in Applied Mechanics and Engineering*, 196(33):3230–3247, 2007.

[41] RI Saye and JA Sethian. Analysis and applications of the voronoi implicit interface method. *Journal of Computational Physics*, 231(18):6051–6085, 2012.

[42] J.C. Simo and T.A. Laursen. An augmented Lagrangian treatment of contact problems involving friction. *Computers & Structures*, 42(1):97 – 116, 1992.

[43] Kurt A Smith, Francisco J Solis, and David Chopp. A projection method for motion of triple junctions by level sets. *Interfaces and Free Boundaries*, 4(3):263–276, 2002.

[44] David P Starinshak, Smadar Karni, and Philip L Roe. A new level set model for multimaterial flows. *Journal of Computational Physics*, 262:1–16, 2014.

[45] P. Wriggers. *Computational Contact Mechanics*. Springer-Verlag Berlin Heidelberg, 2002.